

# Using PCF Files from CAD to PypeServer

---

This document explains the meaning of data in a PCF file and how to structure it to get the best flow of data from your CAD system/workflow into PypeServer.

## Contents

1	Configuring your CAD for exporting the right PCF data.....	2
2	Specifying Scheduled Pipe or Wall Thickness in your PCF file .....	2
2.1	Specifying the NPS size in "ITEM-DESCRIPTION" .....	2
2.2	Specifying NPS sizes in the MATERIALS Section.....	3
3	Intersecting PIPE to PIPE components.....	3
4	Declaring TEE-SET- Types as Pipe (and not fittings).....	4
4.1	Treat all TEE-SET-ON/IN/STUB items as pipe.....	4
4.2	Explicitly define each TEE-SET-ON/IN/STUB item as a pipe.....	4
5	Mapping additional fields .....	4
5.1	Using Cut Templates .....	5
5.2	Mapping OLETs .....	5
5.3	OLET lookup Tables.....	5
5.4	Getting your OLETs to show up as OLETs in PCF.....	6
5.4.1	OLET Example with the set-on HoleSize specified in the ITEM-DESCRIPTION.....	7
5.4.2	Flagging end-cuts and holes so that they must be verified later.....	7
5.5	Mapping Additional data .....	8
5.5.1	Setting Pipe Sizes Pipe Non-Standard Size example .....	8
5.5.2	Mapping Cut Templates to End Cuts for PIPE and the outer end of a TEE-SET- type.....	8
5.5.3	Mapping Cut Templates to the Set-On side of a TEE-SET-ON.....	9
5.5.4	Mapping Cut Templates for the hole created by an OLET .....	9
5.5.5	Creating an OLET Mapping Lookup file for use in PypeServer.....	10
6	One PCF Example .....	11
6.1	The Spool Drawing .....	11
6.2	The PCF File with Comments .....	12
7	CAD Specific Information .....	14
7.1	AutoDesk Fabrication CAD MEP.....	14
7.1.1	Getting OLETs to output as OLET instead of TEE-SET-ON .....	14

## 1 Configuring your CAD for exporting the right PCF data

PypeServer does not provide support for how to export from CAD systems. (There are too many CAD systems and too many add-ons, configurations, and customer work-flows.) Please see the document “Importers” in the PypeServer training system for information PypeServer has gathered on exporting to PCF from various CAD systems.

If you find any helpful videos, please email them to [Support@pypeserver.com](mailto:Support@pypeserver.com) so that we can add them to our Importers document.

## 2 Specifying Scheduled Pipe or Wall Thickness in your PCF file

PCF files do not require the specification of NPS pipe sizes or pipe wall thickness, thus it is up to the user to include information that will allow the PCF importer to determine pipe wall thicknesses.

Without any field mapping, the PCF importer will parse pipe NPS schedule sizes by looking for specific keywords in one of two places. You can also customize the keywords and location by creating a field mapping. The default behavior is discussed in this section. Section 5 discusses field mapping.

NPS schedule sizes are specified by the keyword SCH, or SCHEDULE, followed by a number, or simply the keyword STD, or SCH STD.

### 2.1 Specifying the NPS size in “ITEM-DESCRIPTION”

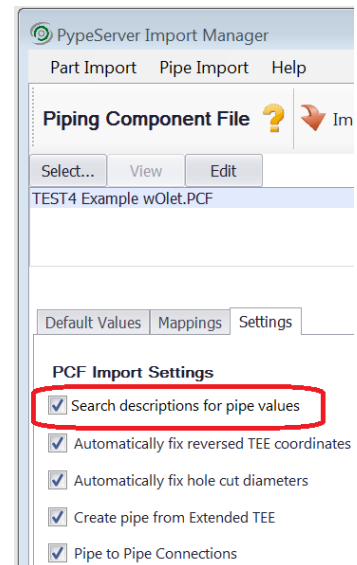
The most common location to locate this keyword (and schedule size) is in the “ITEM-DESCRIPTION” text for a given PIPE (or other parsed component) item.

