

**Guide**



# M4 ISO Isometrics for PTC Creo® Piping

Version 4.2

All rights reserved. No part of this documentation may be reproduced in any manner (print, photocopy or other) without the written permission of CAD Schroer GmbH. CAD Schroer GmbH has made its best effort to ensure that the information in this document is accurate and reliable, but cannot guarantee the accuracy, timeliness, reliability or completeness of any of the information contained herein. CAD Schroer GmbH will not make any warranty nor accept legal responsibility or liability of any kind for consequences resulting from errors or omissions.

Registered Trademarks of CAD Schroer GmbH:  
MEDUSA, STHENO

Trademarks of CAD Schroer GmbH:  
MEDUSA4, STHENO/PRO, MEDEA, MPDS4

Third-Party Products and Trademarks:  
Creo, Pro/ENGINEER, Pro/DETAIL and Pro/TOOLKIT are registered trademarks of PTC, Incorporated.  
All other brand or product names are trademarks or registered trademarks of their respective owners.

December 2023  
Copyright © CAD Schroer GmbH

CAD Schroer GmbH  
Fritz-Peters-Str. 11  
47447 Moers  
Germany

Tel. +49 2841 91 84 - 0  
Fax +49 2841 91 84 - 44  
e-mail: [info@cad-schroer.de](mailto:info@cad-schroer.de)  
[www.cad-schroer.de](http://www.cad-schroer.de)

## Contents

<b>1</b>	<b>Introduction .....</b>	<b>11</b>
<b>2</b>	<b>System Requirements .....</b>	<b>12</b>
<b>3</b>	<b>Installation .....</b>	<b>13</b>
<b>4</b>	<b>M4ISOCreo Installation problems.....</b>	<b>23</b>
4.1	M4ISOCreo will not start when Creo launches .....	23
4.1.1	Creo is not referencing the M4ISOCreo application correctly. ....	23
4.1.2	The license is not installed correctly. ....	24
4.1.3	Wrong HOSTID.....	24
4.1.4	Keyfile Expired.....	25
4.1.5	Keyfile version .....	25
4.2	M4ISO button does not appear on Piping Ribbon.....	25
<b>5</b>	<b>Creating an Isometric .....</b>	<b>27</b>
5.1	Open the M4Iso dialog.....	28
5.2	Pipeline Selection .....	28
5.3	Selecting Spools .....	29
5.4	Reference CSYS .....	30
5.5	Define the Output Directory .....	31
5.6	Mapping Files .....	31
5.7	Drawing Style.....	32
5.8	North Angle.....	32
5.9	Max Count .....	33
5.10	Display Isometrics.....	33
5.11	ISOX File(s) Only .....	34
5.12	Generate the Isometric Drawing .....	34
5.13	Controlling the Isometric Drawing Filename .....	35
5.14	M4 ISO Cutting Allowance .....	36
5.15	Split Point Tab .....	37
5.16	Notes .....	38
5.16.1	Adding Notes to Components/Parts in Creo .....	38
<b>6</b>	<b>ISOREAL Mapping Files .....</b>	<b>40</b>



6.1	isoreal_endtype_map.ptd .....	40
6.1.1	Valid Endtypes.....	41
6.2	isoreal_nominal_size_map.ptd .....	41
6.2.1	Default Sizes.....	42
6.3	isoreal_symbol_map.ptd.....	43
6.3.1	Symbol mapping parameters .....	44
6.3.2	Default Symbols and Classes .....	44
6.4	isoreal_parameter_map.ptd .....	45
<b>7</b>	<b>Creo Environment Variables .....</b>	<b>47</b>
7.1	File handling .....	47
7.2	Creo Drawing Reuse .....	49
7.3	Dialog settings .....	49
7.4	Weld Fittings .....	50
7.5	Miscellaneous .....	50
<b>8</b>	<b>Creo drawing re-use .....</b>	<b>52</b>
<b>9</b>	<b>How to improve the clarity of the Isometric.....</b>	<b>53</b>
<b>10</b>	<b>3D View .....</b>	<b>54</b>
<b>11</b>	<b>M4ISO Configuration and customisation .....</b>	<b>56</b>
11.1	M4ISO Search Path.....	56
11.2	M4Iso Option Store file .....	56
11.2.1	File Structure .....	57
11.2.2	Option Groupings.....	57
11.3	Graphical Representations .....	58
11.3.1	Element Types.....	58
11.3.2	Branch and Bend .....	58
11.4	Creating a Custom Style Definition .....	59
11.5	Isometric units.....	59
11.6	Defining a New Style .....	60
11.7	Isometric Sheet Template .....	61
11.7.1	Isometric Sheet Templates .....	63
11.7.2	3D View definition .....	64

11.7.3 Continuation Sheet .....	65
11.8 View Orientation .....	66
11.9 Dimensioning .....	67
11.9.1 Elevations and Falls .....	71
11.9.2 Show Cut Allowance Tolerance .....	73
11.9.3 Bore label format .....	74
11.9.4 Position Labels .....	74
11.9.5 Terminal Connection Labels .....	76
11.10 References .....	77
11.10.1 Common Settings .....	77
11.10.2 Show References .....	77
11.10.3 Bend References .....	78
11.10.4 Item ID's .....	78
11.10.5 Spool reference padding .....	79
11.10.6 Annotation Border Style .....	79
11.11 Tables .....	80
11.11.1 Parts List .....	80
11.11.2 Spools Table .....	88
11.11.3 Cuts Table .....	91
11.11.4 Welds Table .....	93
11.11.5 Bends Table .....	95
11.12 File Naming .....	100
11.12.1 Creo File names .....	100
11.12.2 CSV files .....	101
11.12.3 NC Files .....	102
11.12.4 Alternative Output directory .....	103
11.13 Creo Intermediate Files .....	103
11.14 Labels and Notes .....	104
11.14.1 Connection Labels .....	104
11.14.2 Header Notes .....	105
11.14.3 Notes Dialog configuration .....	107

11.15	Spec/Pipe Break settings .....	110
11.16	Flow Arrow symbol.....	111
11.17	M4ISOCreo Weld Fittings.....	113
11.17.1	Weld PART definition .....	113
11.17.2	Environment Variables .....	114
11.17.3	Parameter Mappings.....	114
11.17.4	OptionStore settings.....	115
11.17.5	SymbolInfo settings for welds.....	115
11.18	M4ISO Entity Color Settings.....	116
11.18.1	Mixed label colors .....	117
11.18.2	Creo color mapping.....	118
11.18.3	The Advanced Solution .....	120
11.18.4	An Example.....	124
<b>12</b>	<b>Creating custom component symbols .....</b>	<b>129</b>
12.1.1	The M4IsoSymbolInfo File .....	129
12.1.2	Example Symbol Creation .....	135
12.1.3	M4ISO End Conditions .....	140
12.1.4	M4ISO North Symbol Definition .....	145
<b>13</b>	<b>Appendix A – M4ISO Symbols .....</b>	<b>149</b>
13.1	Introduction.....	149
13.2	Built In.....	150
13.2.1	Elbow.....	150
13.2.2	Elbow_teel.....	150
13.2.3	Tee.....	150
13.2.4	Built In symbol dimensions .....	150
13.3	Isometric Symbols .....	151
13.3.1	Bellows .....	151
13.3.2	Block Angle.....	151
13.3.3	Block Offset .....	151
13.3.4	Block Return .....	152
13.3.5	Bush Reducer .....	152

13.3.6 Butt Weld .....	152
13.3.7 Cap .....	152
13.3.8 Cap Flanged .....	153
13.3.9 Cap Round.....	153
13.3.10 Coupling.....	153
13.3.11 Cross.....	153
13.3.12 Elbow .....	153
13.3.13 Elbow Reducing .....	154
13.3.14 Elbow Teed .....	154
13.3.15 Expanding Bellows.....	154
13.3.16 Expansion Joint.....	154
13.3.17 Field Weld .....	154
13.3.18 Filter Angle .....	155
13.3.19 Filter Inline Flanged.....	155
13.3.20 Filter Offset.....	155
13.3.21 Filter Return .....	155
13.3.22 Flame Trap.....	156
13.3.23 Flange (generic) .....	156
13.3.24 Flange Blind .....	156
13.3.25 Flange Blind P2.....	156
13.3.26 Flange Flare .....	157
13.3.27 Flange Lap .....	157
13.3.28 Flange with O-Ring .....	157
13.3.29 Flange with Groove .....	157
13.3.30 Flange Slip-On .....	158
13.3.31 Flange Stub.....	158
13.3.32 Flange Weld Neck.....	158
13.3.33 Flexible Hose .....	158
13.3.34 Flow Arrow .....	159
13.3.35 Flow Meter .....	159
13.3.36 Full Coupling .....	159

13.3.37 Gasket.....	159
13.3.38 Half Coupling.....	160
13.3.39 Half Coupling as Olet .....	160
13.3.40 Hose Coupling.....	160
13.3.41 Latrolet .....	160
13.3.42 Miscellaneous 2 Angle .....	160
13.3.43 Miscellaneous Branch .....	161
13.3.44 Miscellaneous Cross .....	161
13.3.45 Miscellaneous Inline .....	161
13.3.46 Miscellaneous Y .....	161
13.3.47 Nipolet.....	162
13.3.48 Nipple Barrel .....	162
13.3.49 Nipple Running.....	162
13.3.50 Olet .....	162
13.3.51 Orifice.....	163
13.3.52 Penetration.....	163
13.3.53 Penetration Small.....	163
13.3.54 Pipe Hanger .....	163
13.3.55 Pipe Hanger Small .....	164
13.3.56 Plug .....	164
13.3.57 Plug MB .....	164
13.3.58 Reducer Concentric .....	164
13.3.59 Reducer_Concentric Flanged.....	165
13.3.60 Reducer Eccentric.....	165
13.3.61 Reducer Eccentric Flanged.....	165
13.3.62 Screwed Connector.....	165
13.3.63 Setin .....	166
13.3.64 Sight Glass.....	166
13.3.65 Shop Weld.....	166
13.3.66 Socket Weld .....	166
13.3.67 Sockolet .....	167



13.3.68 Spec Break.....	167
13.3.69 Strainer .....	167
13.3.70 Strainer Angle .....	167
13.3.71 Strainer Inline Flanged .....	168
13.3.72 Strainer Offset .....	168
13.3.73 Strainer Return.....	168
13.3.74 Strainer Wye .....	168
13.3.75 Support Guide .....	169
13.3.76 Support Skid.....	169
13.3.77 Sweepolet .....	169
13.3.78 Tee.....	169
13.3.79 Tee 45.....	170
13.3.80 Tee Pitcher.....	170
13.3.81 Tee Set On.....	170
13.3.82 Terminal Equipment .....	170
13.3.83 Thredolet.....	170
13.3.84 Trap Angle.....	171
13.3.85 Trap Offset .....	171
13.3.86 Trap Return .....	171
13.3.87 Tundish .....	171
13.3.88 Tundish Angle .....	172
13.3.89 Union.....	172
13.3.90 Valve (generic) .....	172
13.3.91 Valve 3way.....	172
13.3.92 Valve 4way.....	173
13.3.93 Valve Angle .....	173
13.3.94 Valve Angle Relief .....	173
13.3.95 Valve Ball .....	173
13.3.96 Valve Ball Flange .....	174
13.3.97 Valve Butterfly .....	174
13.3.98 Valve Butterfly Flange .....	174

13.3.99	Valve Check .....	174
13.3.100	Valve Check 2 .....	175
13.3.101	Valve Check 3 .....	175
13.3.102	Valve Check 4 .....	175
13.3.103	Valve Control .....	175
13.3.104	Valve Control SQIN .....	176
13.3.105	Valve Diaphragm .....	176
13.3.106	Valve Diaphragm 2 .....	176
13.3.107	Valve Diaphragm 3 .....	176
13.3.108	Valve Diaphragm Flange .....	177
13.3.109	Valve Gate .....	177
13.3.110	Valve Generic .....	177
13.3.111	Valve Globe .....	177
13.3.112	Valve Needle .....	178
13.3.113	Valve Non-Return .....	178
13.3.114	Valve Plug .....	178
13.3.115	Valve Pressure Reducer .....	178
13.3.116	Valve Reducer .....	179
13.3.117	Valve Reducer 2 .....	179
13.3.118	Valve Reducer 3 .....	179
13.3.119	Valve Relief .....	179
13.3.120	Valve Slide .....	180
13.3.121	Valve Swing Check .....	180
13.3.122	Valve Trap .....	180
13.3.123	Valve Wafer .....	180
13.3.124	Weldolet .....	181
13.3.125	WYE Lat Tee .....	181
<b>14</b>	<b>Appendix B .....</b>	<b>182</b>
14.1	The Pen table configuration .....	182

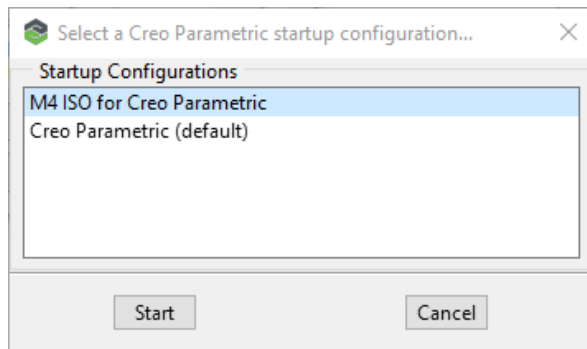
## 1 Introduction

**M4 ISO Isometrics for PTC Creo Piping** is a Creo Toolkit application created by CAD Schroer Group (CSG). This application allows users to automatically create 2D isometric drawings from 3D pipe geometry designed using PTC Creo Piping.

During installation the application is copied into a user specified directory. The installation program declares the toolkit application to Creo by modifying the existing *parametric.psf* file or creates a new file named *M4ISOCreo.psf*.

Start PTC Creo Parametric with the M4ISOCreo toolkit by selecting the *.psf* file created during installation.

### PTC Creo Parametric Start Configurations



## 2 System Requirements

Operating System	Windows 10 Pro 64 Bit Windows 11 Pro
PTC Creo Version	Creo V 6.0 Creo V 7.0 Creo V 8.0 Creo V 9.0 Creo V10.0

The product requires a node-locked license file with three license features:

- cadconvert
- med\_iso
- med\_iso4creo

License servers are not supported.

The license version is CSGGLMD 10.720

Existing M4ISOCreo users of earlier versions will need to upgrade their license to use this version.

### 3 Installation

This section details the installation process for the M4ISO Isometrics for PTC Creo® Piping application software. Note that the software must be installed on the machine nominated for the application. The supplied keyfile is node-locked to this machine only.

The following applies to the entire installation process:

Selecting the **Next >** button will action any changes you make to the dialog and move onto the next page in the process.

Selecting the **< Back** button allows you to step back through the process and make changes as required.

1. Download the installation executable.

You should have been supplied with a link to download the installation software. For the release software, this file is named:

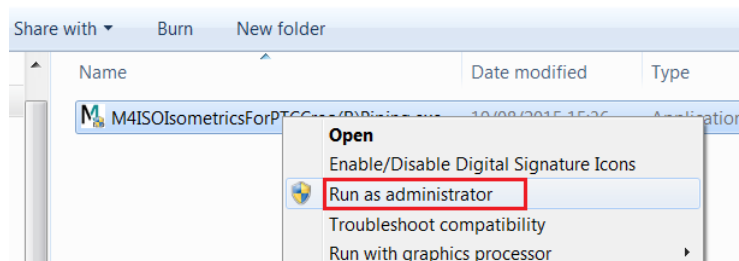
*M4ISOIsometricsForPTCCreo(R)Piping\_420.exe*

2. Run the installation in 'Administrator' mode

The software will require administrator privileges to install the application and to make modifications to the Creo Parametric installation.

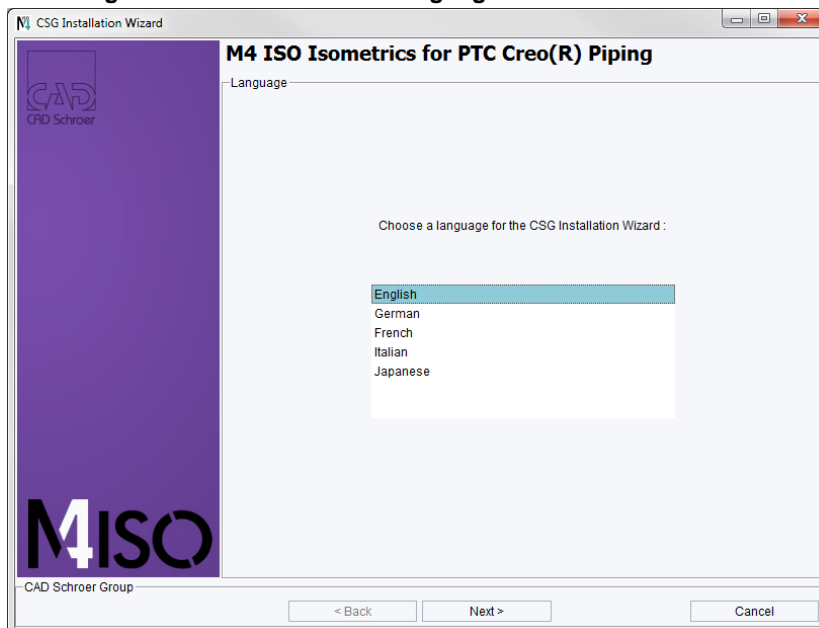
It is not sufficient to log in as a user with administrator privileges, it is also necessary to run the installer software in 'Administrator' mode. This can be achieved by selecting the installer file in a Windows Explorer dialog and then using the right mouse button menu to select 'Run as administrator'

#### Run as administrator

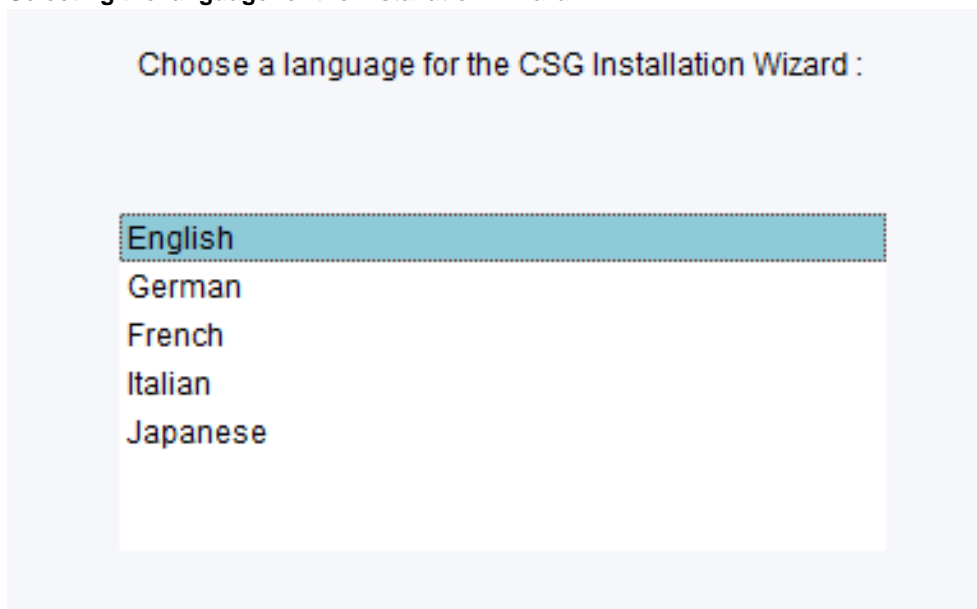


3. The CSG Installation Wizard dialog opens.

#### Choosing the Installation Wizard Language



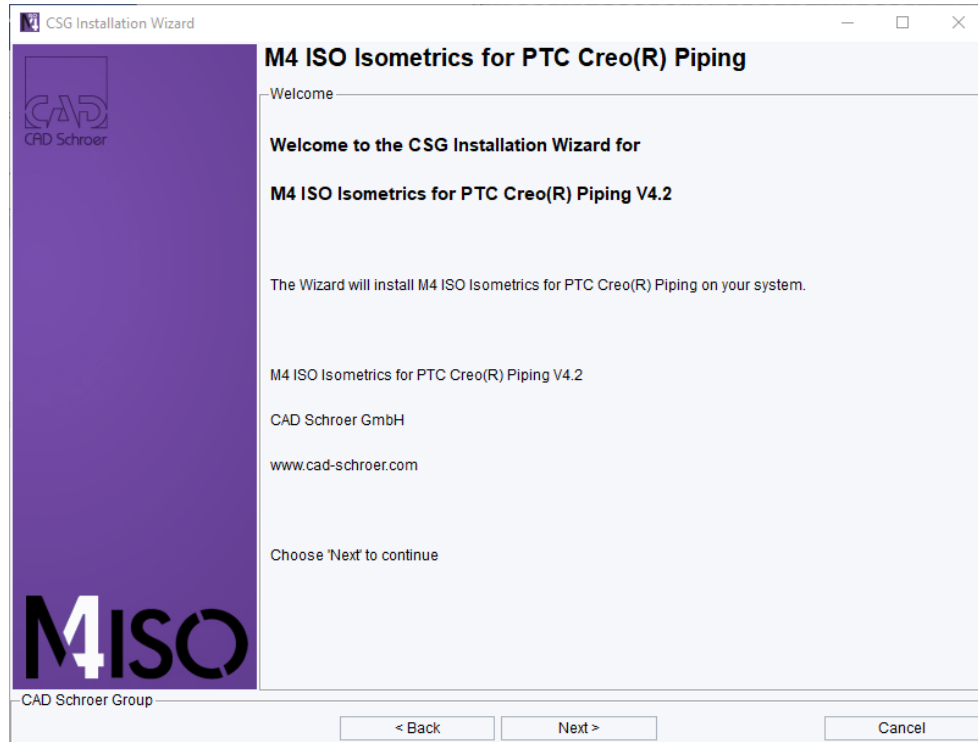
4. Select the installation process language then select Next >  
Selecting the language for the installation Wizard





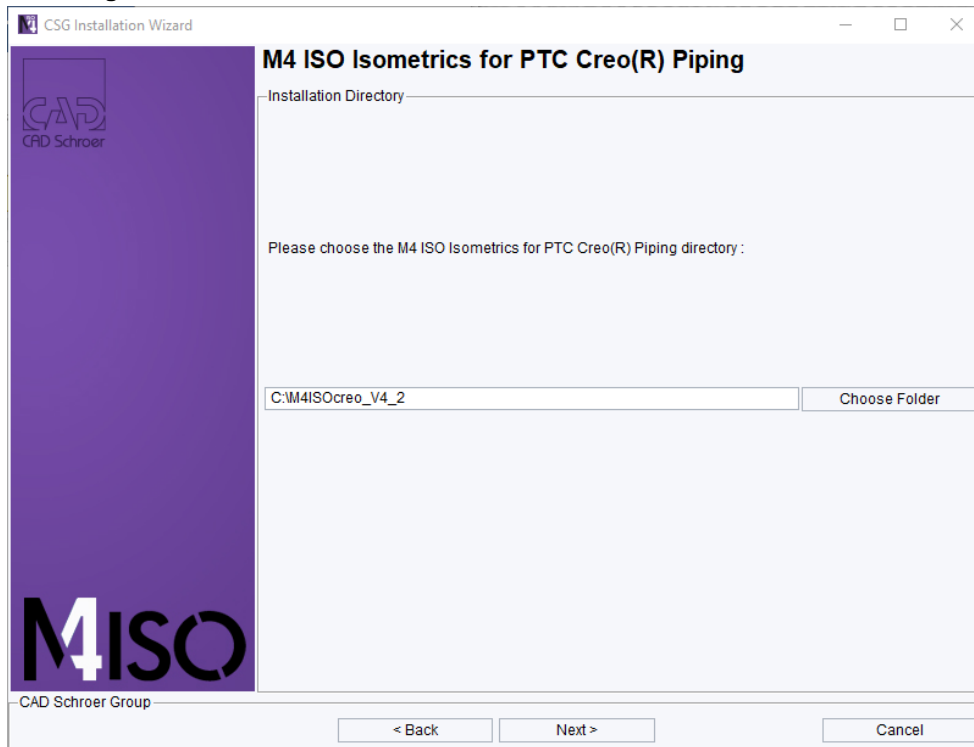
5. The Welcome page is displayed in the language that was selected.

**Installation Welcome Page**



Select Next > to move onto the next page.

6. Choose the application installation folder (default: C:\M4ISOcreo\_V4\_2)  
**Choosing the Installation Folder**



The **Choose Folder** button allows you to select an existing folder to install into. Note that the folder must be empty.

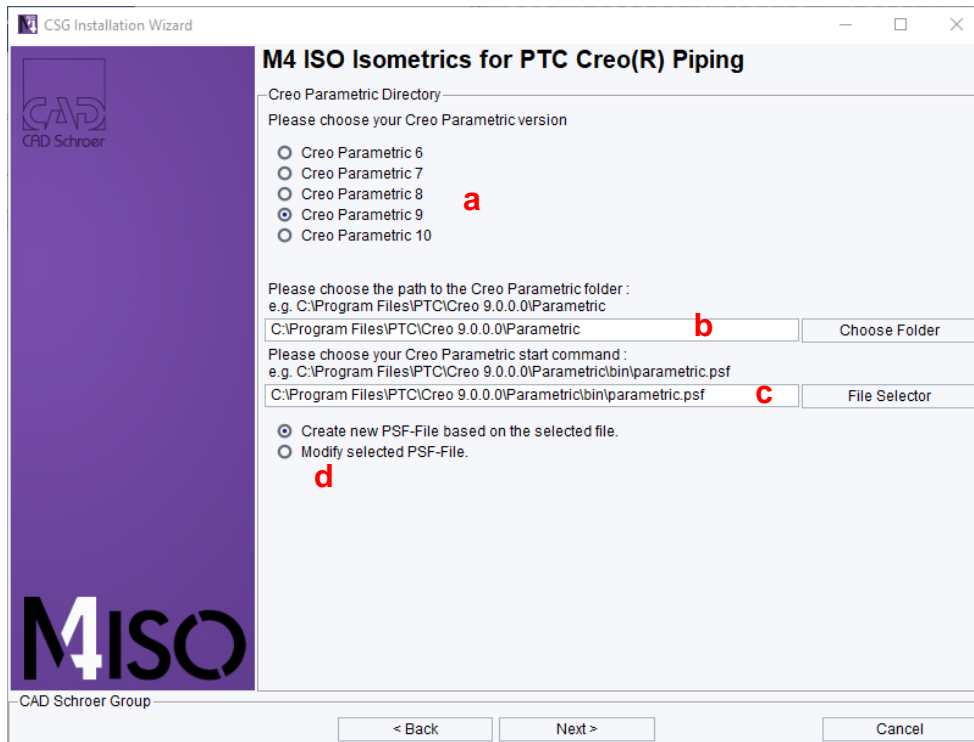
You can also specify the installation path by entering (or modifying) the value in the input field.

## 7. Specify the Creo Parametric Installation settings.

Select :

- Your Creo Parametric version
- Your Creo Parametric folder path
- Your Creo Parametric start file (default: parametric.psf)
- Choose to copy the existing psf file to a new file named M4ISOcreo.psf or modify the existing file identified at c.

### Creo Parametric Settings



CSG Installation Wizard

**M4 ISO Isometrics for PTC Creo(R) Piping**

Creo Parametric Directory

Please choose your Creo Parametric version

☐ Creo Parametric 6

☐ Creo Parametric 7

☐ Creo Parametric 8

☒ **a** Creo Parametric 9

☐ Creo Parametric 10

Please choose the path to the Creo Parametric folder :  
e.g. C:\Program Files\PTC\Creo 9.0.0.0\Parametric

C:\Program Files\PTC\Creo 9.0.0.0\Parametric **b** Choose Folder

Please choose your Creo Parametric start command :  
e.g. C:\Program Files\PTC\Creo 9.0.0.0\Parametric\bin\parametric.psf

C:\Program Files\PTC\Creo 9.0.0.0\Parametric\bin\parametric.psf **c** File Selector

☒ Create new PSF-File based on the selected file.

☐ **d** Modify selected PSF-File.

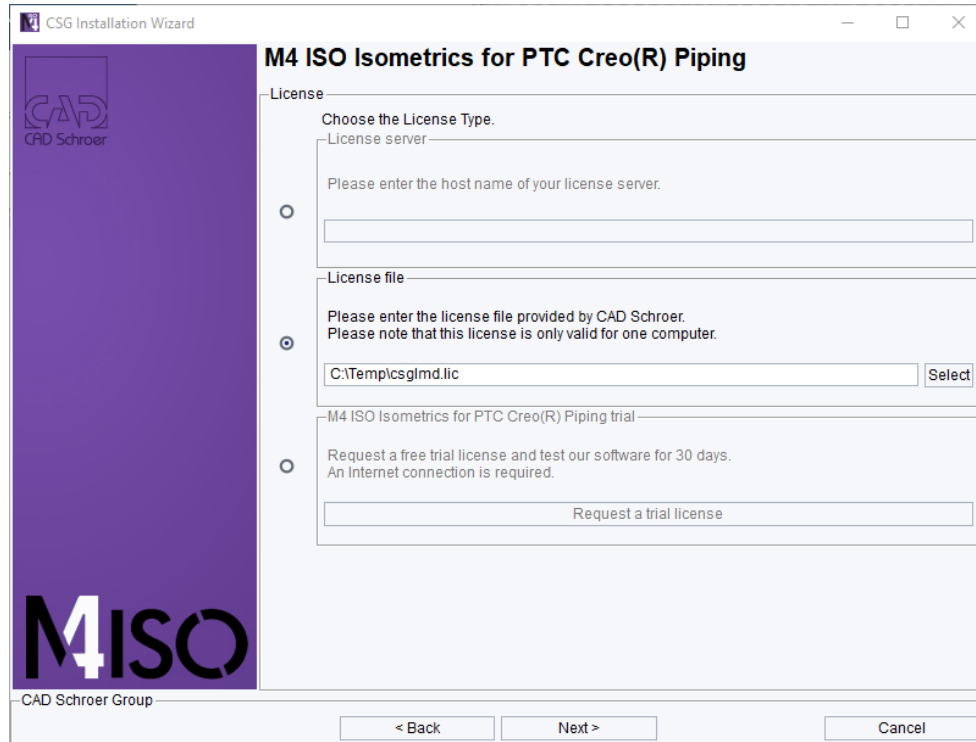
CAD Schroer Group

< Back Next > Cancel

8. Select the license keyfile path.

You will receive the keyfile by email as an attachment. You should save the attachment onto a convenient, accessible place on your installation machine.

**Selecting the license file**

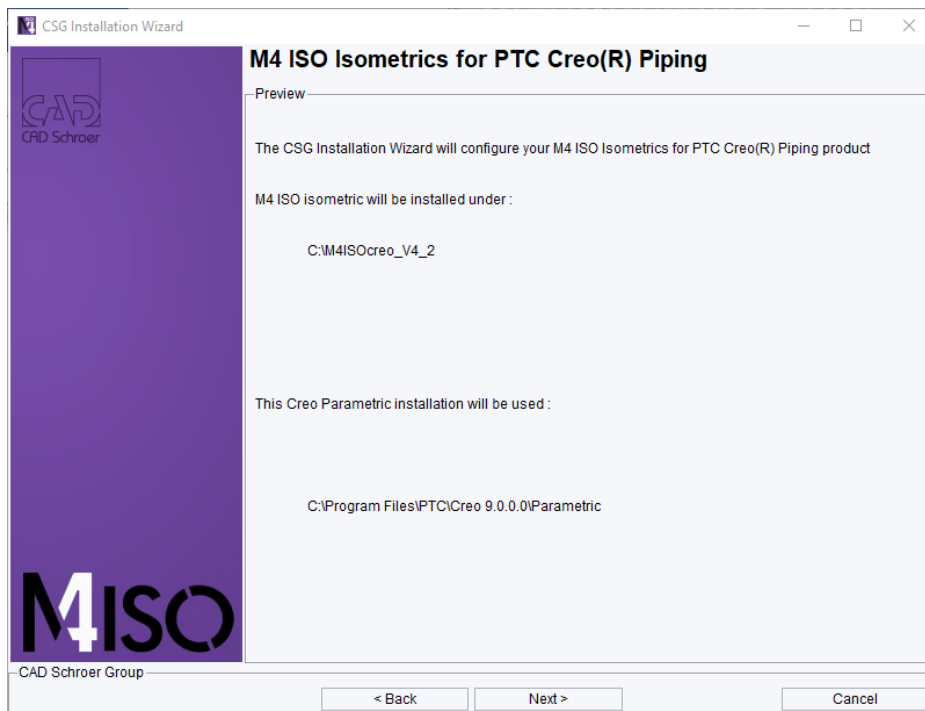


Please use the 'License file' option then use the 'Select' button to find and select the license file you received from CAD Schroer.

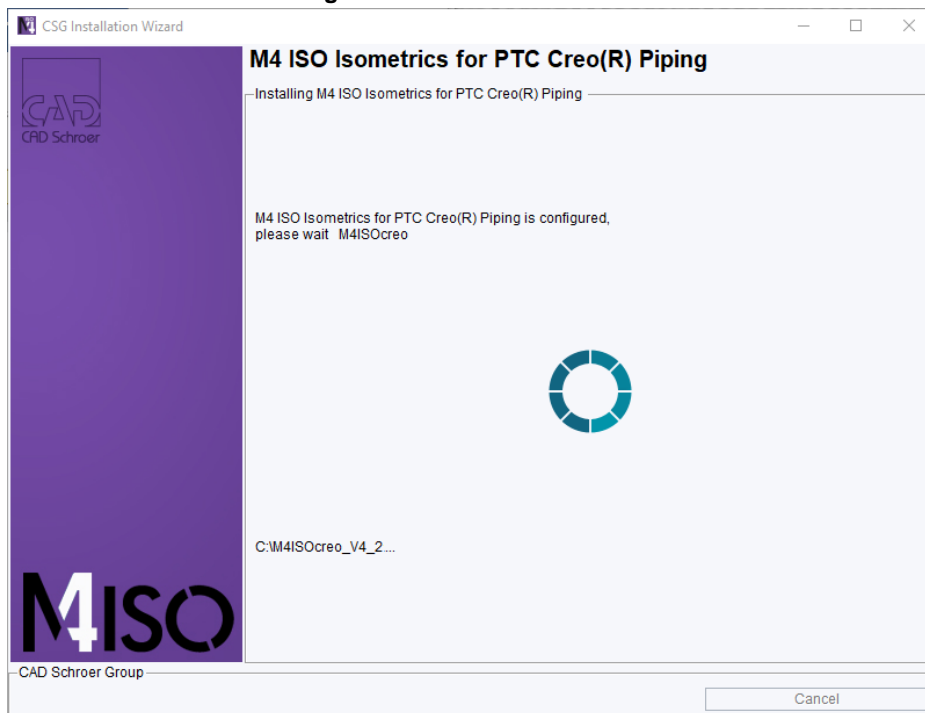
If you are running a trial of the software, use the 'M4ISO Isometrics for PTC Creo(R) Piping trial' option and press the 'Request a trial license' button. This will automatically request and install a 30-day trial license using the internet. You will need to supply some contact details to request the trial license.

## 9. Review the installation details

### Installation Review



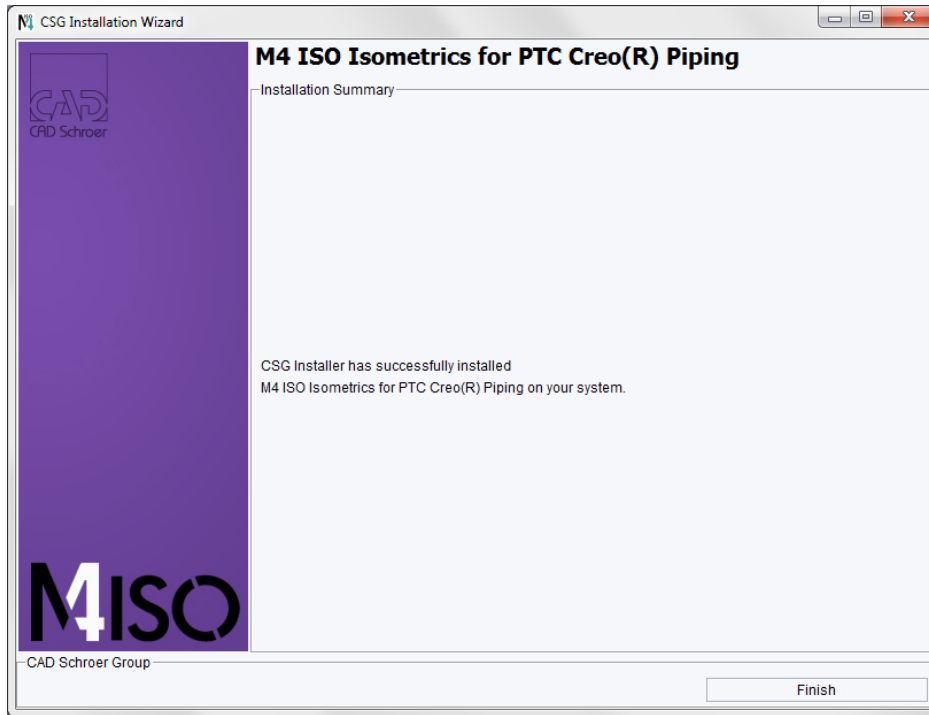
## 10. Wait while installation completes

**Installation Process Running**



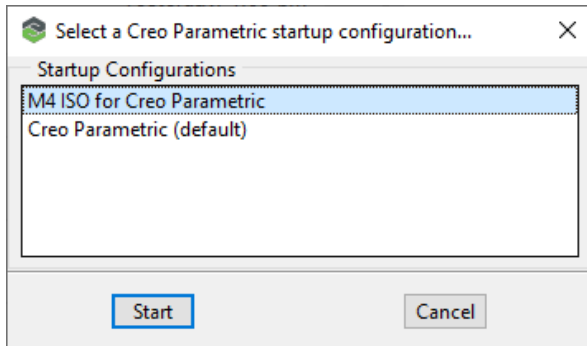
11. When the installation is complete, the Installation Summary page is displayed.  
Select **Finish** to close the installer dialog.

**Installation is Complete**



After successful installation of the software, you can start the M4ISOcreo version of Creo Parametric by selecting the M4ISO for Creo Parametric configuration option during Creo Parametric startup.

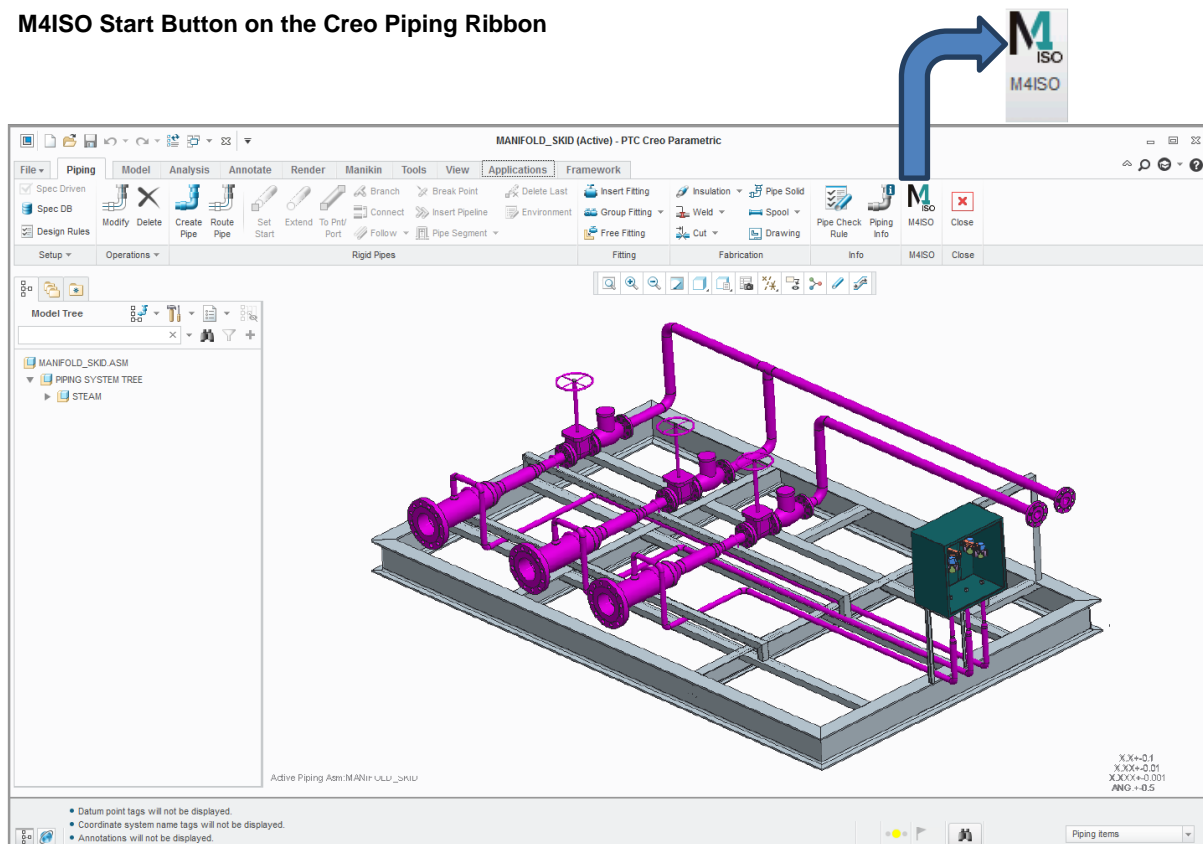
#### Starting M4ISO for Creo



Alternatively, you may want to create a startup batch file (based on the existing parametric.bat file) that references the M4ISOcreo.psf file directly. You could then create a desktop shortcut for this startup file.

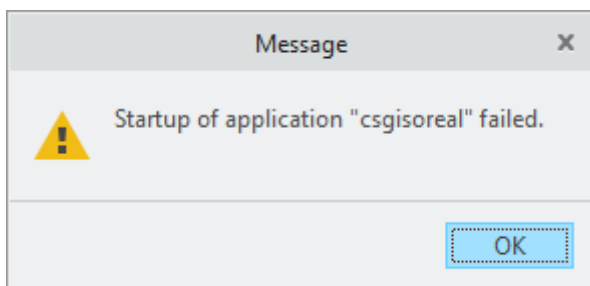
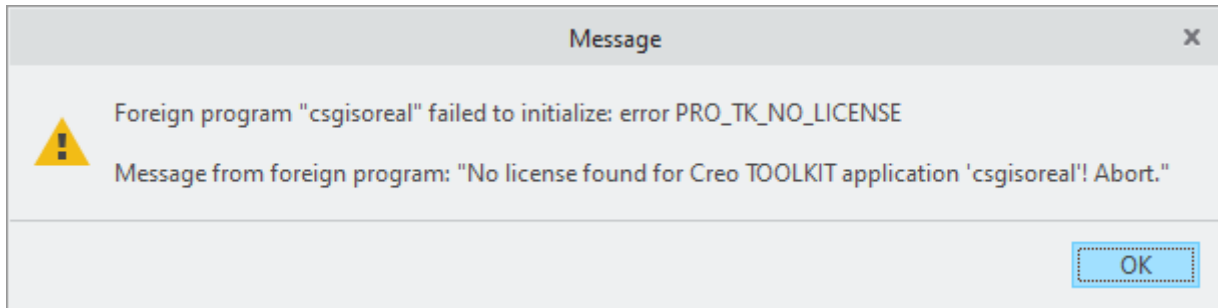
The installation of this application modifies the Creo Piping toolbar by adding a M4ISO button.

#### M4ISO Start Button on the Creo Piping Ribbon



## 4 M4ISOCreo Installation problems

### 4.1 M4ISOCreo will not start when Creo launches



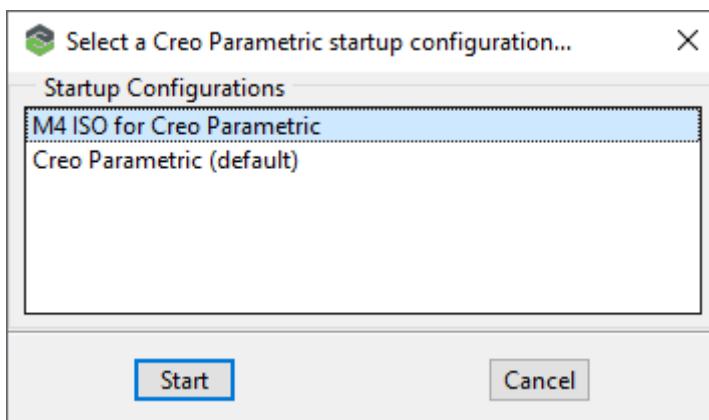
These error messages indicate a problem accessing the license keyfile for M4ISO.

There may be a couple of reasons for this:

#### 4.1.1 Creo is not referencing the M4ISOCreo application correctly.

If you start Creo using the standard method of executing the PARAMETRIC.EXE program using the shortcuts created during installation, then you should be presented with a dialog with a choice of Startup Configurations.

These will normally comprise of "Creo Parametric (default)" or "M4 ISO for Creo Parametric". You must start the "M4 ISO for Creo Parametric" option.



If you have a customised startup process that references a batch script rather than the PARAMETRIC.EXE program, then the M4 ISO startup process will not be executed and the error message will appear. You will need to modify your custom startup

script to correctly reference the M4IsoCreo.psf start-up file.

The reason that the message is appearing on a standard Creo startup is that the Toolkit application is visible to Creo, but the license required to run the application is not being referenced from the Creo startup environment. See section 4.2 below for a solution to this problem.

#### 4.1.2 The license is not installed correctly.

Assuming you have installed the M4ISO software using the default installation path (C:\M4ISOCreo\_V4\_2) and you have referenced the license that was sent to you during installation, then the license will be found under the full pathname:

C:\M4ISOCreo\_V4\_2\medsys\med\license\CSGLMD.LIC

If the file exists, check that the contents of the file contain the following FEATURE lines:

```
FEATURE cadconvert
FEATURE med_iso
FEATURE med_iso4creo
```

If the file does not exist or does not contain the above feature lines, then the license has not been installed correctly. Copy the supplied license file into the above path and restart the Creo application.

#### 4.1.3 Wrong HOSTID

The HOSTID used in the license file must be a match to the network device MAC address of the Creo workstation.

On the Creo workstation, open a Command Prompt window (CMD window) and type the following command:

```
GETMAC /V
```

This will return a table that looks like this:

Connection Name	Network Adapter	Physical Address	Transport Name
Ethernet	Sophos SSL VPN	03-FA-8C-E8-74-2E	Media disconnected
Wireless Networ	Intel(R) Centri	8C-9A-82-58-9D-6B	Media disconnected
Local Area Conn	Realtek PCIe GB	44-EF-B5-5B-9C-44	\Device\NPF{9498AAB...}

From this table we normally use the Physical Address of the Local Area Connection network device (highlighted in the table above). Check that one of the Physical Address on your Creo workstation matches the HOSTID=xxxx reference we have used in your license (the value used in the license file has the hyphen characters re-

moved).

#### 4.1.4 Keyfile Expired

The license keyfile has an expiry date on the FEATURE line, for example:

```
FEATURE med_iso4creo CSGLMD 10.720 31-jul-2024 uncounted \
```

If the keyfile has expired, M4ISOCreo will fail to initialise. In this situation request a new keyfile.

#### 4.1.5 Keyfile version

The license keyfile must be CSGLMD 10.720 on the FEATURE line, for example:

```
FEATURE med_iso4creo CSGLMD 10.720 31-jul-2024 uncounted \
```

If the keyfile is the wrong version, M4ISOCreo may fail to initialise. In this situation request a new keyfile.

## 4.2 M4ISO button does not appear on Piping Ribbon

If Creo starts without error but the M4ISO button does not appear on the Piping ribbon, this indicates that the *PROTKDAT* setting for the M4ISOCreo application has not been initialised.

