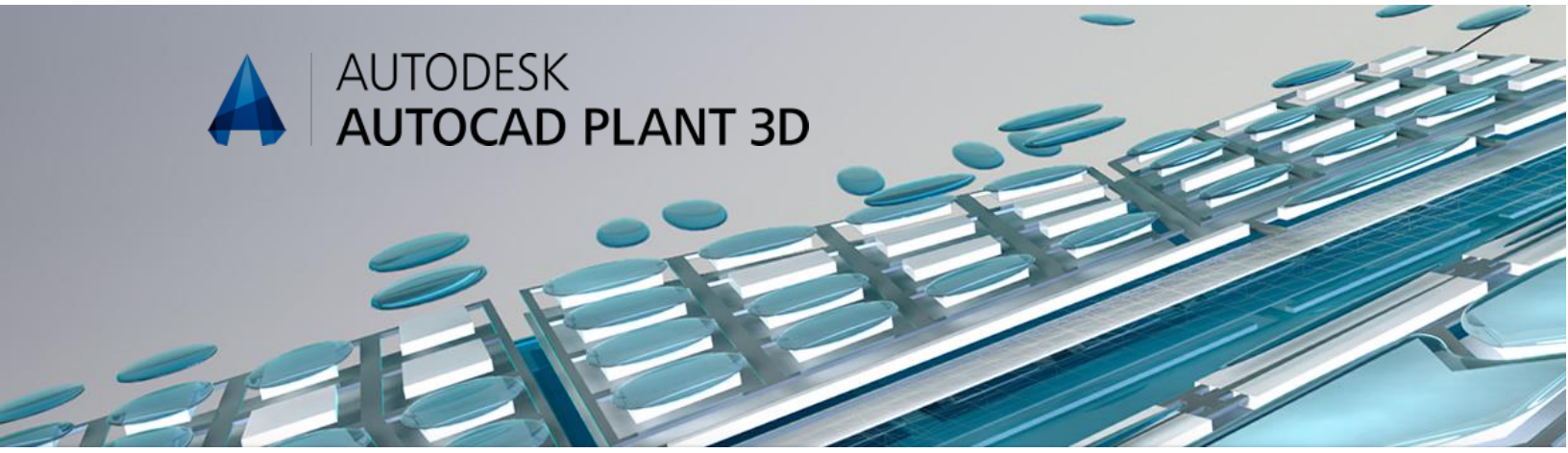
AUTODESK.



Tailoring AutoCAD® P&ID and AutoCAD® Plant 3D

You are tasked with integrating AutoCAD Plant 3D or AutoCAD P&ID into your company's workflow and environment. To integrate AutoCAD Plant 3D, you will have to understand the project structure, P&ID content creation, and aspects of AutoCAD Plant 3D. Use this guide as a reference for managing AutoCAD P&ID and AutoCAD Plant 3D within your company.

Table of Contents

Tailoring AutoCAD® P&ID and AutoCAD® Plant 3D 0

Customization Scope 5

- Reports 5
- Drawing Output 6
- Model Content 6
- Interoperability 6
- 8-fold Path of Success 7
- Walking the Path 8

Project Content 9

- Creation 15
  - Create a Project using the New Project Wizard 15
  - Create a Project Using a Project Template 18
- Database 19
  - Types of Databases 19
  - Database Viewers 20
  - Choosing a Database Type 20
  - Schema 20
  - Cache 21
  - Switching Database Types 21
- Class Structure and Behavior 23
  - Inheritance 23
  - Property Flows 24
  - Custom Class Design (P&ID only) 28
  - Combination (PLANTDEFINECALCPROPERTIES) 28
- Folders 29
  - Filename Format 30
  - Title Blocks 30

Sharing 31
Sharing a Project 32
Sharing Individual Files 33
Project Symbols 34
Going Local 38
P&ID Content 39
Layers 39
Selection Lists 40
Symbols 42
Batch Adding Symbols 47
Connection Point Requirements 48
Symbol Settings 49
Tagging 51
Tag Formats 51
Defining a Tag Format for a family of P&ID Class Definitions 52
Expressions 60
Using Expressions 62
Autogeneration 64
Delimiters 69
Assigning a Tag Format 70
Line Tagging 71
Tagging Prompt Behaviors and the Graphical Style 72
Annotations 80
Simple Annotations 80
Compound Annotations 80
Creating a New Annotation Style 84
Assigning Format Strings to Attribute Definitions 87
Using Expressions in Annotation Format Strings 96
Using Annotations to Edit Underlying Data 99
Principles of Dynamic Sizing 101