#### Example

```
PIPE
END-POINT 34.5570 8.8345 -0.0000 8.0000 PL
END-POINT -37.5680 8.8345 -0.0000 8.0000 PL
ITEM-DESCRIPTION A53B GR B ERW SCH 40
```

This ITEM-DESCRIPTION search location is not enabled by default. You can enable this in Import Settings for the PCF Importer. Change this setting as shown to the right:

Note: Using ITEM-DESCRIPTION will replace the Standard functionality described in the previous section. When mapping (covered in the “Using the Field Mapper” document), this is commonly the preferred method.



## 2.2 Specifying NPS sizes in the MATERIALS Section

By default the PCF importer will follow the PCF specification, which specifies that pipe values and attributes of pipes (such as pipe schedule and wall thickness) be defined in the MATERIALS section of the PCF file<sup>1</sup>. The MATERIALS section is divided into sub-sections that are indexed by an ITEM-CODE or a MATERIAL-IDENTIFIER attribute.

### Example

```
PIPE
  END-POINT 34.5570 8.8345 -0.0000 8.0000 PL
  END-POINT -37.5680 8.8345 -0.0000 8.0000 PL
  MATERIAL-IDENTIFIER 2
  ITEM-DESCRIPTION A53B GR B ERW
MATERIALS
  MATERIAL-IDENTIFIER 2
  DESCRIPTION A53B GR B ERW SCHEDULE 40
```

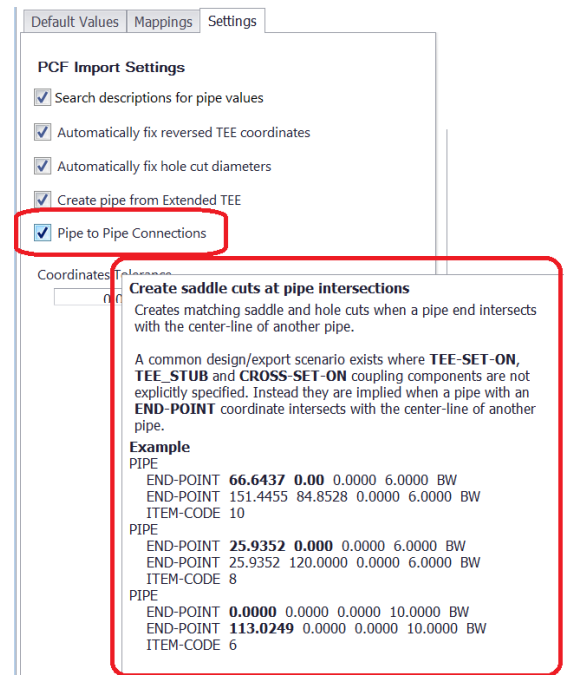
Note: The SCHEDULE keyword can also be SCH

## 3 Intersecting PIPE to PIPE components

Connecting Pipes to Pipes (via saddle cuts) is not defined in the PCF specification, however PypeServer supports this.

To enable this, in the PCF importer settings, check the box circled (shown to the right). Additional description is seen by hovering over the selection.

Pipe to pipe intersections will be interpreted as saddles set on the run pipe, with a hole the ID of the set-on pipe.



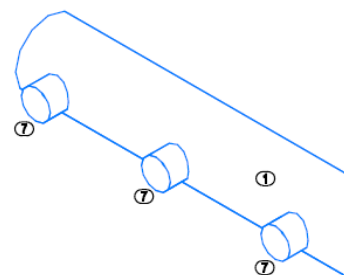
<sup>1</sup> See the PCF Reference Guide (in the training system) for more details—p 27 and p 327.

## 4 Declaring TEE-SET- Types as Pipe (and not fittings)

Customers often specify connecting pipes as TEE-SET-ONs or TEE-SET-INs.

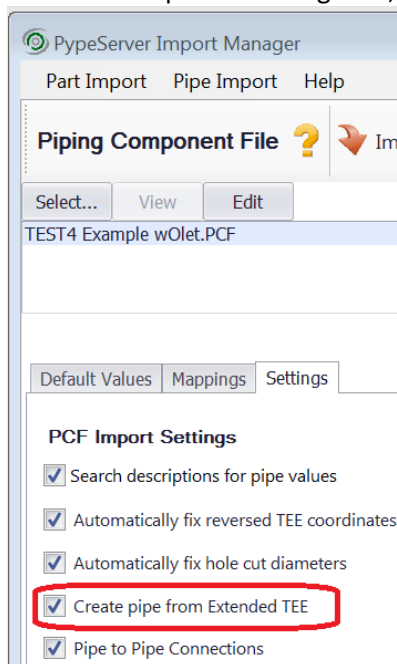
Shown to the right is an example of a 36" pipe with 16" saddle pipes on the run.