The installation process modifies the **config.pro** file in the folder:  
**<Creo\_installation>\Common Files\text**

This process adds the *PROTKDAT* line to initialise the M4IsoCreo toolkit application. For example:

```
protkdat C:\M4ISOcreo_V4_2\isoreal\cfg\protk.dat
```

But some companies automatically overwrite this **config.pro** file with a company standard version.

If the **config.pro** file used on startup does not include the *PROTKDAT* setting for M4ISOCreo, then the M4ISO button will not appear in the ribbon.

Also, be aware that adding this *PROTKDAT* setting into the base **config.pro** means that Creo Parametric users who are not using the M4IsoCreo application will see the CSGISOREAL initialisation failure error messages on startup as described in section 4.1 above. This can be fixed by creating a **config.pro** file that contains the *PROTKDAT* setting in another directory and removing the setting in the base **config.pro**.

During Creo startup, the application searches for **config.pro** files in this order:

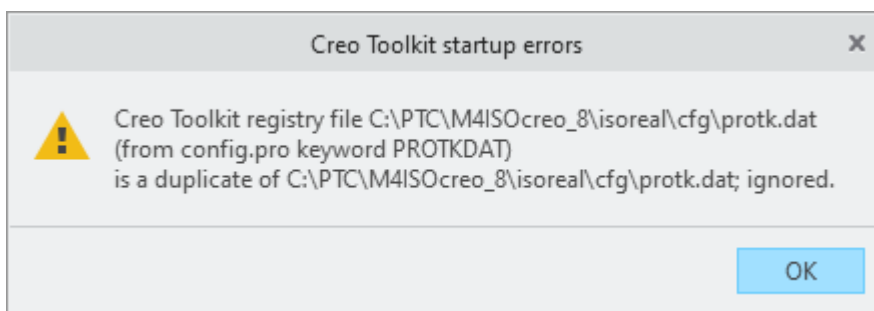
1. The Creo base folder **<Creo\_installation>\Common Files\text**. Commands

- in the **config.pro** file found in this folder will be applied to all Creo users who use that installation of Creo.
2. The Creo startup folder as defined in the Windows shortcut used to launch Creo. Check the shortcut properties to identify this folder. Commands in the **config.pro** file found in this folder will be applied to all users who use that folder as their startup folder – this may be one or more Creo users.
  3. The home directory of the currently logged in user as defined by the Windows environment variables HOMEDRIVE and HOMEPATH. Commands in the **config.pro** file found in this folder will be applied only to that user.

Select one of these folders and create a new file named **config.pro**. If a file of that name already exists, then modify that existing file.

Edit the new **config.pro** file to add the *PROTKDAT* command reference as shown above, then save the file

If more than one **config.pro** file is modified with the *PROTKDAT* command, then on Creo startup the following message (or something similar) may be displayed:



To fix this, remove any surplus *PROTKDAT* definitions from all but one of the **config.pro** files described above. The definition must exist in one of these files but is not normally required in all of them.



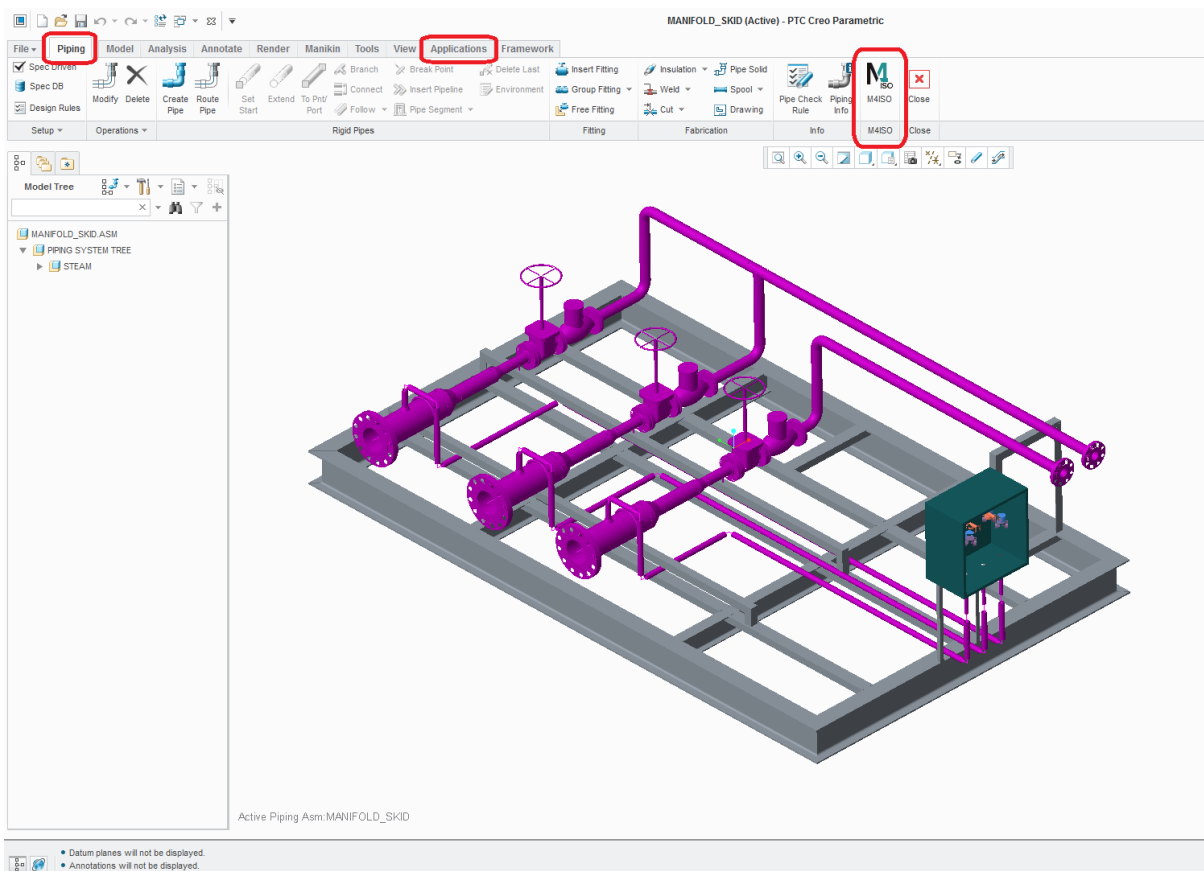
## 5 Creating an Isometric

This section explains how to proceed to create an isometric drawing from 3D pipeline geometry.

After starting PTC Creo Parametric with M4ISOCreo proceed as follows:

Open an existing 3D model of pipeline geometry or create a new one.

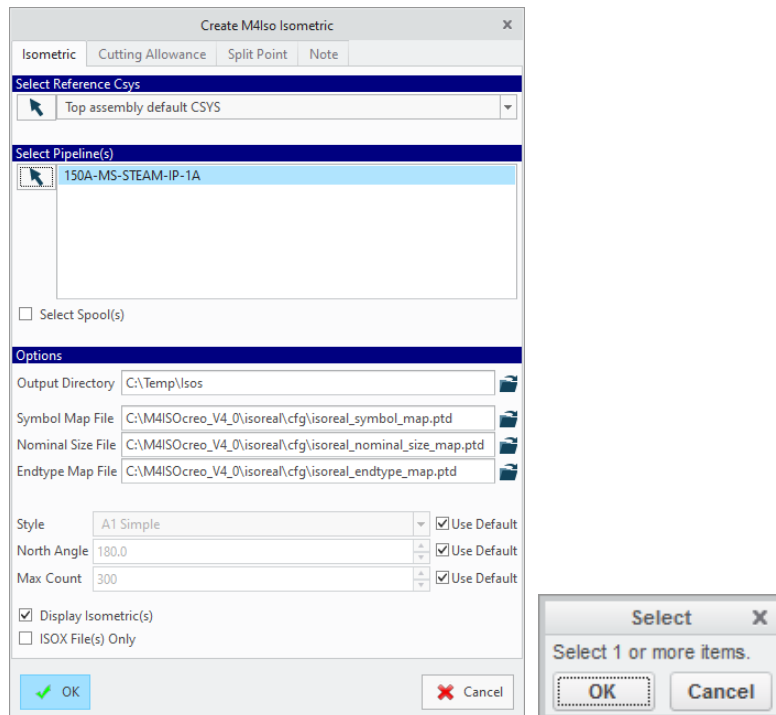
Click on Application > Piping on the ribbon to load the Piping module.  
The Piping ribbon contains the M4 ISO button.



## 5.1 Open the M4Iso dialog

Click on the M4 ISO button.

The *Create M4Iso Isometric* dialog is displayed and a *Select* window appears below the dialog. This dialog always presents with the *Isometric* tab active and automatically goes into pipeline select mode.



## 5.2 Pipeline Selection

Selecting a pipeline to be processed is done by a left click with the mouse on the pipe in the graphics area. The pipeline routing graphics must be visible, or they cannot be selected.

If more than one pipeline is to be selected, hold the **CTRL** key down while clicking the **LMB**. Selected pipelines are shown in the graphics area with a highlighted centreline along the pipeline route.

It is also possible to select pipelines from the *Model Tree* on the left of the screen when the *Pipeline View* option is selected. Selecting a parent node will select all child nodes under the selection.

To apply a selection, click the **OK** button in the *Select* window or click the **MMB**.

When pipelines are selected, they are listed in the *Create M4Iso Isometric* dialog and automatically highlighted. A **LMB** click on the pipeline name deselects/selects that pipe-

line. The **OK** button is available if at least one pipeline is highlighted in the dialog. The **OK** button is disabled if no pipeline is selected.

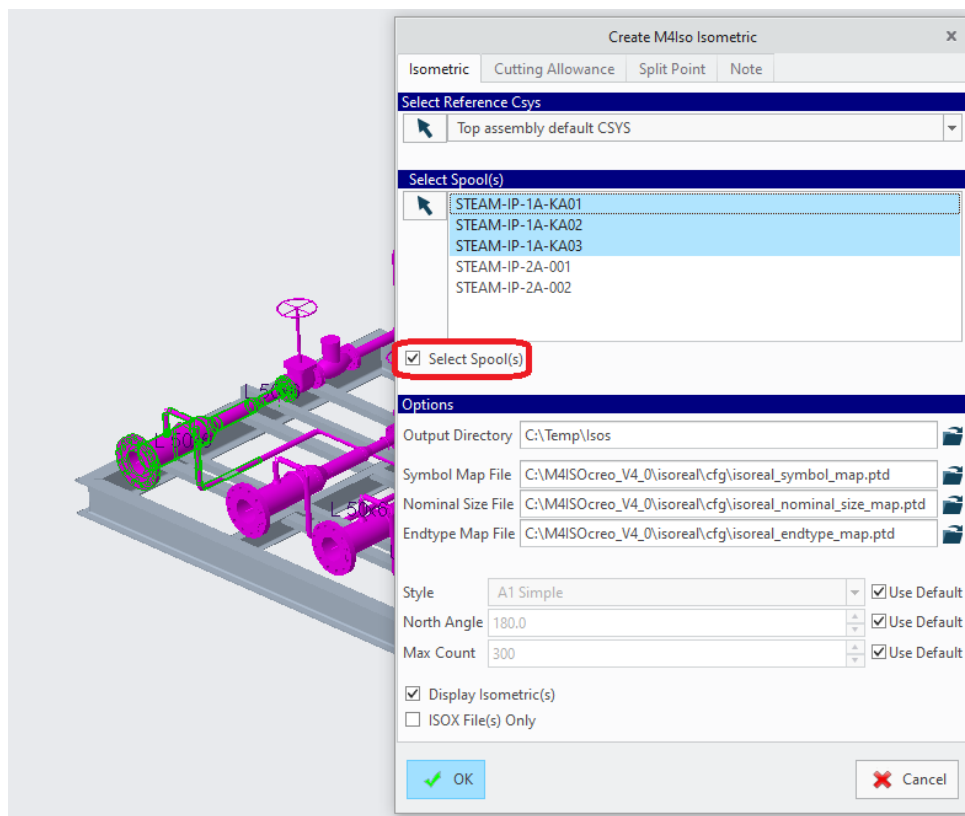
### 5.3 Selecting Spools

The *Create M4 ISO Isometric* dialog contains the *Use Spools* option which creates one separate isometric sheet from each pipe spool. The spools must be defined in the piping assembly using the Creo “Create Spool” command.

Activating the *Select Spools* checkbox lists any known Creo spool features of the selected pipelines.

The spools are automatically displayed as unselected (not highlighted) and the **OK** button is disabled.

#### Select Spool Option in the Create M4 ISO Isometric Dialog



As you select the spools to be processed, the **OK** button becomes active. As the mouse is moved over a spool name in the list, the spool is highlighted in the graphics area.

Please note: If the *Select Spools* checkbox is active then the `M4ISOCREO_DRW_NAME` environment variable setting is ignored.

By default, spool isometrics use the spool name as the drawing filename but you can control the name by setting the `M4ISOCREO_SPOOLDRW_NAME` environment variable (see Controlling the File name of the Isometric Drawing).

If displaying the model 3D View when the *Select Spool(s)* option is selected, then the Creo configuration option “*Display Thick Pipes*” must be set to “*NO*” to only draw the selected Spool. If set to “*YES*” then the whole pipeline is viewed. This is normal Creo behaviour.

## 5.4 Reference CSYS

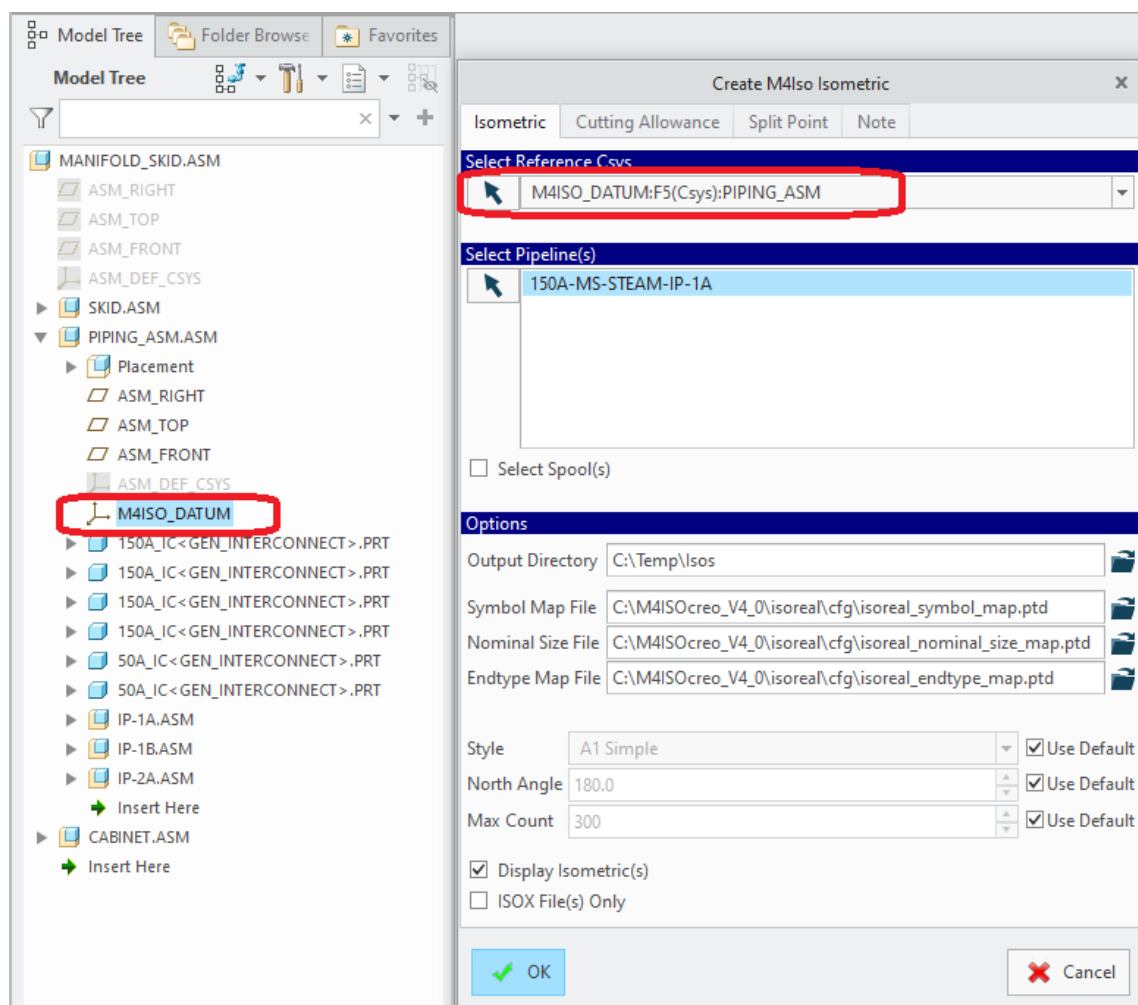
If desired, select a Reference Csys in the model to define the orientation and relative position of the pipelines.

By default, the first CSYS in the assembly is used as the Reference Csys.

By clicking the icon, the user can select a different Csys to use as the reference point. The Y-axis of the selected Csys is aligned to the North Angle direction, while the Z-axis points up on the isometric drawing. All terminal positions reported in the isometric are relative to the selected Csys.

In the example below the user has selected the M4ISO\_DATUM Csys.

Any Csys in the model can be used as a reference point, including the PORTn Csys objects in the piping fittings.



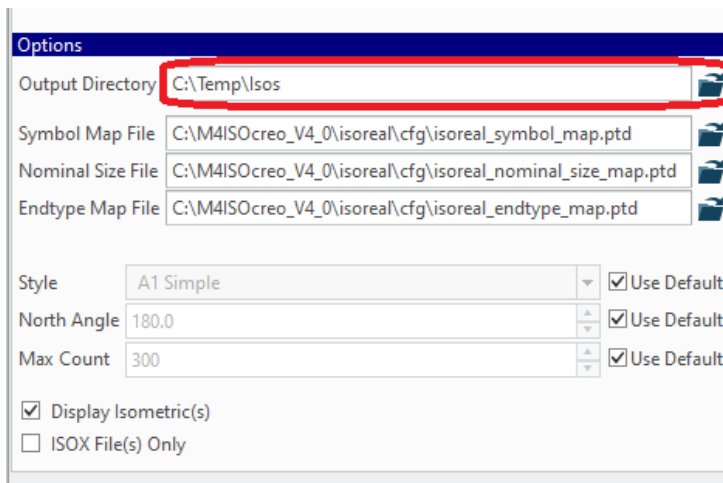
An environment variable can be set to provide a preferred list of CSYS objects to use.

**M4ISOCREO\_CSYS\_DEFAULT= M4ISO,ACS0,ASM\_DEF\_CSYS**

The first Csys in the list that is found in the model assembly is used by the dialog. This can still be changed by the user if desired.

## 5.5 Define the Output Directory

By default, the *Output Directory* is the current working directory, but you can select any other directory from a file browser which is opened using the icon on the right-hand side.



All files created by the M4ISO process will be written to the nominated folder. The user must have read and write access to this folder.

The Output Directory can be preset using an environment variable:

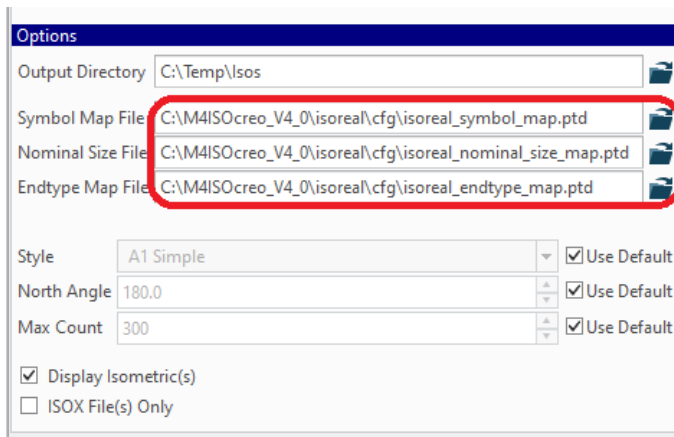
**M4ISOCREO\_OUTDIR=C:\Creo\Isos**

Another environment variable can be set to disable the output directory field so the user cannot change the set value.

**M4ISOCREO\_OUTDIRLOCK=YES**

## 5.6 Mapping Files

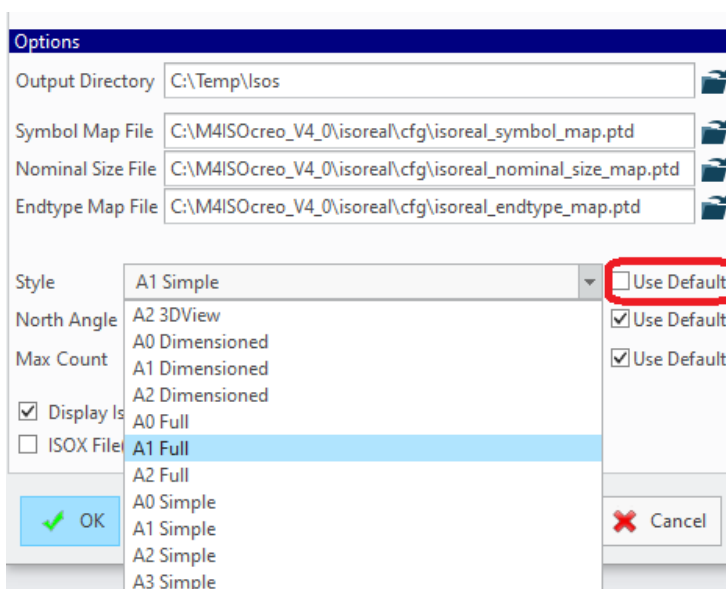
Choose the desired mapping files (for details on the .ptd files see section Creo-M4ISO Mapping Files.).



It is normal to use the supplied defaults, but these can be changed if required by selecting a different file using the browse icons on the right of each entry.

## 5.7 Drawing Style

Select the isometric drawing Style to be used for the isometric; uncheck the *Use Default* checkbox to enable other styles.



The available styles control the size of the drawing, the number of tables created on the isometric drawing, the dimensioning styles and if a 3D view of the pipeline is created.

The default Style presented when the dialog is first opened is controlled by a setting in the M4IsoOptionStore.xml file:

```
<M4IsoCurrentSection name="A1 Simple"/>
```

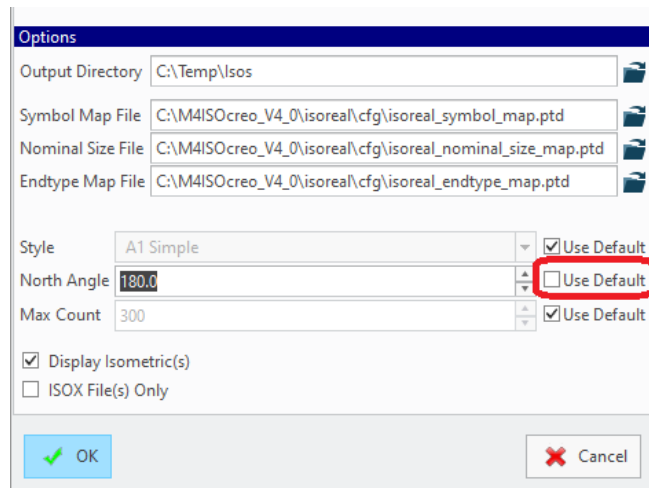
This can be changed to any other known style if required.

## 5.8 North Angle



Set the desired *North Angle* to control the orientation of the isometric drawing; uncheck the *Use Default* checkbox to enable the input box.

The orientation is with respect to the selected Reference Csys. The North direction is the positive Y-axis of the reference Csys.



Multiples of 90 degrees are recommended, but any value in the range 0-359 degrees can be set.

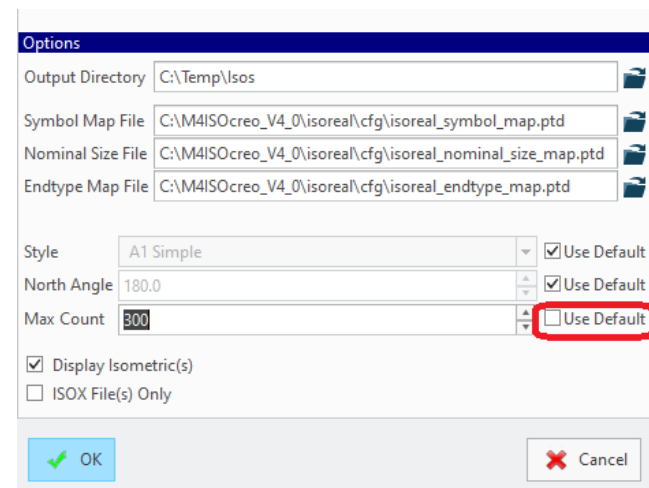
An angle value of 0 sets the positive Y-axis to point to the bottom right corner of the drawing. Other angles are orientated anti-clockwise from this position

Adjustment of the North Angle may be required to get the best isometric view.

## 5.9 Max Count

Set the *Max Count* value to control how much data is drawn on the isometric diagram; uncheck the *Use Default* checkbox to enable the input box.

This setting controls how many items in the isometric are drawn on the diagram before a new page is created. Setting a lower value, such as 50, may make the isometric split into several pages with simpler representations of parts of the pipeline on each page.



A note is automatically added to the split point to indicate the page where the pipe continues.

This setting can be over-ridden by the Split Page settings.

## 5.10 Display Isometrics

Make sure that the *Display Isometric(s)* checkbox is set.

This checkbox is used to display the isometric drawing after processing. By default,

the checkbox is set. The drawing will not be displayed if you uncheck the option. When processing a large number of isometrics, unchecking this option can speed up the processing.

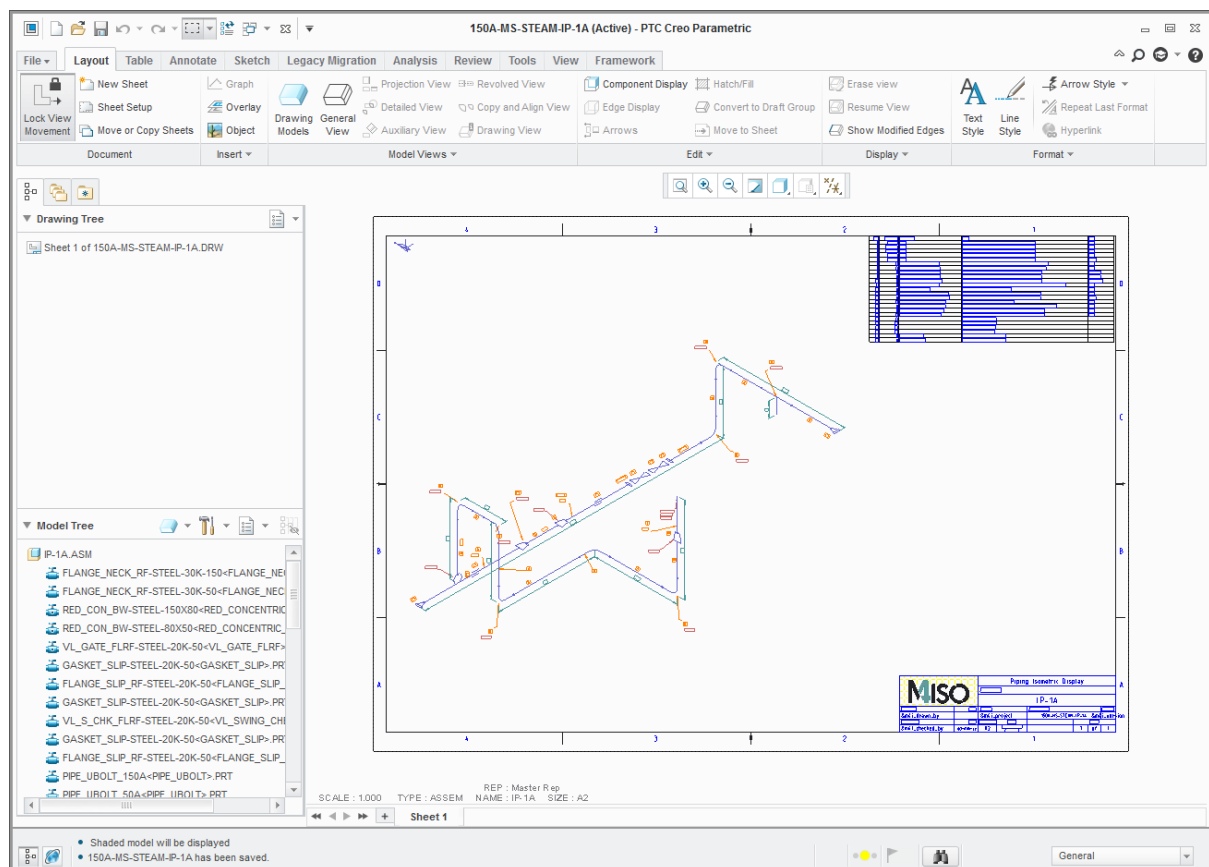
### 5.11 ISOX File(s) Only

This checkbox is used to run only the first part of the Isometric creation process and create the intermediate ISOX file for each pipe (or spool) selected. When this checkbox is enabled, the isometric drawing will not be created.

### 5.12 Generate the Isometric Drawing

Click on the **OK** button to start the creation of the isometric drawing.

The isometric drawing is created and (by default) automatically loaded in PTC Creo Parametric.



The isometric files created are automatically saved into the working directory (or the *Output Directory* if one was set) using the pipeline name as the filename.

It is strongly advised that all pipelines in the Piping assembly model have unique names. If several pipelines use the same pipe name, then multiple versions of the isometric drawing will be created with a different pipeline isometric drawn on each version.

**Please note:** Creo has a limit of how many DRW files it can open simultaneously. When this limit is reached the earliest drawings created will be closed as each new drawing is created.

When processing a large number of isometric drawings from an assembly, unchecking the *Display isometric(s)* checkbox allows M4 ISO to process all the isometric drawings without displaying them in Creo. The Isometric drawings are saved in the nominated *Output Directory* and can be opened manually after processing is complete.

### 5.13 Controlling the Isometric Drawing Filename

By default, the isometric drawing is saved under the name of the pipeline. You can set the environment variable `M4ISOCREO_DRW_NAME` to control the name of the file.

If the *Select Spools* checkbox is active the isometric drawing is saved under the name of the spool. You can set the environment variable `M4ISOCREO_SPOOLDRW_NAME` to control the name of the file (see also section 7 “Creo Environment Variables”).

Both variables may be composed of fixed characters and variable parts. The placeholder names for the variable parts are `<PIPE_NAME>`, `<ASSY_NAME>` and `<SPOOL_NAME>`, they must be enclosed in angle brackets as shown.

For example:

```
set M4ISOCREO_DRW_NAME="ISO-<ASSY_NAME>-<PIPE_NAME>"
```

Placeholder `<SPOOL_NAME>` is not valid for use with `M4ISOCREO_DRW_NAME`.

As an illustration, in the following examples

<code>&lt;ASSY_NAME&gt;</code>	is set to ABC
<code>&lt;PIPE_NAME&gt;</code>	is set to XYZ
<code>&lt;SPOOL_NAME&gt;</code>	is set to 123

Examples:

The environment variable:

```
set M4ISOCREO_DRW_NAME="ISO-<ASSY_NAME>-<PIPE_NAME>"
```

results in the drawing name: ISO-ABC-XYZ

The environment variable:

```
set M4ISOCREO_SPOOLDRW_NAME="ISO-<ASSY_NAME>-<SPOOL_NAME>"
```

results in the drawing name: ISO-ABC-123

If the definition is faulty, M4Iso Creo throws out a warning message and then uses the appropriate default naming convention.

The Creo name size limit is 31 characters. M4IsoCreo will truncate any filename value longer than this when this limit is exceeded. As a result, different pipe isometrics may end up with the same name and may overwrite each other.

## 5.14 M4 ISO Cutting Allowance

At the top of the *Create M4ISO isometric* dialog, there is a tab named *Cutting Allowance*.

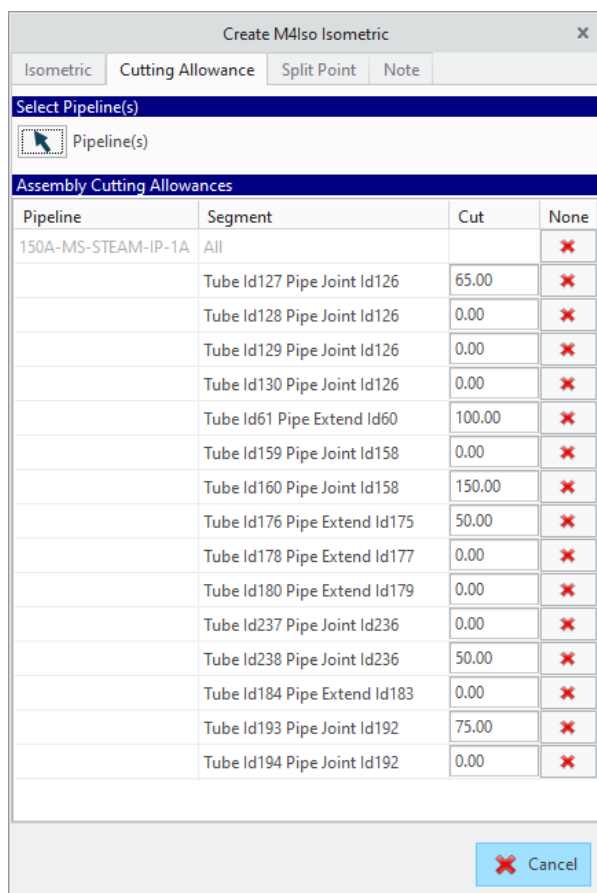
Activating this tab presents the dialog shown below.

A pipeline must be selected when the *Create M4ISO isometric* dialog is opened. Activating the *Cutting Allowance* tab will present a list of all the selected pipeline segments and the current Cutting Allowances assigned to each segment.

Selecting a segment field from the list will highlight the segment in the graphics area. The segment will remain highlighted until it is deselected in the list.

The user can then click in the cut value field to change the value, or press the **X** field to delete the Cutting allowance value (reset to zero). A left-click in the **X** field in the 'All' reference at the top of the pipeline list will reset all the Cutting Allowance values to zero.

### Cutting Allowance dialog



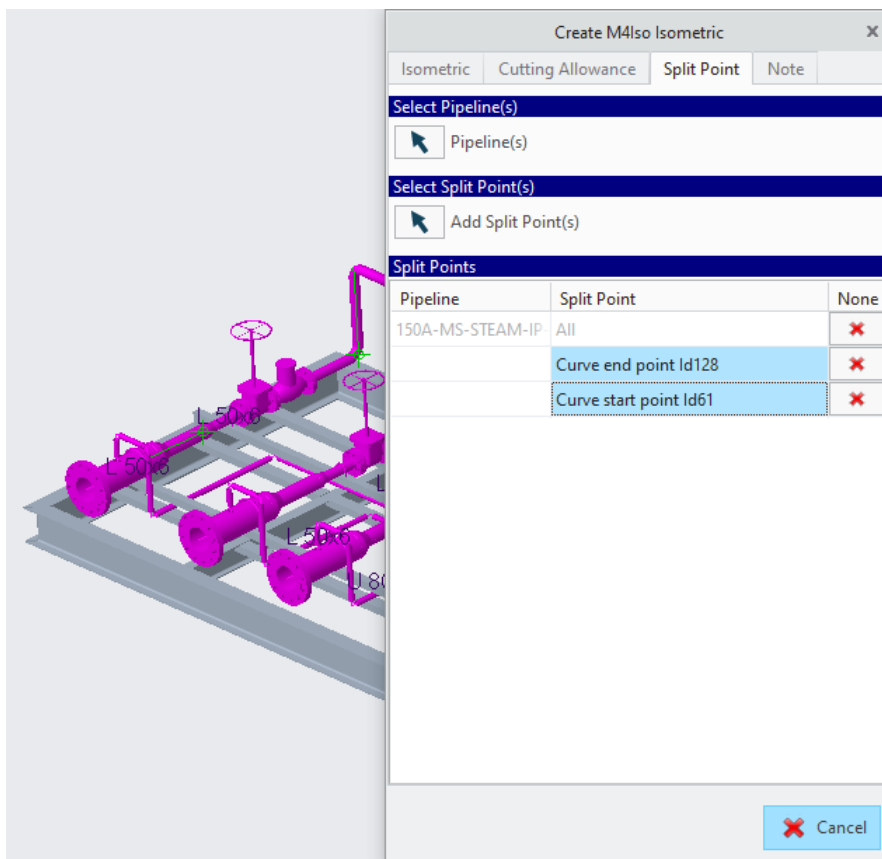
Pipeline	Segment	Cut	None
150A-MS-STEAM-IP-1A	All		X
	Tube Id127 Pipe Joint Id126	65.00	X
	Tube Id128 Pipe Joint Id126	0.00	X
	Tube Id129 Pipe Joint Id126	0.00	X
	Tube Id130 Pipe Joint Id126	0.00	X
	Tube Id61 Pipe Extend Id60	100.00	X
	Tube Id159 Pipe Joint Id158	0.00	X
	Tube Id160 Pipe Joint Id158	150.00	X
	Tube Id176 Pipe Extend Id175	50.00	X
	Tube Id178 Pipe Extend Id177	0.00	X
	Tube Id180 Pipe Extend Id179	0.00	X
	Tube Id237 Pipe Joint Id236	0.00	X
	Tube Id238 Pipe Joint Id236	50.00	X
	Tube Id184 Pipe Extend Id183	0.00	X
	Tube Id193 Pipe Joint Id192	75.00	X
	Tube Id194 Pipe Joint Id192	0.00	X

The assigned (non-zero) Cutting Allowances will be applied in the pipe lengths shown in the *Item Details*, *Cuts* and *Bends* tables on the Isometric diagram (if these are displayed) and can be shown on the Run dimensions.

## 5.15 Split Point Tab

The M4ISOCreo application has a basic mechanism for splitting the isometric into several pages by setting the *Max Count* field to a low value. However, the split point is selected arbitrarily based on the *Max Count* value.

The *Split Point* tab allows the user to nominate a position on the isometric where a new isometric drawing page/sheet should be created. This is independent of the *Max Count* value. If a split point is assigned, then it will be used, regardless of if the *Max Count* value has been reached or not.



More than one split point attribute can be added into a Pipeline. They can be added to either:

- CSYS objects in the pipeline (i.e. the PORT connections on fittings)
- At the end of pipe segments in the pipeline definition.

Split points are added using the *Add Split Points* button and then moving the mouse over the CSYS objects and the pipeline segments. These must be visible in the assembly, or they will not highlight.

A split point can be deleted by a left-click on the ✗ field beside the reference. A left-click in the ✗ field in the *All* reference at the top of the pipeline list will erase all the assigned split points.

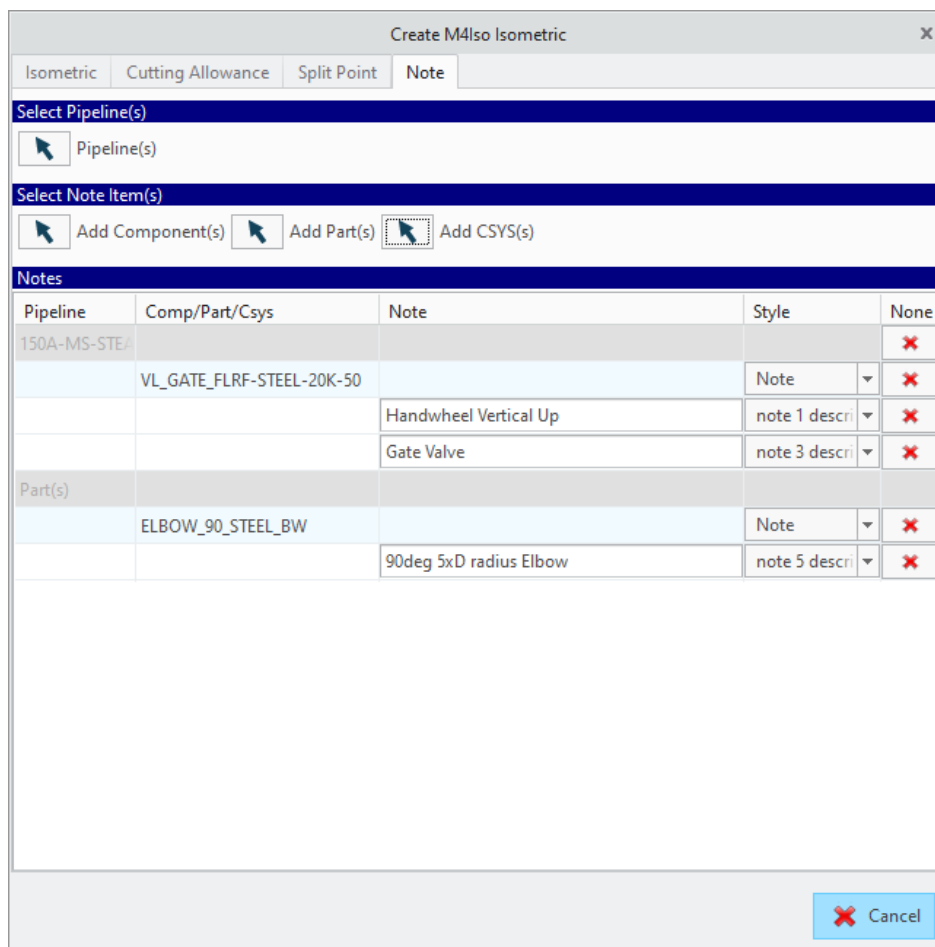
## 5.16 Notes

### 5.16.1 Adding Notes to Components/Parts in Creo

The user can add notes into

- a part
- a port (CSYS) of a part
- a component (instance of a part in an assembly)

This is achieved using the Note Tab in the M4ISO dialog.



Pipeline	Comp/Part/Csys	Note	Style	None
150A-MS-STEEL				✗
	VL_GATE_FLRF-STEEL-20K-50		Note	✗
		Handwheel Vertical Up	note 1 descri	✗
		Gate Valve	note 3 descri	✗
Part(s)				
	ELBOW_90_STEEL_BW		Note	✗
		90deg 5xD radius Elbow	note 5 descri	✗

A note assigned to a Part or a Port will be displayed on each and every component (instance) of that part in the pipeline isometric.

A note assigned to a Component will be displayed only on the Component it was assigned to.

The Note dialog allows the user to select from 10 note styles. More than one note style can be assigned to a Part/CSYS/Component at once.

After selecting the Pipeline(s) that the notes are to be added to, the user then uses the Select Note Item(s) buttons to select the appropriate Component/Part/CSYS. The

selected item then appears in the Notes section. Use the pulldown menu in the Style column to select a note style for that component and a Note input field will appear below the selected item. Enter the Note value to be displayed on that item.

The value assigned for the note can be a literal string of characters, for example:

### **Handwheel Vertical Up**

The value can also reference one or more Creo parameters that have been assigned to the selected items using the ampersand (&) character. For instance, if a note is required regarding the angle of a handwheel spindle on a valve and the valve contains a parameter of SPINDLE\_ANGLE which has a value of 25. Then the note assigned to the valve component can be defined as:

**Valve spindle at &SPINDLE\_ANGLE degrees.**

This would give a result of:

**Valve spindle at 25 degrees.**

Use the red cross icon at the right to delete:

- an individual note,
- all the notes on one item or
- all the notes in the Note dialog.

## 6 ISOREAL Mapping Files

The installation directory *M4ISO > isoreal > cfg* contains three Pro/TABLE files (suffix *.ptd*)

*isoreal\_endtype\_map.ptd*  
*isoreal\_nominal\_size\_map.ptd*  
*isoreal\_symbol\_map.ptd*

These files define the mapping between Pro Creo specifications of pipelines, fittings and endtypes and the labeling of the pipelines, the symbolic representation of fittings in the M4 ISO isometric drawing and the naming of the endtypes in M4 ISO.

A further mapping file can be created if required:

*isoreal\_parameter\_map.ptd*

This file defines the mapping between Creo parameter names and M4 ISO attribute names.

The files can be edited by using the common Pro/TABLE editor.

With Creo V4.0 and above, these files can be created as CSV files (suffix *.csv*) if preferred.

Note that modified versions of the *endtype*, *nominal size* and *symbol* mapping files can be created anywhere on the accessible disk structure as it is possible to select these files by using the browse buttons on the dialog.

But the *parameter* mapping file (if used) must be created in the installation folder.

By default, this would be:

*C:\M4ISO\creo\_V4\_2\isoreal\cfg\isoreal\_parameter\_map.ptd*

### 6.1 isoreal\_endtype\_map.ptd

The *isoreal\_endtype\_map.ptd* file defines the mapping between Creo and M4ISO endtypes. It contains two columns.

**Table 1 Example of the isoreal\_endtype\_map.ptd File**

ENDTYPE	M4ISO_ENDTYPE
BE	bevel_ended
PE	plain
...	...



The first column shows the Creo Piping specification of the endtypes (e.g. BE, PE, SW).

The second column shows the definition of the relevant type in M4 ISO.

There are also two special entries to define a default\_pipe endtype and a forced\_pipe endtype to be used in cases where an endtype mapping cannot be made by the software.

DEFAULT\_PIPE plain  
FORCE\_PIPE plain

The mappings in this file, including the default\_pipe and force\_pipe mapping, can be modified or expanded according to your requirements.

### 6.1.1 Valid Endtypes

The following table lists, in alphabetical order, the valid M4ISO endtypes supplied with the *M4 ISO Isometrics for PTC Creo Piping* application.

**Table 2 Valid Default M4ISO Endtypes**

<b>Valid Endtypes</b>
bevel_ended
butt_weld
compression
flanged
flat_face_flanged
flat_face_gasket
none
plain
raised_face_flanged
raised_face_gasket
ring_joint_gasket
ring_jointed
screwed
screwed_female
screwed_male
slip_on
socket_weld
tongue_and_groove_gasket
tongue_and_grooved

**Please note:** It is possible for administrators to expand this list by modifying the *m4isosymbolinfo.xml* file.

## 6.2 isoreal\_nominal\_size\_map.ptd

The *isoreal\_nominal\_size\_map.ptd* file defines the mapping for pipelines. It contains five

columns.

**Table 3 Example of the isoreal\_nominal\_size\_map.ptd File**

SIZE	M4ISO_NOMINAL_SIZE_MM	M4ISO_DESIGNATION_MM	M4ISO_NOMINAL_SIZE_INCH	M4ISO_DESIGNATION_INCH
6A	6	DN	1/8	NB
8A	8	DN	1/4	NB
10A	10	DN	3/8	NB
...	...	..	...	...
32A	32	DN	"1 1/4"	NB
40A	40	DN	"1 1/2"	NB
...	...	...	...	...

The first column (`SIZE`) is the size specification of the pipes as defined in Creo Piping (e.g. 10A).

The second and third columns define the Metric nominal size and designation used within M4 ISO and displayed on a Metric unit Isometric drawing (e.g. 10 DN).

The 4th and 5th columns define the Inch nominal size and designation used within M4 ISO and displayed on an Inch unit Isometric drawing (e.g. 3/8 NB).

**Please note:** Fractional nominal sizes that include a space character must be enclosed in quote (") marks, e.g. "1 1/2".

The mappings in this file may have to be modified or expanded according to your requirements.

### 6.2.1 Default Sizes

The following tables list, in size order, the default M4ISO\_NOMINAL\_SIZE and M4ISO\_DESIGNATION definitions supplied with the M4 ISO Isometrics for PTC Creo Piping application.