AutoCAD Plant 3D Content 108
Class Structure 108
Selection Lists 108
Equipment Templates 108
Model Organization 110
Setting Up Your Project 111
Controlling Access 112
Coordinate Systems 112
External References 113
Catalogs and Specs 116
Workflow 117
Custom Properties 122
Sharing Tool Palettes 125
Joints 126
Deliverables 131
P&IDs 131
DWFs 131
Title Blocks 132
Orthographics 133
Templates 134
Annotations 135
Isometrics 136
Style Management 136
Title Block Setup 137
Text Styles 139
Dimension Styles 143
Reports 145
Data Manager Views 146
Import/Export 149
Data Manager Reports 150
Report Creator 153

Interoperability 163
Vault 163
Overview 163
Client Installation 164
Using Vault 165
Organizing Files 170
Vault Database 171
Inventor 171
Export to ADSK 171
AutoCAD IMPORT 175
Revit 178
Navisworks 182
Navisworks Navigator 182
Insulation Display 183
Displaying Properties 188
Publishing 190
Reviewing 191
AutoCAD Structural Detailing 192
AutoCAD MEP 192
Contributions 193

Customization Scope

This section will help you identify the scope of the customization required for your company. This is the 20,000 ft. (or 6096 meter) overview that gives a roadmap of what you will touch as you set up the project template. The main goal is to customize AutoCAD Plant 3D so that it creates drawings that look like your existing style and to use additional features your company would like to see.

You should plan your customization with ample time. You should not try to get all the customization done with a project deadline under way. Even though isometrics is listed last, you may need to go through several iterations of checking and review before the drawing matches what your company wants to produce.

While programs like AutoCAD Plant 3D are designed to be flexible, you should customize only what is required to match your company's standard output. While altering class structure or other aspects of the software is possible, it may not be advisable.

As with any customization, you should keep backups of files before making changes. Also, you should change one feature at a time, especially with isometrics, so that you can observe changes step by step.

Reports

Gather a list of reports that your company currently produces as engineering deliverables. While you many have some reports that AutoCAD Plant 3D cannot create, the goal is to identify everything first, and then weed out the items that are not possible. The list may include, but should not be limited to:

- Instrument Index
- Line List
- Equipment List
- Valve list
- Tie-In List
- Piping BOM

Not only do you need a list of the reports, but you need examples. After gathering the reports, you should compile a list of column names that each report uses for future comparison to the project structure.

Drawing Output

Determine what properties or values you need to display in drawings. Look at existing drawings to see, for example, what information is displayed on the P&ID along with the vessel tag: Size, Height, Width, and Design Temperatures? What information is displayed on isometrics: support tags, valve annotations? Create a list of these properties and/or values. Also, gather samples of all the drawings created in a project. If a CAD standard has been developed, that should be in hand.

How are files named? AutoCAD Plant 3D can enforce a file name rule.

Consolidate a list of all of the values used in title blocks. You can save significant time managing the title blocks through AutoCAD Plant 3D.

Model Content

Determine which programs will need access to the models: Revit, ASD, Inventor, AutoCAD, Navisworks? Create a detailed list of programs that may need to interface with the model.

Interoperability

In addition to gathering reports, you also need to know which programs need to access or relate to the P&ID or piping data and which properties those programs are expecting. Are part numbers being used? Will the data be pushed to a purchasing system?