While not defined per the .pcf specification, this is supported in PypeServer two ways, as discussed in the next two subsections.



### 4.1 Treat all TEE-SET-ON/IN/STUB items as pipe

In the PCF Importer Settings tab, check the item circled below.



### 4.2 Explicitly define each TEE-SET-ON/IN/STUB item as a pipe

The other way to declare TEE SET ON/IN/STUB items as pipe is by using the “PSPIPE” keyword in the ITEM-DESCRIPTION (or MATERIAL DESCRIPTION if you are using that field).

```
TEE-SET-ON
CENTRE-POINT -159.5067 -0.0000 120.3750
BRANCH1-POINT -159.5067 -28.0000 120.3750 16.0000 BW
ITEM-CODE 7
ITEM-DESCRIPTION Pipe Saddle - A53 ERW Grade B Carbon Steel PSPIPE SCH Std
```

When importing TEE-SET- types as pipe, the same rules as PIPE apply for the system discerning OD and Wall thickness. In this case the importer is parsing this as an NPS 16 schedule 40 pipe.

## 5 Mapping additional fields

Additional data can be imported from PCF files into PypeServer by using a Mapping Table. These mappings are configured in the PypeServer importer. They:

- Allow specific values in the PCF file to be assigned to fields

- Allow cut templates (templates defined in PypeServer) to be assigned to the cuts defined in the PCF file (endcuts), and also inferred in the PCF file (holes from Set-ons and OLETs).

**What is important to understand first is what data to add to your PCF file. The mapping of that data is a one-time setup that is more complex and can be performed with the help of PypeServer.** Please review “Using the Field Mapper” for a deeper understanding of how field mapping works. In this section, the PCF formatting is discussed.

## 5.1 Using Cut Templates

Please review “Using the Field Mapper” for a deeper understanding of how field mapping works. In this section, the PCF formatting is discussed.

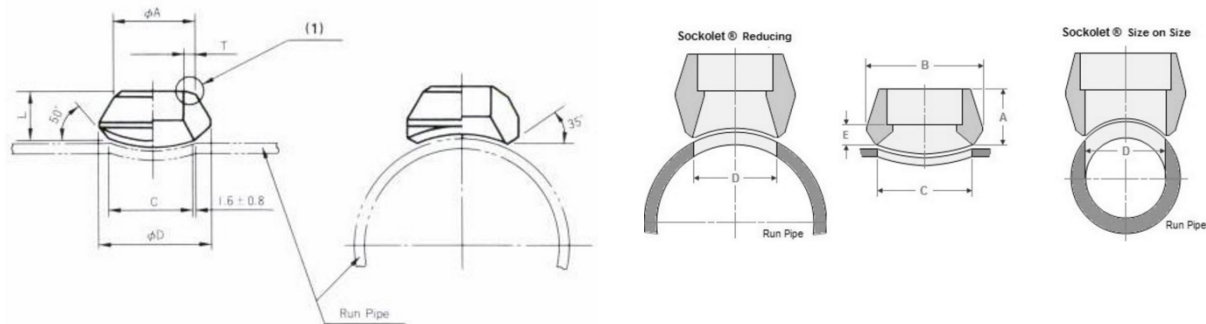
You can assign cut templates to both end-cuts and holes generated by Pipe-to-Pipe Intersections, TEE-SET-types setting on pipe, and OLETs on pipe.

## 5.2 Mapping OLETs

### The Problem

In PCF exports, OLETs typically specify the pipe size as the outward (off the pipe) connection (typically an NPS size). However the OLET set-on hole size at the connection to the run pipe/shell is not specified.

Here are examples of OLETs from [Aesterion](#) showing that the set on hole is not the same as the outward connection.