**Table 4 Default Sizes**

M4ISO_NOMINAL_SIZE_MM	M4ISO_DESIGNATION_MM
6	DN
8	DN
10	DN
15	DN
20	DN
25	DN
32	DN
40	DN
50	DN
65	DN
80	DN
90	DN
100	DN
125	DN
150	DN
200	DN
250	DN
300	DN
350	DN
400	DN
450	DN
500	DN
550	DN

M4ISO_NOMINAL_SIZE_INCH	M4ISO_DESIGNATION_INCH
1/8	NB
1/4	NB
3/8	NB
1/2	NB
3/4	NB
1	NB
"1 1/4"	NB
"1 1/2"	NB
2	NB
"2 1/2"	NB
3	NB
"3 1/2"	NB
4	NB
5	NB
6	NB
8	NB
10	NB
12	NB
14	NB
16	NB
18	NB
20	NB
22	NB

600	DN
650	DN
700	DN
750	DN
800	DN
850	DN
900	DN
1000	DN
1050	DN
1100	DN
1150	DN
1200	DN
1250	DN
1300	DN
1350	DN
1400	DN
1450	DN
1500	DN
1600	DN
1700	DN
1800	DN
1900	DN
2000	DN
2100	DN
2200	DN
2300	DN
2400	DN
2500	DN
2600	DN

24	NB
26	NB
28	NB
30	NB
32	NB
34	NB
36	NB
40	NB
42	NB
44	NB
46	NB
48	NB
50	NB
52	NB
54	NB
56	NB
58	NB
60	NB
64	NB
68	NB
72	NB
76	NB
80	NB
84	NB
88	NB
92	NB
96	NB
100	NB
104	NB

### 6.3 isoreal\_symbol\_map.ptd

The *isoreal\_symbol\_map.ptd* file defines the mapping for fittings. It contains three columns.

**Table 5 Example of the isoreal\_symbol\_map.ptd File**

"FITTING_GENERIC_NAME"	"M4ISO_CLASS"	"M4ISO_SYMBOL"
"coupling_sw"	"full_coupling"	"full_coupling"
"elbow_45_copper_bw"	"elbow"	"elbow"
"flange_slip_rf"	"flange"	"flange"
"gasket_slip"	"gasket"	"gasket"
"nipple_bw"	"nipple"	"nipple"
"pipe_clip"	"support"	"pipe_hanger"
"vl_gate_flff"	"valve"	"valve_gate"
...	...	...

The first column (*FITTING\_GENERIC\_NAME*) contains the generic name of the fitting in Pro Creo. For example: `elbow_45_copper_bw_trim`

The second column (*M4ISO\_CLASS*) defines the classification of the fitting in M4 ISO, for example, *elbow*, *valve* or *flange*. The classification can be broken down into generic or specific symbol references as indicated in the third column (*M4ISO\_SYMBOL*). The classification and symbol references are used to look up the associated M4ISO symbol to be drawn on the isometric drawing.

The mappings in this file may have to be modified or expanded according to your requirements.

(For details on customizing the representation of the isometric symbols, refer to the *Creating Custom Component Symbols* section.)

**Please note:** An automatic mapping to a generic symbol (based on the number of connection points) is executed for all fittings for which a mapping cannot be found.

### 6.3.1 Symbol mapping parameters

It is possible to map Creo parts to M4 ISO isometric symbols using Creo Parameters in the part model.

If these parameters are defined, they will override the values defined in the *isoreal\_symbol\_map.ptd* file.

For example, in a Creo model of a wafer valve: **vl\_wafer\_flff**, define the parameters:

**Name:** **M4ISO\_SYMBOL**,    **type:** **String**,        **value:** **valve\_wafer**

**Name:** **M4ISO\_CLASS**,    **type:** **String**,        **value:** **valve**

These settings will draw the **vl\_wafer\_flff** fitting as a **valve\_wafer** isometric symbol as defined in the *M4IsoSymbolInfo.xml* file.

The dimensioning of this isometric symbol will be controlled using the “*component/valve/dim\_to...*” settings in the *M4IsoOptionStore.xml* file.

### 6.3.2 Default Symbols and Classes

The following table lists, in alphabetical order, a selection of the most common M4ISO\_Class and M4ISO\_Symbol definitions supplied with the *M4 ISO Isometrics for PTC Creo Piping* application.

**Table 6 A selection of M4 ISO Class and Symbol Definitions**

<b>M4ISO_Class</b>	<b>M4ISO_Symbol</b>
<b>cap</b>	cap
<b>coupling</b>	coupling
<b>cross</b>	cross
<b>elbow</b>	elbow
<b>elbow_teel</b>	elbow_teel

flange	flange
flange	flange_wn
flange_blind	flange_blind
full_coupling	full_coupling
half_coupling	half_coupling
miscellaneous_2_angle	miscellaneous_2_angle
miscellaneous_branch	miscellaneous_branch
miscellaneous_cross	miscellaneous_cross
miscellaneous_inline	miscellaneous_inline
miscellaneous_y	miscellaneous_y
olet	half_coupling_as_olet
olet	latrolet
olet	nipolet
olet	olet
olet	sockolet
olet	sweepolet
olet	threadolet
olet	weldolet
reducer_concentric	reducer_concentric
reducer_eccentric	reducer_eccentric
setin	setin
support	pipe_hanger
tee	tee
tee_set_on	tee_set_on
terminal_equipment	terminal_equipment
union	union
valve	valve_ball
valve	valve_butterfly
valve	valve_check
valve	valve_diaphragm
valve	valve_gate
valve	valve_generic
valve	valve_globe
valve	valve_needle
valve	valve_plug
valve_3way	valve_3way
valve_angle	valve_angle

**Please note:** For a full list of the supplied symbols, see Appendix A. It is possible for administrators to expand this list by modifying the *m4isosymbolinfo.xml* file.

## 6.4 isoreal\_parameter\_map.ptd

The *isoreal\_parameter\_map.ptd* file defines the mapping between Creo parameter names and M4ISO attribute names. It contains four columns.

Table 7 Example of the isoreal\_parameter\_map.ptd File

"PARAMETER"	"SOURCE"	"M4ISO_ATTRIBUTE"	"TARGET"
"fitting_code"	"Assembly"	"fitting"	"ItemDetails"
"fitting_code"	"Part"	"fitting"	"ItemDetails"
"sap_number"	"Part"	"fulltype"	"ItemDetails"

"stockno"	"Part"	"stockno"	"ItemDetails"
"GRADE"	"Part"	"my_param1"	"ItemDetails"
...	...	...	...

The first column (*PARAMETER*) is the Creo parameter name to be referenced.

The second column (*SOURCE*) is the Creo entity that the parameter name is to be found in.

It can be one of:

- Assembly
- Part
- Component
- Linestock
- Spool

The third column (*M4ISO\_ATTRIBUTE*) is the attribute name to which the value of the Creo parameter will be assigned. This attribute name can be an existing M4 ISO attribute name, or it can be a new name (for example: my\_param1).

In the M4IsoOptionStore file, the defined attribute names (column 3) can be referenced directly, using the *%attribute name%* notation, for use in the tables or other OptionStore settings.

The fourth column (*TARGET*) is where the attribute will be used in the M4 ISO application.

It can be one of:

- ItemDetails
- PipeComponent
- PipeInformation
- Weld

The ItemDetails target value identifies attributes that can be used in the Item Details (parts list) table.

The PipeComponent target identifies attributes that are available for use in the OptionStore as markers. For example: to identify fittings that should be excluded from the parts list.

The PipeInformation target identifies attributes that are used in the general OptionStore settings. For example: to identify a value to be displayed in a header note.

The Weld target indicates the attributes can be used in the Weld table.

## 7 Creo Environment Variables

Several environment variables are available to enable different functionality within the M4 ISO Isometrics for PTC Creo Piping application.

During installation the product is copied into the user specified directory. The installation program creates a new file named *M4ISOCreo.psf* or modifies an existing *.psf* file; this is defined during installation. The environment variables are usually set in the *.psf* file using the syntax:

ENV=<environment\_variable\_name>=<value>

Environment variables can also be defined in a Windows batch file using the syntax:

SET <environment\_variable\_name>=<value>

### 7.1 File handling

**M4ISOCREO\_DRW\_NAME="ISO-<PIPE\_NAME>"**

The variable M4ISOCREO\_DRW\_NAME can be used to define the isometric drawing name. It can use either one or both of the placeholders <ASSY\_NAME> and <PIPE\_NAME> and any other literal characters.

If this variable is not defined the drawing name defaults to <PIPE\_NAME>.

**M4ISOCREO\_SPOOLDRW\_NAME="ISO-<SPOOL\_NAME>"**

The variable M4ISOCREO\_SPOOLDRW\_NAME can be used to define the isometric drawing name when the *Select Spool(s)* checkbox is enabled. It can use either one or all of the placeholders <ASSY\_NAME> <SPOOL\_NAME> and <PIPE\_NAME> and any other literal characters.

If this variable is not defined the drawing name defaults to <SPOOL\_NAME>.

**M4ISOCREO\_PIPE\_NAME="<PIPE\_NAME>"**

The variable M4ISOCREO\_PIPE\_NAME is used in conjunction with the above 2 environment variables. It can use either one or all of the placeholders <ASSY\_NAME>, <SPOOL\_NAME> and <PIPE\_NAME> and any other literal characters.

It is used by the M4IsoOptionStore setting:

**"diagram/annotation/terminal\_connected\_to\_pipe\_format"**

to ensure the connection label reflects the actual isometric drawing name.

If this variable is not defined the name defaults to <PIPE\_NAME>.

**M4ISOCREO\_OUTDIR=C:\Creo\Isos**

The variable M4ISOCREO\_OUTDIR defines the output directory path for M4ISO files.

If this variable is not defined, then the current working directory is used.

#### **M4ISOCREO\_OUTDIRLOCK=YES**

The variable M4ISOCREO\_OUTDIRLOCK sets the output directory field to be locked (disabled) in the *Create M4Iso Isometric* dialog. Permitted values are YES or NO.

If this variable is not defined the value is assumed to be NO.

#### **M4ISOCREO\_TEMPLATEDIR=wtpub://PDMLINK/Libraries/CreoStandards/formats**

The variable M4ISOCREO\_TEMPLATEDIR sets the directory path where the format (FRM files) and drawing (DRW files) templates are located. This also applies to directories that are managed by a PLM system (e.g., Windchill).

If this variable is not defined, then these templates are read from the %M4ISOCREO\_HOME%\isoreal\templates folder.

#### **M4ISO\_AUXFILES\_DIR=C:\m4iso\data\_files**

The variable M4ISO\_AUXFILES\_DIR defines the output directory path for the CSV and NC files produced by M4ISO. It must be used with the M4IsoOptionStore.xml file settings:

```
<M4IsoOption name="core/files/alternative_aux_files_directory"
              valueType="string" value="$M4ISO_AUXFILES_DIR$"/>
<M4IsoOption name="core/files/use_alternative_aux_files_directory"
              valueType="boolean" value="t"/>
```

If this variable is not defined, then the current M4ISOCREO\_OUTDIR setting is used.

#### **M4ISOCREO\_DEL\_SHEFILES=YES**

This setting causes the Isoreal process to delete the intermediate SHE files created by the M4ISO process. Possible values are YES or NO.

The default is NO if this environment variable is not defined.

#### **M4ISOCREO\_DEL\_DXFFILES=YES**

This setting causes the Isoreal process to delete the intermediate DXF files created by the M4ISO process. Possible values are YES or NO.

The default is NO if this environment variable is not defined.



**M4ISOCREO\_DEL\_ISOXFILES=YES**

This setting causes the Isoreal process to delete the intermediate ISOX files created by the Isoreal M4ISO process. Possible values are YES or NO.

Note: When debugging problems with the M4ISOCreo application, it is often useful to send a copy of the ISOX file to CAD Schroer for analysis.

The default is NO if this environment variable is not defined.

**7.2 Creo Drawing Reuse**

See section 8 for a full explanation of how these settings work.

**M4ISOCREO\_SHEET\_DELETION=NO**

This setting causes the Isoreal process to re-use the isometric drawing files created by the M4ISO process. Possible values are YES or NO. A value of YES means the earlier drawing file is deleted when the isometric is recreated.

The default is YES if this environment variable is not defined.

**M4ISOCREO\_LAYER\_DEFAULT=M4ISO\_ELEMS**

This environment variable is used to assign the M4Iso geometry to a Creo layer named (in this example) M4ISO\_ELEMS. Note that any valid Creo layer name can be used.

If this variable is not defined then the normal Creo import process is used.

**M4ISOCREO\_LAYERS\_TODELETE=M4ISO\_ELEMS,M4ISO\_GRP**

The variable M4ISOCREO\_LAYERS\_TODELETE is used to define a list of comma separated Creo layer names that are to be deleted when the new isometric is drawn. This is used in conjunction with the M4ISOCREO\_SHEET\_DELETION=NO setting.

**7.3 Dialog settings****M4ISOCREO\_ISOX\_ONLY=NO**

The M4ISO dialog has a new checkbox to allow the creation of ISOX files only. This can be enabled by setting the M4ISOCREO\_ISOX\_ONLY value to YES. This will automatically disable the “Display Isometrics” checkbox on the dialog. Possible values are YES or NO.

The default is NO if this environment variable is not defined.

### **M4ISOCREO\_DISPLAY\_ISO=YES**

The “Display Isometrics” button will be enabled if the value of the M4ISOCREO\_DISPLAY\_ISO is set to YES. Possible values are YES or NO.

The default is YES if this environment variable is not defined.

### **M4ISOCREO\_CSYS\_DEFAULT=ACS0,ACS1,ASM\_DEF\_CSYS**

Identifies a Creo CSYS object to be used as the *Reference Csys* for the isometric diagram. The variable can hold one or more CSYS names separated by commas. The processing takes place from left to right and stops at the first named CSYS found in the assembly.

If this variable is not defined the first CSYS in the piping assembly is used as the *Reference Csys*.

## **7.4 Weld Fittings**

### **M4ISOCREO\_WELDPRT\_ID=UPLOAD\_SOFT\_TYPE:M4ISO\_WELD\_PART**

This setting defines the Creo parameter to be used for weld component identification. It's a combination of the Creo parameter name (UPLOAD\_SOFT\_TYPE) and the value (M4ISO\_WELD\_PART) delimited by a colon. Using this example, only parts/components that have the Creo parameter UPLOAD\_SOFT\_PART with the exact value M4ISO\_WELD\_PART are identified as weld components. Note that any valid Creo parameter name and any string value can be used for the identification.

If this variable is not defined then weld part files will not be recognized as welds by the M4ISO application.

### **M4ISOCREO\_WELD\_ID=WELDNO**

This setting references the named Creo parameter (WELDNO) found on the weld parts identified from the M4ISOCREO\_WELDPRT\_ID variable, to be used for the 'Weld ID' in the Weld Table.

If this environment variable is not defined, then weld id's are assigned automatically by the M4ISO application.

## **7.5 Miscellaneous**

### **M4ISOCREO\_TOLERANCE=0.1**

The variable M4ISOCREO\_TOLERANCE is used to define the coincidence tolerance between adjacent components. Any components that are within the tolerance will be assumed to be touching. Any components with a separation larger than the tolerance will be considered to contain a gap, which will force a pipe termination and a new page will be created. Any components which are smaller in length (width) than this tolerance (for example: gaskets) may be ignored and so will not be included on the isometric.

If this variable is not defined, the "core/coincidence\_tolerance" setting in the M4IsoOptionStore.xml file will be used.

### **M4ISOCREO\_RENUMBERING=SAP\_IDNUM**

The variable references a Creo parameter assigned to the parts in the assembly. Note that any parameter name can be used, as long as it contains unique part ID's for the components in the pipeline assembly (type Integer or String). In this example, the values assigned to the Creo parameter named SAP\_IDNUM will be used as the part reference IDs in the item details (partslist) table. This variable also requires the M4IsoOptionStore setting "core/item\_code\_ref/format" to be set to "item\_code".

If this variable is not defined, the M4ISO application will automatically assign part reference IDs.

### **M4ISOCREO\_VALVENUMBER=NO**

Setting the M4ISOCREO\_VALVENUMBER variable to 'YES' causes the CREO parameter 'VALVENUMBER' to be read and its value written as a label for the isometric symbol. Possible values are YES or NO.

The default is NO if this environment variable is not defined.

### **M4ISOCREO\_NO\_FILE\_MESSAGE=YES**

The mapping of Creo endtypes to M4ISO endtypes, depends on the existence of certain class project files in the piping spec folder (\*.inp files). In some situations, where custom piping specs have been created without these files, the isoreal program will generate warning messages that these files do not exist.

This environment variable setting can be used to stop the display of the missing file message. Possible values are YES or NO.

The default is NO if this environment variable is not defined, in which case the missing file message is displayed.

## 8 Creo drawing re-use

This functionality allows the M4ISOCreo application to open a previous, existing version of the Isometric drawing and create a new isometric without deleting any additional notes or model views that may have been added to the earlier isometric drawing.

Four conditions must be met:

1. Define a Creo layer to be used to contain any drafting entities drawn by the Isoreal program. Typically, these would be the lines used in the 3D view of the pipework, if enabled. To define the layer, Creo system option '**def\_layer**' must be set to '**layer\_draft\_grp** *<Creo layer name>*'

For example:

```
def_layer layer_draft_grp M4ISO_GRP
```

2. Group all the isometric geometry created by the M4ISO program to a Creo layer. By default, the M4ISO isometric geometry is displayed in Creo using the layer numbers defined in the M4IsoOptionStore.xml file.

Normally this would be layers 0, 4, 6 and 50-57 inclusive, but does not include the 3D view geometry. By using an environment variable, the isometric geometry layers can be referenced from a single layer.

To define a Creo layer named M4ISO\_ELEMS to contain the isometric geometry, use the environment variable:

```
set M4ISOCREO_LAYER_DEFAULT=M4ISO_ELEMS
```

3. Identify the Creo layers to be deleted when the isometric is recreated. An environment variable called '**M4ISOCREO\_LAYERS\_TODELETE**' is set to a list of comma-separated Creo layer names that are to be deleted when the new isometric is drawn. This would normally be the layers defined in steps 1 and 2 above. For example:

```
set M4ISOCREO_LAYERS_TODELETE= M4ISO_ELEMS, M4ISO_GRP
```

If this variable is not defined, then the isoreal program will assume ALL Creo layers are to be deleted when the isometric is recreated.

4. Tell the M4ISOCREO application that the drawing is to be reused. This is done using an environment variable which defines if the entire sheet is to be deleted (YES) or not (NO). For example:

```
set M4ISOCREO_SHEET_DELETION=NO
```

This value causes the existing isometric drawing to be reused with the identified Creo layers deleted before the new isometric geometry is loaded onto the existing drawing.

If this environment variable does not exist, then the application assumes a value of YES and the entire sheet is deleted each time the M4ISO program is run.

If the value is NO, then the Creo layers identified are deleted from the sheet and the remaining layers are re-used in the new isometric.

Note that any data on the isometric sheet that is not assigned to a Creo layer may be included in the **M4ISOCREO\_LAYER\_DEFAULT** when the sheet gets reloaded. As a result, this data may be deleted. To avoid this, all drawing geometry or annotations that are to be retained should be assigned to a Creo layer that is not in the **TODELETE** list.

## 9 How to improve the clarity of the Isometric

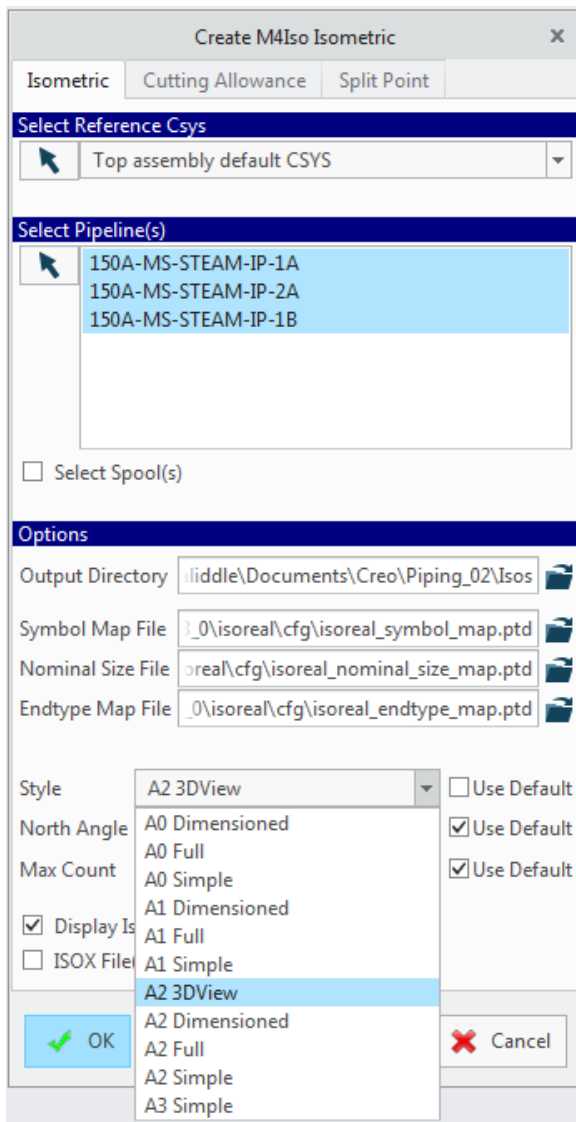
These are suggestions on how the clarity of an isometric could be improved.

1. Create the isometric diagram on a large sheet.
  - A minimum of A2 is recommended to give a reasonable diagram area and space for a 3D view and tables if required.
2. Consider the use of multi-page diagrams.
  - Setting the 'Max Count' value to, for example, 50 will split complex Pipe assemblies onto several sheets. Then each diagram will have more space to expand into.
  - For better control, consider using the Split Point tab to add page breaks.
  - Ensure the diagram size and position on the continuation sheets make the best use of the available space.
3. Simplify the dimensioning.
  - Use only 'Run' style dimensioning.
  - Ensure the component dimension settings are giving the best results.
4. Consider the use of Spools to split up complex pipe assemblies.
  - Each spool is created on its own isometric diagram.
5. Enlarge the symbol definitions to allow more space for dimensions.
  - The use of a 20x20 grid for isometric symbol definitions is recommended.
6. Use line and text styling to improve readability.
  - The use of different colors, text heights and line thickness, especially between geometry and dimensions, can help make the diagram easier to read.

## 10 3D View

The *Style* pulldown menu in the *Options* area of the *Create M4 ISO Isometric* dialog provides the selection choice *A2 3DView*.

### Create M4 ISO Isometric Dialog - 3D View Style Option



Style *A2 3DView* is defined such that the generated isometric drawing contains a 3D view of the piping geometry in addition to the 2D isometric (see below).

<M4ISO\_installationpath>\isoreal\templates.

55

## 11 M4ISO Configuration and customisation

This section details how to configure the M4IsoOptionStore.XML file to create (or modify) a 'Style' definition.

### 11.1 M4ISO Search Path

When searching for M4ISO configuration files, the program looks in specific directories using defined environment variables.

"M4ISO\_PROJ" and "M4ISO\_HOME" are environment variables created during the M4ISO installation process.

"USER\_HOME" is the home directory of the user created from the system environment variables:

`%HOMEDRIVE%%HOMEPATH%`.

In a Creo installation, the environment variables would be set as (for example):

`M4ISO_HOME="C:\<M4ISOcreo_installation>\m4iso"`

`M4ISO_PROJ="C:\<M4ISOcreo_installation>\medproj\m4iso"`

The paths are searched in top to bottom order as follows; the first file found is used:

`%M4ISO_PROJ%\data`

`%M4ISO_PROJ%`

`%M4ISO_HOME%\data`

`%M4ISO_HOME%`

`%USER_HOME%\M4Iso\data`

`%USER_HOME%\M4Iso`

When searching for template sheet files, it uses the same environment variable paths, except that the "data" subdirectory is replaced by "sheets" (for an example, see Section 11.6 "Isometric Sheet Template").

### 11.2 M4Iso Option Store file

The M4ISO product is supplied with several template style definitions that can be used to create isometric diagrams. If these do not meet your needs, then you can create your own template style definition.

In a Creo configuration the template style definitions can be found in:

`%M4ISO_HOME%\data\M4IsoOptionStore.xml`



### 11.2.1 File Structure

The *M4IsoOptionStore* file contains the default option settings for the M4ISO application.

These are defined in the section delimited as:

```
<M4IsoDefaultSection>
...
</M4IsoDefaultSection>
```

Within the *M4IsoDefaultSection*, the settings are configured in specific groupings listed in alphabetical order as follows:

```
name="component/..."
name="core/..."
name="dars_illustrator/..."
name="diagram/..."
name="illustrator/..."
name="qt_sketcher/..."
```

This is followed by the '*M4IsoCoreSection*', this only contains "**core/...**" option definitions.

Some of the values in this section can be modified by the user using the M4Iso user interface. The same options are also defined in the "**core/...**" area of the '*M4IsoDefaultSection*', the values here are used when the user 'Resets' the User Interface.

After this there are a number of '*M4IsoNamedSection*' definitions. These define the available style configurations and contain style specific values for some of the settings defined in the '*M4IsoDefaultSection*'.

The metric default values supplied will create an "A1 Simple" style, that is, an A1 sheet with the isometric diagram and a Parts List only.

The inch default values will create an "ANSI D Simple" style.

### 11.2.2 Option Groupings

The option groupings are used to control various aspects of the M4Iso application.

"**component/...**" settings are defined for each piping component that the M4Iso application can deal with when processing a pipe isometric. Normally these settings are used to describe how dimensioning references a particular piping component symbol.

"**core/...**" settings affect the internal configuration of the M4Iso application. These

values cannot be applied in the ‘*M4IsoNamedSection*’ style definitions. If you add “**core/...**” settings into the ‘*M4IsoNamedSection*’ style definitions, the software will ignore them.

“**dars\_illustrator/...**” settings control data drawn onto the M4ISO isometric sheet. These settings generally control the appearance of lines and text (colour, boldness, etc.) on the M4ISO isometric sheet.

“**diagram/...**” settings control output specific to the Isometric diagram. These will control if an object such as a label or a table is to be displayed, or the dimensioning style to be used.

“**illustrator/...**” settings control common settings for data drawn onto the isometric sheet.

“**qt\_sketcher/...**” settings control the appearance of lines and text in the M4Iso “Interactive” window. (Not available in M4ISOcreo)

## 11.3 Graphical Representations

### 11.3.1 Element Types

In the “**dars\_illustrator/...**” grouping, attributes that create lines and text refer to an “**element\_type**” attribute, for example:

```
<M4IsoOption name="dars_illustrator/item_details_table/line/element_type"
              valueType="string" value="L0"/>
```

The *value=* settings refer to a simple mapping table. The values available are shown in the table below:

Value	Style	value	style
“L0”	Solid line	“T1”	Simple text
“L1”	Dotted line	“TR1”	Rectangular Box Text
“L2”	Chain line	“TS1”	Rounded Box Text
“L3”	Dashed line		

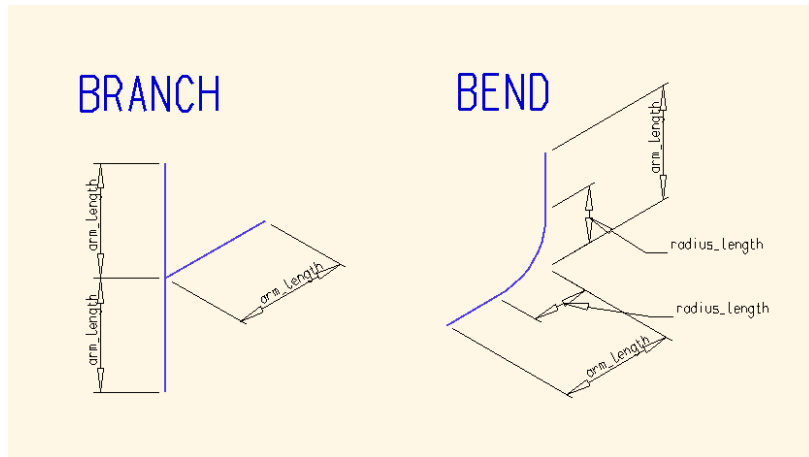
### 11.3.2 Branch and Bend

There are also several settings that can be used to adjust the appearance of branches and bends in the isometric diagram. Adjusting these can aid the clarity of the diagram:

```
<M4IsoOption name="diagram/bend/arm_length" valueType="real" value="40"/>
```

```
<M4IsoOption name="diagram/bend/radius_length" valueType="real" value="20"/>
```

```
<M4IsoOption name="diagram/branch/arm_length" valueType="real" value="20"/>
```



## 11.4 Creating a Custom Style Definition

The example here will go through the process of:

- Consideration of the Units to be used.
- Defining a new style.
- Configuring the template sheet and creating the drawing files to be used in the style.
- Defining the dimension styles to be created on the isometric.
- Creating a parts list table on the isometric.
- Adding notes/labels to components/parts on the isometric.
- Adding notes/labels at end (terminal) points or changes of direction in the pipeline.
- Adding notes to the isometric drawing.
- Adding a Weld List table to the isometric.
- Adding a Spool List table to the isometric.
- Adding a Cut List table to the isometric.
- Adding a Bend Table to the isometric.
- Controlling the filenames and the files used to create the isometric.

## 11.5 Isometric units

The default configuration files supplied with the M4ISOCreo installation are set up for metric millimetre units.

If there is a requirement to use inch units, there are a number of things that will need to be considered.

First, switch off the metric units using the setting:

```
<M4IsoOption name="dars_illustrator/sheet/metric_units"
              valueType="boolean" value="f"/>
```

In the *dars\_illustrator* block of settings, there are several settings that will need to be changed to inch units. These include settings that relate to:

- Text heights
- Line thickness
- Table column widths
- Table base positions
- Dimension snap range
- Elevation fill line spacing
- Header note positions
- Sheet size
- Sheet diagram position and size

There will still be some setting values that can only be defined in metric units. These are normally the values set in the *illustrator* block of settings and the *core* settings.

In the *diagram* block of settings, there are entries that define the units to be displayed in the isometric:

```
<M4IsoOption name="diagram/display_bore_units"
               valueType="string" value="$source$"/>
<M4IsoOption name="diagram/display_length_units"
               valueType="string" value="$source$"/>
<M4IsoOption name="diagram/display_position_units"
               valueType="string" value="$source$"/>
```

There are also similar settings in some of the table definitions:

```
<M4IsoOption name="diagram/bends_table/display_length_units"
               valueType="string" value="$source$"/>
<M4IsoOption name="diagram/cuts_table/display_length_units"
               valueType="string" value="$source$"/>
<M4IsoOption name="diagram/item_details_table/display_length_units"
               valueType="string" value="$source$"/>
```

These values can be set as:

- \$source\$ - uses the units defined in the model, usually mm or in
- mm
- in
- in+frac
- ft+in+frac
- &quot; - a keyword code to display the “ sign

In the examples that follow, the units are assumed to be metric. Specific examples of inch units will be shown when appropriate.

## 11.6 Defining a New Style

If you wish to create your own style definition, then the first thing you must do is create a new '*M4IsoNamed Section*' (alternatively, you can modify one of the existing

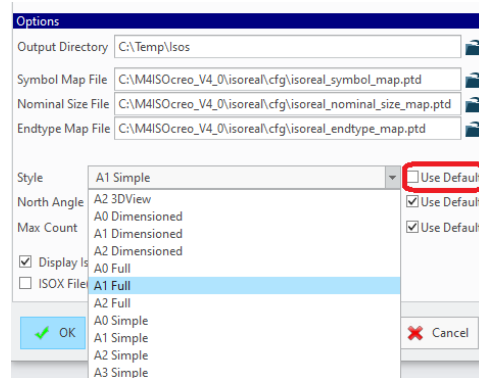
styles).

```
<M4IsoNamedSection name="Custom Full 3D A1">
```

...

```
</M4IsoNamedSection>
```

You must supply a 'name=' attribute at the start of the 'M4IsoNamedSection' definition. The value assigned will be displayed in the M4Iso dialog within the 'Style' pulldown menu for the user to select.



Between the start and end of the 'M4IsoNamedSection' area you define the settings you wish to change. It is good practice to keep the settings in alphabetical order to make them easier to find.

All the settings are 'M4IsoOption' definitions and are usually created on a single line between '<' and '/>'. They normally contain settings for 'name', 'valueType' and 'value'.

Please note: In the following examples, the settings are shown over several lines for clarity. But they would normally be on a single line with spaces as separators.

If you want to have your style definition as the default style presented by the M4Iso dialog, then look for the **M4IsoCurrentSection** setting and change the name value to be the name of your style. For example:

```
<M4IsoCurrentSection name="Custom Full 3D A1"/>
```

## 11.7 Isometric Sheet Template

Commonly, the first setting you would want to change is the template sheet file:

```
<M4IsoOption name="dars_illustrator/sheet/template"
valueType="string" value="Custom_A1_template.she"/>
```

The value of "Custom\_A1\_template.she" is the filename of the template sheet. This file must exist in one of the folders:

```
%M4ISO_PROJ%\sheets
%M4ISO_HOME%\sheets
```

For each template sheet, you must define the sheet size and the area that the isometric diagram will occupy on the sheet:

```
<M4IsoOption name="dars_illustrator/sheet/diagram_position"
               valueType="coord2d" value="80 80"/>
<M4IsoOption name="dars_illustrator/sheet/diagram_size"
               valueType="coord2d" value="500 400"/>
<M4IsoOption name="dars_illustrator/sheet/size"
               valueType="coord2d" value="841 594"/>
```

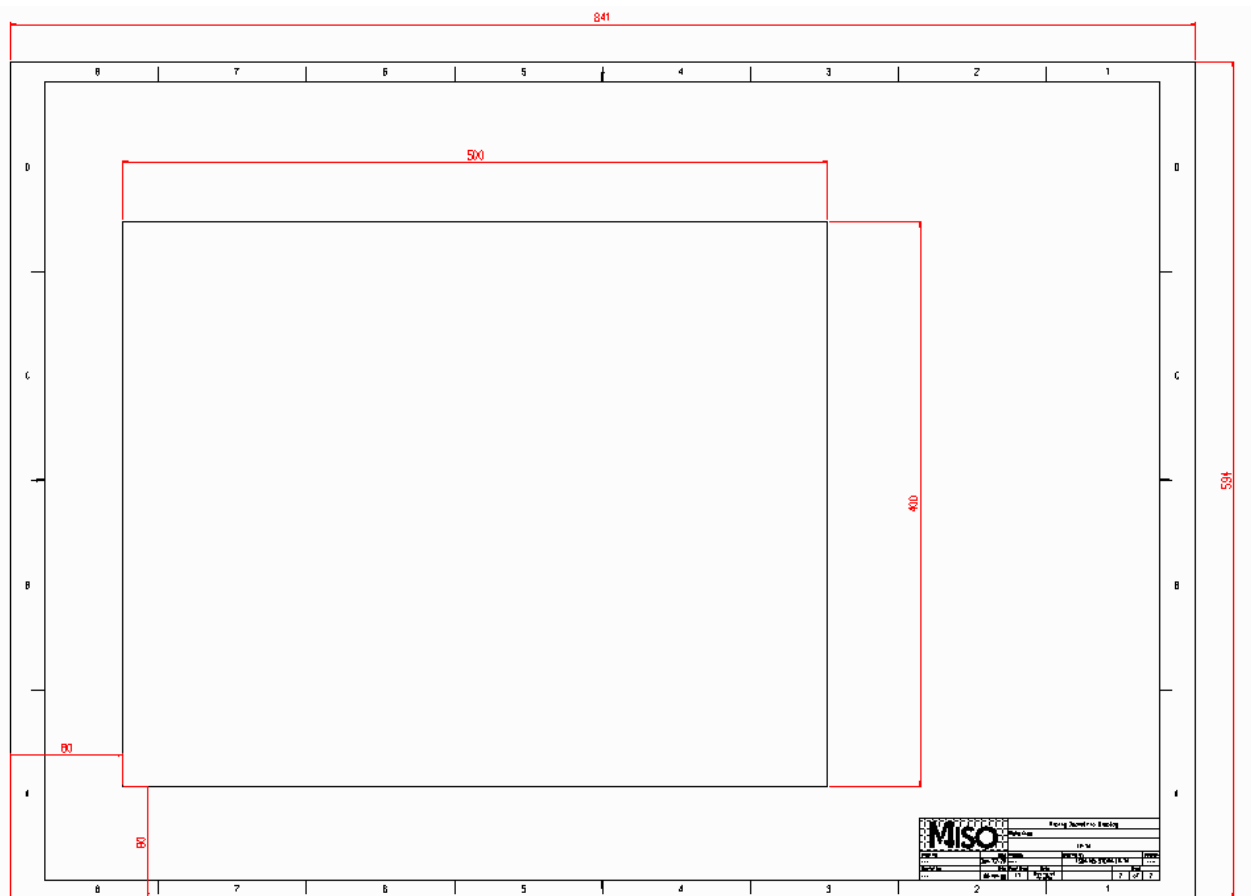
The 'valueType' of "coord2d" dictates that the 'value' must be two numbers representing X and Y values. These may represent a co-ordinate position or dimensions.

The values we have used above tell the application that the isometric diagram area:

is positioned with its bottom left corner at x=80mm, y=80mm,

is of size 500x400mm

is contained on a sheet size of 841x594mm (A1).



For an inch unit template sheet, the sizes are defined in inches, for example:

```
<M4IsoOption name="dars_illustrator/sheet/diagram_position"
               valueType="coord2d" value="3.250 3.250"/>
<M4IsoOption name="dars_illustrator/sheet/diagram_size"
               valueType="coord2d" value="21.0 18.0"/>
<M4IsoOption name="dars_illustrator/sheet/size"
               valueType="coord2d" value="34.0 22.0"/>
<M4IsoOption name="dars_illustrator/sheet/template"
               valueType="string" value="Custom_D_template.she"/>
```

### 11.7.1 Isometric Sheet Templates

In the Creo implementation, M4ISO is supplied with a set of blank template SHE files in the size range A0 to A3 for metric units and E to B in inch units.

These can be found in the folder:

`%M4ISO_HOME%\sheets`

There is a matching selection of Creo Format templates, that contain the Creo drawing border details, and optional Drawing templates. These can be found in the folder:

`%M4ISOCREO_HOME%\isoreal\templates`

It is also possible to access the Creo Format and Drawing templates from a specific folder by setting an environment variable named **M4ISOCREO\_TEMPLATEDIR**. This is especially useful when FMT and DRW templates are managed by a PLM system, such as Windchill. This will not access the SHE template files.

The M4ISO template sheet is referenced by the M4IsoOptionStore.xml settings, and the format file with the same name is referenced automatically when the finished isometric is displayed in Creo.

```
<M4IsoOption name="dars_illustrator/sheet/template"
               valueType="string" value="Custom_A1_template.she"/ >
```

format file: Custom\_A1\_template.fmt

To create a new template sheet, use this procedure:

Select the correct sheet size from the supplied M4ISO template sheets and copy to the new name you want to use. For example: Copy m4iso\_A1\_template.she to Custom\_A1\_template.she

These M4ISO template sheets are blank and do not need to be modified.

In the same way, copy one of the existing FMT files, or one of your own FMT files to the new name used above. Edit the new FMT file as required to define the border details you require.

Or you can, if preferred, create the Creo FMT file from scratch.

If required, copy or create a Creo Drawing file with the same name to define the 3D view (see next section for details). Edit the new DRW file as required.

### 11.7.2 3D View definition

If the template sheet is to include a 3D view of the pipework from the Creo Piping application, then the 3D view *enabled* switch will need to be set to true (value="t"):

```
<M4IsoOption name="dars_illustrator/sheet/view_3d/enabled"  
              valueType="boolean" value="t"/>
```

The 3D viewbox area is defined on a Creo Drawing.

The name of the Drawing must be the same as the name of the M4ISO template sheet.

(e.g. Custom\_A1\_template.drw)

The Drawing file is normally stored in the folder:  
%M4ISOCREO\_HOME%\isoreal\templates (see File Naming section).

The Drawing must contain a layer named: M4ISO\_3DVIEW

On that layer there must be two line elements; 1 horizontal line and 1 vertical line.





The horizontal line defines the width of the viewbox and its horizontal position on the drawing.

The vertical line defines the height of the viewbox and its vertical position on the drawing.

The layer that contains the lines may be hidden so that the definition lines do not appear on the finished isometric drawing.

### 11.7.3 Continuation Sheet

The maximum number of components that can be drawn in a single isometric diagram is controlled by the setting **core/max\_component\_count** whose value is 300 by default. This value can be changed by the user using the Max Count field in the M4Iso dialog.

The use of Split Point markers in the pipe can override the effect of the Max Count value.

If the number of components drawn on a page exceed this value (or a Split Point is found), the components are split up and drawn on one or more continuation sheets. The details of the continuation sheet need to be defined in a similar way to the template sheet:

```
<M4IsoOption name="dars_illustrator/sheet/continuation/diagram_position"
  valueType="coord2d" value="80 80"/>
<M4IsoOption name="dars_illustrator/sheet/continuation/diagram_size"
```

```

valueType="coord2d" value="500 400"/>
<M4IsoOption name="dars_illustrator/sheet/continuation/size"
valueType="coord2d" value="841 594"/>
<M4IsoOption name="dars_illustrator/sheet/continuation/template"
valueType="string" value="custom_A1_continuation_template.she"/>

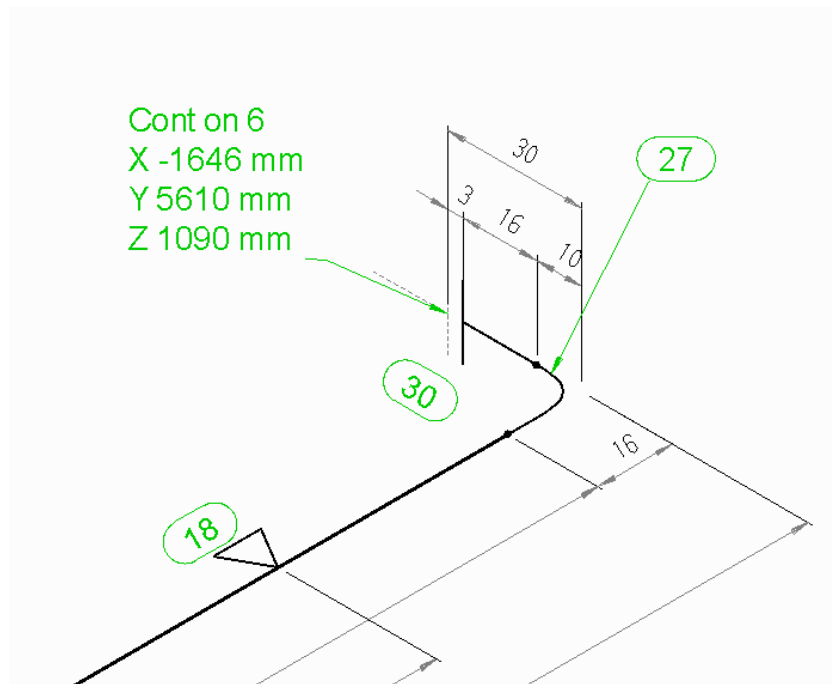
```

When the new page is created, then a continuation label can be defined to identify the sheet that the pipe is continued on:

```

<M4IsoOption name="diagram/annotation/page_continuation_format"
valueType="string" value="Cont on %4&#10;X %1&#10;Y %2&#10;Z %3"/>
<M4IsoOption name="diagram/annotation/show_page_continuation_note"
valueType="boolean" value="t"/>

```



In the page\_continuation\_format string "Cont on %4&#10;X %1&#10;Y %2&#10;Z %3"

%1 represents the X position value,

%2 is the Y position value and

%3 is the Z position value of the page break.

%4 is the page number that the pipe continues on.

The &#10; value is interpreted as a line feed to create multiple lines in the continuation label.

## 11.8 View Orientation

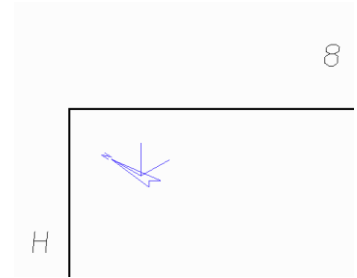
The orientation of the isometric view on the sheet is controlled by the **core/north\_angle** setting. This must be changed in the 'M4IsoCoreSection' definition and cannot be set in the 'M4IsoNamedSection'. The 'north' direction is the positive Y axis direction in the model.

Note that this value can be changed by the user in the M4Iso dialog using the North Angle field.

```
<M4IsoOption name="core/north_angle" valueType="real" value="180"/>
```

The North Arrow symbol is used to indicate the isometric orientation on the sheet. This can be positioned on the sheet using the setting:

```
<M4IsoOption name="dars_illustrator/north_arrow/position"
valueType="coord2d" value="30 550"/>
```

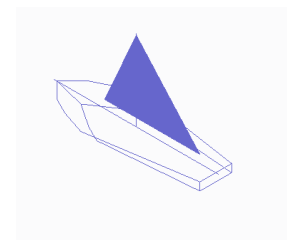
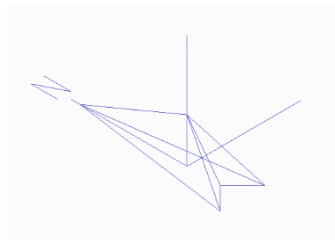
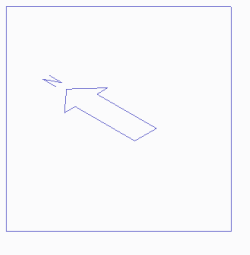


The 'value' setting is the X and Y position on the sheet where the centre of the symbol will be positioned.

To position the symbol on the continuation sheet, use:

```
<M4IsoOption name="dars_illustrator/north_arrow/continuation/position"
valueType="coord2d" value="30 550"/>
```

The style of the North Arrow symbol can be changed, some examples of different North Arrow symbols are shown below. See the section on *Creating Custom Component Symbols* for details on how to achieve this.



The North Angle value orientates the N direction of the symbol to point to a specific corner of the sheet:

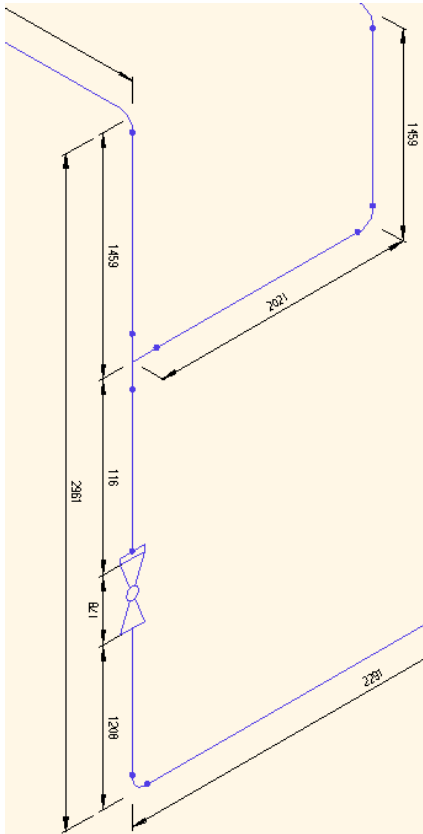
- 0 degrees points at the lower right corner
- 90 degrees points at the upper right corner
- 180 degrees points at the upper left corner (as shown above)
- 270 degrees points at the lower left corner

## 11.9 Dimensioning

Dimensions in the isometric diagram can be displayed as overall dimensions and/or dimensions between components (run level) and/or supports (support level).

```
<M4IsoOption name="diagram/dimension/show_overall_level"
valueType="boolean" value="t"/>
<M4IsoOption name="diagram/dimension/show_run_level"
valueType="boolean" value="t"/>
<M4IsoOption name="diagram/dimension/show_support_level"
valueType="boolean" value="t"/>
```

The 'value' can be set to "t" (true) or "f" (false).



