

How Importing Works

This document describes the process of importing data that describes pipe parts.

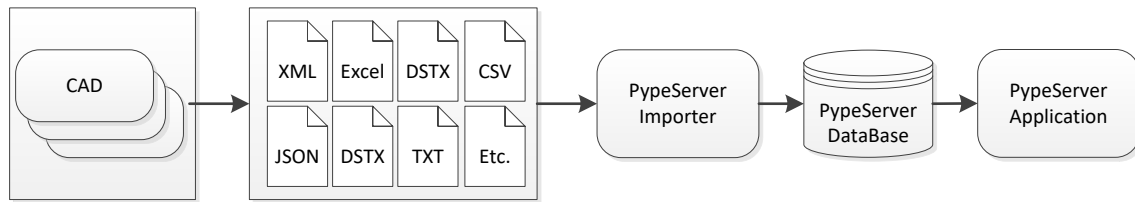
PypeServer creates these importers, though customers may contact PypeServer if they are interested in developing their own importer.

Contents

- Basic data flow.....2
- Types of Import Data (Parameter, GCode, Path and Shape).....2
 - Parameter Imports2
 - GCode/Machine Instructions2
 - Path Import.....2
 - Shape Import2
- How Part and Cut Values are Set.....3
 - Set manually for a specific part3
 - Set Default Values3
 - Use Default Values in PypeServer4
- Selecting Parts to Import.....5
 - Filter down to what you want to import.....5
 - Select exactly what you want to import.....5
- Scheduling Parts (Enterprise Licenses only)5
- The use of the “CAD Import ID” field6
 - Use CAD Import ID Rules6

Basic data flow

This data typically comes from a file exported from a CAD application, though it could come from an excel file, or even a manually created text file.



Types of Import Data (Parameter, GCode, Path and Shape)

Parameter Imports

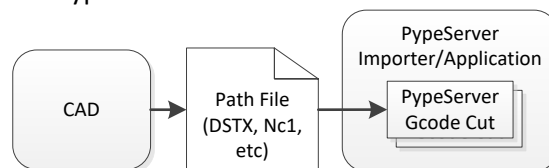
Import data most commonly comes as parameters that define centerline intersections of endcuts and holes. These types of imports convert cuts into standard PyeServer cuts (Mitters, Saddles, Holes, etc).

GCode/Machine Instructions

Some exports define, in GCodes the motions that define the cut. These types of command files can be imported and converted into the PyeServer GCode format. Once in PyeServer these GCode commands define torch action (on, off) and machine motions and feedrates.

Path Import

Some exports define a path around the pipe. DSTX is an example of this. Some of these types of exports are only a single path which can be defined along the ID or OD of the pipe. These types of imports convert the paths into GCode type motion path cuts in PyeServer.

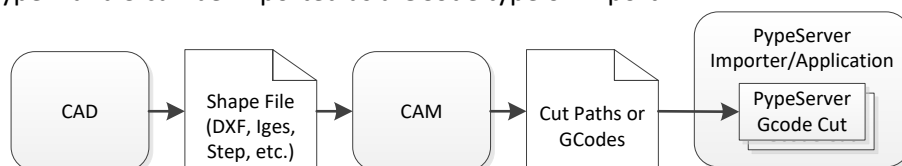


Shape Import

The CAD industry commonly exports data as shapes (dxf, iges, step).

1. Import based on 3D shape: PyeServer does not currently support the direct import of part shapes, though work is in progress on this feature.
2. Import of lines drawn on pipe shapes: Customers do define shapes and export specific lines around pipes, which lines can be converted into cut paths.
3. As with milling systems, other CAM tools can export cut paths, which paths can be imported as cut paths.

The cut paths of type 2 and 3 can be imported as a GCode type of import.



How Part and Cut Values are Set

If a Part or Cut parameter is defined in the importer, then that value will apply to the part or cut. If the importer does not define a value, then the value can be set in this order:

1. In the Importer
2. By manually entering data in a cell for a given part (overrides the import)
3. By setting default values in the Default Values pane of the importer
4. Relying on the default values for Parts and Cuts in PypeServer
5. If no data is required, the value remains blank

Set manually for a specific part

You can directly modify some of the columns for the import. If there were values from the import, you can manually override them by entering data directly into the cells:

Drag a column header here to group by that column						
<input checked="" type="checkbox"/>	Part ID	Customer	Job	Spool Sheet	Name	Design Group
<input type="checkbox"/>	0			CHWS-1	13	
<input type="checkbox"/>	0	Cust X	My Custom Entry 1	CHWS-2	Manual Name entry	
<input type="checkbox"/>	0		My Custom Entry 2	CS-1	1	
<input type="checkbox"/>	0			CR-8	2	
<input type="checkbox"/>	0			CR-7	3	

Special cases:

- The PartID is set by Pypeserver, you cannot set or change that.
- The number of cuts are defined in the import
- You can edit the “CAD Import ID” field when the CAD Import ID Rules checkbox is unchecked

Set Default Values

Use the Default Values pane in the importer to override blank values for parts and cuts. Stated another way: default values will only be inserted in Parts that do not have values already set by the import load, or by the user manually setting specific values in the parts. If you want a part or cut value to be these defaults, then delete the imported or manually entered setting.

Some default values can include a formatting expression as well as default text. A formatting expression consists of multiple text and field names enclosed in braces {}. For example, the default value for the cut Note “Bevel 22.5° Straight Cut” is formatted using the expression “Bevel {BevelAngle}°, {CutType} Cut”. Please see “Custom formatting with field expressions” for more details.

Default Values

Imported From	
Name	
Customer	
Job	
Design Group	
Spool Sheet	
Bar Code	
Notes	
Priority	
Cut Date	
Seam Ang	
Match Pipe Seam	<input type="checkbox"/>
Material Type	
Num Scheduled Parts	0

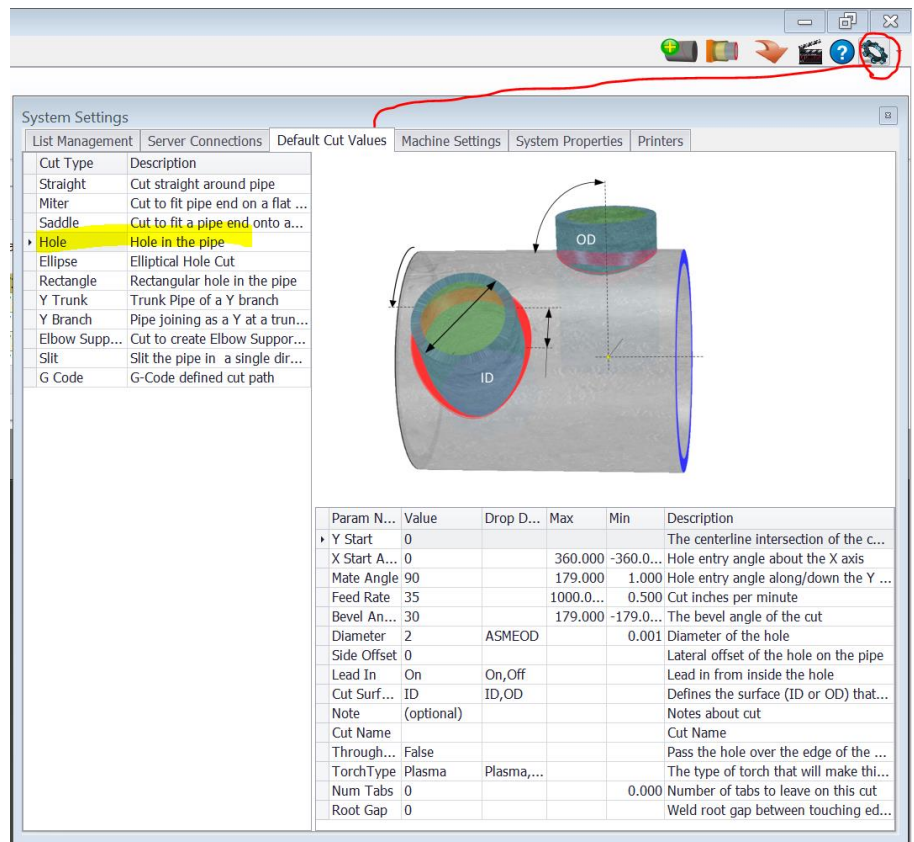
Cut Defaults

Feed Rate	40
Bevel Angle	30
Root Gap	0.0625
Tabs for hole dia >=	20
Num Tabs	4

Use Default Values in PypeServer

If no value is set by the above methods, and the value is required, then the value will be set to the PypeServer default value for the Part or Cut.

Operators can set default values for each cut type—there's a tutorial on that. These default values are set in the server and apply to all users (and all importing). Here is the dialog for setting default values for a new hole cut:



Selecting Parts to Import

The importing system works with filters the same way as other tables in PypeServer. See the “Filtering and Searching” video tutorial for more information on filtering.