OLETs can have the same (NPS) sizes on both ends. However this is not always the case.

Specifying fields of Hole Diameter, Cut-Surface (ID or OD) and bevel are achieved by field mapping or using cut templates (covered in section 5)

## 5.3 OLET lookup Tables

Often data coming out of CAD (and the CAD part catalogs) lacks required fabrication information for cutting the set-on pipe holes for OLETs. To address this, PypeServer offers an OLET mapping tool that gathers OLET information during importing, such as OLET name, OLET Class, OLET Type and OLET outlet size to then match that to an item in an OLET lookup table. The matching item in the table will specify



#### 5.4.1 OLET Example with the set-on HoleSize specified in the ITEM-DESCRIPTION

This sample shows how to add data to the ITEM-DESCRIPTION (or other) field such that the data can be mapped into PypeServer:

```
OLET
CENTRE-POINT 10.4664 8.8345 -0.0000
BRANCH1-POINT 10.4664 5.6154 3.2191 0.7500 PL
ITEM-CODE 10
ITEM-DESCRIPTION Socket Weld Half Coupling HoleSize 0.875 Bevel 37.5
```

Note: the HoleSize keyword can also be HoleDiameter, but then the Item-Description must be mapped to the "HoleDiameter" field on the Cut Tab in the mapping.

The section "[The Need for OLET Cut Templates](#)" in this document explains the need for custom Hole diameters and bevels.

#### 5.4.2 Flagging end-cuts and holes so that they must be verified later

It is not uncommon for the CAD design and importing to be done before the actual OLET supplier or OLET dimensions are known. This section explains how you can require the hole cut to be "Verified" by someone before it can be loaded to the machine.

In the DESCRIPTION of a TEE-SET-ON/IN, or an OLET, add the keyword "VerifyHole", as shown here:

```
OLET
CENTRE-POINT 10.4664 8.8345 -0.0000
BRANCH1-POINT 10.4664 5.6154 3.2191 0.7500 PL
ITEM-CODE 10
ITEM-DESCRIPTION Socket Weld Half Coupling VerifyHole
```

The VerifyHole keyword is then mapped to the "Verify" field in PypeServer for the hole cut, which causes the "Verify" flag to be set in the cut in PypeServer, as shown below.

The screenshot shows the PypeServer interface. On the left, the 'Cuts List' table has columns: Name, Verify, Type, Feed Rate, Y Start, Mate Diam..., and Order. The 'Verify' column is highlighted with a red box, and a red arrow points to it with the text 'You may have to add this column using <right click> "Column Chooser" to see this.' The table contains three rows: 'ID: 40' (Straight, Feed Rate 40, Y Start 4, Mate Diam... 1), 'Hole N4' (Hole, Feed Rate 35, Y Start 2, Mate Diam... 1.25), and 'Hole N3' (Hole, Feed Rate 35, Y Start 0, Mate Diam... 1.25). The 'Verify' column has checkboxes: 'ID: 40' is unchecked, 'Hole N4' is checked, and 'Hole N3' is checked. On the right, the 'Hole' properties panel has fields for Feed Rate (35), ID (Cut Surface), Bevel Angle (37.5), Lead In (On), Num Tabs (0), Root Gap (0), Min Bevel Limit (-60), and TorchType (Plasma). The 'Verify' checkbox is checked and highlighted with a red box.

When the Verify box is checked (for each specific cut), then that cut (and thus the Part or Nesting it is in) cannot be loaded to the machine. The operator must verify the cut and uncheck the verify boxes so that the Cut, Part or Nesting can be loaded to the machine.

Similarly, you can flag endcuts to be verified, though because each end-cut can have specific requirements, this is more often done using end cut templates.

## 5.5 Mapping Additional data

This section discusses how you can directly map data from the PCF file into PypeServer data fields. You will need to use a field mapping to define the keywords (which can be any keywords you want), and then map those keywords to the respective PypeServer fields.