The order of the dimension layout is controlled using the setting:

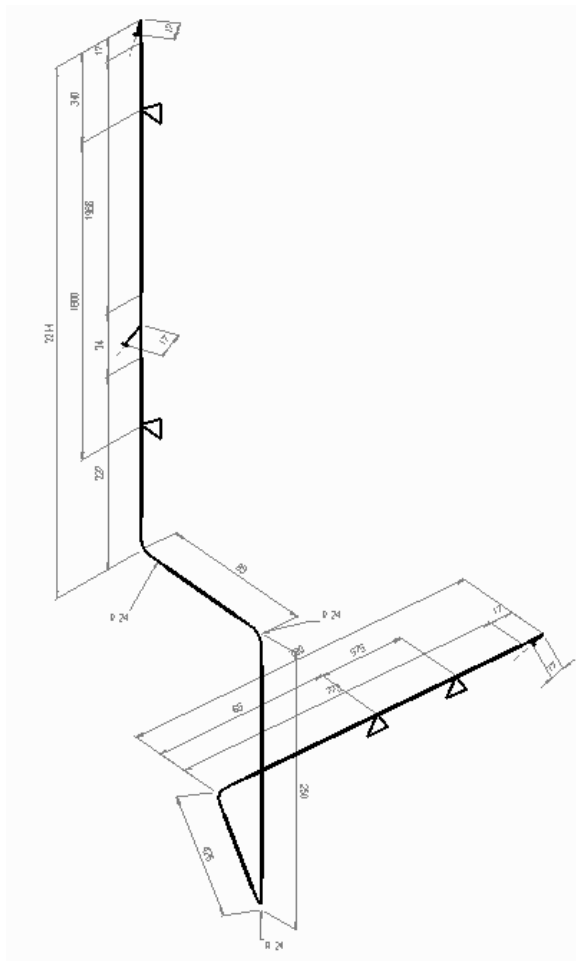
```
<M4IsoOption name="diagram/dimension/order"
  valueType="string" value="run, support, overall"/>
```

The position of dimensions on component symbols can be controlled to either the CP's and/or the datum.

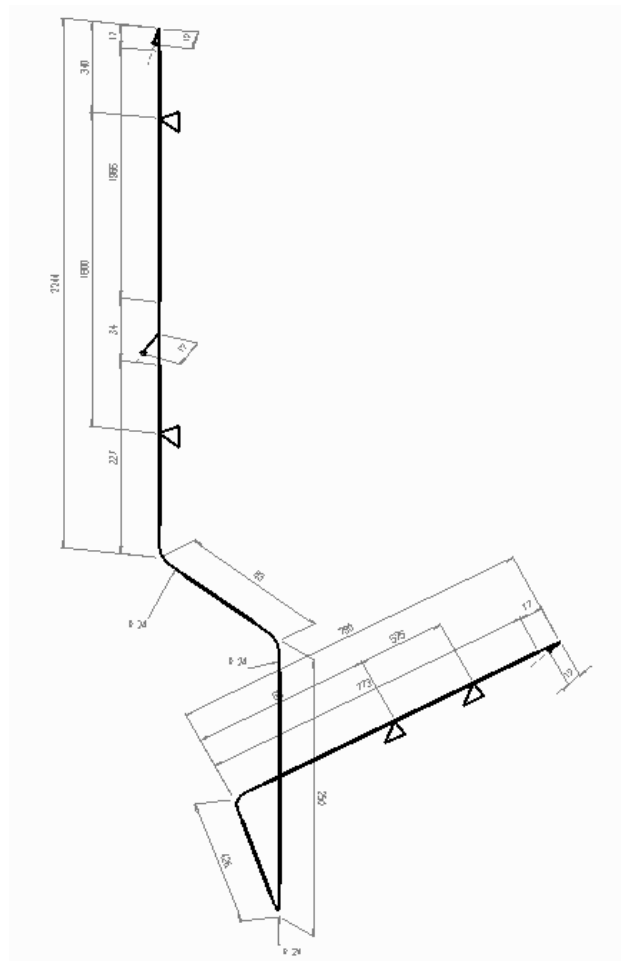
```
<M4IsoOption name="component/branch/dim_to_cps"
  valueType="boolean" value="f"/>
<M4IsoOption name="component/branch/dim_to_datums"
  valueType="boolean" value="t"/>
...
<M4IsoOption name="component/valve/dim_to_cps"
  valueType="boolean" value="t"/>
<M4IsoOption name="component/valve/dim_to_datums"
  valueType="boolean" value="f"/>
```

Normally dimensions are drawn on the same plane as the first symbol in the pipe run. However, this can sometimes make the dimensions difficult to read. A switch is available to force the dimensions onto the horizontal or vertical planes.

```
<M4IsoOption name="diagram/dimension/align_with_skewed_components"
  valueType="boolean" value="f"/>
```

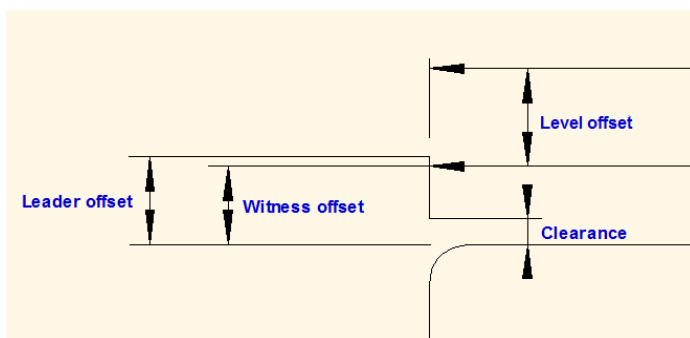


align\_with\_skewed\_components is False



align\_with\_skewed\_components is True

Dimension offset from the pipe geometry is controlled using 4 diagram settings:



```
<M4IsoOption name="diagram/dimension/clearance_offset" valueType="real" value="2"/>
<M4IsoOption name="diagram/dimension/leader_offset" valueType="real" value="27"/>
<M4IsoOption name="diagram/dimension/level_offset" valueType="real" value="15"/>
<M4IsoOption name="diagram/dimension/witness_offset" valueType="real" value="25"/>
```

The number of decimal places to display in a dimension can be controlled using the settings:

```
<M4IsoOption name="diagram/display_precision"
              valueType="integer" value="1"/>
<M4IsoOption name="diagram/display_precision_angle"
              valueType="integer" value="2"/>
<M4IsoOption name="diagram/display_precision_length"
              valueType="integer" value="2"/>
```

For inch unit dimensions, the fractional notation is controlled using:

```
<M4IsoOption name="diagram/display_fractional_precision"
valueType="integer" value="32"/>
```

Values of 4, 8, 16, 32 and 64 are supported.

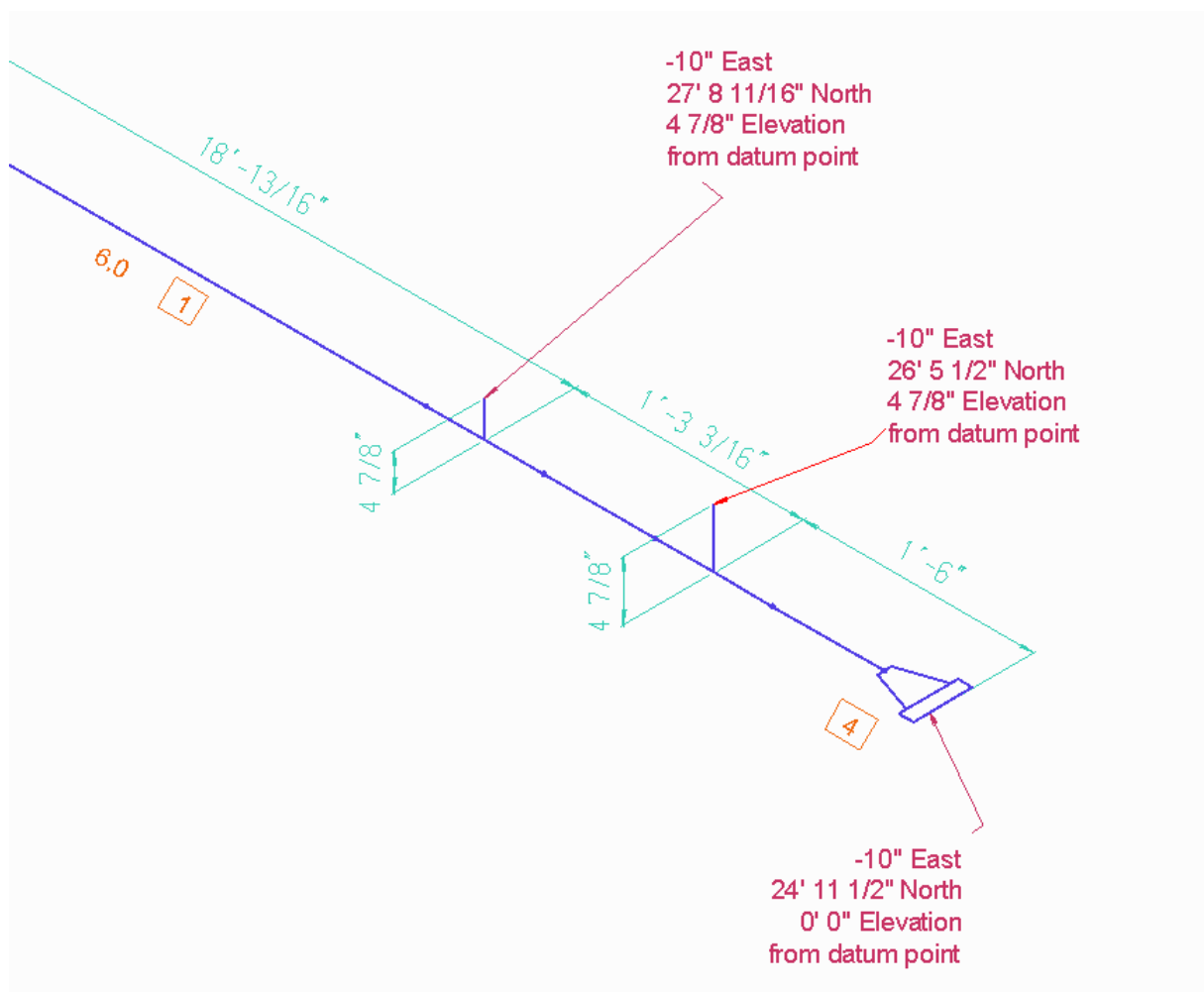
Dimensions will be shown as the largest fractional value. For example 12/16" will be shown as 3/4".

To display length dimensions to a precision of 1/16<sup>th</sup> of an inch as shown below, use the settings:

```
<M4IsoOption name="diagram/display_length_units"
valueType="string" value="ft+in+frac"/>
<M4IsoOption name="diagram/display_fractional_precision"
valueType="integer" value="16"/>
```

By default, zero inch values with fractions are not displayed. If it is required to display the zero inch value, use the setting:

```
<M4IsoOption name="diagram/display_zero_inches_fractional"
valueType="boolean" value="t"/>
```



## 11.9.1 Elevations and Falls

Pipe elevation and fall data can be displayed at non-orthogonal slopes using:

```
<M4IsoOption name="core/elevation/min_out_of_plane_angle"
               valueType="real" value="3"/>

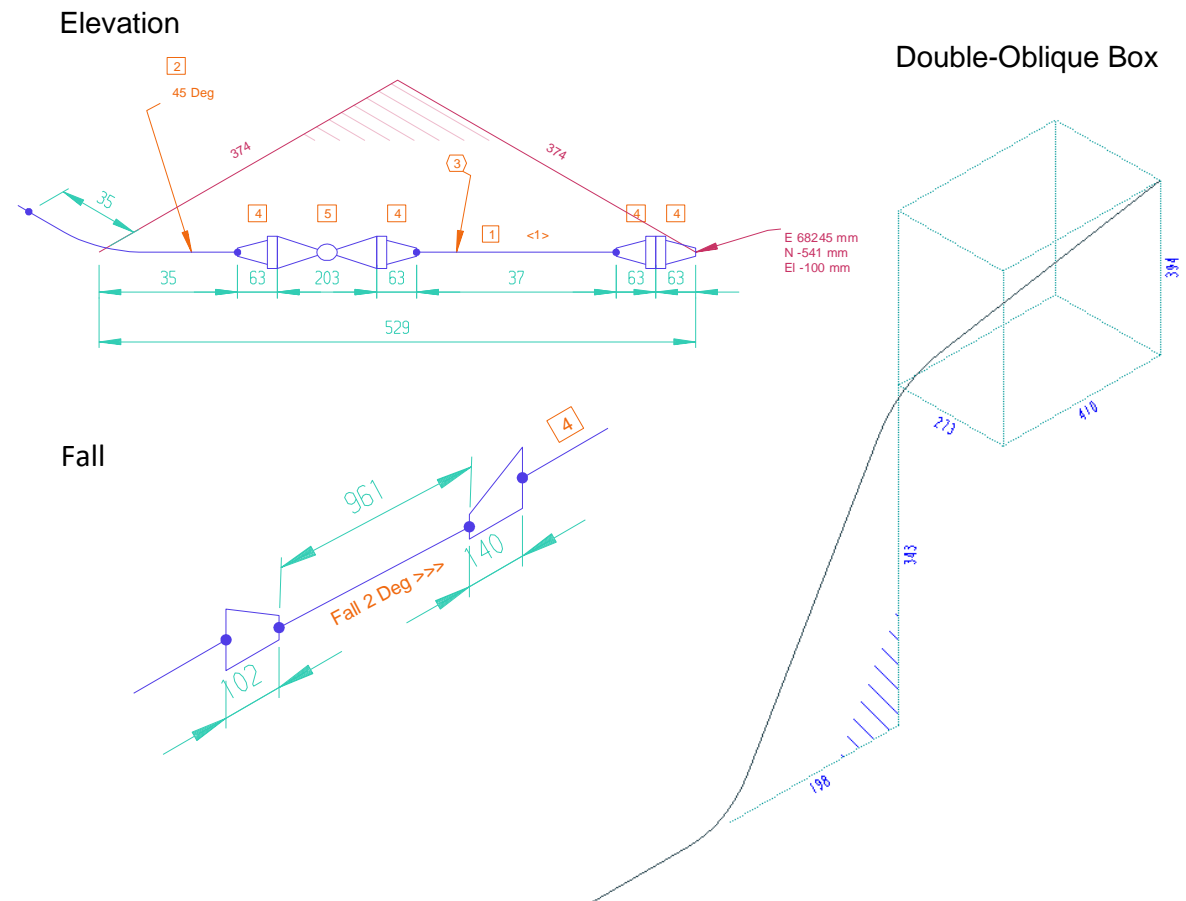
<M4IsoOption name="core/fall/max_angle" valueType="real" value="3"/>

<M4IsoOption name="core/fall/min_angle" valueType="real" value="0.1"/>

<M4IsoOption name="diagram/elevation/show_double_oblique_as_box"
               valueType="boolean" value="f"/>

<M4IsoOption name="diagram/elevation/show_fill" valueType="boolean" value="t"/>

<M4IsoOption name="diagram/show_elevation" valueType="boolean" value="t"/>
```



The appearance of the elevation geometry is controlled using these settings:

```
<M4IsoOption name="dars_illustrator/elevation/fill/boldness"
               valueType="real" value="0"/>

<M4IsoOption name="dars_illustrator/elevation/fill/colour"
               valueType="colour" value="238 187 204 255"/>

<M4IsoOption name="dars_illustrator/elevation/fill/element_type"
               valueType="string" value="L0"/>

<M4IsoOption name="dars_illustrator/elevation/fill/line_angle"
```

```
valueType="real" value="45"/>

<M4IsoOption name="dars_illustrator/elevation/fill/line_spacing"
valueType="real" value="5"/>

<M4IsoOption name="dars_illustrator/elevation/layer" valueType="integer" value="6"/>

<M4IsoOption name="dars_illustrator/elevation/line/colour"
valueType="colour" value="200 50 100 255"/>

<M4IsoOption name="dars_illustrator/elevation/line/element_type"
valueType="string" value="L0"/>

<M4IsoOption name="dars_illustrator/elevation/line/thickness"
valueType="real" value="0"/>

<M4IsoOption name="dars_illustrator/elevation/link/point_function"
valueType="integer" value="58"/>

<M4IsoOption name="dars_illustrator/elevation/text/colour"
valueType="colour" value="200 50 100 255"/>

<M4IsoOption name="dars_illustrator/elevation/text/element_type"
valueType="string" value="T1"/>

<M4IsoOption name="dars_illustrator/elevation/text/font_name"
valueType="string" value="Arial"/>

<M4IsoOption name="dars_illustrator/elevation/text/height" valueType="real" value="2"/>

<M4IsoOption name="diagram/annotation/elevation/border_style"
valueType="text border style" value="none"/>

<M4IsoOption name="diagram/annotation/fall/border_style"
valueType="text border style" value="none"/>

<M4IsoOption name="diagram/display_precision_angle" valueType="integer" value="0"/>

<M4IsoOption name="diagram/elevation/fill_area_fraction" valueType="real" value="0.35"/>
```



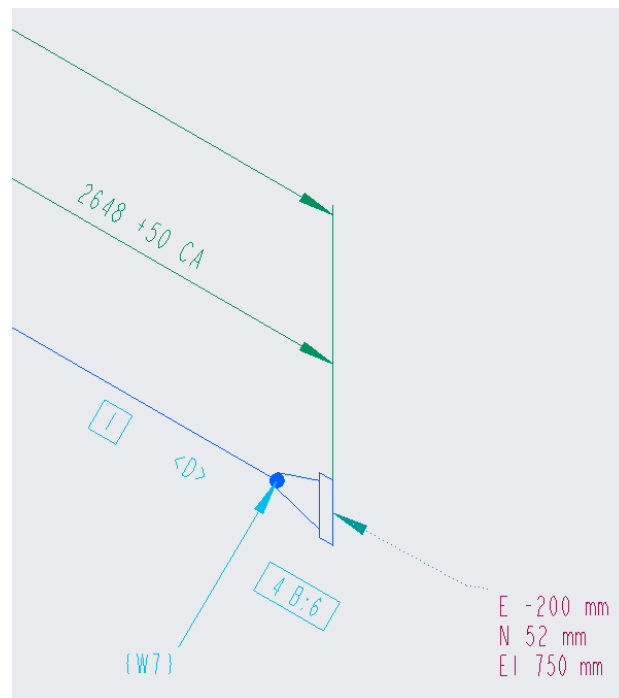
### 11.9.2 Show Cut Allowance Tolerance

```
<M4IsoOption name="diagram/dimension/show_cut_tolerance"
               valueType="boolean" value="f"/>
```

This setting can be used to show a cut allowance "tolerance" string on dimensions - where the tolerance is the Cut Allowance. The output is displayed as Postfix text on the dimension. This additional data is only added to the lowest level of dimensions (also known as "run level" dimensions in M4ISO).

```
<M4IsoOption name="diagram/dimension/ cut_tolerance_format"
               valueType="string" value="+%$tolerance$% CA"/>
```

This setting defines a format string to control the tolerance value shown on the dimension. The **+\$tolerance\$%** will be replaced by the value of the Cut Allowance for that dimension. This follows standard M4ISO functionality of being able to create a format based on some internal variables. Currently **+\$tolerance\$%** is the only supported one.



### 11.9.3 Bore label format

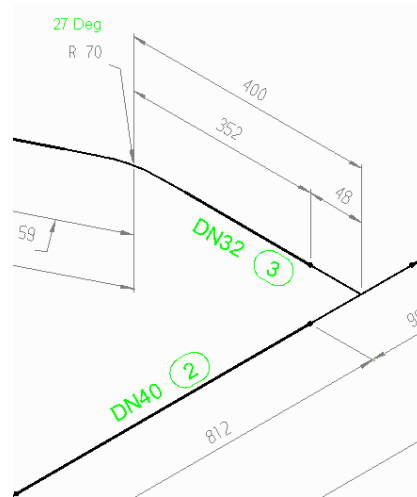
The label shown to indicate piping bore size can be defined using the settings:

```
<M4IsoOption name="diagram/annotation/bore_size/border_style"
  valueType="text border style" value="none"/>
<M4IsoOption name="diagram/annotation/bore_size/format"
  valueType="string" value="%%$size$%%$designation$%%"/>
<M4IsoOption name="diagram/annotation/show_bore_size"
  valueType="boolean" value="t"/>
```

Changing the format to

```
value="%%$designation$%%$size$%%"
```

gives the result shown:



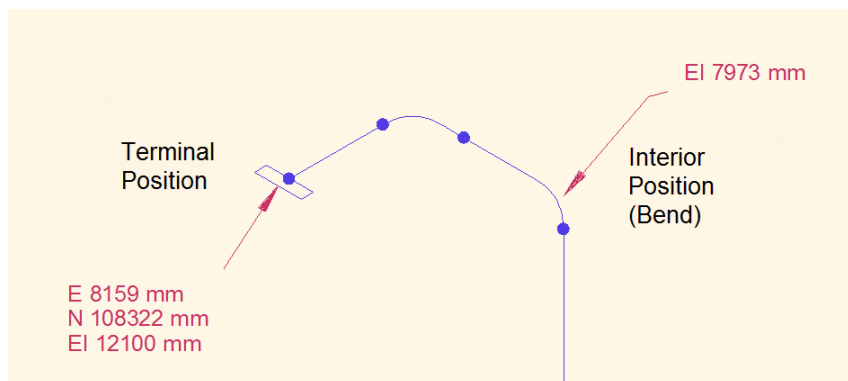
### 11.9.4 Position Labels

Position data can be displayed at bends and at terminal points using:

```
<M4IsoOption name="diagram/annotation/show_interior_position"
  valueType="boolean" value="t"/>
<M4IsoOption name="diagram/annotation/show_terminal_position"
  valueType="boolean" value="t"/>
```

The position data format is controlled using:

```
<M4IsoOption name="diagram/annotation/interior_position_format"
  valueType="string" value="EI %3"/>
<M4IsoOption name="diagram/annotation/terminal_position_format"
  valueType="string" value="E %1&#10;N %2&#10;EI %3"/>
```



In the format string: `value="E %1&#10;N %2&#10;EI %3"/>`

`%1` represents the X value,

`%2` is the Y value and

`%3` is the Z value of the position.

The `&#10;` value is interpreted as a line feed to create multiple lines in the po-

sition

label.

### 11.9.5 Terminal Connection Labels

It is possible to define connection labels on terminal points to identify if they are attached to another pipeline or to a piece of equipment. This relies on the CSYS at the terminal point being co-incident with another CSYS on a different pipeline assembly or on an equipment model. It can also work using connections with skeleton parts.

Use this setting to display the terminal connection labels:

```
<M4IsoOption name="diagram/annotation/show_terminal_connected_to"
               valueType="boolean" value="t"/>
```

To specify the label for equipment connections, use this setting:

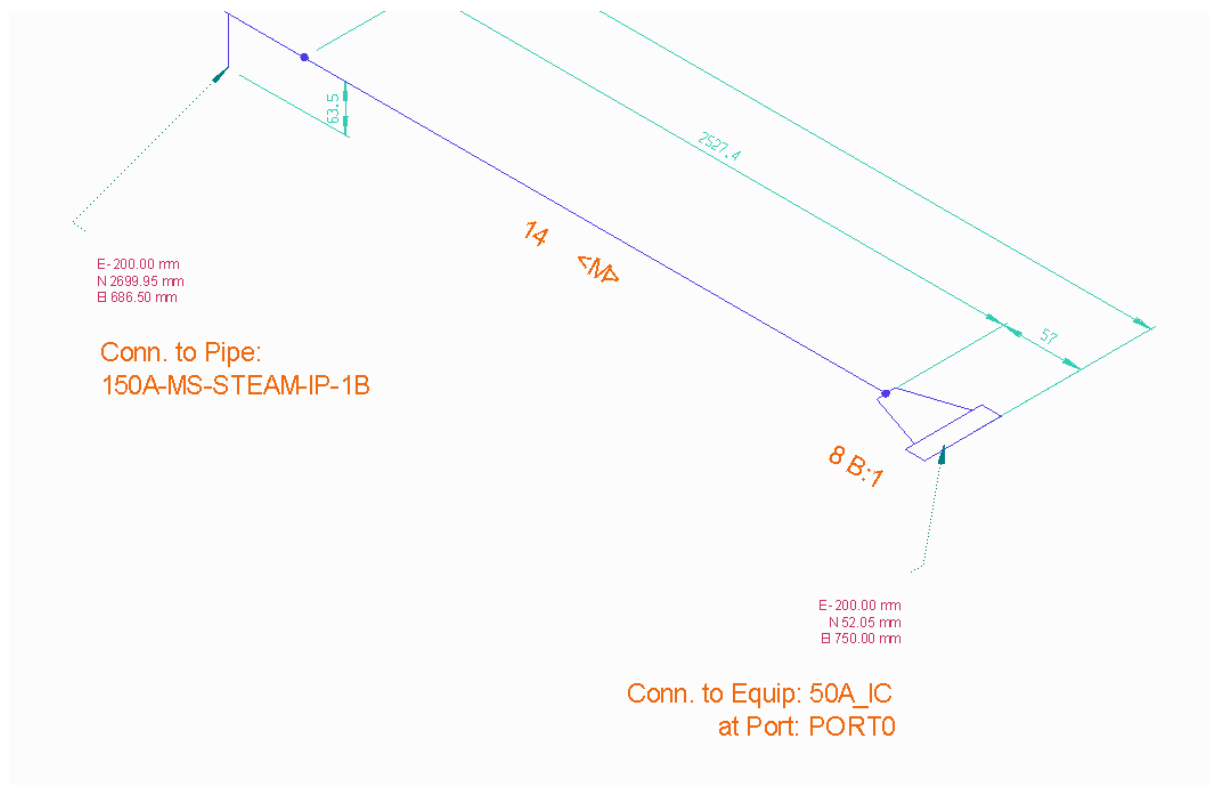
```
<M4IsoOption name="diagram/annotation/terminal_connected_to_equip_format"
               valueType="string" value="Conn. to Equip: %1 &#10; at Port: %2"/>
```

The label uses the %1 reference to hold the equipment name and %2 to reference the connection port name. The character code &#10; is used to force a newline in the label so it appears on more than one line.

To specify the label for pipeline connections, use this setting:

```
<M4IsoOption name="diagram/annotation/terminal_connected_to_pipe_format"
               valueType="string" value="Conn. To Pipe: &#10; %1"/>
```

This label only uses the %1 reference to identify the pipeline name. The character code &#10; is to force a newline in the label.



When the environment variable M4ISOCREO\_DRW\_NAME or M4ISOCREO\_SPOOLDRW\_NAME has been used to redefine the name that the

isometric drawing is saved under, then the names used on the pipe connection labels may not match the isometric drawing name.

Use the environment variable M4ISOCREO\_PIPE\_NAME to ensure that the pipe names referenced in the connection labels are set to match the names used in the isometric drawing name. For example:

```
set M4ISOCREO_DRW_NAME="<ASSY_NAME>"
set M4ISOCREO_PIPE_NAME="<ASSY_NAME>"
```

## 11.10 References

Certain items on the isometric diagram can be identified with a reference ID. The ID can be configured in the Option Store settings to display in different formats. Note that most of these settings are **core** settings and therefore cannot be defined as style settings. If modified, they will apply to all styles.

The items that have reference IDs are:

**bend\_ref**      **cut\_ref**                      **item\_code\_ref**                      **spool\_ref**      **weld\_ref**

### 11.10.1 Common Settings

The following 4 settings are common to all the reference ID's listed above, simply insert the ID name instead of the \*\* characters shown below:

```
<M4IsoOption name="core/**/prefix" valueType="string" value=""/>
<M4IsoOption name="core/**/suffix" valueType="string" value=""/>
<M4IsoOption name="core/**/sequence_type"
    valueType="string" value="numeric"/>
<M4IsoOption name="core/**/custom_sequence_symbols"
    valueType="string" value=""/>
```

The **prefix** and **suffix** settings allow the insertion of any characters before or after the created ID. For example:

```
<M4IsoOption name="core/weld_ref/prefix" valueType="string" value="W"/>
```

generates weld IDs W1, W2, W3,...

The **sequence\_type** setting can be set to one of either "*numeric*", "*alpha*", "*alpha\_no\_o*" or "*custom\_sequence*". Only if the **sequence\_type** is set to "*custom\_sequence*" does the **custom\_sequence\_symbols** setting get used. The characters from **custom\_sequence\_symbols** are used in the order they are defined. For example, if **custom\_sequence\_symbols**, value="AFKM", the ID sequence generated would be A, F, K, M, AA, AF, AK, AM, BA, BF, BK, BM, CA, ..., ZA, ZF, ZK, ZM

### 11.10.2 Show References

The following setting is only available for **cut\_ref**, **spool\_ref** and **weld\_ref** items:

```
<M4IsoOption name="diagram/show_cut_ref" valueType="boolean" value="f"/>
<M4IsoOption name="diagram/show_spool_ref" valueType="boolean" value="f"/>
<M4IsoOption name="diagram/show_weld_ref" valueType="boolean" value="f"/>
```

This setting defines whether the ID reference is displayed on the isometric (value="t") or not (value="f").

### 11.10.3 Bend References

For the bend table only, the following settings can be used to identify the first and last segments in a bent pipeline:

```
<M4IsoOption name="core/bend_ref/first_ref" valueType="string" value="Start"/>
<M4IsoOption name="core/bend_ref/last_ref" valueType="string" value="End"/>
```

The following image shows the resulting bend table output:

Bend Ref	Adj X	Adj Y	Adj Z	Rad
1 Start	0.000	0.000	0.000	
1 2	0.000	-348.000	0.000	178.000
1 End	448.400	-348.000	0.000	
2 Start	0.000	0.000	0.000	
2 2	188.529	0.000	-108.847	178.000
2 End	836.229	0.000	-108.847	
3 Start	0.000	0.000	0.000	
3 2	-436.213	0.000	251.847	178.000
3 End	-940.703	0.000	251.847	

### 11.10.4 Item ID's

As well as the 4 common settings described earlier, the **item\_code\_ref** also has the following settings available:

```
<M4IsoOption name="core/item_code_ref/bolt_prefix"
               valueType="string" value="B:"/>
<M4IsoOption name="core/item_code_ref/gasket_prefix"
               valueType="string" value="G:"/>
<M4IsoOption name="core/item_code_ref/prefix_multiple_refs"
               valueType="boolean" value="t"/>
<M4IsoOption name="core/item_code_ref/format" valueType="string" value="auto"/>
```

The **bolt\_prefix** setting can be set with a value to identify bolt items, similarly the **gasket\_prefix** for gasket items.

The **prefix\_multiple\_refs** setting is used to switch on/off the display of the **bolt\_prefix** and **gasket\_prefix** values.

The **format** setting can have a value of either "auto" or "item\_code".

When set to "auto", the items in the isometric are ID'd according to the **sequence\_type** setting described earlier.

If set to "item\_code", then an item\_code associated with the item is used instead. This is particularly used where a SAP application may have defined a Creo part parameter to be used to identify individual items. This can be referenced using an envi-

renment setting:

**set M4ISOCREO\_RENUMBERING=SAP\_ID\_NUMBER**

In this example, the value assigned to the Creo parameter named SAP\_ID\_NUMBER will then be used as the part reference ID in the item details (partslist) table. Note that any parameter name can be used, as long as it contains unique part ID's for the parts in the pipeline assembly.

### 11.10.5 Spool reference padding

As well as the 4 common settings described earlier, the `spool_ref` also has the following settings available:

```
<M4IsoOption name="core/spool_ref/sequence_width" valueType="integer" value="2"/>
<M4IsoOption name="core/spool_ref/sequence_padding_char"
               valueType="string" value="0"/>
```

These settings control the number of digits to be displayed in the `spool_ref` and the padding character used to prefix the number when the actual number is less than the specified number of digits. For example, setting a width value of "3" and a padding character of "0", the spool reference number "5" would be shown as "005".

### 11.10.6 Annotation Border Style

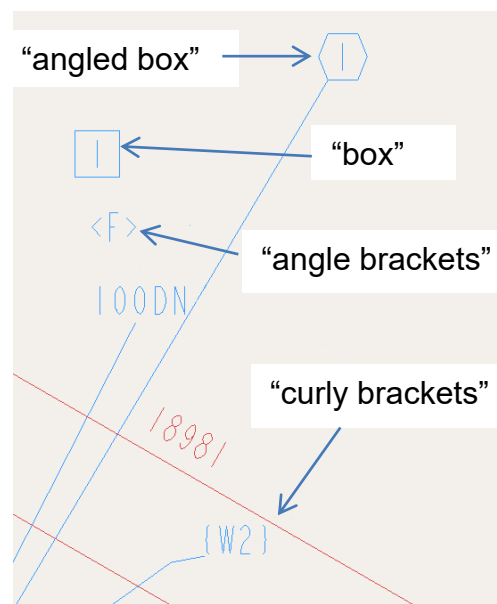
This setting is used on the `cut_ref`, `item_code_ref`, `spool_ref` and `weld_ref` items detailed above, but can also be used on other annotation settings such as **angle**, **bend\_radius**, **bore\_size**, **elevation**, **fall**, **name** and **note**

```
<M4IsoOption name="diagram/annotation/**/border_style"
               valueType="text border style" value=""/>
```

For the **border\_style** setting, the value can be set to one of:

- "angle brackets",
- "angled box",
- "box",
- "curly brackets",
- "none",
- "round box" or
- "square brackets".

If the value is not defined, it defaults to "none".



## 11.11 Tables

M4ISO has several tables that can be displayed on the isometric drawing if required. This section looks at each table in turn and describes how it can be configured and what information can be displayed in the table.

### 11.11.1 Parts List

To display a parts list (item\_details\_table) on the isometric drawing, there are a number of things to be considered:

- The position of the parts list on the sheet
- The number of columns in the parts list and their widths
- The appearance of the parts list (line colours, text height, etc)
- The data to be displayed in the parts list

To make the parts list table visible on the sheet use:

```
<M4IsoOption name="diagram/item_details_table/show_table"
valueType="boolean" value="t"/>
```

The position of the parts list is controlled using:

```
<M4IsoOption name="dars_illustrator/item_details_table/base_position"
valueType="coord2d" value="337.5 400.0"/>
```

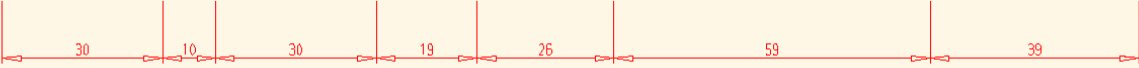
This marks the top left corner of the table.

The number of columns and their widths are defined using:

```
<M4IsoOption name="dars_illustrator/item_details_table/column_widths"
valueType="string" value="30 10 30 20 25 60 40"/>
```

The numeric values shown are the approximate width of each column in the local units (mm in this example).

5		6				
Group	Ref	Item Code	Item Total	Size	Type	Material
Fittings	1	BC0PH65K	21666 mm	50-50	TUBE	ASTMA105
Components	2	BC0OZMGO	11	50-50	90LR-ELBOW 50.0000	ASTMA105
	3	BC0OYU5K	3	50-50	SLIP-ON-FLANGE 50.0000	ASTMA105
	4	BC0P615K	1	50-50	STRAIGHT-TEE 50.0000	ASTMA105
	5	BC0P18JY	1	50-50	BALL-VALVE 50.0000	ASTMA105
Other	6	C4VW0K6Y	1	50-50	CAP	ASTMA105

The appearance of the parts list table is controlled using:

```
<M4IsoOption name="dars_illustrator/item_details_table/line/colour"
valueType="colour" value="0 0 0 255"/>
<M4IsoOption name="dars_illustrator/item_details_table/line/element_type"
valueType="string" value="L0"/>
<M4IsoOption name="dars_illustrator/item_details_table/line/thickness"
valueType="real" value="0"/>
<M4IsoOption name="dars_illustrator/item_details_table/text/colour"
valueType="colour" value="0 0 0 255"/>
<M4IsoOption name="dars_illustrator/item_details_table/text/element_type"
```



```

valueType="string" value="T1"/>
<M4IsoOption name="dars_illustrator/item_details_table/text/font_name"
valueType="string" value="Arial"/>
<M4IsoOption name="dars_illustrator/item_details_table/text/height"
valueType="real" value="2.5"/>

```

Note that on the 'valueType="colour"' objects, the 'value' has 4 numbers representing (in order) Red, Green, Blue and Alpha values in the range 0-255. In M4ISO, the Alpha value is always 255.

The final consideration of the parts list definition is the data to be displayed in the table.

First, we define the Groups to be displayed in the parts list table:

This must be changed in the '*M4IsoCoreSection*' definition and cannot be set in the '*M4IsoNamedSection*'.

```

<M4IsoOption name="core/item_code_table/component_type_groups"
valueType="string">Fittings,bend,pipe_segment;
Components,full_coupling,elbow,flange,reducer_concentric,
tee,valve,valve_angle,olet;
Gaskets,gasket;
Bolts,bolt;
Other
</M4IsoOption>

```

Each grouping is terminated by the semicolon (;) character. The first value in the grouping is the Group title that will appear in the parts list (e.g. 'Fittings'). That is then followed by the component types to be included in that group (e.g. bend, pipe\_segment).

Then we define the data to be displayed in the columns of the parts list table:

```

<M4IsoOption name="diagram/item_details_table/column_format"
valueType="string">Group:%$group/r$%;
Ref:%$item_code_ref$%:r;
Item Code:%$item_code$%;
Item Total:%$item_total$%:r;
Size:%size_1$%-%size_2$%;
Type:%fulltype$%;
Material;
</M4IsoOption>

```

Column Format						
Group	Ref	Item Code	Item Total	Size	Type	Material
Fittings						
	1	BC0PH65K	21666 mm	50-50	TUBE	ASTMA105
Components						
	2	BC0OZMGO	11	50-50	90LR-ELBOW 50.0000	ASTMA105
	3	BC0OYU5K	3	50-50	SLIP-ON-FLANGE 50.0000	ASTMA105
	4	BC0P615K	1	50-50	STRAIGHT-TEE 50.0000	ASTMA105
	5	BC0P18JY	1	50-50	BALL-VALVE 50.0000	ASTMA105
Other						
	6	C4VW0K6Y	1	50-50	CAP	ASTMA105

A column specification consists of:

- a literal heading
- a colon character (:)
- a value attribute reference
- optionally, a second colon and a justification letter
  - 'L' for left,

- 'C' for centre,
- 'R' for right),
- a terminating semicolon character (;)

For example: Item Total: %Item\_total\$:r;

If only the heading is given, then the column shows the value of the attribute with the same name as the heading.

For example: **Material;** is equivalent to **Material: %material%;**

Note: If you do not wish to have the parts list data grouped, then simply omit the '**Group: %\$group/r\$%;**' entry from the **column\_format** definition.

### Common Item Details Attributes

A value specification for the Item Details table consists of literal text strings and specific named attributes enclosed in '%' characters. The common Item Details attributes are listed below:

%\$group\$%	group name, not on a separate row
%\$group/r\$%	group name on a separate row
%\$item_code_ref\$%	item code reference used on the diagram.
%\$item_code\$%	item code for item.
%\$item_count\$%	number of items. Blank if item is measured by length.
%\$item_length\$%	total length of item. Blank if item is counted
%\$item_total\$%	Number of items if item is counted. Length of item if measured by length.
%\$pipe_name\$%	pipe name.
%\$pipe_spec\$%	pipe specification.
%\$item_nominal_length\$%	item length not including any cut allowance
%\$item_cut_allowance\$%	item cut allowance
%\$item_nominal_total\$%	total length of item not including cut allowances

### Attributes for Creo

The following attribute names are available by default in M4ISO for Creo:

%assembly%	From the <i>Assembly Name</i> parameter
%component%	From the <i>Component Name</i> parameter
%code%	From the <i>Material Code</i> parameter
%mccs%	From the <i>MCCS</i> parameter
%specification%	From the <i>Pipe Spec</i> parameter
%description%	From the <i>Material Desc</i> parameter
%size_1%	Bore size of component
%size_2%	Second bore size if different from first
%Type%	Only for pipes, set to "TUBE"
%diameter%	Bolt diameter
%length%	Bolt length

### Mapped Attributes

Any Creo parameter can be mapped to a M4ISO attribute with a target of ItemDetails. That attribute can then be referenced in the ItemDetails table definition by enclosing the attribute name in % characters. For example:

In the isoreal\_parameter\_map.ptd file add the setting:

**"PTC\_COMMON\_NAME" "Part" "ptc\_description" "ItemDetails"**

In the M4IsoOptionStore.xml file use the **ptc\_description** attribute in the Type column of the Item Details table definition:

```
<M4IsoOption name="diagram/item_details_table/column_format"
  valueType="string">Group:%$group/r$%;
                        Ref:%$item_code_ref$%:r;
                        Item Code:%$item_code$%;
                        Item Total:%$item_total$%:r;
                        Size:%size_1%-%size_2%;
                        Type:%ptc_description%;
                        Material;
</M4IsoOption>
```

### Exclude Component from Item Details Table

This setting is used to identify a Creo parameter that can be used to exclude a component from the Item Details table.

You must map the parameter in Creo (SAPUSE) to an attribute in M4ISO (M4ISO\_SAPUSE) using the isoreal\_parameter\_map.ptd file. For example:

**"SAPUSE" "Component" "M4ISO\_SAPUSE" "PipeComponent"**

The target name PipeComponent ensures the attribute is assigned to the component entry in the ISOX file.

Then in the M4IsoOptionStore, identify the attribute using this setting:

```
<M4IsoOption name="diagram/item_details_table/in_MRP_var"
  valueType="string" value="M4ISO_SAPUSE"/>
```

If the attribute has a value of "N" then the component is excluded.

If the attribute does not exist or is not set to "N", then it is included.

Note that there is a slight subtlety for the Item Details table. Namely, if it is only added to 1 component, and there are N components of the same part in the assembly, then the Parts list shows N-1 items in the table.

The following setting controls whether any excluded component is drawn in "ghost" fashion using (normally) a dotted line.

```
<M4IsoOption name="diagram/geometry/show_excluded_as_ghost"
  valueType="boolean" value="t"/>
```

If set to "f" then the excluded component symbol is drawn normally with solid line.

### Automatic Line Wrap in Tables

This setting enables automatic line wrapping when data becomes too large to fit in the column space allocated.

```
<M4IsoOption name="dars_illustrator/table/auto_wrap_columns"
  valueType="boolean" value="t"/>
```

The units for text/margin, text/space and max\_height are read from the current assembly units (mm or inch – mm are shown here).

```
<M4IsoOption name="dars_illustrator/<table>/text/margin"
  valueType="real" value="0.5"/>
```

```

<M4IsoOption name="dars_illustrator/<table>/text/spacing"
               valueType="real" value="1.0"/>
<M4IsoOption name="dars_illustrator/<table>/max_height"
               valueType="real" value="500.0"/>
<M4IsoOption name="dars_illustrator/<table>/max_rows"
               valueType="integer" value="-1"/>

```

Where <table> is one of:

- bend\_table
- cuts\_table
- item\_details\_table
- spools\_table
- welds\_table

The **max\_height** setting controls the physical height of the table before a new table is created. The value specified is the height of the table in the local units. If this setting is used, the **max\_rows** value must be set to "-1".

The **text/margin** setting is used to define a margin above and below the text as it lies inside the row. The value is applied above the text AND below the text.

The **text/spacing** setting is used to define the space between wrapped lines of the text. The value is a size in the local units.

Examples:

In these examples, the text has a box drawn around it to show the effect of the margin and spacing settings. Note that the text naturally has a border around the text, giving a line size of 6.67mm for a single line of 4mm high text. This is not included in the margin/spacing settings.

4mm high text, spacing=0, margin=0

2	16	311AB40150	BOLT
3	18	33AA20050	BOLT
4	18	33AA30050	BOLT
5	2	ELBOW_90_BW	90 LR
		-STEEL-L-50	A234
6	5	ELBOW_90_BW	90 S E
		-STEEL-S-25	GR V

4mm high text, spacing=0, margin=3

2	16	311AB40150	BOLT
3	18	33AA20050	BOLT
4	18	33AA30050	BOLT
5	2	ELBOW_90_BW -STEEL-L-50	90 LR A234
6	5	ELBOW_90_BW -STEEL-S-25	90 S E GR V

The margin is applied above and below the text in the row.

4mm high text, spacing=3, margin=0

2	16	311AB40150	BOLT
3	18	33AA20050	BOLT
4	18	33AA30050	BOLT
5	2	ELBOW_90_BW -STEEL-L-50	90 LR A234
6	5	ELBOW_90_BW -STEEL-S-25	90 S E GR V

The spacing is applied between wrapped lines.

### Attribute Sorting

The contents of the Item\_Details\_table can be sorted based on a parameter value assigned to the parts in the pipe assembly. The example shown below is for SHOP or FIELD fabricated parts, but any type of grouping could be used.

The grouping is controlled by the value assigned to a parameter in Creo. The parameter then must be mapped to an attribute that can be referenced in the M4ISO program.

For simplicity, let us assume there is a Creo parameter named FAB\_TYPE which can have a value of either SHOP or FIELD.

In the isoreal\_parameter\_map.ptd file, map the FAB\_TYPE parameter as follows:  
"FAB\_TYPE" "Component" "fabtype" "PipeComponent"

In the M4IsoOptionStore.xml file, we then define the attribute name that we are using for the grouping. Note this is a core setting which can only be defined in the <M4IsoDefaultSection>.

```
<M4IsoOption name="core/item_code_table/component_attribute_groups_name"
               valueType="string" value="fabtype"/>
```

This setting picks up the fabtype attribute as the group name attribute.

We now need to know what values to sort the grouping on. We do this using the component\_attribute\_groups setting.

In the example below, we have a grouping title of SHOP FABRICATION assigned using the **fabtype** value of SHOP. A second grouping is titled FIELD FABRICATION which is assigned using the **fabtype** value of FIELD. Finally, the **-default** setting assumes any components that do not have a **fabtype** attribute (or have a value other than SHOP or FIELD) will be assigned to the SHOP FABRICATION group.

```
<M4IsoOption name="core/item_code_table/component_attribute_groups"
               valueType="string">
    SHOP FABRICATION,SHOP;
    FIELD FABRICATION,FIELD;
    -default,SHOP;
</M4IsoOption>
```

Finally, we use a copy of the item\_details\_table/column\_format setting to identify where the attribute group title name appears in the table. In this example, it will appear in the Component column.

```
<M4IsoOption name="diagram/item_details_table/attrib_group_column_format"
               valueType="string">
    Group;;
    Ref;;
    Item Total;;
    Component:;%$attribgroup#r$%:c;
    Type;;
    Size;;
</M4IsoOption>
```

Notice how only the component column has an attribute set; the other columns just have their titles. The titles are used to identify them and must be set.

The result will look like this:

Group	Ref	Item Total	SHOP FABRICATION Component	Type	Size
Pipes		1	21454.00 mm	TUBE	WT=3.9mm PIPE, PE SEAMLESS TO API 5L GRADE B
Fittings		2	11	ELBOW_90_BW-STEEL-L-50	90 LR ELBOW, BUTT-WELD SCH 40 STEEL A234 GR WPB CS
		3	1	TEE_STR_BW-STEEL-50X50	TEE, BUTT-WELD STRAIGHT STEEL ASTM A234 GRADE WPB MATCH PIPE
Flanges		4	2	FLANGE_NECK_RF-STEEL-30K-50	FLANGE, RAISED FACE WELD-NECK STEEL ASTM A105
		5	6	FLANGE_SLIP_RF-STEEL-20K-50	FLANGE, SLIP-ON STEEL ASTM A105
Valves		6	3	VL_GATE_FLRF-STEEL-20K-50	GATE VALVE, FLANGED RAISED FACE STEEL ASTM A105
Gaskets		7	6	GASKET_SLIP-STEEL-20K-50	GASKET, 1.5MM FLAT RING CAF GRAPHITED BOTH SIDES
Bolts		8	16	311AB40050	BOLT NUT ASTM C S 106
		9	72	33AA30050	BOLT NUT ASTM C S 111
Group	Ref	Item Total	FIELD FABRICATION Component	Type	Size
Pipes		1	2314.00 mm	TUBE	WT=3.9mm PIPE, PE SEAMLESS TO API 5L GRADE B
Fittings		2	3	ELBOW_90_BW-STEEL-L-50	90 LR ELBOW, BUTT-WELD SCH 40 STEEL A234 GR WPB CS
		3	2	TEE_STR_BW-STEEL-50X50	TEE, BUTT-WELD STRAIGHT STEEL ASTM A234 GRADE WPB MATCH PIPE
Flanges		4	4	FLANGE_NECK_RF-STEEL-30K-50	FLANGE, RAISED FACE WELD-NECK STEEL ASTM A105
		5	2	FLANGE_SLIP_RF-STEEL-20K-50	FLANGE, SLIP-ON STEEL ASTM A105
Valves		6	2	VL_GATE_FLRF-STEEL-20K-50	GATE VALVE, FLANGED RAISED FACE STEEL ASTM A105
Gaskets		7	4	GASKET_SLIP-STEEL-20K-50	GASKET, 1.5MM FLAT RING CAF GRAPHITED BOTH SIDES
Bolts		8	16	311AB40050	BOLT NUT ASTM C S 106
		9	72	33AA30050	BOLT NUT ASTM C S 111

If required, the setting below can be used to force all nuts and bolts items to appear in a particular section of the partslist (for example as Field items):

**<M4IsoOption name="core/item\_code\_table/component\_attribute\_bolt\_group" value="FIELD" valueType="string" value="FIELD"/>**

Note that this is a core setting, so it can only go into the core section of the M4IsoDefaults block. You can't add it into the Style definitions.