Filter down to what you want to import

When you select Import To PypeServer (or Export to File), the importer will only import or export the files that you have not filtered out. This screenshot shows an import list where only 4.5” and 8.625” pipes are going to be imported. Other sizes are filtered out and will not be imported.

CD	Staged Data ▾	Import to PypeServer	Export to File
Drag a column header here to group by that column			
<input type="checkbox"/>	Part ID ▾	Outer Diameter ▾	CAD Import ID
<input type="checkbox"/>		0 8.625	834000999-A
<input checked="" type="checkbox"/>		0 4.500	834000999-B
× <input checked="" type="checkbox"/> [Outer Diameter] = '4.500' Or [Outer Diameter] = '8.625' ▾			

Select exactly what you want to import

The import will only import specific rows if you have any items checked. If nothing is checked, the all the unfiltered rows will be imported.

CD	Staged Data ▾	Import to PypeServer	Export to File
Drag a column header here to group by that column			
<input checked="" type="checkbox"/>	Part ID ▾	Outer Diameter ▾	CAD Import ID
<input checked="" type="checkbox"/>		0 8.625	834000999-A
<input type="checkbox"/>		0 3.500	834000999-C
<input checked="" type="checkbox"/>		0 4.500	834000999-B

Scheduling Parts (Enterprise Licenses only)

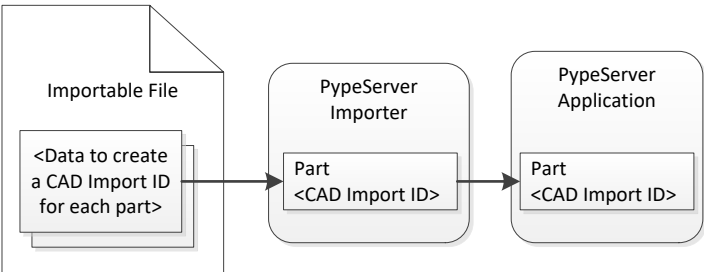
If you have an Enterprise license, you can automatically schedule parts by checking the “Schedule Parts” box in the import toolstrip. And you can manually enter/override the number of parts you want scheduled for each part.

<input checked="" type="checkbox"/> Schedule Parts
Num Parts
2
4
1

The use of the “CAD Import ID” field

If you want the ability to update (re-import) PypeServer Parts after you’ve already imported them (to make fixes), then take the time to understand this section.

The CAD Import ID field uniquely identifies parts that come from CAD. It is created from fields in the file to be imported and used in PypeServer.



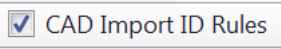
The CAD Import ID is useful for:

- Knowing where the Part came from in your CAD systems.
- Re-importing Parts into PypeServer after you’ve already imported them. (E.g. You import some Parts, then make some changes in CAD, and then you need to update the data in PypeServer.

CAD Import ID Creation

The data comes from the CAD system so there is no way for PypeServer to guarantee that the data is truly unique. That is up to how you configure and use your CAD system. PypeServer can work with you to create the CAD Import ID from data in the exported file.

When importing you will see this checkbox in the import menu:



Use CAD Import ID Rules

Using CAD Import ID Rules ensures that each Part imported will have its own unique CAD Import ID. This will allow you to easily update your Parts in PypeServer if you make changes in CAD.

When CAD Import ID Rules are turned off, the importing system will not check the database for Parts in the database that match the CAD Import IDs of the Parts being imported. With the rules off, importing will not update any parts (will create new parts instead), and may load parts with duplicate CAD Import IDs.

If you are not maintaining uniqueness in your Part exports, then you may need to turn off the CAD Import ID Rule checking to allow duplicate CAD Import IDs. When more than one Part in the PypeServer database has the same CAD Import ID, they cannot be updated by re-importing.

Parts that are being updated (by applying CAD Import ID rules) will NOT have scheduled parts changed in any way. This is why you see the Num Parts column in yellow for parts being updated.

Staged Data		Import to PypeServer	Export to File	<input checked="" type="checkbox"/> Schedule Parts
tag a column header here to group by that column				
<input type="checkbox"/>	Part ID	<input type="checkbox"/>	CAD Import ID	Num Parts
<input type="checkbox"/>	258	<input type="checkbox"/>	834000999-A	1
<input type="checkbox"/>	259	<input type="checkbox"/>	834000999-C	1
<input type="checkbox"/>	260	<input type="checkbox"/>	834000999-B	1