### 5.5.1 Setting Pipe Sizes Pipe Non-Standard Size example

This sample shows the use of a keyword “WallThickness” (case insensitive) to define the pipe’s non-standard (non-NPS) wall thickness.

```
PIPE
END-POINT 34.5570 8.8345 -0.0000 8.0000 PL
END-POINT -37.5680 8.8345 -0.0000 8.0000 BW
ITEM-CODE 2 Main Run
ITEM-DESCRIPTION WallThickness .275 A53
FABRICATION-ITEM
```

You will need to use a field mapping to define the “WallThickness” keyword, and then map that keyword to the respective PypeServer Wall Thickness field.

NOTE: for PIPE and TEE-SET- types, a Cut Template is a more powerful option because you will need to specify values for both ends of the pipe (for PIPES) and additionally for TEE-SET-types the hole created on the mate pipe.

### 5.5.2 Mapping Cut Templates to End Cuts for PIPE and the outer end of a TEE-SET- type

In PCF (exported from most PCF CAD importers) the endpoints (the end cuts) are specified as a type. From most CAD systems this is a two letter description defining the cut type. Shown below are PL and BW. (The meaning of these is defined in the CAD system). These can be mapped to cut templates in PypeServer, and thus the meaning can be augmented or clarified in PypeServer—based on the cut templates set up in PypeServer.

```
PIPE
END-POINT 34.5570 8.8345 -0.0000 8.0000 PL
END-POINT -37.5680 8.8345 -0.0000 8.0000 BW
ITEM-CODE 2 Main Run
ITEM-DESCRIPTION SCH 40 A53
FABRICATION-ITEM
```



In preparing your PCF file, CAD operators need to make sure to properly define their end-cuts with the proper names (like PL and BW) so that PypeServer's cut templates can put the right information on the end-cuts. Typically what is defined in the cut templates for these end-cuts are the Bevel Angles. Other fields may be set as well, such as root-gap, feedrate, tabs, etc.

### 5.5.3 Mapping Cut Templates to the Set-On side of a TEE-SET-ON

TEE-SET-ONs are more complex than PIPE types, in that a) the PCF data does not define the set-on end cut type, and b) the hole information is not provided at all.

Here is an example of a TEE-SET-ON where the set-on saddle end-cut and the hole are each defined by different cut templates.

```
TEE-SET-ON
CENTRE-POINT 10.4664 8.8345 -0.0000
BRANCH1-POINT 10.4664 5.6154 3.2191 0.7500 PL
SKEY TESO
ITEM-CODE 10
ITEM-DESCRIPTION Socket Weld Half Coupling HoleCutTemplate HoleType1 MateSaddleCutTemplate
SaddleCutType4 SCH 40
```

Notes:

- The HoleCutTemplate and MateSaddleCutTemplate identifiers can be any names you choose. They are defined when creating the mapping.
- When mapping, these cut templates can be in different libraries. This allows users to have a HoleCutTemplate library which is primarily used for OLETs.

### 5.5.4 Mapping Cut Templates for the hole created by an OLET

The section "[The Need for OLET Cut Templates](#)" in this document explains the need for custom Hole diameters and bevels.

This sample shows how to add a cut template name to the ITEM-DESCRIPTION (or other) field such that the cut template will provide cut information for the OLET hole—including hole diameter, bevel angle, and cut surface (ID or OD).

```
OLET
CENTRE-POINT 10.4664 8.8345 -0.0000
BRANCH1-POINT 10.4664 5.6154 3.2191 0.7500 PL
ITEM-CODE 10
ITEM-DESCRIPTION Socket Weld Half Coupling HoleCutTemplateName OLET1
```

In the example above the name HoleCutTemplateName can be any keyword. This field is followed by the cut template name for the given OLET type.

Notes about Cut Templates libraries and multiple Mappings

- Mappings can refer to one or more Cut Template libraries.

- This means that if you use one set of OLET cut template names in your PCF file, you could create different mappings for different OLET providers.
- Cut Template libraries can be referred to by multiple Mappings.
  - You can have multiple Cut Templates Libraries and can map any given field to any given library. What this means is that you can create an Olet-only cut template library. This is useful for maintaining a list of OLETs.

### 5.5.5 Creating an OLET Mapping Lookup file for use in PypeServer

In many CAD cases the CAD operator adds an OLET to a model, and selects the outflow diameter, but

- The set-on diameter is not specified in the catalog, and
- The user has no way to specify the set on diameter.