With the setting above active you should see the nuts and bolts appear in the FIELD section of the partslist.

### Subassembly parts listing

Some component parts in a Creo pipeline can be assembly parts containing several different child parts. If required, it is possible to list the child parts in the parts list as indented references of the assembly. This functionality was originally developed to list the contents of hanger support assemblies.

To achieve this, the parent assembly and the child parts must be identified using a common Creo parameter name. The parameter name can be any unique, valid Creo parameter name. In the examples below, the parameter name PARENT\_CHILD will be used.

For the Assembly component, the PARENT\_CHILD parameter must be given the value "PARENT".

For each child part, the PARENT\_CHILD parameter must be given the name of the assembly it belongs to.

To identify the child parts in each assembly, the M4ISO software matches the name used in the PARENT\_CHILD parameter with an assembly that has the "PARENT" value.

Associated with this functionality, there are 3 environment variables.

The first one must be used to enable the functionality:

**M4ISOCREO\_HANGER\_EXTENSION=YES**

The second environment variable is used to identify the Creo parameter name that is used to match the child parts with their parent assembly:

**M4ISOCREO\_HANGER\_PARAM\_ID=PARENT\_CHILD**

Finally, there is an environment variable to set a tolerance whereby assemblies sitting on (or near) a pipe centreline can be identified:

**M4ISOCREO\_HANGER\_TOLERANCE=0.5**

The value given for the tolerance is in the local model units.

To display the child part details in the ItemDetailsTable, we must modify the M4IsoOptionStore.xml file to add the sub\_column\_format details.

For the assembly and its child parts, the sub\_column\_format setting defines where the child part attributes appear in the table:

```
<M4IsoOption name="diagram/item_details_table/sub_column_format"
                                     valueType="string">
    Group;;
    Ref;;
    Item Total;;
    Component;
    Type:%Type% %description%;
    Size;;
</M4IsoOption>
```

This setting indicates that for each child part of the assembly, only the TYPE column is defined using the attributes %Type% %description%

You can add other attributes into other columns as required.

Note that the child parts do not have their own part number – they take the value from their parent assembly.

### 11.11.2 Spools Table

The definition for a Spools Table is created in a similar manner to the Parts List. There are several things to be considered:

- The position of the Spool Table on the sheet



- The number of columns in the Spool Table and their widths
- The appearance of the Spool Table (line colours, text height, etc)
- The data to be displayed in the Spool Table

To make the spools table visible on the sheet use:

```
<M4IsoOption name="diagram/spools_table/show_table"
  valueType="boolean" value="t"/>
<M4IsoOption name="diagram/show_spool_ref"
  valueType="boolean" value="t"/>
```

The position of the spools table is controlled using:

```
<M4IsoOption name="dars_illustrator/spools_table/base_position"
  valueType="coord2d" value="390 250"/>
```

The appearance of the spools table is controlled using:

```
<M4IsoOption name="dars_illustrator/spools_table/layer"
  valueType="integer" value="56"/>
<M4IsoOption name="dars_illustrator/spools_table/line/colour"
  valueType="colour" value="0 0 0 255"/>
<M4IsoOption name="dars_illustrator/spools_table/line/element_type"
  valueType="string" value="L0"/>
<M4IsoOption name="dars_illustrator/spools_table/line/thickness"
  valueType="real" value="0"/>
<M4IsoOption name="dars_illustrator/spools_table/max_rows"
  valueType="integer" value="-1"/>
<M4IsoOption name="dars_illustrator/spools_table/text/colour"
  valueType="colour" value="0 0 0 255"/>
<M4IsoOption name="dars_illustrator/spools_table/text/element_type"
  valueType="string" value="T1"/>
<M4IsoOption name="dars_illustrator/spools_table/text/font_name"
  valueType="string" value="Arial"/>
<M4IsoOption name="dars_illustrator/spools_table/text/height"
  valueType="real" value="4"/>
```

The contents of the spools table are controlled using:

```
<M4IsoOption name="dars_illustrator/spools_table/column_widths"
  valueType="string" value=""/>
<M4IsoOption name="diagram/spools_table/column_format"
  valueType="string" value=""/>
<M4IsoOption name="diagram/spools_table/show_headings"
  valueType="boolean" value="t"/>
```

## Spools table attributes

\$spool\_ref\$ - spool reference used on the diagram.

\$pipe\_name\$ - pipe name.

\$pipe\_spec\$ - pipe spec for spool.

This example spool table format:

```
<M4IsoOption name="diagram/spools_table/column_format" valueType="string"
  value=" :%$spool_ref$%;
        ;
        :%$pipe_name$%/%$spool_ref$%;
        ;
        Spools:Pipe spool %$spool_ref$% to specification %$spool_spec$%;" />
```

Results in the following spool table layout:

			Spools
S-1-P		PIPE2/S-1-P	Pipe spool S-1-P to specification ASTM A105
S-2-P		PIPE2/S-2-P	Pipe spool S-2-P to specification ASTM A105

Note: Spools are controlled using M4IsoConnectionDefinition settings in the M4IsoSymbolInfo.xml file

### 11.11.3 Cuts Table

In a similar manner, a Cuts Table can be added to the drawing.

There are several things to be considered:

- The position of the Cuts Table on the sheet
- The number of columns in the Cuts Table and their widths
- The appearance of the Cuts Table (line colours, text height, etc)
- The data to be displayed in the Cuts Table

To make the cuts table visible on the sheet use:

```
<M4IsoOption name="diagram/cuts_table/show_table"
                    valueType="boolean" value="t"/>
<M4IsoOption name="diagram/show_cut_ref"
                    valueType="boolean" value="t"/>
```

The position of the cuts table is controlled using:

```
<M4IsoOption name="dars_illustrator/cuts_table/base_position"
                    valueType="coord2d" value="390 125"/>
```

The appearance of the cuts table is controlled using:

```
<M4IsoOption name="dars_illustrator/cuts_table/layer"
                    valueType="integer" value="54"/>
<M4IsoOption name="dars_illustrator/cuts_table/line/colour"
                    valueType="colour" value="0 0 0 255"/>
<M4IsoOption name="dars_illustrator/cuts_table/line/element_type"
                    valueType="string" value="L0"/>
<M4IsoOption name="dars_illustrator/cuts_table/line/thickness"
                    valueType="real" value="0"/>
<M4IsoOption name="dars_illustrator/cuts_table/max_rows"
                    valueType="integer" value="-1"/>
<M4IsoOption name="dars_illustrator/cuts_table/text/colour"
                    valueType="colour" value="0 0 0 255"/>
<M4IsoOption name="dars_illustrator/cuts_table/text/element_type"
                    valueType="string" value="T1"/>
<M4IsoOption name="dars_illustrator/cuts_table/text/font_name"
                    valueType="string" value="Arial"/>
<M4IsoOption name="dars_illustrator/cuts_table/text/height"
                    valueType="real" value="4"/>
```

### Cut Table Attributes

\$cut\_ref\$ - cut reference used on diagram.

\$cut\_length\$ - total length of cut.

\$end\_prep\$ - list of all end preparations, separated by commas.

\$end\_prep/F\$ - first end preparation.

\$end\_prep/L\$ - last end preparation. If there is only one, show "(!)".

If there are more than two, show "(\*)" followed by the last end prep.

%%pipe\_name%% - pipe name.

%%pipe\_spec%% - pipe specification

The contents of the cuts table are controlled using:

```

<M4IsoOption name="dars_illustrator/cuts_table/column_widths"
               valueType="string" value="15 25 50"/>
<M4IsoOption name="diagram/cuts_table/column_format"
               valueType="string" value="Ref:%%$cut_ref$%;
               Length:%%$cut_length$%;
               End Prep:%%$end_prep/F$% - %%$end_prep/L$%;"/>
<M4IsoOption name="diagram/cuts_table/show_headings"
               valueType="boolean" value="t"/>

```

The above settings would result in the following table layout:

Ref	Length	End Prep
1	423	bevelled - bevelled
2	1058	bevelled – bevelled

Note: Cuts are controlled using M4IsoConnectionType settings in the M4IsoSymbolInfo.xml file.

### Cut Start label

```

<M4IsoOption name="diagram/annotation/show_cut_start"
               valueType="boolean" value="f"/>

```

This setting is used to show a label to indicate where the cut piece starts from. It also requires that the Creo parameter M4ISO\_STARTBEND is set on one of the pipe segments/bends in the cut. If this parameter is not set, then the label will not display.

It uses the same style as a Note.

### 11.11.4 Welds Table

In a similar manner, a Welds Table can be added to the drawing.

There are several things to be considered:

- The position of the Welds Table on the sheet
- The number of columns in the Welds Table and their widths
- The appearance of the Welds Table (line colours, text height, etc)
- The data to be displayed in the Welds Table

To make the Welds table visible on the sheet use:

```
<M4IsoOption name="diagram/welds_table/show_table"
  valueType="boolean" value="t"/>
<M4IsoOption name="diagram/show_weld_ref"
  valueType="boolean" value="t"/>
```

The position of the Welds table is controlled using:

```
<M4IsoOption name="dars_illustrator/welds_table/base_position"
  valueType="coord2d" value="500 160"/>
```

The appearance of the Welds table is controlled using:

```
<M4IsoOption name="dars_illustrator/welds_table/layer"
  valueType="integer" value="57"/>
<M4IsoOption name="dars_illustrator/welds_table/line/colour"
  valueType="colour" value="0 0 0 255"/>
<M4IsoOption name="dars_illustrator/welds_table/line/element_type"
  valueType="string" value="L0"/>
<M4IsoOption name="dars_illustrator/welds_table/line/thickness"
  valueType="real" value="0"/>
<M4IsoOption name="dars_illustrator/welds_table/max_rows"
  valueType="integer" value="-1"/>
<M4IsoOption name="dars_illustrator/welds_table/text/colour"
  valueType="colour" value="0 0 0 255"/>
<M4IsoOption name="dars_illustrator/welds_table/text/element_type"
  valueType="string" value="T1"/>
<M4IsoOption name="dars_illustrator/welds_table/text/font_name"
  valueType="string" value="Arial"/>
<M4IsoOption name="dars_illustrator/welds_table/text/height"
  valueType="real" value="4"/>

<M4IsoOption name="diagram/welds_table/column_format"
  valueType="string" value="Weld:%$weld_ref$%;
  Welder: ;
  N.D.T.: ;"/>
```

## Welds Table Attributes

\$weld\_ref\$ - Weld reference used on diagram.

The contents of the Welds table are controlled using:

```
<M4IsoOption name="dars_illustrator/welds_table/column_widths"
               valueType="string" value="20 30 30"/>
<M4IsoOption name="diagram/welds_table/column_format"
               valueType="string"
               value="Weld:%$weld_ref$%;
                    Welder: ;
                    N.D.T.: ;"/>
<M4IsoOption name="diagram/welds_table/show_headings"
               valueType="boolean" value="t"/>
```

The above settings would result in the following table layout:

Weld	Welder	N.D.T.
1		
2		

## Mapped Attributes

Any Creo parameter can be mapped to a M4ISO attribute with a target of Weld. That attribute can then be referenced in the Weld table definition by enclosing the attribute name in % characters. For example:

In the isoreal\_parameter\_map.ptd file add the setting:

```
"WELD_TYPE" "Part" "WELDTYPE" "Weld"
```

In the M4IsoOptionStore.xml file use the **WELDTYPE** attribute in the Weld table definition:

```
<M4IsoOption name="dars_illustrator/welds_table/column_widths"
               valueType="string" value="20 30 100 50"/>
<M4IsoOption name="diagram/welds_table/column_format"
               valueType="string"
               value="Weld:%$weld_ref$%;
                    Welder: ;
                    N.D.T.: ;
                    Weld Type:%WELDTYPE%;"/>
<M4IsoOption name="diagram/welds_table/show_headings"
               valueType="boolean" value="t"/>
```

Weld	Welder	N.D.T.	Weld Type
1			Butt Weld
2			Socket Weld

Note:

See section 11.17 for further details on welds in Creo

### 11.11.5 Bends Table

In a similar manner, a Bends Table can be added to the drawing. There are several things to be considered:

- The position of the Bends Table on the sheet
- The number of columns in the Bends Table and their widths
- The appearance of the Bends Table (line colours, text height, etc)
- The data to be displayed in the Bends Table

To make the Bends table visible on the sheet use:

```
<M4IsoOption name="diagram/bends_table/show_table"
  valueType="boolean" value="t"/>
```

The position of the Bends table is controlled using:

```
<M4IsoOption name="dars_illustrator/bends_table/base_position"
  valueType="coord2d" value="500 160"/>
```

The appearance of the Bends table is controlled using:

```
<M4IsoOption name="dars_illustrator/bends_table/layer"
  valueType="integer" value="54"/>
<M4IsoOption name="dars_illustrator/bends_table/line/colour"
  valueType="colour" value="0 0 0 255"/>
<M4IsoOption name="dars_illustrator/bends_table/line/element_type"
  valueType="string" value="L0"/>
<M4IsoOption name="dars_illustrator/bends_table/line/thickness"
  valueType="real" value="0"/>
<M4IsoOption name="dars_illustrator/bends_table/max_rows"
  valueType="integer" value="-1"/>
<M4IsoOption name="dars_illustrator/bends_table/text/colour"
  valueType="colour" value="0 0 0 255"/>
<M4IsoOption name="dars_illustrator/bends_table/text/element_type"
  valueType="string" value="T1"/>
<M4IsoOption name="dars_illustrator/bends_table/text/font_name"
  valueType="string" value="Arial"/>
<M4IsoOption name="dars_illustrator/bends_table/text/height"
  valueType="real" value="4"/>
```

### Bends Table Attributes

\$pipe\_name\$ - The name of the pipe.

\$pipe\_spec\$ - The spec name of the pipe.

\$cut\_ref\$ - The cut reference of the cut pipe containing the bend

\$bend\_ref\$ - The bend reference.

The following attributes can have the "/u" switch added before the trailing "\$" sign. Then a unit value will be appended to the output. (e.g. "8.1" becomes "8.1 mm" with the /u switch added). The unit value displayed depends on the "diagram/display\_position\_units" setting in the M4ISOOptionStore file (unless specified otherwise below).

\$adjusted_length\$	the adjusted length of a pipe segment including any cut allowance
---------------------	-------------------------------------------------------------------

\$bend_radius\$	The bend radius in units based on the "diagram/display_length_units" setting.
\$cut_allowance\$	The cut allowance value added to a pipe segment
\$nominal_length\$	The nominal length of a pipe segment
\$nominal_position\$	The nominal position of the bend.
\$nominal_position_x\$	The x coordinate of the nominal position.
\$nominal_position_y\$	The y coordinate of the nominal position.
\$nominal_position_z\$	The z coordinate of the nominal position.
\$relative_adjusted_position\$	The position of the bend, relative to the start, including the cut allowances.
\$relative_adjusted_position_x\$	The x coordinate of the relative adjusted position.
\$relative_adjusted_position_y\$	The y coordinate of the relative adjusted position.
\$relative_adjusted_position_z\$	The z coordinate of the relative adjusted position.
\$adjusted_position\$	The position of the bend, relative to the diagram origin, including the cut allowances.
\$adjusted_position_x\$	The x coordinate of the adjusted position.
\$adjusted_position_y\$	The y coordinate of the adjusted position.
\$adjusted_position_z\$	The z coordinate of the adjusted position.

The contents of the Bends table are controlled using:

```

<M4IsoOption name="dars_illustrator/bends_table/column_widths"
                valueType="string" value=""/>
<M4IsoOption name="diagram/bends_table/column_format"
                valueType="string">Bend Ref: %$pipe_name$%_%$cut_ref$%_%$bend_ref$%;
                                Rel Adj X: %$relative_adjusted_position_x$%;
                                Rel Adj Y: %$relative_adjusted_position_y$%;
                                Rel Adj Z: %$relative_adjusted_position_z$/u$%;
                                Radius: %$bend_radius/u$%;
                                </M4IsoOption>
<M4IsoOption name="diagram/bends_table/show_headings"
                valueType="boolean" value="t"/>
<M4IsoOption name="core/bend_ref/custom_sequence_symbols"
                valueType="string" value=""/>
<M4IsoOption name="core/bend_ref/first_ref" valueType="string" value="START"/>
<M4IsoOption name="core/bend_ref/last_ref" valueType="string" value="END"/>
<M4IsoOption name="core/bend_ref/prefix" valueType="string" value=""/>
<M4IsoOption name="core/bend_ref/sequence_type"
                valueType="string" value="numeric"/>
<M4IsoOption name="core/bend_ref/suffix" valueType="string" value=""/>

```

The above settings would result in the following table layout:

Bend Ref	Rel Adj X	Rel Adj Y	Rel Adj Z	Radius
L99-50-A3A_1_START	0	0	0 mm	
L99-50-A3A_1_END	0	624	0 mm	
L99-50-A3A_2_START	0	0	0 mm	
L99-50-A3A_2_2	0	0	424 mm	150 mm
L99-50-A3A_2_3	700	0	424 mm	200 mm
L99-50-A3A_2_4	700	0	-276 mm	300 mm
L99-50-A3A_2_END	700	300	-276 mm	

### Bend data output to file

The bend data written to the Bend table can also be output to a CSV file by using the setting:



```
<M4IsoOption name="diagram/bends_table/save_as_csv"
              valueType="boolean" value="t"/>
```

### Bend data output to NC file

The Bend data can also be output to a file in order to drive an NC machine using the following settings:

```
<M4IsoOption name="diagram/bends_table/nc_template_file"
              valueType="string" value="NCTemplate.txt"/>
<M4IsoOption name="diagram/bends_table/save_for_nc"
              valueType="boolean" value="t"/>
```

The format of the output for the NC machine is controlled by the commands in the **nc\_template\_file**.

The file must contain 3 sections marked using the keywords:

```
$$$ HEADER $$$
$$$ BODY $$$
$$$ FOOTER $$$
```

Comments in the file are identified with a # character. If the # character is to be used in the NC output, it should be preceded with a \ character like so: \#.

The data in the NC output is created using any of the attributes defined for the item details. Also, there are a set of bend attributes associated with each section of the output file.

For the **\$\$\$ HEADER \$\$\$** section the available attributes are:

%%\$assy_name\$%	<i>The Creo assembly name</i>
%%\$bend_count\$%	<i>Number of bends in cut.</i>
%%\$cut_ref\$%	<i>The cut reference id</i>
%%\$cut_length\$%	<i>The length of the cut (including cutting allowances)</i>
%%\$date\$%	<i>The current date</i>
%%\$iso_name\$%	<i>The isometric drawing name</i>
%%\$pipe_name\$%	<i>The pipeline name</i>
%%\$pipe_spec\$%	<i>The pipe spec name</i>
%%\$spool_name\$%	<i>The Creo spool name</i>
%%\$straight_count\$%	<i>Number of straights in cut.</i>
%%\$time\$%	<i>The current time</i>
%%<attribute>\$%	<i>The name of any attribute in the item details associated with the pipe</i>

In the **\$\$\$ BODY \$\$\$** section, the following attributes are available:

%%\$adjusted_cp1\$%	Position of CP1, adjusted for cut allowance, in the form "(x, y, z)".
%%\$adjusted_cp1_x\$%	
%%\$adjusted_cp1_y\$%	
%%\$adjusted_cp1_z\$%	
%%\$adjusted_cp2\$%	Position of CP2, adjusted for cut allowance, in the

	form "(x, y, z)".
%%\$adjusted_cp2_x\$%	
%%\$adjusted_cp2_y\$%	
%%\$adjusted_cp2_z\$%	
%%\$adjusted_datum\$%	Position of bend datum, adjusted for cut allowance, in the form "(x, y, z)".
%%\$adjusted_datum_x\$%	
%%\$adjusted_datum_y\$%	
%%\$adjusted_datum_z\$%	
%%\$adjusted_dim_length\$%	Distance from datum to previous datum, adjusted for cut allowance.
%%\$adjusted_seg_length\$%	Distance along pipe centreline from centre of bend to centre of previous bend, adjusted for cut allowance.
%%\$adjusted_straight_length\$%	Distance along straight section of pipe, adjusted for cut allowance.
%%\$arc_length\$%	Length of bend along centreline.
%%\$arm_length\$%	Distance from datum to start of straight section.
%%\$assy_name\$%	
%%\$bend_angle\$%	
%%\$bend_radius\$%	
%%\$cut_ref\$%	
%%\$date\$%	
%%\$iso_name\$%	
%%\$nominal_cp1\$%	
%%\$nominal_cp1_x\$%	
%%\$nominal_cp1_y\$%	
%%\$nominal_cp1_z\$%	
%%\$nominal_cp2\$%	
%%\$nominal_cp2_x\$%	
%%\$nominal_cp2_y\$%	
%%\$nominal_cp2_z\$%	
%%\$nominal_datum\$%	
%%\$nominal_datum_x\$%	
%%\$nominal_datum_y\$%	
%%\$nominal_datum_z\$%	
%%\$nominal_dim_length\$%	Nominal distance from datum to previous datum.
%%\$nominal_seg_length\$%	Nominal distance along pipe centreline from centre of bend to centre of previous bend.
%%\$nominal_straight_length\$%	Nominal length of straight section of pipe.
%%\$pipe_name\$%	
%%\$pipe_spec\$%	
%%\$rotation_angle\$%	
%%\$spool_name\$%	
%%\$time\$%	
%%<attribute>\$%	The name of any attribute in the item details associated with the pipe

Angles are in degrees. Lengths and positions are in the length units of the bend table.

Finally, the **\$\$\$ FOOTER \$\$\$** section has the following attributes available:

%%\$adjusted_cp\$%	Position of the end of the last straight section, ad-
--------------------	-------------------------------------------------------

	justed for cut allowance, in the form (x, y, z).
%%\$adjusted_cp_x\$%	
%%\$adjusted_cp_y\$%	
%%\$adjusted_cp_z\$%	
%%\$adjusted_dim_length\$%	Distance from end of last straight section to previous datum, adjusted for cut allowance.
%%\$adjusted_seg_length\$%	Distance along pipe centreline from end of last straight section to centre of previous bend, adjusted for cut allowance.
%%\$adjusted_straight_length\$%	Length of last straight section of pipe, adjusted for cut allowance.
%%\$assy_name\$%	
%%\$cut_ref\$%	
%%\$date\$%	
%%\$iso_name\$%	
%%\$nominal_cp\$%	Nominal position of the end of the last straight section, in the form "(x, y, z)".
%%\$nominal_cp_x\$%	
%%\$nominal_cp_y\$%	
%%\$nominal_cp_z\$%	
%%\$nominal_dim_length\$%	Nominal distance from end of last straight section to previous datum.
%%\$nominal_seg_length\$%	Nominal distance along pipe centreline from end of last straight section to centre of previous bend.
%%\$nominal_straight_length\$%	Nominal length of last straight section of pipe.
%%\$pipe_name\$%	
%%\$pipe_spec\$%	
%%\$spool_name\$%	
%%\$time\$%	
%%<attribute>%	The name of any attribute in the item details associated with the pipe.

Formatting of the attribute values can be achieved using { } with attribute references.

Real values: {10.3} = for numbers only; a width of 10 characters with 3 decimal places  
' 123.456'

Integer values: {6} = a width of 6 characters ' 456'

Padded values: {X5} = a width of 5 characters with preceding spaces filled with the specified character (X in this example) 'XXX45'

Justification: normally right justified (positive), use negative values for left justified  
{-5} = '12 '

The formatting only works on attributes and the format must immediately follow the first % character:

%%{10.3}cut\_length%, %%{6.2}\$bend\_angle\$%

Any non-comment lines in any section will be output as literally written with any attributes interpreted according to entered formats.

An example NC template file could look like this:

```
# Template NC Bend File
$$$ HEADER $$$
\# NC Bend File generated on %%date$% %%time$%
\# Isometric: %%iso_name$%
\# Outside Diameter=%%OutsideDiameter%, wall Thickness: %%wallthickness%
\# Cut ref: %%{03}$cut_ref$%
\# Cut Length: %%$cut_length$%
```

```

\# No of Bends: %$bend_count$, No of Straights: %$straight_count$
\#
\# AdjX pos, AdjY pos, AdjZ pos, Adj Len | Angle, Radius, Rotation
#
# 0.000, 0.000, 0.000, 0.000 | 0.00, 0.000, 0.00
#
$$$ BODY $$$
    %{8.3}$adjusted_datum_x$, %{8.3}$adjusted_datum_y$, %{8.3}$adjusted_datum_z$,
    %{8.3}$adjusted_straight_length$ | %{6.2}$bend_angle$, %{7.3}$bend_radius$,
    %{6.2}$rotation_angle$
#
$$$ FOOTER $$$
    %{8.3}$adjusted_cp_x$, %{8.3}$adjusted_cp_y$, %{8.3}$adjusted_cp_z$,
    %{8.3}$adjusted_straight_length$ | 0.00, 0.000, 0.00
\#
\# Completed on: %$Date$ %$Time$
\# Pipe: %$pipe_name$_%{03}$cut_ref$
#
#EOF

```

Note: in the above example, the width of the page has caused some lines to be wrapped. Those **highlighted** should be on a single line.

Typically, the commands above would produce an output file that looked like this:

```

# NC Bend File generated on 19 10 2022 13:26:21
# Isometric: ISO_HV_CONN
# Outside Diameter=25.4, wall Thickness: 2.3
# Cut ref: 001
# Cut Length: 5764.646
# No of Bends: 7, No of Straights: 8
#
# AdjX pos, AdjY pos, AdjZ pos, Adj Len | Angle, Radius, Rotation
# 0.000, 0.000, 0.000, 0.000 | 0.00, 0.000, 0.00
# -230.000, 0.000, 0.000, 150.000 | 90.00, 80.000, 0.00
# -230.000, 1400.000, 0.000, 1240.000 | 90.00, 80.000, 180.00
# -430.000, 1400.000, 0.000, 40.000 | 90.00, 80.000, -90.00
# -430.000, 1400.000, -800.000, 640.000 | 90.00, 80.000, -90.00
# -430.000, 3200.000, -800.000, 1640.000 | 90.00, 80.000, 180.00
# -430.000, 3200.000, -1400.00, 440.000 | 90.00, 80.000, -90.00
# -80.000, 3200.000, -1400.00, 190.000 | 90.00, 80.000, 180.00
# -80.000, 3200.000, -2025.00, 545.000 | 0.00, 0.000, 0.00
#
# Completed on: 19 10 2022 13:26:21
# Pipe: HV_CONN_001

```

Header section  
 Body section  
 Footer section

By default, the file created is given the same name as the associated isometric drawing, but the suffix **\_nc.txt** is used for the file type. The output file name can be specified using the **nc\_filename** setting (see the 'File Naming' section)

## 11.12 File Naming

### 11.12.1 Creo File names

By default, the isometric drawing is saved under the name of the pipeline. You can set the environment variable **M4ISOCREO\_DRW\_NAME** to control the name of the file.

If creating SPOOL isometric drawings, the file is saved under the name of the spool. You can set the environment variable **M4ISOCREO\_SPOOLDRW\_NAME** to control the name of the file.

Both variables may be composed of fixed characters and variable parts. The keywords for the variable parts are *PIPE\_NAME*, *ASSY\_NAME* and *SPOOL\_NAME*, they must be enclosed in angle brackets.

Placeholder *SPOOL\_NAME* is not valid for **M4ISOCREO\_DRW\_NAME**.

*ASSY\_NAME* uses the name of the assembly (e.g. ABC), *PIPE\_NAME* the name of the pipe (e.g. XYZ) and *SPOOL\_NAME* uses the name of the spool (e.g. 123).

Examples:

The environment variable:

```
set M4ISOCREO_DRW_NAME=ISO-<ASSY_NAME>--<PIPE_NAME>
```

results in the drawing name: **ISO-ABC-XYZ**

The environment variable:

```
set M4ISOCREO_SPOOLDRW_NAME=ISO-<ASSY_NAME>--<SPOOL_NAME>
```

results in the drawing name: **ISO-ABC-123**

If the environment variable value is faulty, M4IsoCreo uses the default naming convention and displays a warning message.

The Creo name size limit is 31 characters. M4IsoCreo truncates the combined string when the limit is reached.

The environment variable:

```
set M4ISOCREO_OUTDIR="C:\Temp\Isos"
```

defines the output directory path for M4ISO files. If not set, then the current working directory is used.

The environment variable:

```
set M4ISOCREO_OUTDIRLOCK=YES
```

sets the output directory field to be locked in the *Create M4Iso Isometric* dialog. Permitted values are YES or NO.

The environment variable:

```
set M4ISOCREO_TEMPLATEDIR="C:\M4ISO_config\Sht_Templates"
```

sets the directory path for format (FRM files) and drawing (DRW files) templates.

**Please note:** This also applies to directories that are managed by a PLM system (e.g., Windchill).

Using this M4IsoOptionStore setting to display the terminal connection labels:

```
<M4IsoOption name="diagram/annotation/show_terminal_connected_to"
valueType="boolean" value="t"/>
```

then the isometric will show the name of the pipeline that the isometric is connected to (if known). Normally this name is the pipeline label, but if the environment variable **M4ISOCREO\_DRW\_NAME** has been set, then the default pipename used may not match the isometric drawing name the pipe is drawn on.

To make sure the pipename matches the isometric name use the environment variable: **M4ISOCREO\_PIPE\_NAME**

This uses the same notation format as the **M4ISOCREO\_DRW\_NAME** setting.

Normally it would emulate this setting. For example:

```
set M4ISOCREO_DRW_NAME=<ASSY_NAME>
```

```
set M4ISOCREO_PIPE_NAME=<ASSY_NAME>
```

## 11.12.2 CSV files

Individual table output can be saved to a CSV file using these settings:

```

<M4IsoOption name="diagram/bends_table/save_as_csv"
               valueType="boolean" value="t"/>
<M4IsoOption name="diagram/cuts_table/save_as_csv"
               valueType="boolean" value="t"/>
<M4IsoOption name="diagram/item_details_table/save_as_csv"
               valueType="boolean" value="t"/>
<M4IsoOption name="diagram/spools_table/save_as_csv"
               valueType="boolean" value="t"/>
<M4IsoOption name="diagram/welds_table/save_as_csv"
               valueType="boolean" value="t"/>

```

The following setting controls if existing CSV files are purged (t) or not (f) when a new isometric is created. Only those files that match the generated output name of the pipe being processed are purged.

```

<M4IsoOption name="core/files/purge_csv_files"
               valueType="boolean" value="t"/>

```

### 11.12.3 NC Files

```

<M4IsoOption name="diagram/bends_table/nc_template_file"
               valueType="string" value="NCTemplate.txt"/>

```

This specifies the name of a template file that contains instructions to format the output NC data file. This file must exist in the `%M4ISOCREO_HOME%\m4isodata` folder.

An example template file named **NCTemplate.txt** is supplied during installation. You may modify this file or create your own with a different filename.

The `save_for_nc` setting dictates if an NC data file is output (t) or not (f)

```

<M4IsoOption name="diagram/bends_table/save_for_nc"
               valueType="boolean" value="f"/>

```

The `nc_filename` setting controls the name of the NC data file created using known parameters in the isometric data.

```

<M4IsoOption name="diagram/bends_table/nc_filename"
               valueType="string">
NC-%$pipe_name$%--%$cut_ref$%-OD%OutsideDiameter%.txt
</M4IsoOption>

```

The `nc_filename_purge_filter` setting defines a wildcard filter to be used to purge existing NC data files when a new isometric is created. Only those files that match the generated output name of the pipe being processed are purged.

```

<M4IsoOption name="diagram/bends_table/nc_filename_purge_filter"
               valueType="string"
               value="NC-%$pipe_name$%-* .txt"/>

```

### 11.12.4 Alternative Output directory

These two settings control the location and use of an alternative folder separate to the output folder specified in the M4ISO dialog.

```
...  
<M4IsoOption name="core/files/alternative_aux_files_directory"  
              valueType="string" value="$M4ISO_NC_OUTDIR$"/>  
<M4IsoOption name="core/files/use_alternative_aux_files_directory"  
              valueType="boolean" value="t"/>
```

Only CSV and NC output files are stored in this auxiliary folder. The location value can be a literal pathname or a reference to an existing environment variable (the environment variable must be enclosed in \$ characters as shown above).

### 11.13 Creo Intermediate Files

During the processing from M4ISO into Creo, a number of intermediate files are created. These normally are not required once the Isometric is completed and can be erased. There are a number of environment variables to automatically delete intermediate files created for/by M4ISO.

Delete the intermediate SHE files created by the M4ISO process:

```
SET M4ISOCREO_DEL_SHEFILES=YES
```

Delete the intermediate DXF files created by the M4ISO process:

```
SET M4ISOCREO_DEL_DXFFILES=YES
```

Delete the ISOX data file created by the ISOREAL process for the M4ISO process:

```
SET M4ISOCREO_DEL_ISOXFILES=YES
```

Note: When debugging problems with the M4ISOCreo application, it is often useful to send a copy of the ISOX file to CAD Schroer for analysis.

## 11.14 Labels and Notes

### 11.14.1 Connection Labels

Labels can be added to termination points to identify the equipment or pipeline that the terminal connects to.

This mechanism works by looking at the CSYS object connected to the end of the selected pipeline that is not part of the selected pipeline. It then identifies the connected equipment or pipeline name for that CSYS. Obviously, if the pipeline being processed is not connected to anything, then no label will be created.

Use this setting to turn on the connection labels:

```
<M4IsoOption name="diagram/annotation/show_terminal_connected_to"
valueType="boolean" value="t"/>
```

Use this setting to define the equipment connection label:

```
<M4IsoOption name="diagram/annotation/terminal_connected_to_equip_format"
valueType="string" value="Conn. to Equip: %1 atPort %2"/>
```

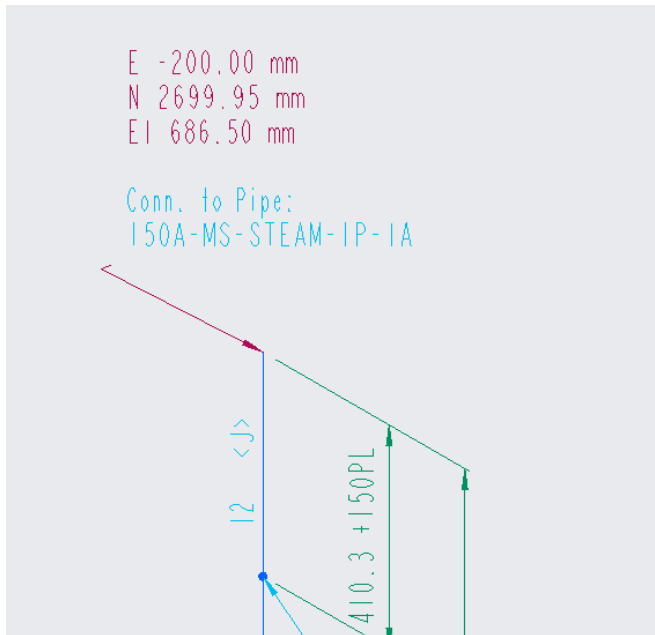
Use this setting to define the pipeline connection label:

```
<M4IsoOption name="diagram/annotation/terminal_connected_to_pipe_format"
valueType="string" value="Conn. To Pipe:&#10;%1"/>
```

The characters **&#10;** are used to force a new line in the label to make it appear over 2 or more lines.







### 11.14.2 Header Notes

M4ISO has the potential to define a maximum of 9 header notes that can be positioned on the isometric sheet. These header notes are not associated with components or parts but can be applied anywhere on the isometric drawing. All the header notes will have the same style definition, but each note can be positioned individually.

The appearance of all the header notes is controlled using these settings:

```
<M4IsoOption name="dars_illustrator/header/box_text/element_type"
  valueType="string" value="TR1"/>
<M4IsoOption name="dars_illustrator/header/round_box_text/element_type"
  valueType="string" value="TS1"/>
<M4IsoOption name="dars_illustrator/header/text/colour"
  valueType="colour" value="0 0 255 255"/>
<M4IsoOption name="dars_illustrator/header/text/element_type"
  valueType="string" value="T1"/>
<M4IsoOption name="dars_illustrator/header/text/font_name"
  valueType="string" value="Arial"/>
<M4IsoOption name="dars_illustrator/header/text/height"
  valueType="real" value="4"/>
<M4IsoOption name="dars_illustrator/header/text/layer"
  valueType="integer" value="6"/>
<M4IsoOption name="diagram/header/border_style"
  valueType="text border style" value="none"/>
```

To make the header notes visible on the isometric sheet use:

```
<M4IsoOption name="diagram/header/show_headers"
  valueType="boolean" value="t"/>
```

Each header note is defined using a *fieldn* reference, where *n* is a number in the range 1 to 9.

The position of each note is controlled using:

```
<M4IsoOption name="dars_illustrator/header/fieldn/position"
```

**valueType="coord2d" value="xpos ypos"/>**

Where “xpos ypos” are the x and y coordinates of the note datum point on the isometric sheet.

The content of each header note is defined using:

**<M4IsoOption name="diagram/header/fieldn/format"  
valueType="string">Literal String value</M4IsoOption>**

Where *Literal String value* is the exact characters to be output in the header note and any attribute names that are to be referenced, enclosed in % signs.

For example, a *Literal String value* of: Pipe Name: %\$pipe\_name\$%

Would typically produce the result: Pipe Name: Steam\_Demo-SP01

The format can also be left, center or right justified using the modifiers :l, :c and :r (default is :l).

For example: **value="Pipe Name:%\$pipe\_name\$%:r "**

### Header Note Attributes

The following internal attributes are available for the header note:

\$model\_name\$ - The name of the isox file used to create the isometric diagram.

\$pipe\_name\$ - Pipe name

\$pipe\_spec\$ - Pipe specification

In Creo the **isoreal\_parameter\_map.ptd** file can be modified to add Creo parameters from the ASSEMBLY or ROUTE SPOOL features. They must be given the target value “PipeInformation”. These can then be referenced in the header definition.

For example:

We need to create 3 notes on the isometric drawing to report the pipe name, a general note on the fabrication process and the spool ID.

In the Creo Piping Assembly, create a parameter named TEST with the string value: All welds to be ground.

In a ROUTE SPOOL feature, create a parameter named SPTEST with the string value: SPOOL SP01

In the **isoreal\_parameter\_map.ptd** file add these lines:

```
"TEST"      "Assembly"  "WC_TEST"    "PipeInformation"
"SPTEST"    "Spool"     "WC_SPTTEST" "PipeInformation"
```

In the header format definition add these lines to define the notes as field1, field2 and field3:

```
<M4IsoOption name="dars_illustrator/header/field1/position"
valueType="coord2d" value="220.0 275.0"/>
<M4IsoOption name="diagram/header/field1/format"
valueType="string">Pipe Name:%$pipe_name$% </M4IsoOption>
```

```

<M4IsoOption name="dars_illustrator/header/field2/position"
  valueType="coord2d" value="220.0 268.0"/>
<M4IsoOption name="diagram/header/field2/format"
  valueType="string">NOTE:%wc_test% </M4IsoOption>

<M4IsoOption name="dars_illustrator/header/field3/position"
  valueType="coord2d" value="220.0 261.0"/>
<M4IsoOption name="diagram/header/field3/format"
  valueType="string">Spool ID:%wc_sptest% </M4IsoOption>

```

The result on the isometric drawing would typically look like this:

Pipe Name: STEAM-DEMO-SPOOL  
 NOTE: All welds to be ground.  
 Spool ID: SPOOL SPOOL

### 11.14.3 Notes Dialog configuration

The notes available in the Notes tab of the Create M4ISO Isometric dialog are defined in the M4IsoOptionStore.xml file using the following settings:

General properties for generic note labels:

```

<M4IsoOption name="dars_illustrator/note/box_text/element_type"
  valueType="string" value="TR1"/>
<M4IsoOption name="dars_illustrator/note/line/colour"
  valueType="colour" value="240 105 15 255"/>
<M4IsoOption name="dars_illustrator/note/line/element_type"
  valueType="string" value="L0"/>
<M4IsoOption name="dars_illustrator/note/line/thickness"
  valueType="real" value="0.01"/>
<M4IsoOption name="dars_illustrator/note/link/point_function"
  valueType="integer" value="58"/>
<M4IsoOption name="dars_illustrator/note/round_box_text/element_type"
  valueType="string" value="TS1"/>
<M4IsoOption name="dars_illustrator/note/text/element_type"
  valueType="string" value="T1"/>

```

Note specific properties are defined as follows. The place marker <NOTE> is replaced by the correct note name, in the range **note\_1** to **note\_9** inclusive.

```

<M4IsoOption name="dars_illustrator/<NOTE>/layer" valueType="integer" value="6"/>
<M4IsoOption name="dars_illustrator/<NOTE>/text/colour"
  valueType="colour" value="240 105 15 255"/>
<M4IsoOption name="dars_illustrator/<NOTE>/text/font_name"
  valueType="string" value="Arial"/>
<M4IsoOption name="dars_illustrator/<NOTE>/text/height"
  valueType="real" value="0.2"/>

```

```

<M4IsoOption name="diagram/annotation/<NOTE>/border_style"
               valueType="text border style" value="box"/>
<M4IsoOption name="diagram/annotation/<NOTE>/description"
               valueType="string" value="Note description"/>

```

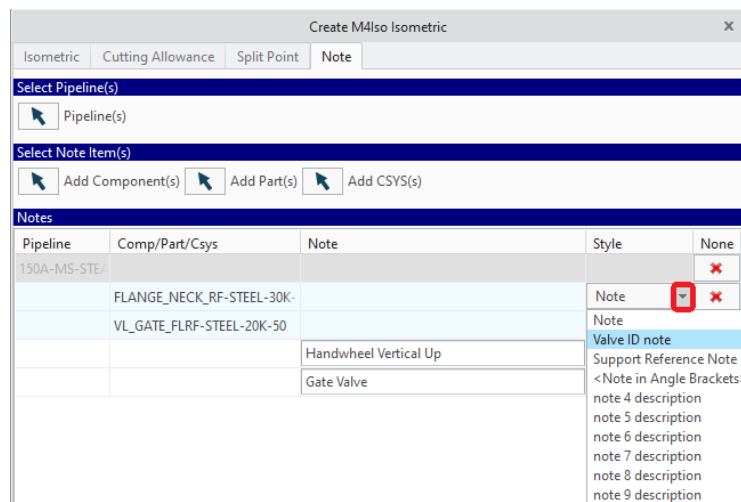
The description value should be meaningful description of the purpose or style of the note. These descriptions are displayed in the pulldown menu of note types that the user can select from when creating a note.

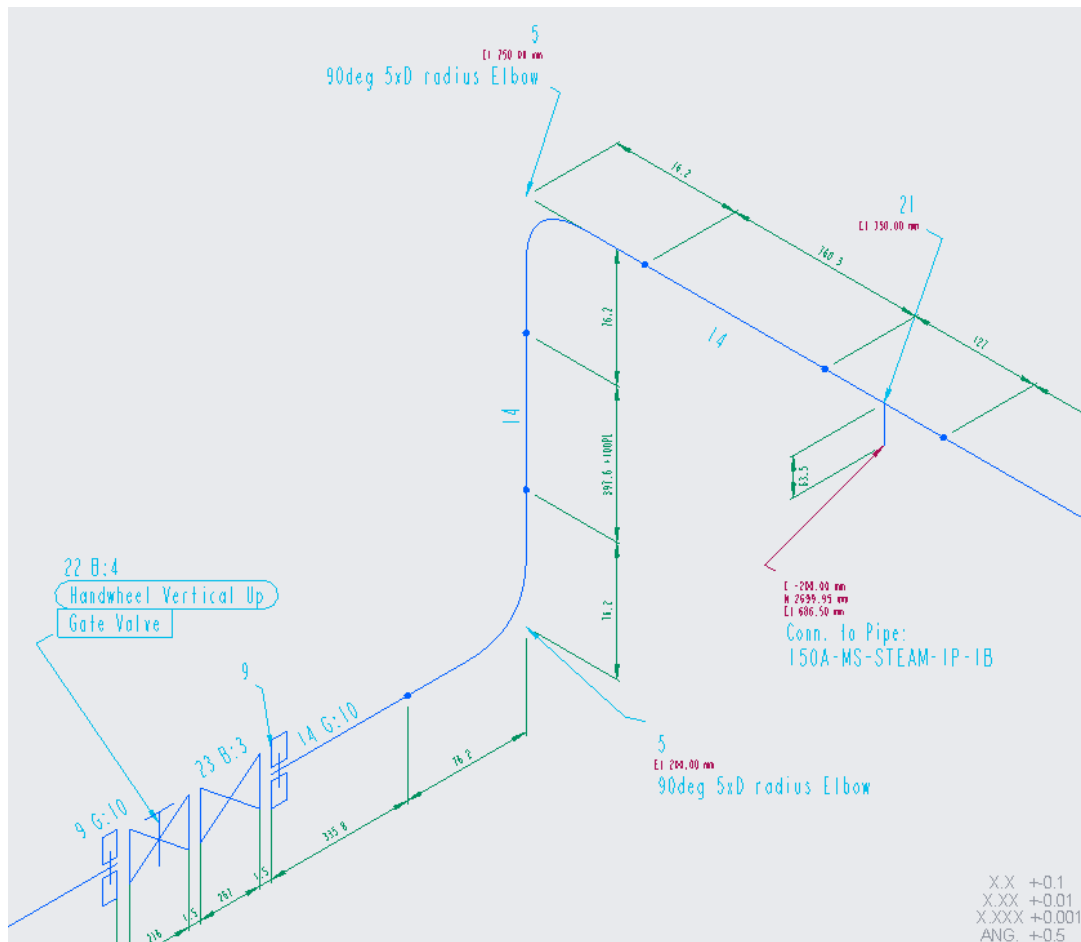
For example:

```

<M4IsoOption name="diagram/annotation/note_1/description"
               valueType="string" value="Valve ID note"/>
<M4IsoOption name="diagram/annotation/note_2/description"
               valueType="string" value="Support Reference Note"/>
<M4IsoOption name="diagram/annotation/note_3/description"
               valueType="string" value="<Note in Angle Brackets>"/>

```

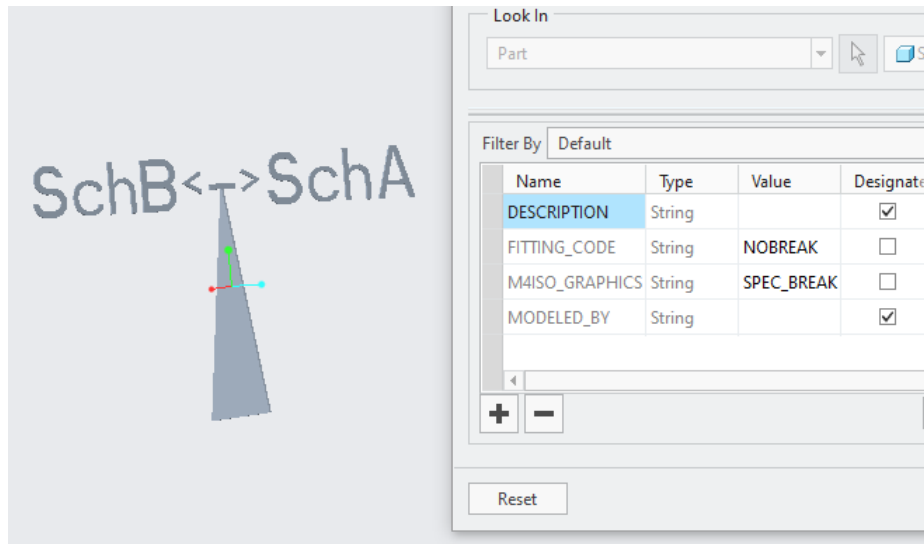




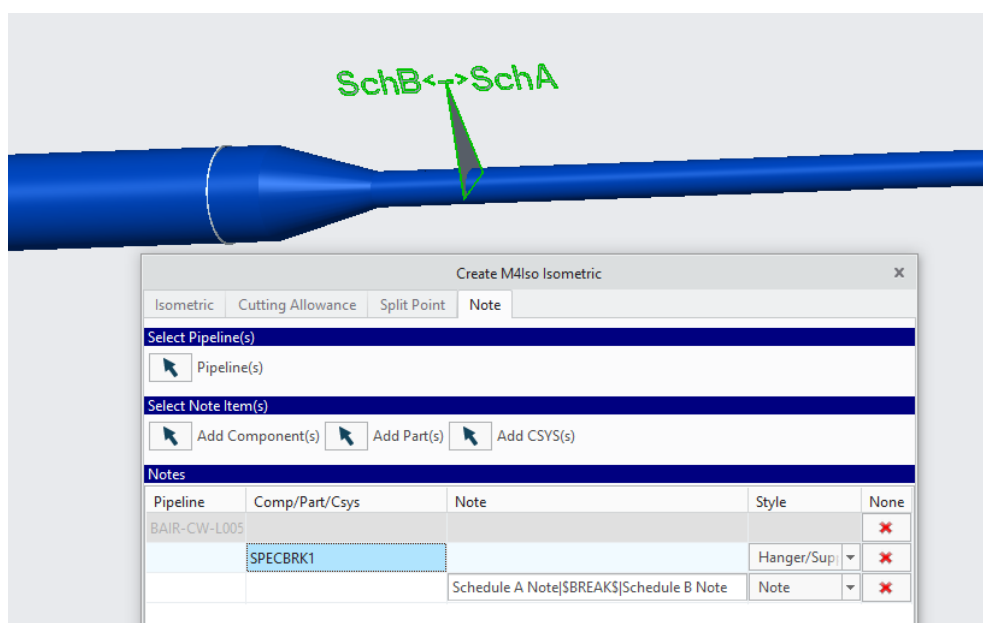
The image above shows the results of different styles of notes added to the Component (valve) and Part (elbow) items. Notice how the note value added to the elbow part appears on both elbows in the Isometric.

### 11.15 Spec/Pipe Break settings

A Creo in-line (NOBREAK) fitting can be mapped to a M4ISO symbol (specbrk) to identify spec or vendor breaks in the pipeline. The fitting must have a parameter named M4ISO\_GRAPHICS with a value of "SPEC\_BREAK".



After the part has been added into the pipeline assembly, the Component entity of the part can have a note associated with it. The note should be of the format: "Side A note|\$BREAK\$|Side B note". The literal keyword code |\$BREAK\$| is used by the software to identify the split between the note to be displayed on side A (PORT0) and the note to be displayed on side B (PORT1). The content of note A and note B is user definable, but the complete value cannot exceed 80 characters, including the keyword code.



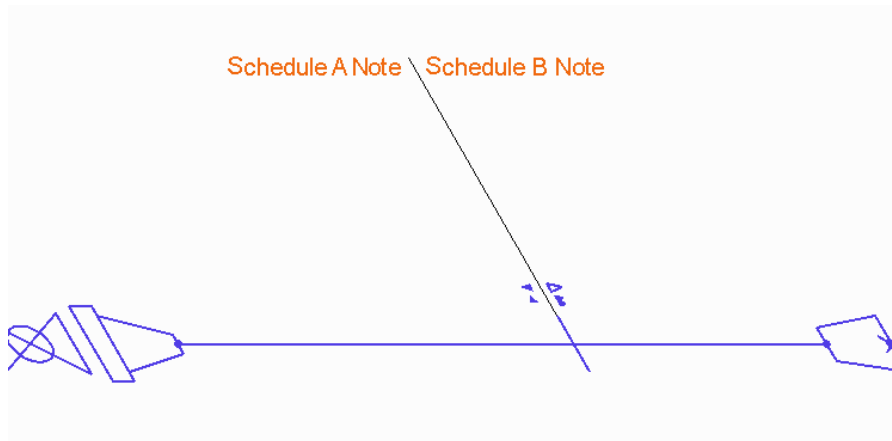
The Creo generic part name for the break part must be mapped to the M4ISO symbol

definition that will be displayed on the isometric in the isoreal\_symbol\_map.ptd file:

**"SPECBRK1" "specbrk" "specbrk"**

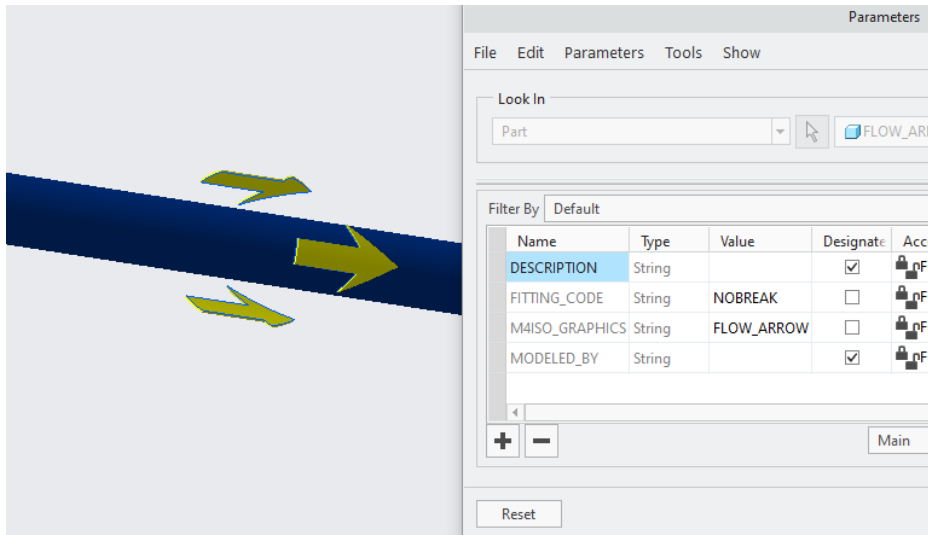
When the isometric is processed, the note appearance is controlled using the appropriate note settings, depending on which note style was selected. The following settings are used to control the appearance of the line from the symbol to the notes, the orientation of the notes and the spacing between the notes.

```
<M4IsoOption name="dars_illustrator/break/line/colour"
               valueType="colour" value="240 105 15 255"/>
<M4IsoOption name="dars_illustrator/break/line/element_type"
               valueType="string" value="L1"/>
<M4IsoOption name="dars_illustrator/break/line/thickness"
               valueType="real" value="0"/>
<M4IsoOption name="diagram/break/notes/keep_horizontal"
               valueType="boolean" value="f"/>
<M4IsoOption name="diagram/break/notes/spacing"
               valueType="real" value="0"/>
```



### 11.16 Flow Arrow symbol

A Creo in-line (NOBREAK) fitting can be mapped to a M4ISO symbol (flow\_arrow) to identify the direction of fluid flow in the pipeline. The fitting must have a parameter named M4ISO\_GRAPHICS with a value of "FLOW\_ARROW".

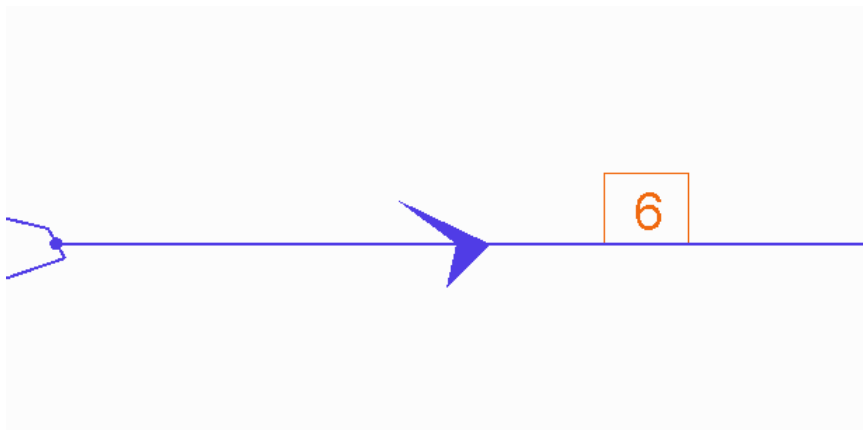


The flow arrow part must be mapped to the M4ISO symbol so that it can be displayed on the isometric. In the isoreal\_symbol\_map.ptd file add the line:

**"FLOW\_ARROW" "flow\_arrow" "flow\_arrow"**

Where "FLOW\_ARROW" is the Creo generic file name of the flow arrow part. Note that there must be a part inserted into the pipeline assembly, in the correct orientation, to identify the flow direction. The ISOREAL/M4ISO software is not able to identify the Creo assigned flow direction.

The isometric symbol is:





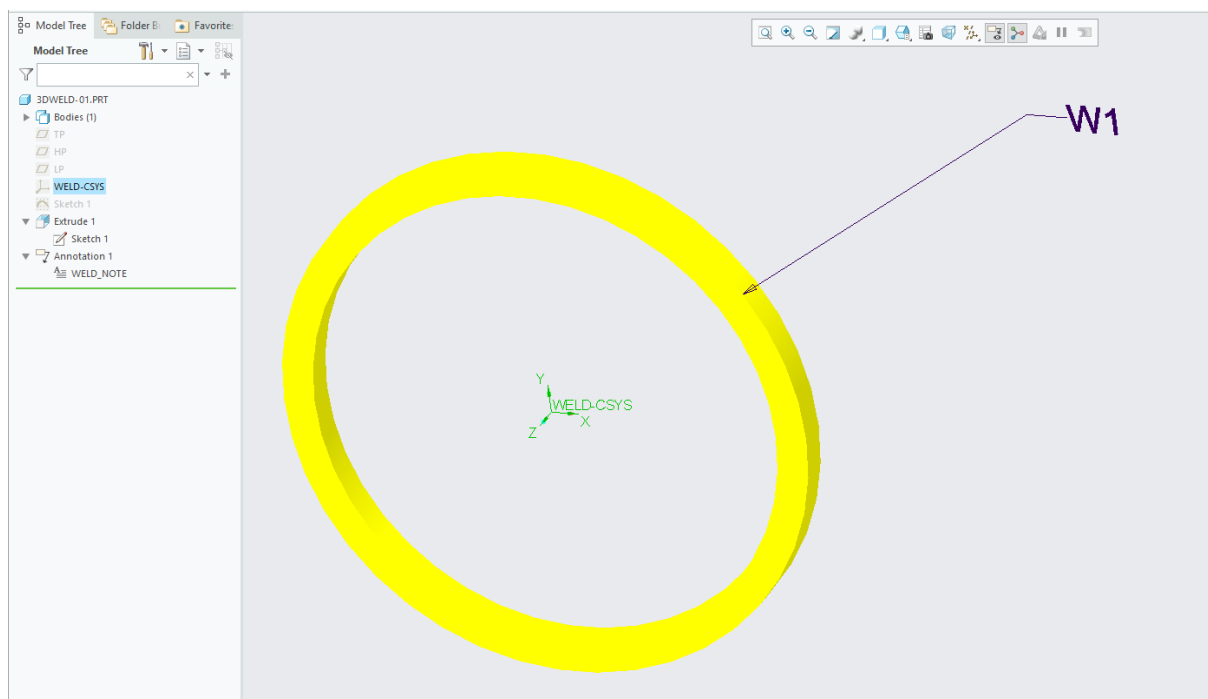
## 11.17 M4ISOCreo Weld Fittings

There is a restriction in the Creo Toolkit libraries that does not allow the interrogation of weld cut features in a pipeline assembly.

To work around this problem, a number of customers have implemented weld parts, which are inserted like other fittings (valves, elbows, etc.) into the pipeline.

To correctly deal with these fittings and recognise them as welds, the M4ISOCreo application has been modified with some extra functionality.

### 11.17.1 Weld PART definition



Each weld part has to be unique. A family part definition for welds will not work.

The simplest layout is shown above – a simple ring with a single CSYS named WELD-CSYS.

The weld part must contain these parameters:

PART_TYPE=M4ISO_WELD_PART	(identify the part as an M4ISO weld fitting)
WELD_SW_FW=SW or FW	(Shop Weld or Field Weld)
WELD_TYPE=BW, FW or SW	(Butt Weld, Fillet Weld or Socket Weld)
WELD_NUMBER=Wn	(Example: W1, W2, W23, etc.)

The weld part may also contain other parameters if required.

WELD_SHRINK=value	(the shrinkage expected/allowed during welding e.g. 0.5)
WELD_QUALITY_LEVEL=value	(a code indicating quality required)
WELD_SHAPE=value	(A weld shape code, for example: V, U or I)
WELD_THICKNESS=value	(The weld thickness value – e.g. 4.0)

Any other user-defined parameter.

### 11.17.2 Environment Variables

#### Weld Part ID

In order to identify the weld parts to M4ISO, the following environment variable is required:

```
set M4ISOCREO_WELDPRT_ID=PART_TYPE:M4ISO_WELD_PART
```

The M4ISOCREO\_WELDPRT\_ID environment variable is used to identify those Parts/Components found in the Piping assembly that are welds. The value consists of 2 parts separated by a colon character (:). The first part is the Creo parameter name that has been added to the Weld part/component. In the example, a parameter name of PART\_TYPE is used, but any parameter name can be used.

The second part of the setting is the value that has been assigned to the parameter to identify the part as a weld. The example shows a value of M4ISO\_WELD\_PART.

In this example, only those Parts/Components found in the pipeline assembly that contain a parameter named PART\_TYPE which has the value M4ISO\_WELD\_PART will be considered as weld fittings by M4ISO.

#### Weld numbers

The weld parts can contain a Creo parameter that is used to number each weld. The following environment variable is used to define the Creo parameter used to number the welds.

```
set M4ISOCREO_WELD_ID=WELD_NUMBER
```

The defined Creo parameter can be set on any weld Part/Component identified in the pipe assembly to define the weld number for that fitting. If the parameter does not exist on a weld fitting, then the standard method of assigning weld numbers will be used for that fitting.

### 11.17.3 Parameter Mappings

In order to access the Creo parameters so that they can be displayed in the Weld Table, it is necessary to map the CREO parameters to M4ISO attribute names with a target of "Weld" in the isoreal\_parameter\_map.ptd file.

The format of the mapping is:

```
"<CREO_PARAMETER_NAME>" "<CREO_TYPE_ID>" "<M4ISO_ATTRIBUTE_NAME>" "Weld"
```

Where:

**<CREO\_PARAMETER\_NAME>** is the parameter name to be mapped.

**<CREO\_TYPE\_ID>** is one of PART, COMPONENT or ASSEMBLY where the parameter is defined.

**<M4ISO\_ATTRIBUTE\_NAME>** is an M4ISO attribute name that will be used to hold the value.

**Weld** is the target identifier that makes the attribute available to the Weld Table

```
"WELD_SW_FW" "Part" "M4I_WELD_SW_FW" "Weld"
"WELD_SHRINK" "Part" "M4I_WELD_SHRINK" "Weld"
"WELD_TYPE" "Part" "M4I_WELD_TYPE" "Weld"
"WELD_SHAPE" "Part" "M4I_WELD_SHAPE" "Weld"
```

The M4ISO attribute name can be the same as the Creo parameter name (if it hasn't already been used elsewhere) or any unique name can be used. In the example above, the attribute name is based on the parameter name but has been prefixed with the characters **M4I\_**.

#### 11.17.4 OptionStore settings

The main OptionStore setting for weld parts would be used to control the appearance of the [welds\\_table/ column\\_format](#) to display the mapped attributes. For example:

```
<M4IsoOption name="diagram/welds_table/column_format"
  valueType="string">Weld:%$weld_ref$%;
  Shape:%M4I_WELD_SHAPE%;
  FW/SW:%M4I_WELD_FW_SW%;
  Shrink:%M4I_WELD_SHRINK%;
  Type:%M4I_WELD_TYPE%;
</M4IsoOption>
```

Would produce a table that could appear like:

Weld	Shape	FW/SW	Shrink	Type
W1	V	SW	0.9	BW
W2	V	SW	0.9	SW
W3	V	FW	0.9	BW

#### 11.17.5 SymbolInfo settings for welds

Towards the bottom of the M4IsoSymbolInfo.xml file, there are a small number of definitions for weld symbols. These relate to the WELD\_TYPE parameter in the weld part fittings:

```
<M4IsoWeldDefinition name="butt_weld" symbol="blob" field_symbol="hexagon_with_cross"/>
<M4IsoWeldDefinition name="fillet_weld" symbol="blob" field_symbol="hexagon_with_cross"/>
<M4IsoWeldDefinition name="socket_weld" symbol="blob_and_cup"
  field_symbol="hexagon_and_cup_with_cross"/>
```

The weld definition maps the weld type onto specific symbols for example:

SHOP welds - symbol="blob"

FIELD welds - field\_symbol="hexagon\_with\_cross"

For this to work correctly, you should map the Creo weld type codes to the name values used in the SymbolInfo file. This is done in the **isoreal\_endtype\_map.ptd** file like so:

BW butt\_weld  
 SW socket\_weld  
 FW fillet\_weld

It is also possible to miss out the **isoreal\_endtype\_map.ptd** definition and add the Creo weld type code directly in the M4IsoWeldDefinition like so:

```
<M4IsoWeldDefinition name="FW" symbol="blob" field_symbol="hexagon_with_cross"/>
```

## 11.18 M4ISO Entity Color Settings

In M4ISO there are several entities that have a color attribute. Colours are assigned as an array of 4 numbers that represent values for Red, Green, Blue and Alpha in that order. The colour values are in the range 0 to 255 with 0 being no colour and 255 being maximum colour. In M4ISO the Alpha value is ALWAYS 255.

By varying the amounts of the 3 primary colours, any shade of colour can be created.

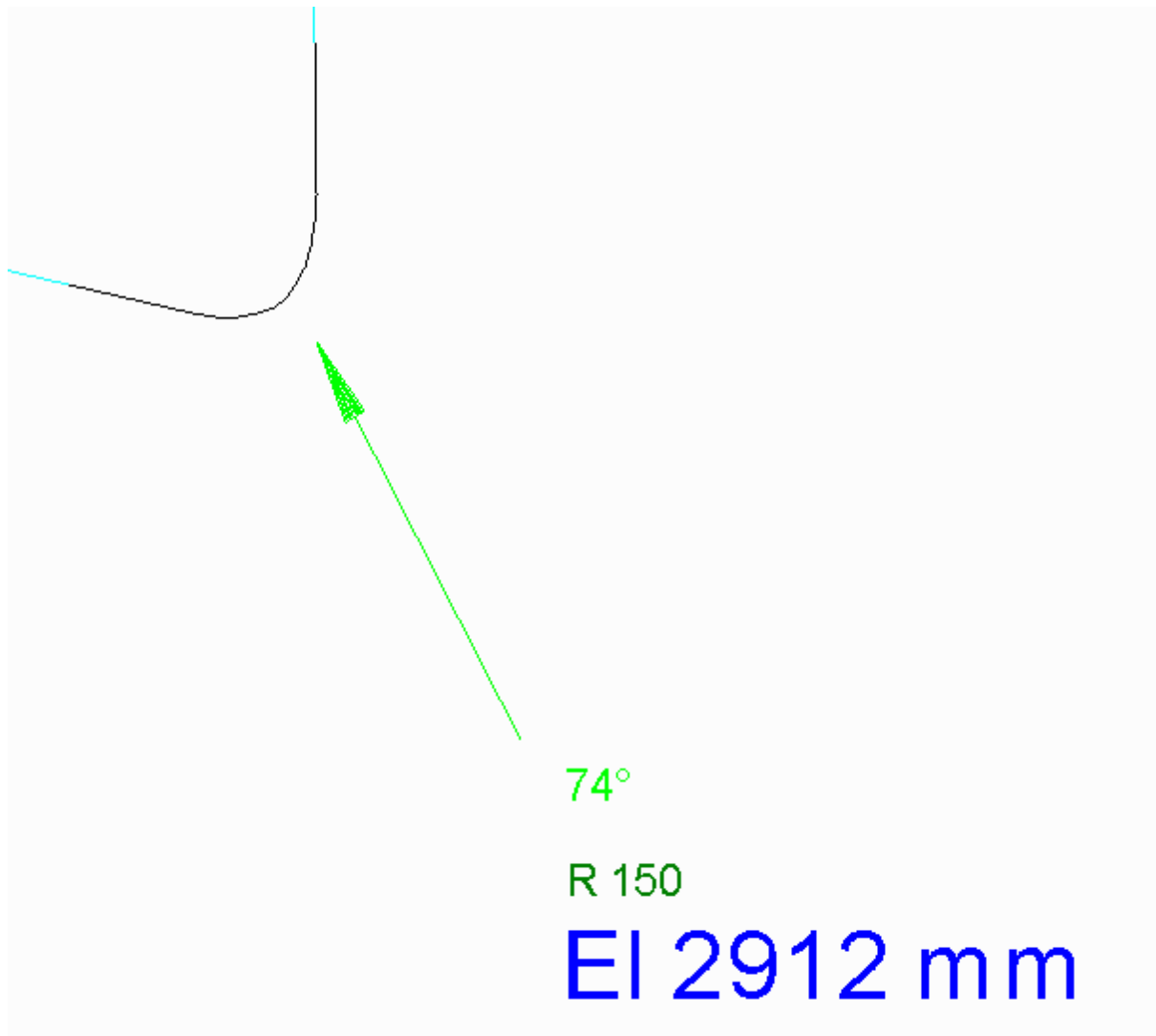
For example:

Color		Red	Green	Blue	Alpha
Black		0	0	0	255
Yellow		255	215	0	255
Pink		255	192	203	255
Orange		255	127	0	255
Green		0	255	0	255
Dark Red		140	0	0	255
Sky Blue		135	206	255	255
Purple		135	104	205	255
Dark Blue		0	0	140	255
Cyan		10	255	255	255
Dark Green		0	140	0	255

### 11.18.1 Mixed label colors

During creation of the notes and labels on an isometric, M4ISO will reuse existing label arrows and simply add the text. This can result in several different color labels using the arrow from the first label created.

For example:



This is a deliberate process to avoid unnecessary clutter in the isometric diagram.

### 11.18.2 Creo color mapping

In the M4ISOcreo application, the definition of colors is complicated by the fact that there is a data exchange process between the M4ISO application and Creo.

M4ISO defines color and thickness to specific element types. For example, a solid thin line will have a color and boldness (thickness) attribute.

DXF defines color and thickness based on the Layer an element is drawn on.


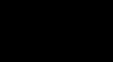






Creo defines a color attribute that geometry can be assigned to. The thickness of the geometry is defined in a pen table of up to 8 pens that assigns a pen thickness to a System color. Users can specify their own custom colors, but these will only ever be drawn in Pen 1. Therefore, to assign different pen thicknesses, the geometry must be assigned to a System Color.

Our solution is to:

1. Map the M4ISO elements to DXF colors.
2. Map the DXF colors to Creo System Colors.
3. Create a pen table to map the Creo System Colors to a thickness.

M4ISOCreo is supplied with a default mapping solution based on the standard colors defined in the M4IsoOptionStore.xml file and the default color numbers used by the AutoCAD DXF format.

#### M4IsoOptionStore colors:

Entity	Color	Red	Green	Blue	Alpha
Annotation, Notes		240	105	15	255
Table Lines, North Angle, Weld/Bends Table Text		0	0	0	255
Ghost Lines		138	123	238	255
Components, Weld Symbols		80	60	230	255
Dimensions		50	205	180	255
Pipe Segments		120	50	50	255
Elevation Line & Text		200	50	100	255
Spool/Items/Cuts Table Text, Header Text		0	0	255	255

The M4ISO drawing file is mapped to the DXF formats using the files:

- C:\M4ISOCreo\_V4\_2\cadconvert\cfg\med2dxf\_lstyle.map for lines.
- C:\M4ISOCreo\_V4\_2\cadconvert\cfg\med2dxf\_tstyle.map for texts.



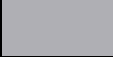
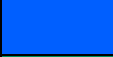




### The DXF Colors

Entity	DXF Color Number	
Annotation, Notes	30	
Table Lines, North Angle, Weld/Bends Table Text	7	
Ghost Lines	181	
Components, Pipe Segments, Weld Symbols	173	
Dimensions	135	
Elevation Hatching	252	
Elevation Line & Text	13	
Spool/Items/Cuts Table Text, Header Text	5	

The DXF file is mapped to the Creo colors using the file:

- *C:\M4ISOcreo\_V4\_2\isorea\cfg\M4IsoImport.pro*

### The Creo Color mapping

Entity	Creo System Color name	Color
Annotation, Notes	SECTION_COLOR	
Table Lines, North Angle, Weld/Bends Table Text	LETTER_COLOR	
Ghost Lines	DIMMED_COLOR	
Components, Pipe Segments, Weld Symbols	CURVE_COLOR	
Dimensions	SHEETMETAL_COLOR	
Elevation Hatching	LETTER_COLOR	
Elevation Line & Text	HIGHLIGHT_COLOR	
Spool/Items/Cuts Table Text, Header Text	LETTER_COLOR	

This mapping is taken from the M4IsoImport.pro file supplied at M4ISOCreo V4.2. The Creo colors shown in the third column are based on the “Default Theme” from Creo.

### Creo Pen Table

The final part of the supplied solution is the definition of different line thicknesses when plotting the drawing.

At M4IsoCreo V4.2, two template pen definitions are supplied:

*C:\M4ISOcreo\_V4\_2\isorea\cfg\M4IsoThin.pnt:*

! PenTable

pen 1 thickness .050 cm; drawing\_color; curve\_color;

pen 2 thickness .025 cm; letter\_color; section\_color; attention\_color; highlight\_color;

*C:\M4ISOcreo\_V4\_2\isorea\cfg\M4IsoThick.pnt:*

! PenTable

pen 1 thickness .070 cm; drawing\_color; curve\_color;

pen 2 thickness .035 cm; letter\_color; section\_color; attention\_color; highlight\_color;

There are several ways to specify which one of the pen tables to use.

- The Creo configuration option '**pen\_table\_file**' can be set to reference one of the above pen table files.
- The default Creo pen table file can be overwritten (or modified) to include the required mapping.
- A plotter configuration can be modified to include the required mapping.

The default Creo pen table file is normally held in the current working directory and is named ***table.pnt***.

Note: The Creo System color names visible in the Creo dialogs and the names used in the pen table are different.

## Summary

If you are using M4ISOCreo without making any modifications to the layers and/or colors used in the M4IsoOptionStore.xml file, then the situation described above will work in most cases.

If you have modified the layers and colors used in the M4IsoOptionStore, or if you wish to have specific geometry drawn at a different thickness, then you may need to consider using your own mapping files as described in the next section.

### 11.18.3 The Advanced Solution

#### Overview

If the colors used in the M4IsoOptionStore are modified, then the default mapping supplied with M4IsoCreo V4.1 will not work as the DXF color numbers referenced in the M4IsoImport.pro file may no longer be valid.

There is also a complication that the supplied default mapping does not consider that lines that use the same color may need to be shown as different thicknesses on the plot.

In these situations, it may be necessary to configure a custom mapping for each step of the conversion from M4ISO to Creo.

The custom mapping may be as simple as mapping the new DXF colors in the *M4IsoImport.pro* file. Or it may be more complicated by the need to pass different line



widths through to the plot. In this situation a full mapping will be necessary.

### M4ISO to DXF map

The M4ISO to DXF exchange is processed using the CADCONVERT utility included with M4ISOCreo.

Cadconvert uses 2 mapping files to convert line (*med2dxf\_lstyle.map*) and text (*med2dxf\_tstyle.map*) elements to DXF. There are 2 files as the format is slightly different to accommodate the different attributes between line and text elements, but the idea is broadly the same.

The mapping file is split into columns, delimited by the colon (:) character. The left half defines the M4ISO element to be mapped and the right half defines the DXF element to be created. Any line beginning with a double hyphen (--) is ignored by the processor.

The settings in this file can be used to control the format of the created DXF entities, either by entering literal values or by using the symbolic reference: @src which will use the value from the source file.

For example:

Line mapping

```
-- M4ISO
--LineStyle      : Layer : Colour: LineType  : Bold  :
-----+-----+-----+-----+-----+
                :  50   :   1   :    L1     :    1   :
```

```
AutoCAD
LineStyle      : Layer : Colour: Bold :Transfer:
-----+-----+-----+-----+-----+
DOTTED         : @src  : @src  : @src  :    1    :
```

In this example, lines in the M4ISO drawing that lie on layer 50, color 1, boldness 1 and are of type L1 will be mapped onto the DXF linestyle named DOTTED. They will keep the layer number, color and boldness (thickness) value they had in the M4ISO drawing.

Similarly with the Text mapping (there is an extra column for fonts):

```
M4ISO
TextStyle      : Layer  :Colour : TextType : Font  : Bold  :
-----+-----+-----+-----+-----+
                :  54   :   4    :    T1    : 1000  :    1   :

TextStyle      : FontFile : Layer  : Colour : Bold  : Transfer :
-----+-----+-----+-----+-----+-----+
    T1_1000     : ARIAL.ttf : @SRC   : @SRC   : @SRC  :    1     :
```

In the Text mapping example, any text element on the M4ISO drawing that is on layer 54, is color 4, boldness 4, font 1000 and type T1 will be mapped to a DXF text style

of T1\_1000 using font ARIAL.ttf. The layer, color and bold values will be kept from the M4ISO values.

The 'Transfer' column is used to control if an entity is converted (1) or not (2).

The default supplied mapping files are deliberately left open to allow most entities to transfer without too much control. If desired, a very fine control can be implemented by adding extra lines to control the transfer of entities.

There is also a configuration file: *C:\M4ISOcreo\_V4\_2\cadconvert\cfg\cadconvert.cfg*

The setting in this file control the DXF output such as the DXF version, how dimensions are created, etc.

There is also the possibility to create a 'real' mapping file during the transfer process. By switching this option ON, a mapping file will be created showing the line and text entities that were picked up from the M4ISO drawing and the entities that were created in the DXF file. This 'real' map can be useful for customising the transfer process.

To turn on the 'real' mapping, edit the cadconvert.cfg file and modify the lines:

**MAPLOG = ON**

**MAPFILE = C:\Temp\M4ISO\_cadconvert.map**

After saving the file, create an isometric through the M4ISO application. A file named *M4ISO\_cadconvert.map* will be created in the *C:\Temp* folder. This file will detail the actual mapping employed to create the isometric. For example:

```
CSGCONVTYPE : CSGMED42DXF
CSGMAPPINGTYPE : CSGLINESTYLEMAPPING
CSGENCODING : ANSI_1252
@FRE : 57 : 4 : LO : 1 : LO : @SRC : @SRC : @SRC : 1 :
@FRE : 54 : 4 : LO : 1 : LO : @SRC : @SRC : @SRC : 1 :
@FRE : 56 : 4 : LO : 1 : LO : @SRC : @SRC : @SRC : 1 :
@FRE : 55 : 4 : LO : 1 : LO : @SRC : @SRC : @SRC : 1 :
@FRE : 53 : 5 : LO : 1 : LO : @SRC : @SRC : @SRC : 1 :
@FRE : 51 : 5 : LO : 1 : LO : @SRC : @SRC : @SRC : 1 :
@FRE : 6 : 2 : LO : 1 : LO : @SRC : @SRC : @SRC : 1 :
LFUN : 0 : : LFUN : : LFUN : LFUN : @SRC : @SRC : 1 :
@FRE : 6 : 6 : L1 : 1 : DOTTED : @SRC : @SRC : @SRC : 1 :
@FRE : 4 : 1 : LDM : 1 : LDM : @SRC : @SRC : @SRC : 1 :
@FRE : 4 : 1 : LEA : 1 : LEA : @SRC : @SRC : @SRC : 1 :
@FRE : 52 : 5 : LO : 2 : LO : @SRC : @SRC : @SRC : 1 :
@FRE : 50 : 5 : LO : 1 : LO : @SRC : @SRC : @SRC : 1 :

CSGCONVTYPE : CSGMED42DXF
CSGMAPPINGTYPE : CSGTEXTSTYLEMAPPING
CSGENCODING : ANSI_1252
@FRE : 6 : 7 : T1 : 1000 : 1 : T1_1000 : ARIAL.ttf : @SRC : @SRC : @SRC : 1 :
@FRE : 57 : 4 : T1 : 1000 : 1 : T1_1000 : ARIAL.ttf : @SRC : @SRC : @SRC : 1 :
@FRE : 54 : 4 : T1 : 1000 : 1 : T1_1000 : ARIAL.ttf : @SRC : @SRC : @SRC : 1 :
@FRE : 54 : 7 : T1 : 1000 : 1 : T1_1000 : ARIAL.ttf : @SRC : @SRC : @SRC : 1 :
@FRE : 56 : 7 : T1 : 1000 : 1 : T1_1000 : ARIAL.ttf : @SRC : @SRC : @SRC : 1 :
@FRE : 55 : 7 : T1 : 1000 : 1 : T1_1000 : ARIAL.ttf : @SRC : @SRC : @SRC : 1 :
@FRE : 6 : 2 : T1 : 1000 : 1 : T1_1000 : ARIAL.ttf : @SRC : @SRC : @SRC : 1 :
@FRE : 6 : 2 : TR1 : 1000 : 1 : TR1_1000 : ARIAL.ttf : @SRC : @SRC : @SRC : 1 :
@FRE : 6 : 3 : T1 : 1000 : 2 : T1_1000 : ARIAL.ttf : @SRC : @SRC : @SRC : 1 :
@FRE : 4 : 1 : TDM : 0 : 2 : TDM : txt.shx : @SRC : @SRC : @SRC : 1 :
```

## DXF to Creo

M4ISOCreo has its own mapping file that it uses in preference to the normal Creo

mapping file:

*C:\M4ISOcreo\_V4\_2\isorea\cfg\M4IsoImport.pro*

The supplied file has this mapping:

!This is a DXF import mapping file listing color mapping.

! Notes in table cells

map\_color 5 LETTER\_COLOR

! Table frames

map\_color 7 LETTER\_COLOR

! Connector representation (previous element)

map\_color 181 LETTER\_COLOR

! Hatches

map\_color 252 LETTER\_COLOR

! Position, fall, oblique, fitting nominal sizes.

map\_color 13 HIGHLIGHT\_COLOR

! Annotations: Item numbers, tube nominal sizes, angles etc.

map\_color 30 SECTION\_COLOR

! Dimensions

map\_color 135 SHEETMETAL\_COLOR

! Tubes, fittings

map\_color 173 CURVE\_COLOR

This mapping is based on the color settings in the M4IsoOptionStore.xml file being mapped to specific DXF color numbers.

For example: the color attribute for parts table lines and text is defined as:

```
<..."dars_illustrator/item_details_table/line/color" valueType="color" value="0 0 0 255"/>
<..."dars_illustrator/item_details_table/text/color" valueType="color" value="0 0 255 255"/>
```

These colors then map in the DXF file as DXF colors 7 and 5 respectively.

The M4ISOImport.pro file then maps these DXF color numbers to specific Creo colors, in this example, both 7 and 5 are mapped to LETTER\_COLOR

If the M4ISOImport.pro file doesn't exist, then the import to Creo will use the filename referenced in the Creo configuration setting:

**intf2d\_in\_dxf\_mapping\_file**

It is possible to get Creo to generate a report on the mappings used when a DXF file is imported.

By using the File Open command to manually open a DXF file previously created by the M4ISO program, the 'Import DXF' dialog that Creo displays has a 'Save Mappings' button that creates a file of the elements found in the DXF file and how they were mapped into Creo. This file could be used as the basis for any modification of the M4ISOImport.pro file.

Note the saved mappings will be written to the file referenced in the **intf2d\_in\_dxf\_mapping\_file** setting in the Creo configuration. If there is no file de-

defined for this setting, then the mapping will be written to the working directory as **dximport.pro**.

## 11.18.4 An Example

### The Problem

A customer wants the Pipe Segment geometry to be shown in a thicker line than the other geometry. How can we achieve this?

### A Solution

For the DXF to Creo exchange, the mapping process is primarily based on the colors of elements. We need to identify the elements in the M4ISO drawing and assign them to the colors we want in Creo.

While color mapping from M4ISO to DXF can be done, it may be easier to understand if we map layers in M4ISO to colors in DXF.

Creo has a restriction that line thickness can only be applied to the Creo System Colors. If a line is assigned a non-System Color, then it can only be drawn in Pen 1. So ideally, we want to ensure that we map our geometry to use the Creo System Colors and that the Pipe Segment lines use a unique System Color. There is also a Creo restriction that for plotting/printing a maximum of 8 pens can be defined in the plotter pen table, although normally only 4 are used.

### The M4IsoOptionStore.xml setup

Decide which layers are going to be assigned to different features in the isometric drawing.

For example:

We are going to assign layers to the elements created on an isometric drawing. We can then map these layers to use a specific color in the DXF file. Note that layers are integer numbers and can be in the range 0-1023. We will use layers 50, 60, 65, 70, 75, 80 and 85.

Layer 50 will be assigned to the frame lines on tables, the North Symbol icon and break lines:

```
name="dars_illustrator/bends_table/layer" valueType="integer" value="50"
name="dars_illustrator/cuts_table/layer" valueType="integer" value="50"
name="dars_illustrator/item_details_table/layer" valueType="integer" value="50"
name="dars_illustrator/spools_table/layer" valueType="integer" value="50"
name="dars_illustrator/welds_table/layer" valueType="integer" value="50"
name="dars_illustrator/north_symbol/layer" valueType="integer" value="50"
```

Layer 60 will be assigned to the lines and text used in annotations and notes:

```
name="dars_illustrator/angle/layer" valueType="integer" value="60"
name="dars_illustrator/annotation/layer" valueType="integer" value="60"
name="dars_illustrator/note/layer" valueType="integer" value="60"
name="dars_illustrator/note_1/layer" valueType="integer" value="60"
```

```
name="dars_illustrator/note_2/layer" valueType="integer" value="60"
name="dars_illustrator/note_3/layer" valueType="integer" value="60"
name="dars_illustrator/note_4/layer" valueType="integer" value="60"
name="dars_illustrator/note_5/layer" valueType="integer" value="60"
name="dars_illustrator/note_6/layer" valueType="integer" value="60"
name="dars_illustrator/note_7/layer" valueType="integer" value="60"
name="dars_illustrator/note_8/layer" valueType="integer" value="60"
name="dars_illustrator/note_9/layer" valueType="integer" value="60"
```

Layer 65 will be assigned for dimension lines and text:

```
name="dars_illustrator/dimension/layer" valueType="integer" value="65"
```

Layer 70 will be assigned for elevation lines and text:

```
name="dars_illustrator/elevation/layer" valueType="integer" value="70"
```

Layer 75 will be assigned to ghost line geometry:

```
name="dars_illustrator/component/ghost/layer" valueType="integer" value="75"
name="dars_illustrator/pipe_segment/ghost/layer"
valueType="integer" value="75"
```

Layer 80 will be assigned to component fittings and welds:

```
name="dars_illustrator/component/layer" valueType="integer" value="80"
name="dars_illustrator/connection/layer" valueType="integer" value="80"
```

Layer 85 will be assigned to the Pipe Segment geometry:

```
name="dars_illustrator/pipe_segment/layer" valueType="integer" value="85"
```

## M4ISO to DXF Mapping Files

Now we can map the layers onto the colors in the DXF file.

We could reference DXF color index values here, but we are going to directly address the Creo colors using the RGB notation. To achieve this, we must edit the `C:\M4ISOCreo_V4_2\cadconvert\cfg\cadconvert.cfg` file and set the **USETRUECOLOR** setting to **ON**. This setting can be found at the bottom of the supplied file.

We now map the M4ISO layers we have defined in the `M4IsoOptionStore.xml` file to the RGB colors used in Creo. In this example the color values have been taken from the Graphics colors of the Light Theme.

Elements	Layer Number	RGB value	Creo color name
Annotations, Table Text	50, 60	0, 0, 240	LETTER_COLOR
Components, Welds	80	0, 153, 90	SHEETMETAL_COLOR
Dimensions	65	0, 153, 204	SECTION_COLOR
Table frames and Others	50, 0	34, 34, 38	GEOMETRY_COLOR
Pipe Segments	85	153, 102, 51	DATUM_COLOR
Ghost Lines	75	194, 194, 204	HIDDEN_COLOR

Elevations	70	240, 192, 0	QUILT_COLOR
------------	----	-------------	-------------

To map the lines, edit the file:

*C:\M4ISOcreo\_V4\_2\cadconvert\cfg\med2dxf\_lstyle.map*

```
-- MEDUSA 4
-- Style : Layer : Colour: Type : Bold : Style : Layer : Colour : Bold : Trans:
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
LPRM : 92 : : : : @src : LPRM : @src : @src : 2 :
--
-- Frame line
@FRE : 0 : : : : L0 : @SRC : RGB 34, 34, 38 : @src : 1 :
DOTTED_THIN: 0 : : : : DOTTED: @SRC : RGB 34, 34, 38 : @SRC : 1 :
--
-- Dimensions
@FRE : 65 : : LDM : : LDM : @SRC : RGB 0, 153, 204 : @SRC : 1 :
@FRE : 65 : : LEA : : LEA : @SRC : RGB 0, 153, 204 : @SRC : 1 :
--
-- Annotation labels
@FRE : 60 : : : : L0 : @SRC : RGB 0, 0, 240 : @SRC : 1 :
--
-- Tables
@FRE : 50 : : : : L0 : @SRC : RGB 34, 34, 38 : @SRC : 1 :
--
-- Elevations
@FRE : 70 : : : : L0 : @SRC : RGB 240, 192, 0 : @SRC : 1 :
--
-- Ghost Lines
@FRE : 75 : : : : DOTTED: @SRC : RGB 194, 194, 204 : @SRC : 1 :
--
-- Components and welds
@FRE : 80 : : : : L0 : @SRC : RGB 0, 153, 90 : @SRC : 1 :
--
-- Pipes
@FRE : 85 : : : : L0 : @SRC : RGB 153, 102, 51 : @SRC : 1 :
--
-- Others
: : : : : @src : @src : RGB 34, 34, 38 : @src : 1 :
-- EOF
```

To map the texts, edit the file:

*C:\M4ISOcreo\_V4\_2\cadconvert\cfg\med2dxf\_tstyle.map*

```
-- MEDUSA 4
-- Style : Layer : Colour: Type : Font : Bold : Style : Font : Layer : Colour : Bold:Trans:
-----+-----+-----+-----+-----+-----+-----+-----+-----+-----+
: : : : : : : : : : : :
--
-- Tables
@FRE : 50 : : T1 : 1000 : :T1_1000 : ARIAL.ttf: @SRC : RGB 0, 0, 240:@SRC: 1 :
--
-- Dimensions
@FRE : 65 : : TDM : 0 : :TDM : txt.shx : @SRC : RGB 0, 153, 204:@SRC: 1 :
--
-- Annotation labels
@FRE : 60 : : TS1 : 1000 : :TS1_1000: ARIAL.ttf: @SRC : RGB 0, 0, 240:@SRC: 1 :
@FRE : 60 : : T1 : 1000 : :T1_1000 : ARIAL.ttf: @SRC : RGB 0, 0, 240:@SRC: 1 :
--
-- Elevations
@FRE : 70 : : T1 : 1000 : :T1_1000 : ARIAL.ttf: 70T : RGB 240, 192, 0:@SRC: 1 :
--
-- Others
: : : : : : @src : : @src : @src : @src: 1 :
-- EOF
```

## DXF to Creo Mapping File

M4ISOCreo has its own mapping file that it uses in preference to the normal Creo mapping file:

*C:\M4ISOcreo\_V4\_2\isoreal\cfg\M4\solImport.pro*

If this file doesn't exist, then it will use the filename referenced in the Creo configuration setting:

## intf2d\_in\_dxf\_mapping\_file

In the *M4ISOImport.pro* file, we map the colors we assigned to the DXF file to the Creo System Colors we have chosen to use, like so:

```
map_color    BLUE          LETTER_COLOR
map_color    WHITE         GEOMETRY_COLOR
map_color    34,34,38      GEOMETRY_COLOR      ! Table Lines and Others
map_color    194,194,204   HIDDEN_COLOR         ! Ghost Fittings
map_color    0,153,204     SKETCH_COLOR           ! Dimensions
map_color    240,192,0     QUILT_COLOR              ! Elevations
map_color    0,153,90      SHEETMETAL_COLOR        ! Components and Welds
map_color    153,102,51    DATUM_COLOR             ! Pipe Segments
map_color    0,0,240       LETTER_COLOR            ! Text/Annotations
!
map_font     ARIAL.ttf     font
map_font     txt           font
!
! EOF
```

## Pen Table File

The final part of the mapping is the creation and assignment of a Pen Table file. This is used to assign plotter/printer line thickness values to specific Creo System Colors. Note that the assigned line thickness is only visible on the printed version of the Creo drawing. It is not visible in the drawing displayed in the Creo graphics screen (although a line weight can be assigned by selecting the line in the drawing and using the line properties dialog to set a weight value).

The pen table file works by assigning a pen number to the Creo color name. Several colors can be assigned to a single pen by separating the color names with a semicolon character. Any colors not assigned to a pen will use Pen 1 by default.

The mapping is further complicated because the Creo color names used by the Pen Table do not match the System Color names used in Creo. Appendix B contains a map of Creo System color names to Pen Table color names.

For our example, we will use three pens numbered 1, 2 and 3.

Pen 1 will have a line thickness of 0.15mm (0.015cm). This will be used to draw the dimensions, text/annotations and tables, ghost fittings and elevations.

Pen 2 will have a line thickness of 0.50mm (0.05cm). This will be used to draw the fittings (components/welds).

Pen 3 will have a line thickness of 1.0mm (0.10cm). This will be used to draw the Pipe Segment lines.

```
pen 1 thickness .015 cm; letter_color; section_color; drawing_color; half_tone_color;
quilt_color;
pen 2 thickness .050 cm; attention_color;
pen 3 thickness .100 cm; datum_color;
```

These settings are normally placed in a file named **table.pnt**.