However, it is in many cases possible to output a UID for each OLET size. Users can output a list of these OLET IDs and use those IDs to create a PypeServer lookup table that will assign set-on diameter sizes. (Customers can do this in Revit).

The output format to create is as follows, where each line represents a specific OLET part ID.

```
[
{"BevelAngle":0.0,"CutName":"123781740374809","CutType":"Hole","Mate1Diameter":1.125,"YStart":0.0}
{"BevelAngle":0.0,"CutName":"20d8e-b56f-4fb5-2379-5c","CutType":"Hole","Mate1Diameter":1.25,"YStart":0.0}
]
```

- The CutName can be letters and numbers and dashes, no spaces.
- The Mate1Diameter is the set-on (or set-in) diameter that the user wants to cut on the run pipe that the OLET is attaching to.

That lookup file can be then be loaded into PypeServer. PypeServer support can help with that. Once this file is loaded into PypeServer, then the OLETs UNIQUE-COMPONENT-IDENTIFIER can be mapped to set-on hole cuts for OLETs so that the importing is automatic.

## 6 One PCF Example

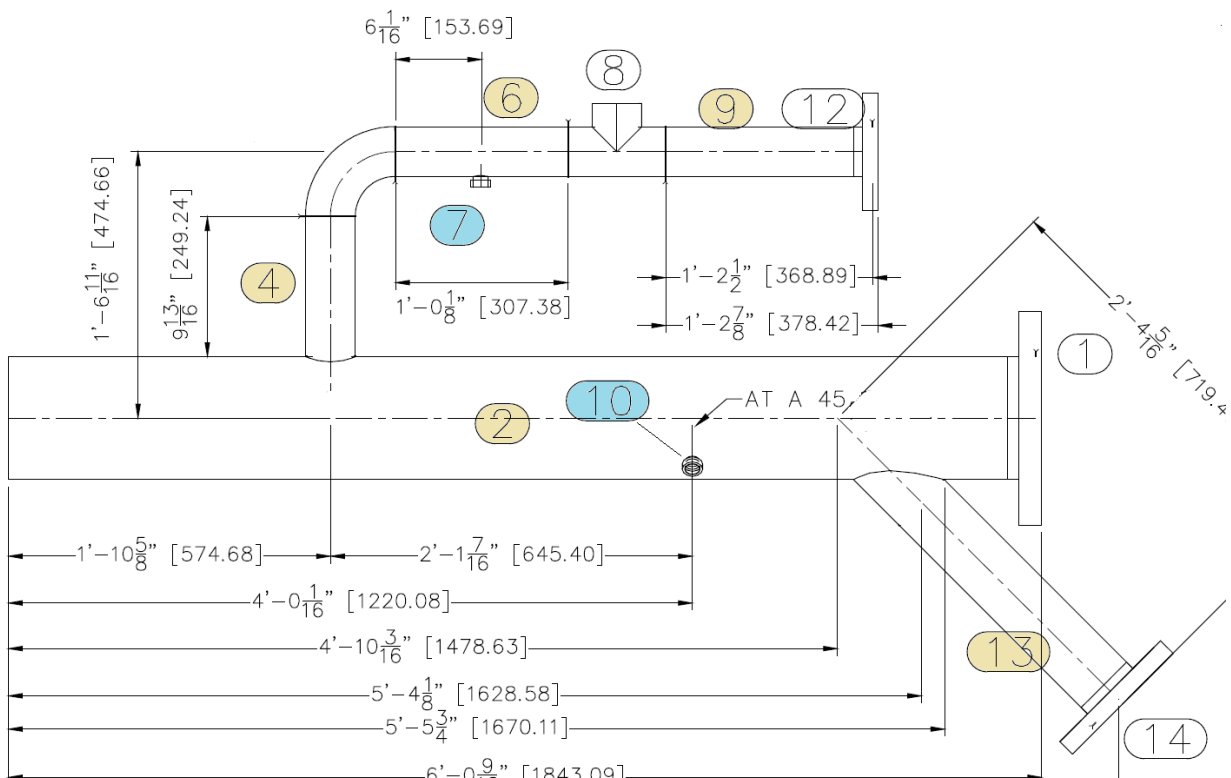
There are many ways to configure PCF export data for this import example. Much of that depends on where your data is coming from, what details you want to come from CAD vs. leaving some details to the Machine Operator, your specific workflows, etc.