To ensure these settings are used, it is necessary to configure Creo to use this file using the configuration setting:

**pen\_table\_file** <file pathname>\table.pnt

If PDF files are to be produced, then another configuration setting is necessary to ensure the PDF export uses the pen table:

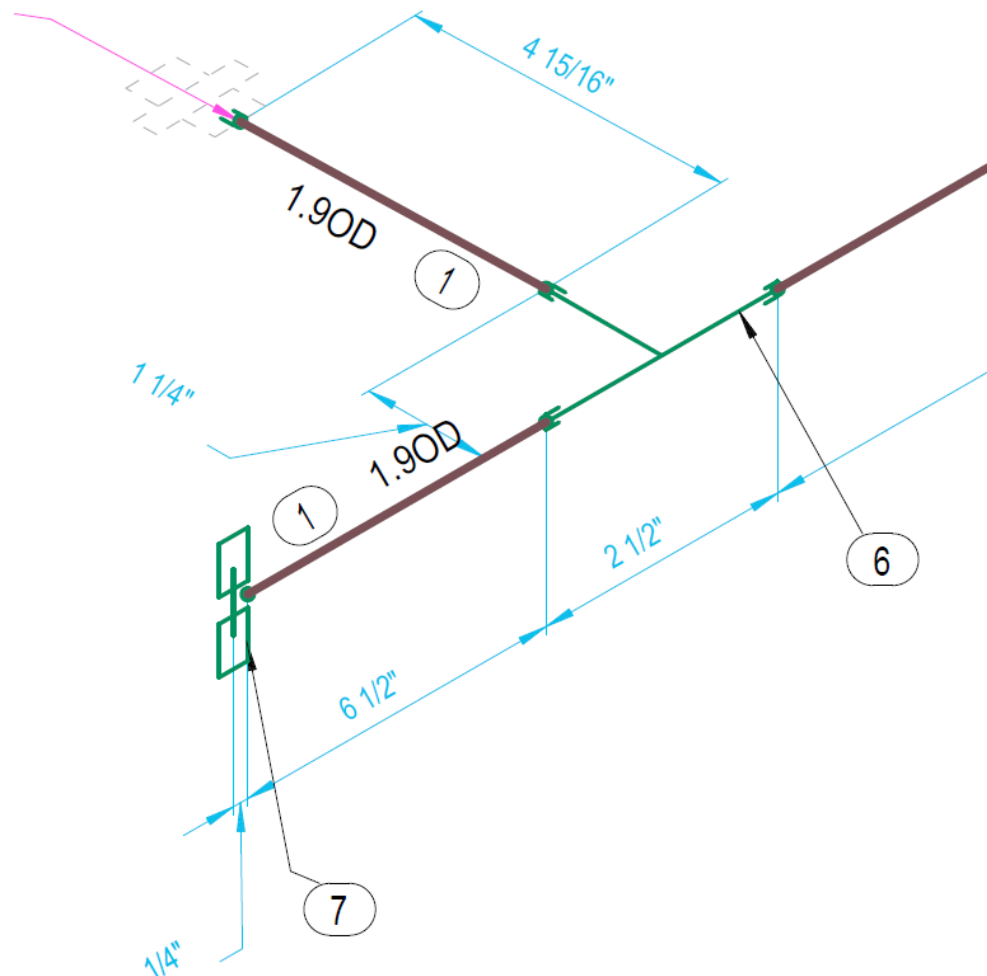
**pdf\_use\_pentable** yes

### Create an Isometric Print

We are now ready to proceed with the new configuration.

Processing an isometric and then printing it (in this case to a PDF file) will produce an output that looks like this:

Cont on page 2  
X -6 3/16"  
Y 7 7/8"  
Z 16"





## 12 Creating custom component symbols

### 12.1.1 The M4IsoSymbolInfo File

The M4IsoSymbolInfo.xml file is in the <m4iso product>\data directory and can be edited using a plain text editor. This file contains the definitions of all the components and symbols M4ISO can use. The file is also used to:

- control the automatic creation of Spools within the M4ISO application (as opposed to spools created in Creo).
- to identify the start and end of Cut segments of pipe.
- to identify where welds are in a pipe.

Note that the 'as supplied' file is not comprehensive, and you may want to add extra component symbol definitions to represent your own fittings.

#### The File syntax

The M4IsoSymbolInfo.xml file is defined in sections as follows:

- M4IsoSymbolDefinition
- M4IsoSymbolDescriptor
- M4IsoConnectionType
- M4IsoConnectionDefinition
- M4IsoWeldDefinition
- M4IsoConnectionSymbolDescriptor
- M4IsoNorthSymbolDescriptor

#### M4IsoSymbolDefinition

- name = a unique name for the symbol
- class = an internal code to define the type of symbol
- componentType = a group name that the symbol belongs to
- isWeld = "Yes" or "No"
- descriptor = the name of the *M4IsoSymbolDescriptor* to create the symbol

Example:

```
<M4IsoSymbolDefinition name="sweepolet" class="$olet$"  
                        componentType="olet" descriptor="sockolet"/>
```

The "class" attribute can be one of "\$bend\$", "\$branch\$", "\$gasket\$", "\$olet\$", "\$plane\_component\$", "\$pipe\_segment\$", "\$support\$" and "\$terminal\$". If not specified it will default to "\$plane\_component\$".

Note that "\$olet\$", "\$support\$" and "\$terminal\$" are all special forms of plane components.

Except for the "\$plane\_component\$" class, there is a built-in symbol descriptor for each component class, bearing the same name as the class with which it is compatible. These are: "\$bend\$", "\$branch\$", "\$gasket\$", "\$pipe\_segment\$", and "\$terminal\$".

The "componentType" attribute specifies the basic type of the component the symbol represents (e.g. "valve") and is used to group items in the item details list and to control dimension placement for that group of symbols. If the "componentType" is not specified it will default to the definition name.

Note that the "name" and "componentType" attributes are used to define the M4ISO\_SYMBOL and M4ISO\_CLASS parameters respectively in the mapping of Creo parts to isometric symbols.

The "isWeld" parameter is used to indicate that a symbol represents a weld, so the instance will be included in the welds list and not in the item details list. It defaults to "No".

The "descriptor" parameter specifies the descriptor to be used to create (draw) this symbol on the isometric. It is only used for plane component symbols (i.e. with class "\$plane\_component\$", "\$olet\$", "\$support\$" or "\$terminal\$"). If not specified, it defaults to the definition name.

### M4IsoSymbolDescriptor

- name = a unique name for the symbol description definition
- M4IsoDatum2D
  - x = a decimal number
  - y = a decimal number
- M4IsoCP2D
  - x = a decimal number
  - y = a decimal number
  - dx = one of -1.0, 0.0 or 1.0
  - dy = one of -1.0, 0.0 or 1.0
  - datum = normally 1
  - big\_end = "true"
  - M4IsoValidConnectionType
    - Name = a valid *M4IsoConnectionType* name
- M4IsoDiamond
  - x0 = a decimal number
  - y0 = a decimal number
  - x1 = a decimal number
  - y1 = a decimal number
  - x2 = a decimal number

- y2 = a decimal number
- x3 = a decimal number
- y3 = a decimal number
- M4IsoPolygon2D
  - size = a positive integer number (the number of points in the polygon)
  - lineStyle = one of solid or dotted
  - fillStyle = one of none or solid
  - M4IsoPoint2D
    - x = a decimal number
    - y = a decimal number
- M4IsoPolyline2D
  - size = a positive integer number (the number of points in the line)
  - lineStyle = one of solid or dotted
  - M4IsoPoint2D
    - x = a decimal number
    - y = a decimal number

Example:

```
<M4IsoSymbolDescriptor name="miscellaneous_inline">
  <M4IsoDatum2D x="0.00" y="0.00"/>
  <M4IsoCP2D x="-10.00" y="0.00" dx="-1.00" dy="0.00" datum="1"/>
  <M4IsoCP2D x="10.00" y="0.00" dx="1.00" dy="0.00" datum="1"/>
  <M4IsoDiamond x0="10.50" y0="-8.50" x1="-10.50" y1="-8.50"
    x2="-10.50" y2="8.50" x3="10.50" y3="8.50"/>
  <M4IsoPolygon2D size="4" lineStyle="solid" fillStyle="none">
    <M4IsoPoint2D x="-10.00" y="8.00"/>
    <M4IsoPoint2D x="10.00" y="8.00"/>
    <M4IsoPoint2D x="10.00" y="-8.00"/>
    <M4IsoPoint2D x="-10.00" y="-8.00"/>
  </M4IsoPolygon2D>
  <M4IsoPolyline2D size="2" lineStyle="dotted">
    <M4IsoPoint2D x="-10.00" y="0.00"/>
    <M4IsoPoint2D x="10.00" y="0.00"/>
  </M4IsoPolyline2D>
</M4IsoSymbolDescriptor>
```

The M4IsoSymbolDescriptor syntax defines the geometric properties of a 2D symbol for a component of class "\$plane\_component\$", including the connection points, datums, enclosing boundary diamond, polygon outlines and polylines.

For asymmetric symbols such as weld neck flanges, the M4IsoCP2D definition can have an M4IsoValidConnectionType defined on one end to help the software orientate the symbol correctly. For example:

```
<M4IsoSymbolDescriptor name="flange_wn">
  <M4IsoDatum2D x="0.00" y="0.00"/>
  <M4IsoCP2D x="0.00" y="0.00" dx="-1.00" dy="0.00" datum="1"/>
  <M4IsoCP2D x="20.00" y="0.00" dx="1.00" dy="0.00" datum="1">
    <M4IsoValidConnectionType name="butt_weld"/>
  </M4IsoCP2D>
</M4IsoSymbolDescriptor>
```

```

    <M4IsoValidConnectionType name="compression"/>
    <M4IsoValidConnectionType name="screwed"/>
    <M4IsoValidConnectionType name="slip_on"/>
    <M4IsoValidConnectionType name="socket_weld"/>
  </M4IsoCP2D>
  ...
</M4IsoSymbolDescriptor>

```

For asymmetric symbols where the input and output connection points are different sizes, (such as with a reducer), one of the connection points needs to be marked as being the big\_end to help orientate the symbol correctly. For example:

```

<M4IsoSymbolDescriptor name="reducer_concentric">
  <M4IsoDatum2D x="0.00" y="0.00"/>
  <M4IsoCP2D x="-7.50" y="0.00" dx="-1.00" dy="0.00" datum="1" big_end="true"/>
  <M4IsoCP2D x="7.50" y="0.00" dx="1.00" dy="0.00" datum="1"/>
  ...
</M4IsoSymbolDescriptor>

```

For asymmetric symbols where the input and output connection points are not in the same plane, (such as with an eccentric reducer) the connection points and datum need to form a 90degree angle. For example:

```

<M4IsoSymbolDescriptor name="reducer_eccentric">
  <M4IsoDatum2D x="20.00" y="0.00"/>
  <M4IsoCP2D x="0.00" y="0.00" dx="-1.00" dy="0.00" datum="1" big_end="true"/>
  <M4IsoCP2D x="20.00" y="-2.50" dx="1.00" dy="0.00" datum="2"/>
  ...
</M4IsoSymbolDescriptor>

```

### M4IsoConnectionType

- name = a unique name to identify the end/connection type
- endPrepName = a description of the connection type as displayed in the tables.
- cutEnd = "Yes" or "No"

Example:

```

<M4IsoConnectionType name="butt_weld" endPrepName="Butt" cutEnd="Yes"/>

```

The M4IsoConnectionType defines the name and end preparation for a connection, and specifies if it is a cut end, so it can potentially be shown in the cuts table.

### M4IsoConnectionDefinition

- female = First end valid *M4IsoConnectionType*
- male = Second end valid *M4IsoConnectionType*
- continuesSpool = "Yes" or "No"
- continuesTube = "Yes" or "No"
- isWeld = "Yes" or "No"
- symbol = valid *M4IsoConnectionSymbolDescriptor* to represent the joint

Example:

```

<M4IsoConnectionDefinition female="bevel_ended"

```

```
male="butt_weld"
continuesSpool="Yes"
continuesTube="Yes"
isWeld="Yes"
symbol="blob"/>
```

The M4IsoConnectionDefinition syntax defines a valid connection between two connection types. It also defines the connection symbol and specifies if the connection continues a spool and/or a tube or is a welded connection. Note that the order of female/male connection end types is not important.

### M4IsoWeldDefinition

- name = a unique name for the weld
- symbol = A *M4IsoConnectionSymbolDescriptor* to be used for shop welds
- field\_symbol = A *M4IsoConnectionSymbolDescriptor* to be used for field welds

Example:

```
<M4IsoWeldDefinition name="butt_weld" symbol="blob"
field_symbol="hexagon_with_cross"/>
```

The M4IsoWeldDefinition syntax defines a weld symbol to be used in a specific Creo scenario. It defines the connection symbol descriptor to be used for shop and field weld types.

### M4IsoConnectionSymbolDescriptor

- name = a unique name for the connection symbol descriptor
- M4IsoConnectionGlyph
  - name = a valid in-built glyph name (see section 12.1.3)
- M4IsoPolygon2D
  - size = a positive integer number (the number of points in the polygon)
  - lineStyle = one of solid or dotted
  - fillStyle = one of none or solid
  - M4IsoPoint2D
    - x = a decimal number
    - y = a decimal number
- M4IsoPolyline2D
  - size = a positive integer number (the number of points in the line)
  - lineStyle = one of solid or dotted
  - M4IsoPoint2D
    - x = a decimal number
    - y = a decimal number

Example:

```
<M4IsoConnectionSymbolDescriptor name="dbar">
  <M4IsoConnectionGlyph name="double_bar"/>
</M4IsoConnectionSymbolDescriptor>
```

```

<M4IsoConnectionSymbolDescriptor name="screwed">
  <M4IsoPolygon2D size="5" lineStyle="solid" fillStyle="none">
    <M4IsoPoint2D x="2.00" y="2.00"/>
    <M4IsoPoint2D x="0.00" y="2.00"/>
    <M4IsoPoint2D x="2.00" y="0.00"/>
    <M4IsoPoint2D x="0.00" y="-2.00"/>
    <M4IsoPoint2D x="2.00" y="-2.00"/>
  </M4IsoPolygon2D>
</M4IsoConnectionSymbolDescriptor>

```

The M4IsoConnectionSymbolDescriptor defines the glyphs (graphics) or geometry to be drawn at a connection point.

### M4IsoNorthSymbolDescriptor

- M4IsoProjectedGeometry
  - M4IsoPolygon3D
    - size = a positive integer number (the number of points in the polygon)
    - lineStyle = one of solid or dotted
    - fillStyle = one of none or solid
    - M4IsoPoint3D
      - x = a decimal number
      - y = a decimal number
      - z = a decimal number
  - M4IsoPolyline3D
    - size = a positive integer number (the number of points in the line)
    - lineStyle = one of solid or dotted
    - M4IsoPoint3D
      - x = a decimal number
      - y = a decimal number
      - z = a decimal number
- M4IsoFlatGeometry
  - alignmentAxis = one of Z, East or North
  - M4IsoPolyline2D
    - size = a positive integer number (the number of points in the line)
    - lineStyle = one of solid or dotted
    - M4IsoPoint2D
      - x = a decimal number
      - y = a decimal number

The M4IsoNorthSymbolDescriptor syntax defines both the 3D projected geometry and the flat 2D representation of the North symbol.

For an example see section 12.1.4

## 12.1.2 Example Symbol Creation

For this worked example, we will create symbol definitions for 2 components.

1. An in-line Strainer
2. A Wafer Valve

When modifying the M4IsoSymbolInfo.xml file it is recommended that any new entries are added into their appropriate existing sections in alphabetical order. This makes them easier to find and avoids unnecessary duplication.

First, we need to set up a '*M4IsoSymbolDefinition*' name reference as shown below.

```
<M4IsoSymbolDefinition name="strainer" descriptor="strainer"/>
<M4IsoSymbolDefinition name="valve_wafer" componentType="valve"
                        descriptor="valve_wafer"/>
```

The '*M4IsoSymbolDefinition*' has a *descriptor* reference which is a pointer to the *name* reference in the '*M4IsoSymbolDescriptor*' (below).

The '*M4IsoSymbolDescriptor*' is used to define the profile of the component as it will appear in the isometric diagram.

The *M4IsoSymbolDescriptor* for the strainer component is as follows.

```
<M4IsoSymbolDescriptor name="strainer">
  <M4IsoDatum2D x="0.00" y="0.00"/>
```

The M4IsoDatum2D defines the location of a datum in the symbol. There can be any number of datums in a symbol. Datums are numbered in the order they are defined.

```
<M4IsoCP2D x="-4.00" y="0.00" dx="-1.00" dy="0.00" datum="1"/>
<M4IsoCP2D x="4.00" y="0.00" dx="1.00" dy="0.00" datum="1"/>
```

The M4IsoCP2D specifies the position, direction and associated datum of the connection points on the component.

```
<M4IsoDiamond x0="4.50" y0="-3.50" x1="-4.50" y1="-3.50"
              x2="-4.50" y2="3.50" x3="4.50" y3="3.50"/>
```

The M4IsoDiamond defines the corners of a bounding area (which is usually a rectangle). This is used to aid the placement of annotations so that they do not overlap the component.

```
<M4IsoPolygon2D size="4" lineStyle="solid" fillStyle="none">
  <M4IsoPoint2D x="-4.00" y="3.00"/>
  <M4IsoPoint2D x="4.00" y="3.00"/>
  <M4IsoPoint2D x="4.00" y="-3.00"/>
  <M4IsoPoint2D x="-4.00" y="-3.00"/>
</M4IsoPolygon2D>
```

The M4IsoPolygon2D defines a closed polygon shape consisting of a fixed number of points (size=n) and of a particular lineStyle and fillStyle. Each point in the polygon is defined using the M4IsoPoint2D definition to mark its x/y position.

```
<M4IsoPolyline2D size="2" lineStyle="solid">
  <M4IsoPoint2D x="-4.00" y="3.00"/>
  <M4IsoPoint2D x="-2.86" y="2.15"/>
</M4IsoPolyline2D>
```

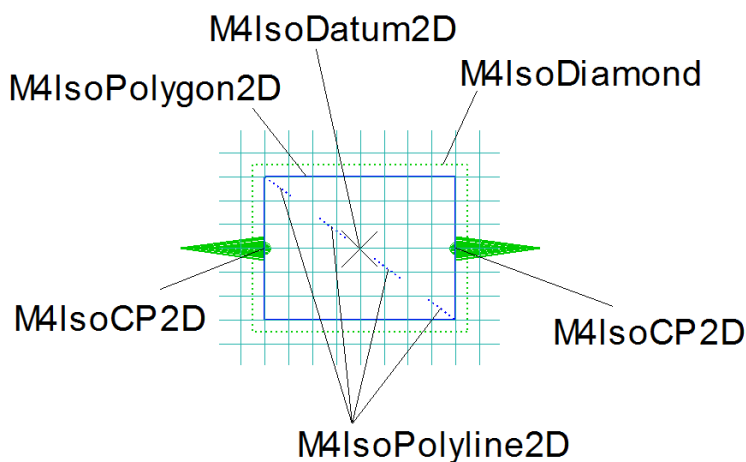
The M4IsoPolyline2D defines an open polyline of a fixed number of points (size=) and a particular lineStyle. Each point in the polyline is defined using the M4IsoPoint2D definition to mark its x/y position.

```

<M4IsoPolyline2D size="2" lineStyle="solid">
  <M4IsoPoint2D x="-1.71" y="1.28"/>
  <M4IsoPoint2D x="-0.58" y="0.41"/>
</M4IsoPolyline2D>
<M4IsoPolyline2D size="2" lineStyle="solid">
  <M4IsoPoint2D x="0.58" y="-0.41"/>
  <M4IsoPoint2D x="1.71" y="-1.28"/>
</M4IsoPolyline2D>
<M4IsoPolyline2D size="2" lineStyle="solid">
  <M4IsoPoint2D x="2.86" y="-2.15"/>
  <M4IsoPoint2D x="4.00" y="-3.00"/>
</M4IsoPolyline2D>
</M4IsoSymbolDescriptor>

```

The *M4IsoSymbolDescriptor* definition for the strainer results in the following representation of the component. (A 1mm grid is drawn to clearly show the positioning)



Only the elements shown in the BLUE line will be drawn on the isometric diagram. The Datum, CP's and Diamond will only be used programmatically for positioning purposes.

The *M4IsoSymbolDescriptor* for the Wafer Valve is as follows.

```

<M4IsoSymbolDescriptor name="valve_wafer">
  <M4IsoDatum2D x="0.00" y="0.00"/>
  <M4IsoCP2D x="-10.00" y="-0.00" dx="-1.00" dy="-0.00" datum="1"/>
  <M4IsoCP2D x="10.00" y="-0.00" dx="1.00" dy="-0.00" datum="1"/>
  <M4IsoDiamond x0="10.50" y0="-4.50" x1="-10.50" y1="-4.50"
    x2="-10.50" y2="4.50" x3="10.50" y3="4.50"/>
  <M4IsoPolygon2D size="3" lineStyle="solid">
    <M4IsoPoint2D x="-10.00" y="-4.00"/>
    <M4IsoPoint2D x="-10.00" y="4.00"/>
    <M4IsoPoint2D x="10.00" y="0.00"/>
  </M4IsoPolygon2D>
  <M4IsoPolyline2D size="2" lineStyle="solid">
    <M4IsoPoint2D x="10.00" y="4.00"/>
    <M4IsoPoint2D x="10.00" y="-4.00"/>
  </M4IsoPolyline2D>
</M4IsoSymbolDescriptor>

```

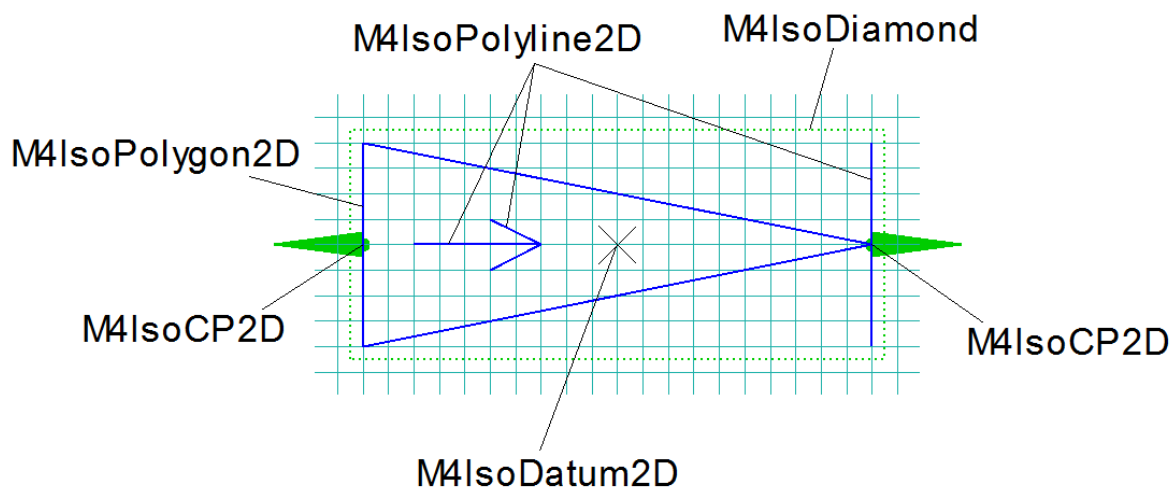


```

        <M4IsoPoint2D x="-8.00" y="0.00"/>
        <M4IsoPoint2D x="-3.00" y="0.00"/>
    </M4IsoPolyline2D>
    <M4IsoPolyline2D size="3" lineStyle="solid">
        <M4IsoPoint2D x="-5.00" y="1.00"/>
        <M4IsoPoint2D x="-3.00" y="0.00"/>
        <M4IsoPoint2D x="-5.00" y="-1.00"/>
    </M4IsoPolyline2D>
</M4IsoSymbolDescriptor>

```

The *M4IsoSymbolDescriptor* definition for the Wafer Valve results in the following representation of the component. (A 1mm grid is drawn to clearly show the positioning)



Only the elements shown in the BLUE line will be drawn on the isometric diagram. The Datum, CP's and Diamond will only be used programmatically for positioning purposes.

### Linking the Isometric Symbol to the Piping Component in Creo

The name used to define the M4ISO isometric symbol is linked to the Creo Piping component using the mapping file:

```
%M4ISOCREO_HOME%\isoreal\cfg\isoreal_symbol_map.ptd
```

Assuming the components, as defined in Creo Piping, are named "strainer\_ff" for the strainer and "vl\_wafer\_flff" for the wafer valve, then the mappings used in the file would be:

```

"strainer_ff" "strainer" "strainer"
"vl_wafer_flff" "valve" "valve_wafer"

```

The new symbol definitions would now be ready to use in the M4ISO application after the Creo application has been restarted.

It is also possible to map Creo components to M4ISO isometric symbols using Creo Parameters in the component model. For example, in the Creo component model vl\_wafer\_flff, we can use the parameters:

Name: **M4ISO\_SYMBOL** , type: *String*, value: ***valve\_wafer***

Name: **M4ISO\_CLASS**, type: *String*, value: ***valve***

If these are defined, they will override any values defined in the *isoreal\_symbol\_map.ptd* file.

## Graphical Attributes

The new symbol definitions will be drawn in the isometric diagram using the symbol definition defined.

However, when dimensioned, they will take on a random value of either `dim_to_cps` or `dim_to_datums`. If you wish to force a value, then you must add appropriate entries to the **M4IsoOptionStore** file as follows:

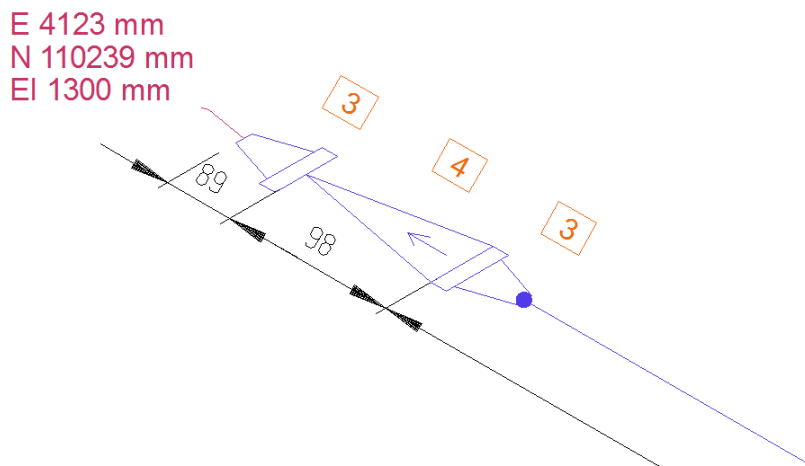
```
<M4IsoOption name="component/strainer/dim_to_cps"
               valueType="boolean" value="t"/>
<M4IsoOption name="component/strainer/dim_to_datums"
               valueType="boolean" value="f"/>
```

The Wafer Valve does not need to be defined as it is already included under the generic grouping for valves.

These entries can be included under the '*M4IsoDefaultSection*' attributes or as part of a set of '*M4IsoNamedSection*' attributes.

## Results

The figure below shows the Wafer Valve as it appears in an isometric diagram using the settings we have defined in the process above:



### 12.1.3 M4ISO End Conditions

The end conditions used in M4ISO are defined in 2 sections of the M4IsoSymbolInfo.xml file:

- M4IsoConnectionType
- M4IsoConnectionDefinition

Each of these sections are described in the following notes.

#### M4IsoConnectionType

The section defines the symbolic name of an end type, its proper name and if it is a cut end or not.

The syntax is:

```
<M4IsoConnectionType
    name="endtype_symbolic_name"
    endPrepName="proper_name"
    cutEnd="Yes"
/>
```

For the **name=** setting, "endtype\_symbolic\_name" is a string used to reference the connection type in the M4ISO application (primarily in the M4IsoSymbolInfo.xml file)

For the **endPrepName=** setting, "proper\_name" is a string used to display the full, proper name of the end type when it is written to the Isometric sheet, normally in the Cuts Table. This proper\_name can be the name as it appears in the local language.

The **cutEnd="Yes"** setting is used to define when a pipe is displayed in the Cuts Table. If this setting is not defined, then a value of "No" is assumed.

#### Defining CUT conditions

Any pipe that has both end types with the cutEnd="Yes" defined will be listed in the Cuts Table. The **continuesTube="Yes"** setting in the M4IsoConnectionDefinition section also has a bearing on the appearance of pipes in the Cuts Table

By default, the following end types are defined as cutEnds

- Bevel\_ended
- Butt\_weld
- Compression
- Field\_weld
- None
- Plain

All other end types (for example: flanged or screwed) do not have the cutEnd setting.

For example:

A pipe length with end conditions of "plain" and "butt\_weld" will be displayed in the Cuts Table.

A pipe length with end conditions of “butt\_weld” and “screwed” will not be displayed in the Cuts Table.

### M4IsoConnectionDefinition

This section defines what happens when 2 end conditions meet along the pipeline.

The syntax of these definitions is:

```
<M4IsoConnectionDefinition
    female="endtype_symbolic_name"
    male="endtype_symbolic_name"
    continuesSpool="Yes"
    continuesTube="Yes"
    isWeld="Yes"
    symbol="connection_symbol"
/>
```

The **male/female=** setting uses the “endtype\_symbolic\_name” string reference from the M4IsoConnectionType section.

The **continuesSpool=“Yes”** setting is used to identify when a connection (usually a weld) continues a length of Spool pipeline. If this setting is not used in the M4IsoConnectionDefinition, then a value of “No” is assumed and any spool will terminate at the specified connection.

The **continuesTube=“Yes”** setting is used to identify when a connection continues a length of Cut pipeline. If this setting is not used in the M4IsoConnectionDefinition, then a value of “No” is assumed.

The **isWeld=“Yes”** setting is used to identify when two pipe end types are welded together. If this setting is not used in the M4IsoConnectionDefinition, then a value of “No” is assumed.

The **symbol=“connection\_symbol”** setting is used to identify a symbol to be drawn in the isometric to graphically highlight the type of connection. The “*connection\_symbol*” value points to the name of the M4IsoConnectionSymbolDescriptor definition for the symbol. This may be an in-built ‘glyph’ or a custom symbol definition.

### Defining SPOOL conditions

M4ISO identifies where a spool starts and ends using the end conditions it encounters along the pipe.

A spool is a collection of pipeline components, which are normally fully welded and delivered as a single piece. Spool boundaries are where things are not welded e.g. flanged connections, Valves etc.

For example, a spool would start at a weldneck flange, may continue over some lengths of pipe, reducers, elbows and bends until it terminates in another weldneck flange. The only requirement is that all the instances between the 2 flanges are (nor-

mally) welded together.

#### Defining WELD conditions

The ***isWeld="Yes"*** setting is used to define when a connection is to be welded.

The use of this setting tells M4ISO where to place weld identifiers in the isometric drawing and ensures the Weld Table (if displayed) contains the correct number of rows.

### Linking the M4ISO End Condition names to the values defined in Creo

The name used to define the M4ISO End Condition is linked to the Creo names using the mapping file:

*%M4ISOCREO\_HOME%\isoreal\cfg\isoreal\_endtype\_map.ptd*

The mappings used in the file are in the format:

ENDTYPE	M4ISO_ENDTYPE
BE	bevel_ended
PE	plain
PL	plain
SC	screwed
TE	screwed
SW	socket_weld
SWLD	socket_weld
BW	butt_weld
BWLD	butt_weld
SO	slip_on
FL	flanged
FF	flat_face_flanged
FLFF	flat_face_flanged
FLRF	raised_face_flanged
FLRTJ	ring_jointed
FLRT	ring_jointed
FLTG	tongue_and_grooved
FLGT	tongue_and_grooved
FLGR	tongue_and_grooved
GKFF	flat_face_gasket
GKRF	raised_face_gasket
GKRTJ	ring_joint_gasket
GKRT	ring_joint_gasket
GKTG	tongue_and_groove_gasket
STUBIN	butt_weld
DEFAULT_PIPE	plain
FORCE_PIPE	plain

If you have created any new endtype codes in your Creo piping system, then you must add them into this file with a mapping to the equivalent M4ISO endtype name.

The DEFAULT\_PIPE and FORCE\_PIPE settings are designed to ensure that any undefined pipe ends are set to a value – plain by default. The default value can be changed if required by modifying these entries in the file.



## Connection Symbols

This section defines the graphical representation of connection symbols along the isometric pipe line.

The syntax of these definitions is:

```
<M4IsoConnectionSymbolDescriptor name="connection_symbol">
  <M4IsoConnectionGlyph name="glyph_name"/>
  <M4IsoPolygon2D size="6" lineStyle="solid" fillStyle="solid">
    <M4IsoPoint2D x="1.00" y="0.00"/>
    <M4IsoPoint2D x="0.5" y="-0.8660"/>
    ...
  </M4IsoPolygon2D>
  <M4IsoPolyline2D size="2" lineStyle="solid">
    <M4IsoPoint2D x="-1.5" y="-1.5"/>
    <M4IsoPoint2D x="1.5" y="1.5"/>
  </M4IsoPolyline2D>
</M4IsoConnectionSymbolDescriptor>
```

The ***name="connection\_symbol"*** setting is used to identify the symbol. This name is used as the reference from the M4IsoConnectionDefinition ***symbol=*** setting.

If the symbol is defined using the in-built glyph definitions, then only the appropriate M4IsoConnectionGlyph settings need to be used.

If suitable glyphs are not available, then a custom symbol graphic can be defined using the M4IsoPolygon2D and/or M4IsoPolyline2D definitions.

## Valid Glyphs

The M4ISO application has a few in-build symbols called glyphs. These can be referenced individually or together to form a simple graphic symbol.

<u>Graphic</u>	<u>Glyph Name</u>
	Bar
●	Filled_circle
□	Cup
○	Circle
×	Cross
	Double_bar
□	Square



### 12.1.4 M4ISO North Symbol Definition

The M4ISO North Symbol is a graphic that uses 3D and 2D geometry in its representation.

The syntax of the definition is:

```
<M4IsoNorthSymbolDescriptor>
  <M4IsoProjectedGeometry>
    <M4IsoPolygon3D size="4" lineStyle="solid" fillStyle="none">
      <M4IsoPoint3D x="0.00" y="0.48" z="0.0"/>
      ...
    </M4IsoPolygon3D>
    <M4IsoPolyline3D size="4" lineStyle="solid">
      <M4IsoPoint3D x="-0.03" y="0.55" z="0.0"/>
      ...
    </M4IsoPolyline3D>
  </M4IsoProjectedGeometry>

  <M4IsoFlatGeometry alignmentAxis="direction">
    <M4IsoPolyline2D size="2" lineStyle="solid">
      <M4IsoPoint2D x="0.00" y="0.00"/>
      ...
    </M4IsoPolyline2D>
  </M4IsoFlatGeometry>
</M4IsoNorthSymbolDescriptor>
```

The **M4IsoProjectedGeometry** section defines the 3D graphics using the **M4IsoPolygon3D** and **M4IsoPolyline3D** commands.

The **M4IsoFlatGeometry** sections define the 2D geometry that appears on a specified plane identified by the **alignmentAxis="direction"** setting. The value "direction" must be one of:

- "Z"                      This view is on the plane of the isometric sheet and does not rotate
- "East"                   This view is 60degrees clockwise from the Z axis and rotates with the north angle
- "North"                   This view is 60degrees anticlockwise from the Z axis and rotates with the north angle

The 2D geometry is defined by the **M4IsoPolyline2D** and **M4IsoPolygon2D** commands. In the supplied North Symbol definition, only the **M4IsoPolyline2D** command is used.

The position and size of the North Symbol is controlled in the M4IsoOptionStore.xml file using the settings:

```
<M4IsoOption name="dars_illustrator/north_symbol/position"
              valueType="coord2d" value="23 393"/>
```

```
<M4IsoOption name="dars_illustrator/north_symbol/scale_factor"
              valueType="real" value="15"/>
```

The 'position' value places the datum point (x=0, y=0, z=0) of the North Symbol.

The 'scale\_factor' value controls the magnification of the symbol on the sheet about its datum point.

Note that the creation of 3D and/or 2D graphics is optional. The symbol could contain only 2D graphics or only 3D graphics if required.

## North Symbol Examples

The following section details some simple examples of North Symbols.

The images on the right show the geometry being created in each section, highlighted in green.

### Default North Symbol

```
<M4IsoNorthSymbolDescriptor>
```

```
<M4IsoProjectedGeometry>
```

```
<M4IsoPolygon3D size="4" lineStyle="solid" fillStyle="none">
```

```
<M4IsoPoint3D x="0.00" y="0.48" z="0.0"/>
```

```
<M4IsoPoint3D x="-0.10" y="-0.25" z="0.0"/>
```

```
<M4IsoPoint3D x="0.00" y="-0.15" z="0.0"/>
```

```
<M4IsoPoint3D x="0.10" y="-0.25" z="0.0"/>
```

```
</M4IsoPolygon3D>
```

```
<M4IsoPolyline3D size="4" lineStyle="solid">
```

```
<M4IsoPoint3D x="-0.03" y="0.55" z="0.0"/>
```

```
<M4IsoPoint3D x="-0.03" y="0.67" z="0.0"/>
```

```
<M4IsoPoint3D x="0.03" y="0.55" z="0.0"/>
```

```
<M4IsoPoint3D x="0.03" y="0.67" z="0.0"/>
```

```
</M4IsoPolyline3D>
```

```
<M4IsoPolyline3D size="3" lineStyle="solid">
```

```
<M4IsoPoint3D x="0.00" y="0.48" z="0.0"/>
```

```
<M4IsoPoint3D x="0.0" y="0.00" z="0.2"/>
```

```
<M4IsoPoint3D x="0.00" y="-0.15" z="0.0"/>
```

```
</M4IsoPolyline3D>
```

```
<M4IsoPolyline3D size="3" lineStyle="solid">
```

```
<M4IsoPoint3D x="-0.10" y="-0.25" z="0.0"/>
```

```
<M4IsoPoint3D x="0.0" y="0.00" z="0.2"/>
```

```
<M4IsoPoint3D x="0.10" y="-0.25" z="0.0"/>
```

```
</M4IsoPolyline3D>
```

```
</M4IsoProjectedGeometry>
```

```
<M4IsoFlatGeometry alignmentAxis="East">
```

```
<M4IsoPolyline2D size="2" lineStyle="solid">
```

```
<M4IsoPoint2D x="0.00" y="0.00"/>
```

```
<M4IsoPoint2D x="0.51" y="0.00"/>
```

```
</M4IsoPolyline2D>
```

```
</M4IsoFlatGeometry>
```

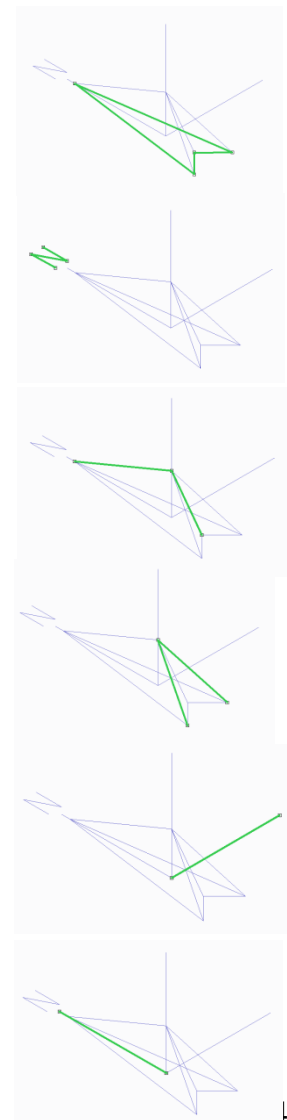
```
<M4IsoFlatGeometry alignmentAxis="North">
```

```
<M4IsoPolyline2D size="2" lineStyle="solid">
```

```
<M4IsoPoint2D x="0.00" y="0.00"/>
```

```
<M4IsoPoint2D x="0.52" y="0.00"/>
```

```
</M4IsoPolyline2D>
```



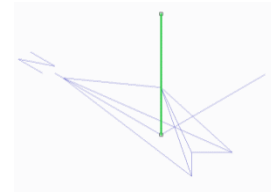
```

</M4IsoFlatGeometry>

<M4IsoFlatGeometry alignmentAxis="Z">
  <M4IsoPolyline2D size="2" lineStyle="solid">
    <M4IsoPoint2D x="0.00" y="0.00"/>
    <M4IsoPoint2D x="0.51" y="0.00"/>
  </M4IsoPolyline2D>
</M4IsoFlatGeometry>

</M4IsoNorthSymbolDescriptor>

```



## A generic North Arrow symbol

```

<M4IsoNorthSymbolDescriptor>

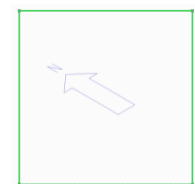
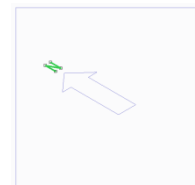
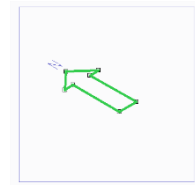
  <M4IsoProjectedGeometry>
    <M4IsoPolygon3D size="7" lineStyle="solid" fillStyle="none">
      <M4IsoPoint3D x="0.00" y="0.48" z="0.0"/>
      <M4IsoPoint3D x="-0.20" y="0.30" z="0.0"/>
      <M4IsoPoint3D x="-0.10" y="0.30" z="0.0"/>
      <M4IsoPoint3D x="-0.10" y="-0.25" z="0.0"/>
      <M4IsoPoint3D x="0.10" y="-0.25" z="0.0"/>
      <M4IsoPoint3D x="0.10" y="0.30" z="0.0"/>
      <M4IsoPoint3D x="0.20" y="0.30" z="0.0"/>
    </M4IsoPolygon3D>

    <M4IsoPolyline3D size="4" lineStyle="solid">
      <M4IsoPoint3D x="-0.03" y="0.55" z="0.0"/>
      <M4IsoPoint3D x="-0.03" y="0.67" z="0.0"/>
      <M4IsoPoint3D x="0.03" y="0.55" z="0.0"/>
      <M4IsoPoint3D x="0.03" y="0.67" z="0.0"/>
    </M4IsoPolyline3D>
  </M4IsoProjectedGeometry>

  <M4IsoFlatGeometry alignmentAxis="Z">
    <M4IsoPolygon2D size="4" lineStyle="solid">
      <M4IsoPoint2D x="0.90" y="0.90"/>
      <M4IsoPoint2D x="-0.90" y="0.90"/>
      <M4IsoPoint2D x="-0.90" y="-0.90"/>
      <M4IsoPoint2D x="0.90" y="-0.90"/>
    </M4IsoPolygon2D>
  </M4IsoFlatGeometry>

</M4IsoNorthSymbolDescriptor>

```



## A Yacht symbol

&lt;M4IsoNorthSymbolDescriptor&gt;

&lt;M4IsoProjectedGeometry&gt;

&lt;M4IsoPolygon3D size="9" lineStyle="solid" fillStyle="none"&gt;

&lt;M4IsoPoint3D x="0.00" y="0.50" z="0.0"/&gt;

&lt;M4IsoPoint3D x="-0.10" y="0.4" z="0.0"/&gt;

&lt;M4IsoPoint3D x="-0.15" y="0.3" z="0.0"/&gt;

&lt;M4IsoPoint3D x="-0.2" y="0.15" z="0.0"/&gt;

&lt;M4IsoPoint3D x="-0.1" y="-0.5" z="0.0"/&gt;

&lt;M4IsoPoint3D x="0.1" y="-0.5" z="0.0"/&gt;

&lt;M4IsoPoint3D x="0.2" y="0.15" z="0.0"/&gt;

&lt;M4IsoPoint3D x="0.15" y="0.3" z="0.0"/&gt;

&lt;M4IsoPoint3D x="0.1" y="0.4" z="0.0"/&gt;

&lt;/M4IsoPolygon3D&gt;

&lt;M4IsoPolygon3D size="7" lineStyle="solid" fillStyle="none"&gt;

&lt;M4IsoPoint3D x="0.00" y="0.35" z="-0.05"/&gt;

&lt;M4IsoPoint3D x="-0.1" y="0.2" z="-0.05"/&gt;

&lt;M4IsoPoint3D x="-0.15" y="0.1" z="-0.05"/&gt;

&lt;M4IsoPoint3D x="-0.1" y="-0.5" z="-0.05"/&gt;

&lt;M4IsoPoint3D x="0.1" y="-0.5" z="-0.05"/&gt;

&lt;M4IsoPoint3D x="0.15" y="0.1" z="-0.05"/&gt;

&lt;M4IsoPoint3D x="0.1" y="0.2" z="-0.05"/&gt;

&lt;/M4IsoPolygon3D&gt;

&lt;M4IsoPolyline3D size="2" lineStyle="solid"&gt;

&lt;M4IsoPoint3D x="0.0" y="0.50" z="0.0"/&gt;

&lt;M4IsoPoint3D x="0.0" y="0.35" z="-0.05"/&gt;

&lt;/M4IsoPolyline3D&gt;

&lt;M4IsoPolyline3D size="2" lineStyle="solid"&gt;

&lt;M4IsoPoint3D x="-0.1" y="-0.50" z="0.0"/&gt;

&lt;M4IsoPoint3D x="-0.1" y="-0.50" z="-0.05"/&gt;

&lt;/M4IsoPolyline3D&gt;

&lt;M4IsoPolyline3D size="2" lineStyle="solid"&gt;

&lt;M4IsoPoint3D x="0.1" y="-0.50" z="0.0"/&gt;

&lt;M4IsoPoint3D x="0.1" y="-0.50" z="-0.05"/&gt;

&lt;/M4IsoPolyline3D&gt;

&lt;M4IsoPolygon3D size="3" lineStyle="solid" fillStyle="solid"&gt;

&lt;M4IsoPoint3D x="0.0" y="0.0" z="0.50"/&gt;

&lt;M4IsoPoint3D x="0.0" y="-0.40" z="0.05"/&gt;

&lt;M4IsoPoint3D x="0.0" y="0.20" z="0.05"/&gt;

&lt;/M4IsoPolygon3D&gt;

&lt;/M4IsoProjectedGeometry&gt;

&lt;M4IsoFlatGeometry alignmentAxis="North"&gt;

&lt;M4IsoPolyline2D size="2" lineStyle="solid"&gt;

&lt;M4IsoPoint2D x="-0.52" y="0.00"/&gt;

&lt;M4IsoPoint2D x="0.52" y="0.00"/&gt;

&lt;/M4IsoPolyline2D&gt;

&lt;/M4IsoFlatGeometry&gt;

&lt;M4IsoFlatGeometry alignmentAxis="Z"&gt;

&lt;M4IsoPolyline2D size="2" lineStyle="solid"&gt;

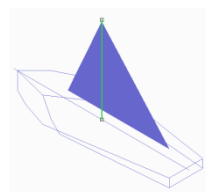
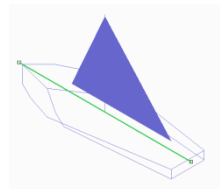
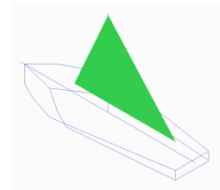
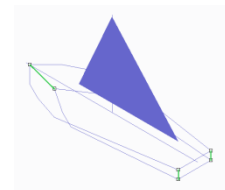
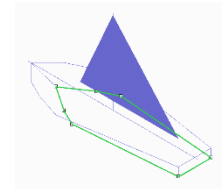
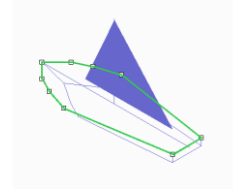
&lt;M4IsoPoint2D x="0.00" y="0.00"/&gt;

&lt;M4IsoPoint2D x="0.51" y="0.00"/&gt;

&lt;/M4IsoPolyline2D&gt;

&lt;/M4IsoFlatGeometry&gt;

&lt;/M4IsoNorthSymbolDescriptor&gt;



## 13 Appendix A – M4ISO Symbols

### 13.1 Introduction

The following pages contain representations of the isometric symbols supplied with M4ISO in the **M4IsoSymbolInfo.xml** file.

Each symbol entry contains the **M4IsoSymbolDefinition** entry followed by the diagram of the symbol, or a reference to another entry used to represent the symbol.

The blue lines are the geometry of the symbol that will be displayed by M4ISO. These are normally created using **M4IsoPolygon2D** and **M4IsoPolyline2D** objects

The dashed green line around the geometry is the **M4IsoDiamond** which is a bounding box to contain the geometry. This is used by the software to avoid overlapping geometry in the isometric and is not shown in the isometric.