### 6.1 The Spool Drawing

Here is a simple spool, with the PCF file shown below.

PyteServer:

- Will import items 2, 4, 6, 9 and 13.
- Will use items 7 and 10 (OLETs) to create holes in parts 2 and 6 respectively.



## 6.2 The PCF File with Comments

This PCF file shows numerous ways to map data. Users typically settle on a few features to meet their importing needs.

Notes on the file below:

- Some lines have been removed for clarity
- When mapping, all keywords in **bold red** can be changed to any keyword you want.
- Keywords in **bold blue** are default behaviors for the importer, though by mapping they can be changed to different names as well.
- The sample shows most of the fields being mapped in ITEM-DESCRIPTION. This is what most of our customers use due to restrictions in how data (parts) flow into their CAD systems from part catalogs. However you can use other fields for this data.

```
ISOGEN-FILES ISOGEN.FLS
UNITS-BORE INCH
UNITS-CO-ORDS INCH
UNITS-BOLT-LENGTH INCH
UNITS-BOLT-DIA INCH
UNITS-WEIGHT LBS
PIPE
  END-POINT 34.5570 8.8345 -0.0000 8.0000 PL
  END-POINT -37.5680 8.8345 -0.0000 8.0000 BW
  ITEM-CODE 2 Main Run
  ITEM-DESCRIPTION SCH 402 A53
  SPOOL-IDENTIFIER SPOOL13
  ITEM-ATTRIBUTE0 CID=[2041]
TEE-SET-ON
  CENTRE-POINT -14.9430 8.8345 -0.0000
  BRANCH1-POINT -14.9430 22.9595 -0.0000 3.0000 PL
  SKEY TESO
  ITEM-CODE 4 Tee on 2
  ITEM-DESCRIPTION SCH 40 SaddleSetOnCutTemplate Saddle14 HoleCutTemplateName Hole15
  SPOOL-IDENTIFIER SPOOL2
  ITEM-ATTRIBUTE0 CID=[2875]
PIPE
  END-POINT 1.7211 27.5220 -0.0000 3.0000 BW
  END-POINT -10.3805 27.5220 -0.0000 3.0000 BW
  ITEM-CODE 6 BVxBV with Olet
  ITEM-DESCRIPTION SCH 40 A53B GR B ERW
  SPOOL-IDENTIFIER SPOOL2
  ITEM-ATTRIBUTE0 CID=[2041]
OLET
  CENTRE-POINT -4.4042 27.5220 -0.0000
  BRANCH1-POINT -4.4042 25.5320 -0.0000 0.7500 PL
  SKEY TESO
```

<sup>2</sup> SCH or SCHEDULE defines the pipe NPS size. This is found in the ITEM-DESCRIPTION by checking the feature to look for Schedule in the ITEM-DESCRIPTION. (This is default behavior without mapping. You can change this with mapping.)

<sup>3</sup> By default this is mapped to the PypeServer Spool Sheet field. You can change this with a mapping.

<sup>4</sup> In the Mapping in this example, we've mapped SaddleSetOnCutTemplate to identify our saddle cut templates for set-on saddles. Saddle1 is one of the cut templates in our EndCutTemplates library.