The **Pn** text elements identify the **M4IsoCP2D** connection points of the symbol, with their position identified with a cross (X).

The **Dn** text elements identify the position of the **M4IsoDatum2D** points defined in the symbol, with their position identified with a cross (X).

There is also a smaller unnamed datum cross positioned at the x=0, y=0 point in the diagram.

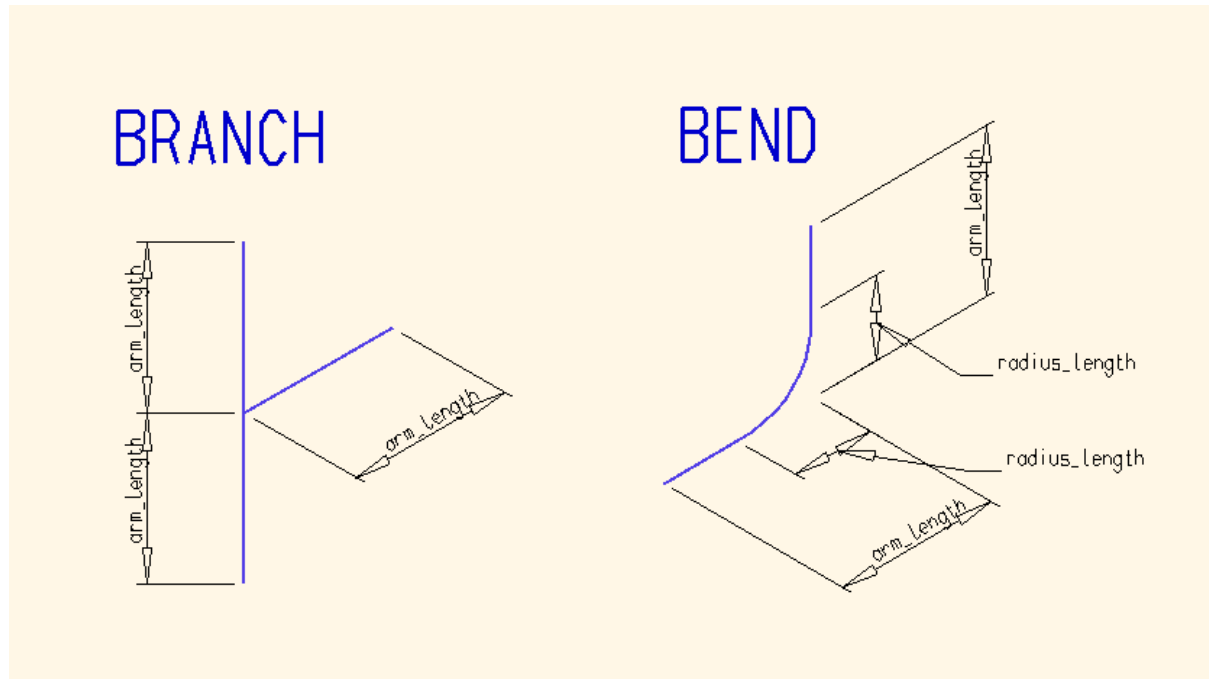
Note that it is common to have the marker crosses from different entities positioned at the same point on the diagram. The Pn, Dn texts and the datum cross are not shown in the isometric.

Please refer to the M4IsoSymbolInfo.xml file supplied with the M4ISOCreo installation for the definition of each symbol.

## 13.2 Built In

There are several symbols which are not fully defined in the **M4IsoSymbolInfo.xml** file but have a simple representation that is built into the M4ISO software.

In these instances the **M4IsoSymbolDescriptor** for the symbol is not available.



### 13.2.1 Elbow

Elbows are represented using the Bend definition shown above.

```
<M4IsoSymbolDefinition name="elbow" class="$bend$" descriptor="$bend$"/>
```

### 13.2.2 Elbow\_teed

Teed Elbows are represented using the Branch definition shown above.

```
<M4IsoSymbolDefinition name="elbow_teed" class="$branch$" descriptor="$branch$"/>
```

### 13.2.3 Tee

Tees are represented using the Branch definition shown above.

```
<M4IsoSymbolDefinition name="tee" class="$branch$" descriptor="$branch$"/>
```

### 13.2.4 Built In symbol dimensions

The dimensions of these built in symbols are defined in the M4IsoOptionStore.xml file

```
<M4IsoOption name="diagram/bend/arm_length" valueType="real" value="30"/>
<M4IsoOption name="diagram/bend/radius_length" valueType="real" value="20"/>

<M4IsoOption name="diagram/branch/arm_length" valueType="real" value="20"/>
```

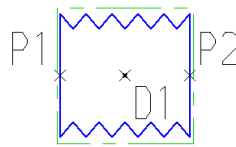
### 13.3 Isometric Symbols

A selection of the isometric symbols available with this version of M4ISOCreo are listed below in alphabetical order.

Each definition contains the M4IsoSymbolDefinition line for the symbol as defined in the M4IsoSymbolInfo.xml file. The M4ISO\_CLASS and M4ISO\_SYMBOL references are also given (for mapping purposes) and a picture of the symbol is shown.

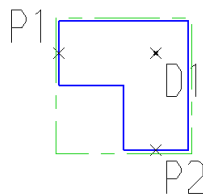
#### 13.3.1 Bellows

```
<M4IsoSymbolDefinition name="bellows" componentType="bellows"/>
M4ISO_CLASS="bellows", M4ISO_SYMBOL="bellows"
```



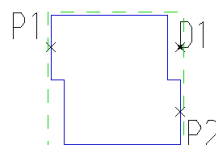
#### 13.3.2 Block Angle

```
<M4IsoSymbolDefinition name="block_angle"/>
M4ISO_CLASS="block_angle", M4ISO_SYMBOL="block_angle"
```



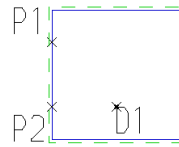
#### 13.3.3 Block Offset

```
<M4IsoSymbolDefinition name="block_offset"/>
M4ISO_CLASS="block_offset", M4ISO_SYMBOL="block_offset"
```



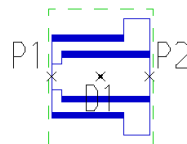
### 13.3.4 Block Return

```
<M4IsoSymbolDefinition name="block_return"/>
  M4ISO_CLASS="block_return", M4ISO_SYMBOL="block_return"
```



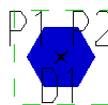
### 13.3.5 Bush Reducer

```
<M4IsoSymbolDefinition name="bush_reducer" componentType="bush_reducer"/>
  M4ISO_CLASS="bush_reducer", M4ISO_SYMBOL="bush_reducer"
```



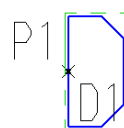
### 13.3.6 Butt Weld

```
<M4IsoSymbolDefinition name="butt_weld" isWeld="Yes"/>
  M4ISO_CLASS="butt_weld", M4ISO_SYMBOL="butt_weld"
```



### 13.3.7 Cap

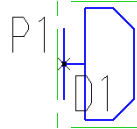
```
<M4IsoSymbolDefinition name="cap" componentType="cap"/>
  M4ISO_CLASS="cap", M4ISO_SYMBOL="cap"
```





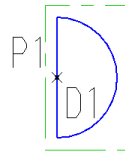
### 13.3.8 Cap Flanged

```
<M4IsoSymbolDefinition name="cap_fl" componentType="cap"/>
  M4ISO_CLASS="cap", M4ISO_SYMBOL="cap_fl"
```



### 13.3.9 Cap Round

```
<M4IsoSymbolDefinition name="cap_round" componentType="cap"/>
  M4ISO_CLASS="cap", M4ISO_SYMBOL="cap_round"
```



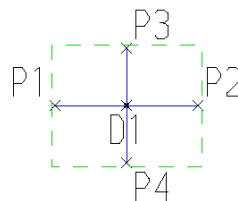
### 13.3.10 Coupling

```
<M4IsoSymbolDefinition name="coupling" descriptor="full_coupling"/>
  M4ISO_CLASS="coupling", M4ISO_SYMBOL="coupling"
```

See Full Coupling

### 13.3.11 Cross

```
<M4IsoSymbolDefinition name="cross"/>
  M4ISO_CLASS="cross", M4ISO_SYMBOL="cross"
```



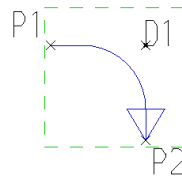
### 13.3.12 Elbow

```
<M4IsoSymbolDefinition name="elbow" class="$bend$" descriptor="$bend$"/>
  M4ISO_CLASS="elbow", M4ISO_SYMBOL="elbow"
```

Built In symbol

### 13.3.13 Elbow Reducing

```
<M4IsoSymbolDefinition name="elbow_reducing"/>
  M4ISO_CLASS="elbow_reducing", M4ISO_SYMBOL="elbow_reducing"
```



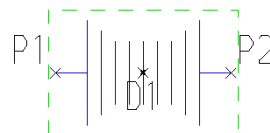
### 13.3.14 Elbow Teed

```
<M4IsoSymbolDefinition name="elbow_teed" class="$branch$" descriptor="$branch$"/>
  M4ISO_CLASS="elbow_teed", M4ISO_SYMBOL="elbow_teed"
```

Built In symbol

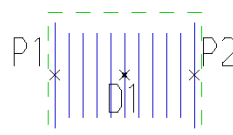
### 13.3.15 Expanding Bellows

```
<M4IsoSymbolDefinition name="expanding_bellows" componentType="bellows"/>
  M4ISO_CLASS="bellows", M4ISO_SYMBOL="expanding_bellows"
```



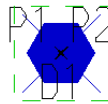
### 13.3.16 Expansion Joint

```
<M4IsoSymbolDefinition name="expansion_joint" componentType="bellows"/>
  M4ISO_CLASS="bellows", M4ISO_SYMBOL="expansion_joint"
```



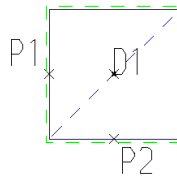
### 13.3.17 Field Weld

```
<M4IsoSymbolDefinition name="field_weld" isWeld="Yes"/>
  M4ISO_CLASS="field_weld", M4ISO_SYMBOL="field_weld"
```



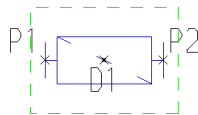
### 13.3.18 Filter Angle

```
<M4IsoSymbolDefinition name="filter_angle"/>
  M4ISO_CLASS="filter_angle", M4ISO_SYMBOL="filter_angle"
```



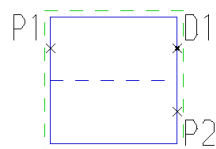
### 13.3.19 Filter Inline Flanged

```
<M4IsoSymbolDefinition name="filter_inline_fl" componentType="strainer"/>
  M4ISO_CLASS="strainer", M4ISO_SYMBOL="filter_inline_fl"
```



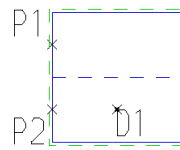
### 13.3.20 Filter Offset

```
<M4IsoSymbolDefinition name="filter_offset"/>
  M4ISO_CLASS="filter_offset", M4ISO_SYMBOL="filter_offset"
```



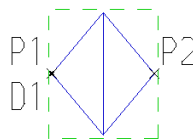
### 13.3.21 Filter Return

```
<M4IsoSymbolDefinition name="filter_return"/>
  M4ISO_CLASS="filter_return", M4ISO_SYMBOL="filter_return"
```



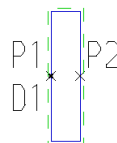
### 13.3.22 Flame Trap

```
<M4IsoSymbolDefinition name="flame_trap" />
  M4ISO_CLASS="flame_trap", M4ISO_SYMBOL="flame_trap"
```



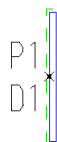
### 13.3.23 Flange (generic)

```
<M4IsoSymbolDefinition name="flange" componentType="flange" />
  M4ISO_CLASS="flange", M4ISO_SYMBOL="flange"
```



### 13.3.24 Flange Blind

```
<M4IsoSymbolDefinition name="flange_blind" componentType="flange" />
  M4ISO_CLASS="flange", M4ISO_SYMBOL="flange_blind"
```



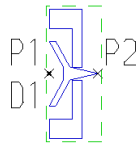
### 13.3.25 Flange Blind P2

```
<M4IsoSymbolDefinition name="flange_blind_p2" componentType="flange" />
  M4ISO_CLASS="flange", M4ISO_SYMBOL="flange_blind_p2"
```



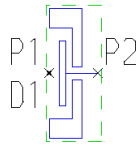
### 13.3.26 Flange Flare

```
<M4IsoSymbolDefinition name="flange_flare" componentType="flange"/>
  M4ISO_CLASS="flange", M4ISO_SYMBOL="flange_flare"
```



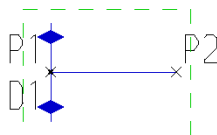
### 13.3.27 Flange Lap

```
<M4IsoSymbolDefinition name="flange_lap" componentType="flange"/>
  M4ISO_CLASS="flange", M4ISO_SYMBOL="flange_lap"
```



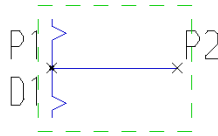
### 13.3.28 Flange with O-Ring

```
<M4IsoSymbolDefinition name="flange_oring" componentType="flange"/>
  M4ISO_CLASS="flange", M4ISO_SYMBOL="flange_oring"
```



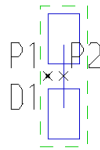
### 13.3.29 Flange with Groove

```
<M4IsoSymbolDefinition name="flange_rg" componentType="flange"/>
  M4ISO_CLASS="flange", M4ISO_SYMBOL="flange_rg"
```



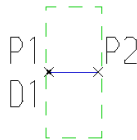
### 13.3.30 Flange Slip-On

```
<M4IsoSymbolDefinition name="flange_slipon" componentType="flange"/>
  M4ISO_CLASS="flange", M4ISO_SYMBOL="flange_slipon"
```



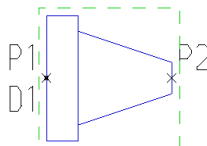
### 13.3.31 Flange Stub

```
<M4IsoSymbolDefinition name="flange_stub" componentType="flange"/>
  M4ISO_CLASS="flange", M4ISO_SYMBOL="flange_stub"
```



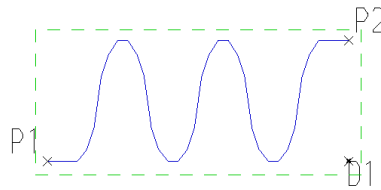
### 13.3.32 Flange Weld Neck

```
<M4IsoSymbolDefinition name="flange_wn" componentType="flange"/>
  M4ISO_CLASS="flange", M4ISO_SYMBOL="flange_wn"
```



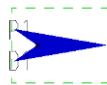
### 13.3.33 Flexible Hose

```
<M4IsoSymbolDefinition name="flex_hose"/>
  M4ISO_CLASS="flex_hose", M4ISO_SYMBOL="flex_hose"
```



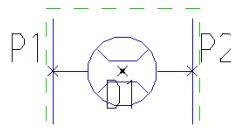
### 13.3.34 Flow Arrow

```
<M4IsoSymbolDefinition name="flow_arrow"/>
M4ISO_CLASS="flow_arrow", M4ISO_SYMBOL="flow_arrow"
```



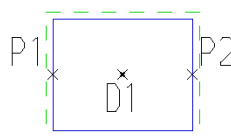
### 13.3.35 Flow Meter

```
<M4IsoSymbolDefinition name="flow_meter"/>
M4ISO_CLASS="flow_meter", M4ISO_SYMBOL="flow_meter"
```



### 13.3.36 Full Coupling

```
<M4IsoSymbolDefinition name="full_coupling"/>
M4ISO_CLASS="full_coupling", M4ISO_SYMBOL="full_coupling"
```



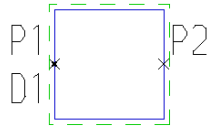
### 13.3.37 Gasket

```
<M4IsoSymbolDefinition name="gasket" class="$gasket$"/>
M4ISO_CLASS=" gasket", M4ISO_SYMBOL=" gasket"
```

Built In symbol

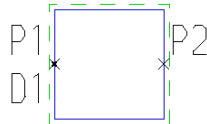
### 13.3.38 Half Coupling

```
<M4IsoSymbolDefinition name="half_coupling"/>
  M4ISO_CLASS="half_coupling", M4ISO_SYMBOL="half_coupling"
```



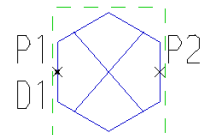
### 13.3.39 Half Coupling as Olet

```
<M4IsoSymbolDefinition name="half_coupling_as_olet" class="$olet$"
  componentType="olet"/>
  M4ISO_CLASS="olet", M4ISO_SYMBOL="half_coupling_as_olet"
```



### 13.3.40 Hose Coupling

```
<M4IsoSymbolDefinition name="hose_coupling"/>
  M4ISO_CLASS="hose_coupling", M4ISO_SYMBOL="hose_coupling"
```



### 13.3.41 Latrolet

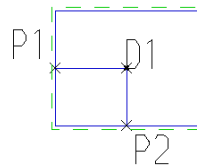
```
<M4IsoSymbolDefinition name="latrolet" class="$olet$" componentType="olet"
  descriptor="half_coupling_as_olet"/>
  M4ISO_CLASS="olet", M4ISO_SYMBOL="latrolet"
```

See Half Coupling as Olet

### 13.3.42 Miscellaneous 2 Angle

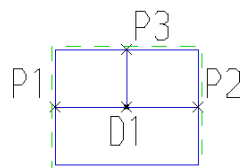
```
<M4IsoSymbolDefinition name="miscellaneous_2_angle"/>
  M4ISO_CLASS="miscellaneous_2_angle", M4ISO_SYMBOL="miscellaneous_2_angle"
```





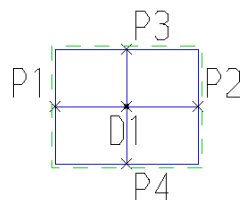
### 13.3.43 Miscellaneous Branch

```
<M4IsoSymbolDefinition name="miscellaneous_branch"/>
M4ISO_CLASS="miscellaneous_branch", M4ISO_SYMBOL="miscellaneous_branch"
```



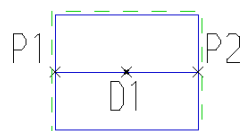
### 13.3.44 Miscellaneous Cross

```
<M4IsoSymbolDefinition name="miscellaneous_cross"/>
M4ISO_CLASS="miscellaneous_cross", M4ISO_SYMBOL="miscellaneous_cross"
```



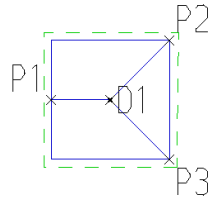
### 13.3.45 Miscellaneous Inline

```
<M4IsoSymbolDefinition name="miscellaneous_inline"/>
M4ISO_CLASS="miscellaneous_inline", M4ISO_SYMBOL="miscellaneous_inline"
```



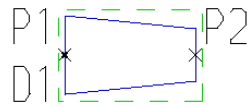
### 13.3.46 Miscellaneous Y

```
<M4IsoSymbolDefinition name="miscellaneous_y"/>
M4ISO_CLASS="miscellaneous_y", M4ISO_SYMBOL="miscellaneous_y"
```



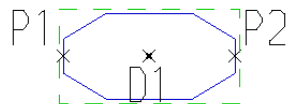
### 13.3.47 Nipolet

```
<M4IsoSymbolDefinition name="nipolet" class="$olet$" componentType="olet"/>
  M4ISO_CLASS="olet", M4ISO_SYMBOL="nipolet"
```



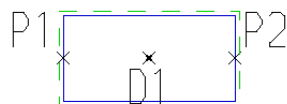
### 13.3.48 Nipple Barrel

```
<M4IsoSymbolDefinition name="nipple_barrel"/>
  M4ISO_CLASS="nipple_barrel", M4ISO_SYMBOL="nipple_barrel"
```



### 13.3.49 Nipple Running

```
<M4IsoSymbolDefinition name="nipple_running"/>
  M4ISO_CLASS="nipple_running", M4ISO_SYMBOL="nipple_running"
```



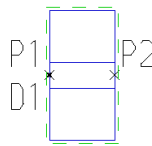
### 13.3.50 Olet

```
<M4IsoSymbolDefinition name="olet" class="$olet$" componentType="olet"
  descriptor="sockolet"/>
  M4ISO_CLASS="olet", M4ISO_SYMBOL="olet"
```

See Sockolet

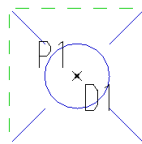
### 13.3.51 Orifice

```
<M4IsoSymbolDefinition name="orifice"/>
M4ISO_CLASS="orifice", M4ISO_SYMBOL="orifice"
```



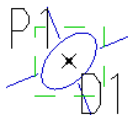
### 13.3.52 Penetration

```
<M4IsoSymbolDefinition name="penetration" class="$support$"
    componentType="support"/>
M4ISO_CLASS="support", M4ISO_SYMBOL="penetration"
```



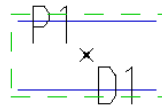
### 13.3.53 Penetration Small

```
<M4IsoSymbolDefinition name="penetration_small" class="$support$"
    componentType="support"/>
M4ISO_CLASS="support", M4ISO_SYMBOL="penetration_small"
```



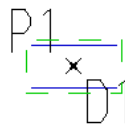
### 13.3.54 Pipe Hanger

```
<M4IsoSymbolDefinition name="pipe_hanger" class="$support$"
    componentType="support"/>
M4ISO_CLASS="support", M4ISO_SYMBOL="pipe_hanger"
```



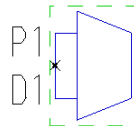
### 13.3.55 Pipe Hanger Small

```
<M4IsoSymbolDefinition name="pipe_hanger_small" class="$support$"  
    componentType="support" />  
M4ISO_CLASS="support", M4ISO_SYMBOL="pipe_hanger_small"
```



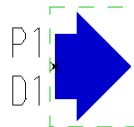
### 13.3.56 Plug

```
<M4IsoSymbolDefinition name="plug"/>  
M4ISO_CLASS="plug", M4ISO_SYMBOL="plug"
```



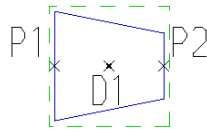
### 13.3.57 Plug MB

```
<M4IsoSymbolDefinition name="plug_mb"/>  
M4ISO_CLASS="plug_mb", M4ISO_SYMBOL="plug_mb"
```



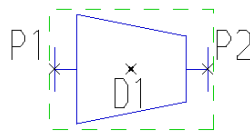
### 13.3.58 Reducer Concentric

```
<M4IsoSymbolDefinition name="reducer_concentric"  
    componentType="reducer_concentric"/>  
M4ISO_CLASS="reducer_concentric", M4ISO_SYMBOL="reducer_concentric"
```



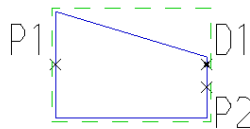
### 13.3.59 Reducer\_Concentric Flanged

```
<M4IsoSymbolDefinition name="reducer_concentric_fl"
                        componentType="reducer_concentric"/>
M4ISO_CLASS="reducer_concentric", M4ISO_SYMBOL="reducer_concentric_fl"
```



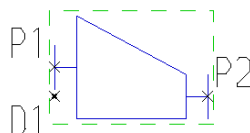
### 13.3.60 Reducer Eccentric

```
<M4IsoSymbolDefinition name="reducer_eccentric"
                        componentType="reducer_eccentric"/>
M4ISO_CLASS="reducer_eccentric", M4ISO_SYMBOL="reducer_eccentric"
```



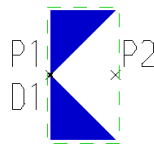
### 13.3.61 Reducer Eccentric Flanged

```
<M4IsoSymbolDefinition name="reducer_eccentric_fl"
                        componentType="reducer_eccentric"/>
M4ISO_CLASS="reducer_eccentric", M4ISO_SYMBOL="reducer_eccentric_fl"
```



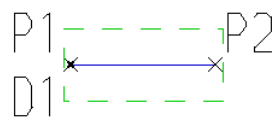
### 13.3.62 Screwed Connector

```
<M4IsoSymbolDefinition name="screwed_connector"/>
M4ISO_CLASS="screwed_connector", M4ISO_SYMBOL="screwed_connector"
```



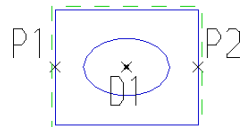
### 13.3.63 Setin

```
<M4IsoSymbolDefinition name="setin" class="$olet$" />
  M4ISO_CLASS="olet", M4ISO_SYMBOL="setin"
```



### 13.3.64 Sight Glass

```
<M4IsoSymbolDefinition name="sight_glass" />
  M4ISO_CLASS="sight_glass", M4ISO_SYMBOL="sight_glass"
```



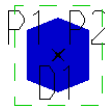
### 13.3.65 Shop Weld

```
<M4IsoSymbolDefinition name="shop_weld" />
  M4ISO_CLASS="shop_weld", M4ISO_SYMBOL="shop_weld"
```



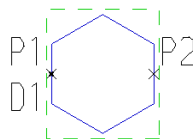
### 13.3.66 Socket Weld

```
<M4IsoSymbolDefinition name="socket_weld" />
  M4ISO_CLASS="socket_weld", M4ISO_SYMBOL="socket_weld"
```



### 13.3.67 Sockolet

```
<M4IsoSymbolDefinition name="sockolet" class="$olet$" componentType="olet"/>
  M4ISO_CLASS="olet", M4ISO_SYMBOL="sockolet"
```



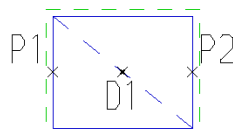
### 13.3.68 Spec Break

```
<M4IsoSymbolDefinition name="specbrk"/>
  M4ISO_CLASS="specbrk", M4ISO_SYMBOL="specbrk"
```



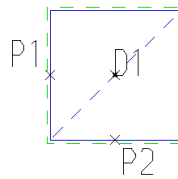
### 13.3.69 Strainer

```
<M4IsoSymbolDefinition name="strainer" componentType="strainer"/>
  M4ISO_CLASS="strainer", M4ISO_SYMBOL="strainer"
```



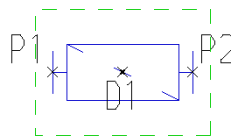
### 13.3.70 Strainer Angle

```
<M4IsoSymbolDefinition name="strainer_angle" componentType="strainer"/>
  M4ISO_CLASS="strainer", M4ISO_SYMBOL="strainer_angle"
```



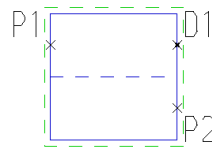
### 13.3.71 Strainer Inline Flanged

```
<M4IsoSymbolDefinition name="strainer_inline_fl" componentType="strainer"/>
  M4ISO_CLASS="strainer", M4ISO_SYMBOL="strainer_inline_fl"
```



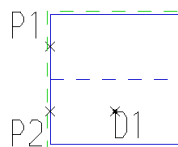
### 13.3.72 Strainer Offset

```
<M4IsoSymbolDefinition name="strainer_offset" componentType="strainer"/>
  M4ISO_CLASS="strainer", M4ISO_SYMBOL="strainer_offset"
```



### 13.3.73 Strainer Return

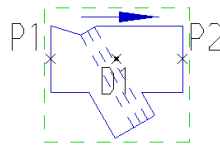
```
<M4IsoSymbolDefinition name="strainer_return" componentType="strainer"/>
  M4ISO_CLASS="strainer", M4ISO_SYMBOL="strainer_return"
```



### 13.3.74 Strainer Wye

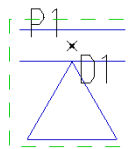
```
<M4IsoSymbolDefinition name="strainer_wye" componentType="strainer"/>
  M4ISO_CLASS="strainer", M4ISO_SYMBOL="strainer_wye"
```





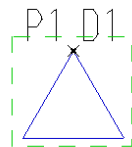
### 13.3.75 Support Guide

```
<M4IsoSymbolDefinition name="support_guide" class="$support$"
                        componentType="support"/>
M4ISO_CLASS="support", M4ISO_SYMBOL="support_guide"
```



### 13.3.76 Support Skid

```
<M4IsoSymbolDefinition name="support_skid" class="$support$"
                        componentType="support"/>
M4ISO_CLASS="support", M4ISO_SYMBOL="support_skid"
```



### 13.3.77 Sweepolet

```
<M4IsoSymbolDefinition name="sweepolet" class="$olet$"
                        componentType="olet" descriptor="sockolet"/>
M4ISO_CLASS="olet", M4ISO_SYMBOL="sweepolet"
```

See Sockolet

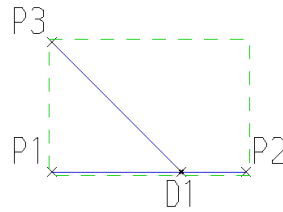
### 13.3.78 Tee

```
<M4IsoSymbolDefinition name="tee" class="$branch$" descriptor="$branch$"/>
M4ISO_CLASS="tee", M4ISO_SYMBOL="tee"
```

Built In symbol

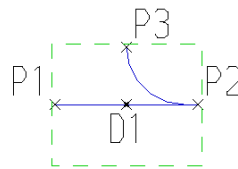
### 13.3.79 Tee 45

```
<M4IsoSymbolDefinition name="tee_45" class="$branch$" />
M4ISO_CLASS="tee_45", M4ISO_SYMBOL="tee_45"
```



### 13.3.80 Tee Pitcher

```
<M4IsoSymbolDefinition name="tee_pitcher" class="$branch$" />
M4ISO_CLASS="tee_pitcher", M4ISO_SYMBOL="tee_pitcher"
```



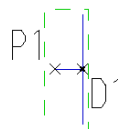
### 13.3.81 Tee Set On

```
<M4IsoSymbolDefinition name="tee_set_on" class="$olet$"
componentType="olet" descriptor="setin"/>
M4ISO_CLASS="olet", M4ISO_SYMBOL="tee_set_on"
```

See Setin

### 13.3.82 Terminal Equipment

```
<M4IsoSymbolDefinition name="terminal_equipment" class="$terminal$"
componentType="terminal"/>
M4ISO_CLASS="terminal", M4ISO_SYMBOL="terminal_equipment"
```



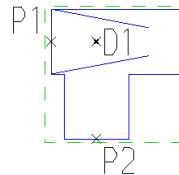
### 13.3.83 Threadolet

```
<M4IsoSymbolDefinition name="threadolet" class="$olet$" componentType="olet"
descriptor="sockolet"/>
M4ISO_CLASS="olet", M4ISO_SYMBOL="threadolet"
```

See Sockolet

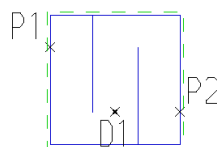
### 13.3.84 Trap Angle

```
<M4IsoSymbolDefinition name="trap_angle"/>
M4ISO_CLASS="trap_angle", M4ISO_SYMBOL="trap_angle"
```



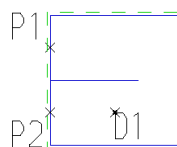
### 13.3.85 Trap Offset

```
<M4IsoSymbolDefinition name="trap_offset"/>
M4ISO_CLASS="trap_offset", M4ISO_SYMBOL="trap_offset"
```



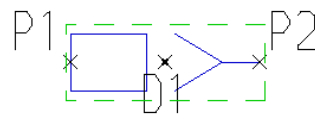
### 13.3.86 Trap Return

```
<M4IsoSymbolDefinition name="trap_return"/>
M4ISO_CLASS="trap_return", M4ISO_SYMBOL="trap_return"
```



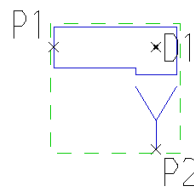
### 13.3.87 Tundish

```
<M4IsoSymbolDefinition name="tundish"/>
M4ISO_CLASS="tundish", M4ISO_SYMBOL="tundish"
```



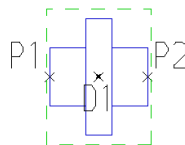
### 13.3.88 Tundish Angle

```
<M4IsoSymbolDefinition name="tundish_angle"/>
M4ISO_CLASS="tundish_angle", M4ISO_SYMBOL="tundish_angle"
```



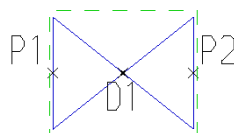
### 13.3.89 Union

```
<M4IsoSymbolDefinition name="union"/>
M4ISO_CLASS="union", M4ISO_SYMBOL="union"
```



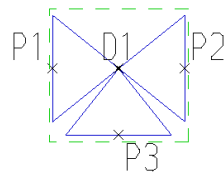
### 13.3.90 Valve (generic)

```
<M4IsoSymbolDefinition name="valve" componentType="valve"/>
M4ISO_CLASS="valve", M4ISO_SYMBOL="valve"
```



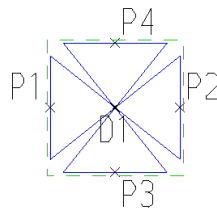
### 13.3.91 Valve 3way

```
<M4IsoSymbolDefinition name="valve_3way" componentType="valve "/>
M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_3way"
```



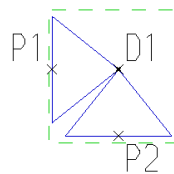
### 13.3.92 Valve 4way

```
<M4IsoSymbolDefinition name="valve_4way" componentType="valve" />
  M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_4way"
```



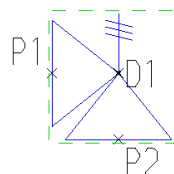
### 13.3.93 Valve Angle

```
<M4IsoSymbolDefinition name="valve_angle" componentType="valve" />
  M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_angle"
```



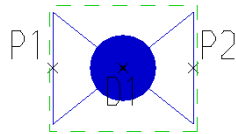
### 13.3.94 Valve Angle Relief

```
<M4IsoSymbolDefinition name="valve_angle_relief" componentType="valve" />
  M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_angle_relief"
```



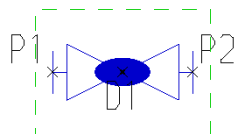
### 13.3.95 Valve Ball

```
<M4IsoSymbolDefinition name="valve_ball" componentType="valve" />
  M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_ball"
```



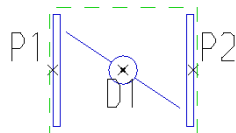
### 13.3.96 Valve Ball Flange

```
<M4IsoSymbolDefinition name="valve_ball_fl" componentType="valve"/>
  M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_ball_fl"
```



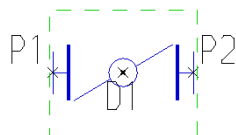
### 13.3.97 Valve Butterfly

```
<M4IsoSymbolDefinition name="valve_butterfly" componentType="valve"/>
  M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_butterfly"
```



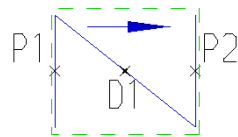
### 13.3.98 Valve Butterfly Flange

```
<M4IsoSymbolDefinition name="valve_butterfly_fl" componentType="valve"/>
  M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_butterfly_fl"
```



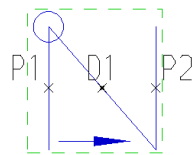
### 13.3.99 Valve Check

```
<M4IsoSymbolDefinition name="valve_check" componentType="valve"/>
  M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_check"
```



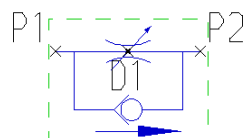
### 13.3.100 Valve Check 2

```
<M4IsoSymbolDefinition name="valve_check2" componentType="valve"/>
M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_check2"
```



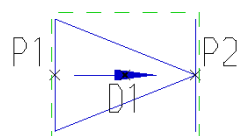
### 13.3.101 Valve Check 3

```
<M4IsoSymbolDefinition name="valve_check3" componentType="valve"/>
M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_check3"
```



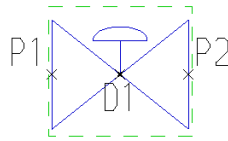
### 13.3.102 Valve Check 4

```
<M4IsoSymbolDefinition name="valve_check4" componentType="valve"/>
M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_check4"
```



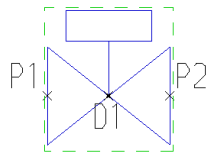
### 13.3.103 Valve Control

```
<M4IsoSymbolDefinition name="valve_control" componentType="valve"/>
M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_control"
```



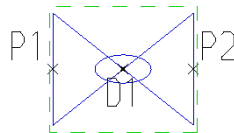
### 13.3.104 Valve Control SQIN

```
<M4IsoSymbolDefinition name="valve_control_sqin" componentType="valve"/>
M4ISO_CLASS="valv", M4ISO_SYMBOL="valve_control_sqin"
```



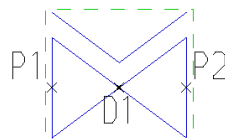
### 13.3.105 Valve Diaphragm

```
<M4IsoSymbolDefinition name="valve_diaphragm" componentType="valve"/>
M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_diaphragm"
```



### 13.3.106 Valve Diaphragm 2

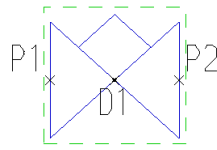
```
<M4IsoSymbolDefinition name="valve_diaphragm2" componentType="valve"/>
M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_diaphragm2"
```



### 13.3.107 Valve Diaphragm 3

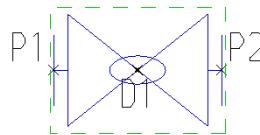
```
<M4IsoSymbolDefinition name="valve_diaphragm3" componentType="valve"/>
M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_diaphragm3"
```





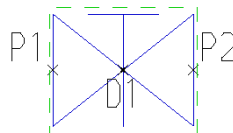
### 13.3.108 Valve Diaphragm Flange

```
<M4IsoSymbolDefinition name="valve_diaphragm_fl" componentType="valve"/>
  M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_diaphragm_fl"
```



### 13.3.109 Valve Gate

```
<M4IsoSymbolDefinition name="valve_gate" componentType="valve"/>
  M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_gate"
```



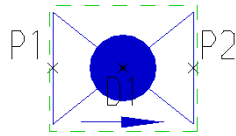
### 13.3.110 Valve Generic

```
<M4IsoSymbolDefinition name="valve_generic" componentType="valve"
  descriptor="valve"/>
  M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_generic"
```

See Valve

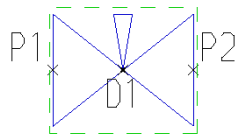
### 13.3.111 Valve Globe

```
<M4IsoSymbolDefinition name="valve_globe" componentType="valve"/>
  M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_globe"
```



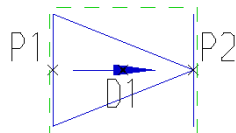
### 13.3.112 Valve Needle

```
<M4IsoSymbolDefinition name="valve_needle" componentType="valve"/>
  M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_needle"
```



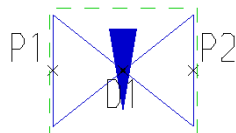
### 13.3.113 Valve Non-Return

```
<M4IsoSymbolDefinition name="valve_nonreturn" componentType="valve"/>
  M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_nonreturn"
```



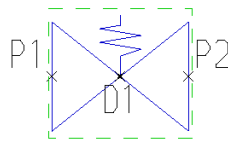
### 13.3.114 Valve Plug

```
<M4IsoSymbolDefinition name="valve_plug" componentType="valve"/>
  M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_plug"
```



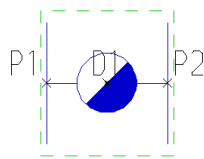
### 13.3.115 Valve Pressure Reducer

```
<M4IsoSymbolDefinition name="valve_presred" componentType="valve"/>
  M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_presred"
```



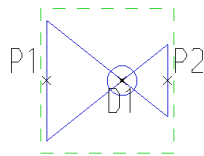
### 13.3.116 Valve Reducer

```
<M4IsoSymbolDefinition name="valve_reducer" componentType="valve"/>
M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_reducer"
```



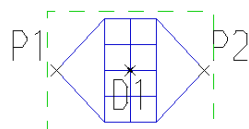
### 13.3.117 Valve Reducer 2

```
<M4IsoSymbolDefinition name="valve_reducer2" componentType="valve"/>
M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_reducer_2"
```



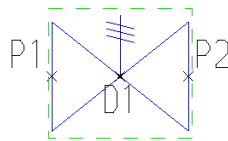
### 13.3.118 Valve Reducer 3

```
<M4IsoSymbolDefinition name="valve_reducer3" componentType="valve"/>
M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_reducer_3"
```



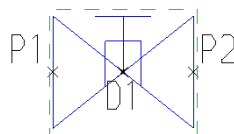
### 13.3.119 Valve Relief

```
<M4IsoSymbolDefinition name="valve_relief" componentType="valve"/>
M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_relief"
```



### 13.3.120 Valve Slide

```
<M4IsoSymbolDefinition name="valve_slide" componentType="valve"/>
M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_slide"
```



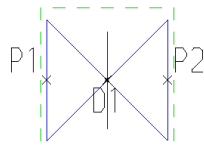
### 13.3.121 Valve Swing Check

```
<M4IsoSymbolDefinition name="valve_swingcheck" componentType="valve"
                        descriptor="valve_check2"/>
M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_swingcheck"
```

See Valve Check 2

### 13.3.122 Valve Trap

```
<M4IsoSymbolDefinition name="valve_trap" componentType="valve"/>
M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_trap"
```



### 13.3.123 Valve Wafer

```
<M4IsoSymbolDefinition name="valve_wafer" componentType="valve"
                        descriptor="valve_butterfly"/>
M4ISO_CLASS="valve", M4ISO_SYMBOL="valve_wafer"
```

See Valve Butterfly

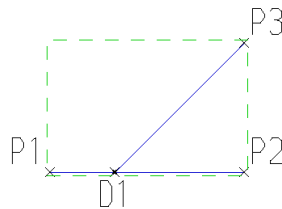
### 13.3.124 Weldolet

```
<M4IsoSymbolDefinition name="weldolet" class="$olet$" componentType="olet"
    descriptor="sockolet"/>
    M4ISO_CLASS="olet", M4ISO_SYMBOL="weldolet"
```

See Sockolet

### 13.3.125 WYE Lat Tee

```
<M4IsoSymbolDefinition name="ylat"/>
    M4ISO_CLASS="ylat", M4ISO_SYMBOL="ylat"
```



## 14 Appendix B

### 14.1 The Pen table configuration

```

! =====
! Pen Table System Colors
! =====
!
! Creo System Color           ==> Pen Table Color Name
! Geometry                     ==> drawing_color
! Curve                        ==> curve_color*
! Manufacturing Volume          ==> magenta_color
! Quilt                         ==> quilt_color
! Datum                        ==> datum_color
! Preselection Highlight       ==> presel_highlight_color*
! Preview Geometry             ==> preview_geom_color*
! Secondary Preview Geom       ==> secondary_preview_color*
! Letter                       ==> letter_color*
! Hidden Line                  ==> half_tone_color
! Highlight-Primary            ==> highlight_color
! Selected                     ==> selected_color*
! Secondary Selected           ==> secondary_selected_color*
! Sheet Metal                  ==> attention_color
! Sketched Curve               ==> section_color
! Highlight-Secondary          ==> dimmed_color
! Highlight-Edge               ==> edge_highlite_color
!
! *These colors are reserved for specific purposes and may not be assigned to other entities/uses.
!
!
! =====
! Default line patterns:
! =====
! pen (#) pattern (1st line length) (1st space length) (2nd linelength) (2nd space length) (units)
! pen 3 pattern 0.1 0.05 0.025 0.05 in ( ____ - ____ - ____ - ____ - ____ _ )
!
! =====
! Default line thickness:
! =====
! pen (#) thickness (value) (units)
! pen 1 thickness 0.05 in
! pen 2 thickness 0.5 cm
!
! =====
! Default pen colors:
! =====
! pen (#) color (unit% of red) (unit% of green) (unit% of blue)
! pen 7 color 0.0 0.5 1.0
!
! =====

```

```

! Mapping entities to other pens:
! =====
! pen (#) (system_color_name)
! pen 1 dimmed_color
! pen 2 letter_color background_color
! pen 3 section_color; dimmed_color
!
! =====
! Pen Table Examples
! =====
!
! pen 1 color 0.0 0.0 0.7; highlight_color
! pen 2 thickness 0.5 cm; letter_color; dimmed_color
! pen 5 pattern 1.0 0.1 0.5 0.01 in; color 1.0 0.0 1.0; drawing color
!
!
! *Note: If a pen table file called table.pnt is in the working directory,
! Pro/ENGINEER will automatically use it when creating plot files even though
! it is not explicitly selected by the user.
!
!
! =====
! Pen Table Default Weight
! =====
! Set in config.pro
!
! pen1_line_weight 4
! pen2_line_weight 1
! pen3_line_weight 2
! pen4_line_weight 3
! pen5_line_weight 2
! pen6_line_weight 3
! pen7_line_weight 1
! pen8_line_weight 4
!
!
! Each increment value equals a value of 0.005 inches.
!
! pen 1 thickness 0.020 in; color 0.0 0.0 0.0
! pen 2 thickness 0.005 in; color 0.0 0.0 0.0
! pen 3 thickness 0.010 in; color 0.0 0.0 0.0
! pen 4 thickness 0.015 in; color 0.0 0.0 0.0
! pen 5 thickness 0.010 in; color 0.0 0.0 0.0
! pen 6 thickness 0.015 in; color 0.0 0.0 0.0
! pen 7 thickness 0.005 in; color 0.0 0.0 0.0
! pen 8 thickness 0.020 in; color 0.0 0.0 0.0
!
! Source: page 306. Kelley, David S. "Chapter 8 Creating a Pro/Engineer Drawing."
! ProENGINEER 2001 Instructor. 1st. McGraw-Hill, 2001. ISBN 0072510072

```

```

! =====
! Suggested Pen Table Mapping
! =====
!
! Pen 1 Mapping:
! * Outline Geometry, Section Text / Arrows, Object Lines
! * White Entities
! * Plot with solid lines: Visible geometry, Cross-section cutting plane arrows and text,
! * Drawing format and boundary, Tag text,
! * Centerline line font with white color, Brown portion of datum planes;
! * Plot with phantom lines: Cross-section cutting plane lines
!
! Pen 2 Mapping:
! * Dimension Lines and Leaders, Cross Hatching, Text**
! * Yellow Entities
! * Plot with solid lines: Dimension lines, Geometric tolerance lines,
! * All text (except cross-section text),
! * Balloon notes, Cross-hatching, Centerline line font with yellow color, Leaders;
! * Plot with centerlines: Axes and centerlines
! (line_style_standard = std_ansi**)
!
! Pen 3 Mapping:
! * Hidden Lines
! * Gray Entities
! * Plot with dashed lines with phantom font: Hidden lines
!
! Pen 4 Mapping:
! * Drawing Frame
! * Red Entities (Primary Highlight (Dark Red), Selected (Red), Secondary Selected (Orange))
! * Plot with solid lines: Spline surface grid (does not plot in drawings)
!
! Pen 5 Mapping:
! * Sheet Metal
! * Green Entities
! * Plot with solid lines: Sheet metal color entities
!
! Pen 6 Mapping:
! * Sketch Lines, Section
! * Cyan Entities
! * Plot with solid lines: Sketcher section entities
!
! Pen 7 Mapping:
! * Tangent Edges (Dimmed e.g. Tangent Edges)
! * Dark Gray Entities
! * Plot with solid lines: Toggled sections, Grayed dimensions and text, Dimmed tangent edges,
! * Dark gray portion of datum planes
!
! Pen 8 Mapping:
! * Text** (Spline Surface Grid) [use for drawing perimeter border in format]
! * Blue Entities
! * Plot with solid lines: Spline surface grid

```



```
! (line_style_standard = std_iso**)
!  
! **Note that the text pen is reassigned according to line_style_standard set  
!      in the drawing config file.  
!  
! Source:  
! PTC Suggested Technique for Creating Plotter Configuration Files  
! http://support.ptc.com/cs/cs\_27/howto/plt737/plt737.htm  
!
```