<sup>5</sup> In the Mapping in this example, we've mapped HoleCutTemplateName to identify the cut template name for holes. Hole1 is one of the cut templates in our HoleCutTemplates library. (Note that we can map different names to different libraries.)

```

ITEM-CODE 7 Olet on 6
ITEM-DESCRIPTION Socket Weld Half Coupling HoleCutTemplateName OLET16
SPOOL-IDENTIFIER SPOOL2
ITEM-ATTRIBUTE0 CID=[2875]
TEE7
END-POINT 8.5336 27.5220 -0.0000 3.0000 PL
END-POINT 1.7836 27.5220 -0.0000 3.0000 PL
CENTRE-POINT 5.1586 27.5220 -0.0000
BRANCH1-POINT 5.1586 30.9020 -0.0000 3.0000 PL
SKEY TEPL
ITEM-CODE 8 Pulled Tee
ITEM-DESCRIPTION Straight Tee
SPOOL-IDENTIFIER SPOOL2
ITEM-ATTRIBUTE0 CID=[2047]
PIPE
END-POINT 23.0570 27.5220 -0.0000 3.0000 BW
END-POINT 8.5961 27.5220 -0.0000 3.0000 Bw
ITEM-CODE 9 BVxBV
ITEM-DESCRIPTION SCH 40 A53B GR B ERW
SPOOL-IDENTIFIER SPOOL2
OLET
CENTRE-POINT 10.4664 8.8345 -0.0000
BRANCH1-POINT 10.4664 5.6154 3.2191 0.7500 PL
SKEY TESO
ITEM-CODE 10
ITEM-DESCRIPTION Socket Weld Half Coupling HoleDiameter 0.8758 BevelAngle 37.59 VerifyHoleCut10
FABRICATION-ITEM
PIPING-SPEC Piping Materials
SPOOL-IDENTIFIER SPOOL1
ITEM-ATTRIBUTE0 CID=[2875]
TEE-SET-ON
CENTRE-POINT 20.6457 8.8345 -0.0000
BRANCH1-POINT 40.3656 -10.8854 -0.0000 4.0000 PL
SKEY TESO
ITEM-CODE 13 Tilted Branch on 2
ITEM-DESCRIPTION BRANCH WELD 45 PSPIPE11 SCH 40
SPOOL-IDENTIFIER SPOOL1
MATERIAL-IDENTIFIER 1
ITEM-ATTRIBUTE0 CID=[2875]
MATERIALS
MATERIAL-IDENTIFIER 1
DESCRIPTION A53B GR B ERW SCH 4012

```

<sup>6</sup> Here we use the same HoleCutTemplateName as for the TEE-SET-ON. The OLET1 is one of our OLETs in our HoleCutTemplates library.

<sup>7</sup> A pulled Tee—ignored by the importer

<sup>8</sup> HoleDiameter is mapped to the hole diameter field in the PCF mapping. If this keyword is found, it will set the diameter of the hole to this value instead of the 0.75 specified in the BRANCH1-POINT line.

<sup>9</sup> BevelAngle keyword is mapped to the PyeServer Bevel Angle.

<sup>10</sup> The VerifyHoleCut keyword is mapped to set the verify flag for the cut.

<sup>11</sup> The PPSPIPE keyword declares this as a pipe (though in most cases you just set all TEE-SET- types to be recognized as pipe. (See [Using Tee Set types as Pipe](#))

<sup>12</sup> Because in this example we've set the importer to look at the ITEM-DESCRIPTION to find the Schedule size, the importer will not also look here. This would work if it you unchecked the "Search descriptions for pipe values" setting. Use one or the other methods in your PCF files.

## 7 CAD Specific Information

### 7.1 AutoDesk Fabrication CAD MEP



This section covers some problems that customers have faced in the past. Please also refer to the document "Importers" to see additional information about exporting .pcf from AutoDesk Fabrication CAD MEP.

#### 7.1.1 Getting OLETs to output as OLET instead of TEE-SET-ON

This can be done a couple of ways.

One, just put the PCF SKey in.

<https://forums.autodesk.com/autodesk/attachments/autodesk/371/16550/1/Isogen%20Symbol%20Keys.pdf>

Description	Owner Inform...	Revision
 Weldolet Extra Strong	v1	VBS_2018-03-1
 Weldolet	v1	VBS_2017-09-0

Catalogue Item File "Weldolet"

?

X

Other

General

Options

Information

Design Flow

Item

Manufacturing

Costing

Ancillaries

Item No

Qty 1

Service None

Service Type 4: Pipework

Section None

Drawing

Zone

E-Tag

Alternate 0

Status 0: Design

Order 26x16 STD BW UNIV ANVILET

Code VBS-111614-1029789

Pallet

PCF SKey WTBW

OK

Cancel

The first two letters are the item and the last two are the connection.

The other way is to use wildcards for the connection and have the Pipework End Type pick up the last two. (This keeps your PCF Skey list a lot shorter.)