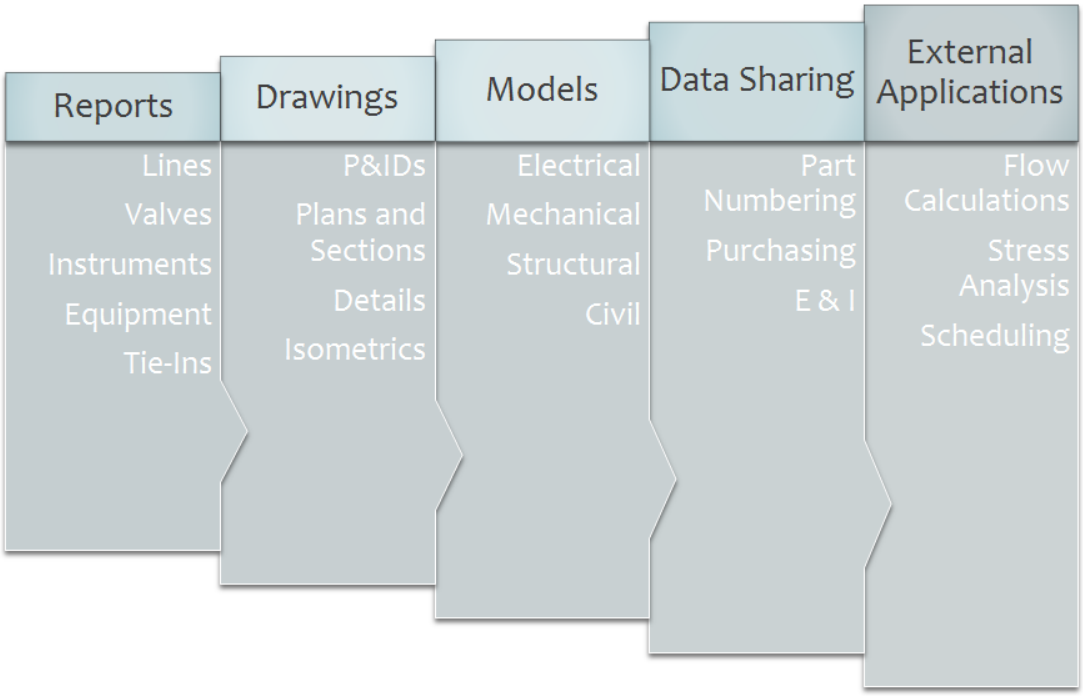


Figure 1: Areas of Impact

8-fold Path of Success

The graphic below shows the order in which the customization needs to be done.

1. Project Settings – program options, locations, joints, mappings
2. Class Structure – properties, types of tags, and class setup
3. Drawing Templates – dimension styles, text styles, layers and reports
4. P&ID Symbols – adding company symbols, combining standards
5. Annotations – text callouts
6. Catalog Content – Bulk part library with custom properties (if needed)
7. Specs – Model part insertion
8. Isometrics – Customize output to implement company standards

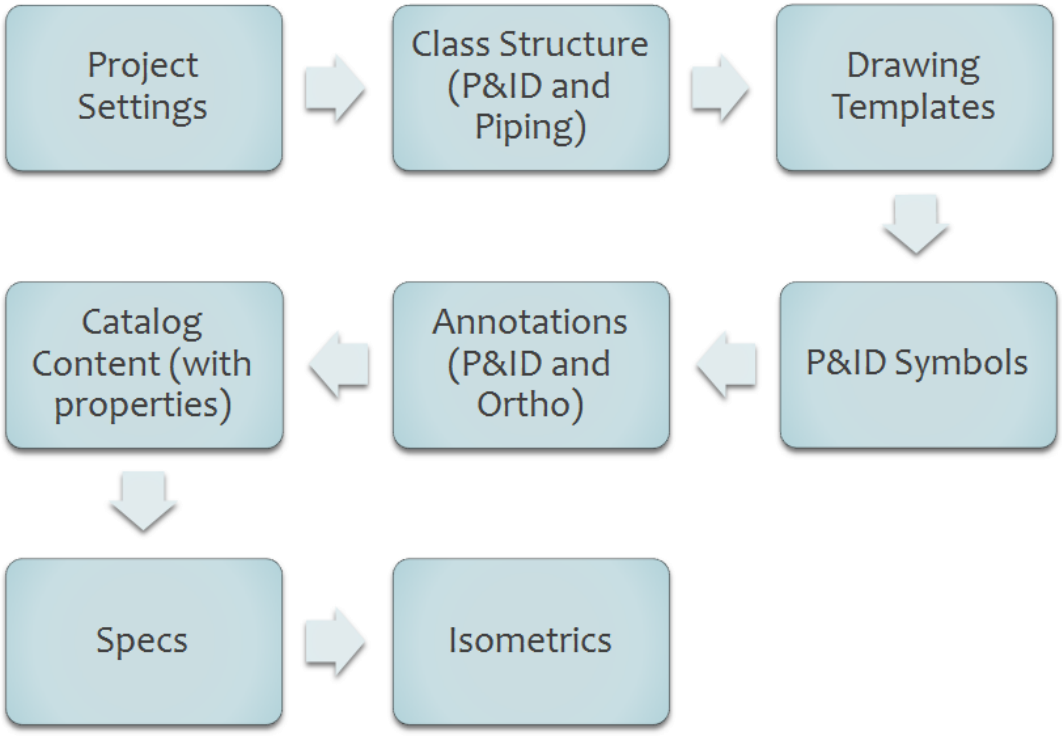


Figure 2: Path of Success

Walking the Path

Throughout this manual, several programs will be used (most of them free). Recommended programs to install are:

7-Zip	A free zip application. 7z is the most compressed zip type.	http://www.7-zip.org/download.html
Foxe	A well-developed xml editor. Tutorial here: http://www.pdoteam.com/2012/08/editing-xml-files/	http://www.firstobject.com/dn_editor.htm
SQLite Expert Personal	A free SQLite database editor that renders guids appropriately	http://sqliteexpert.com/download.html

Project Content

Understanding the folders and content within a project forms a basis for creating and locating a template project. A project folder consists of several files that contain the data and the structure of an AutoCAD P&ID project. The project is organized into parts which are utilized by aspects of the program. Because of this separation, AutoCAD P&ID can read and use an AutoCAD Plant 3D project, and an AutoCAD Plant 3D project can include P&IDs. The project.xml file in the project folder contains links to all of the parts for its project.

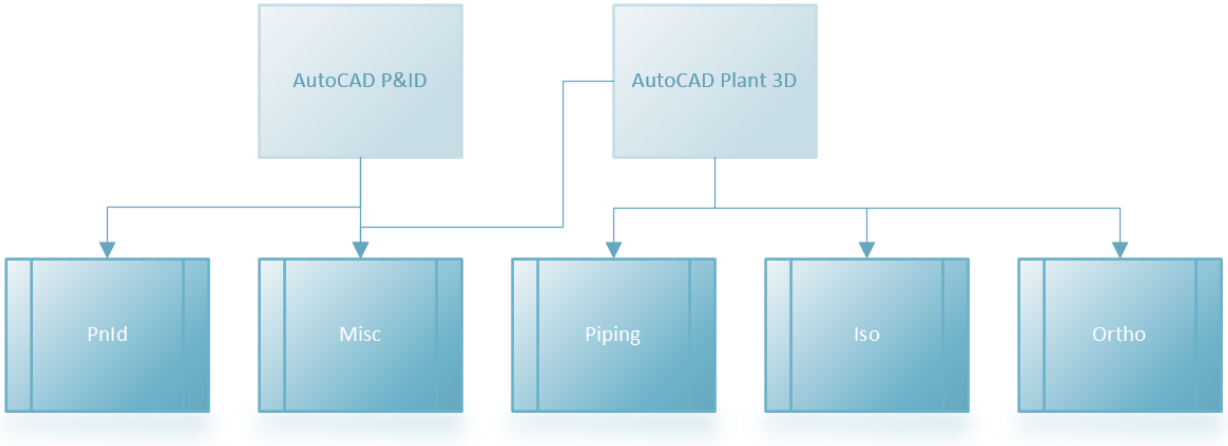


Figure 3: Program Structured into Parts

Each project part consists of a few files – an .xml, a .dcf, and a .dcfx.

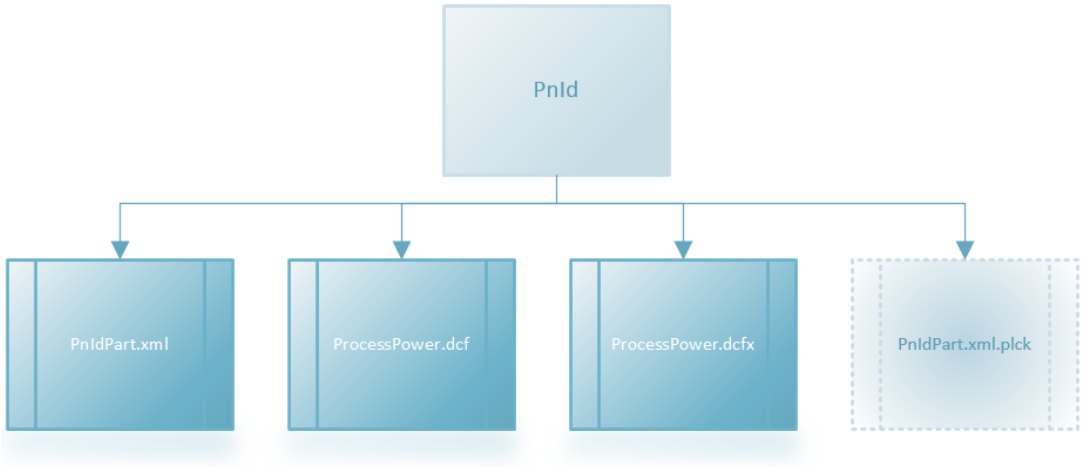


Figure 4: Part Structured into Files

The part files contain different types of information:

.xml	Project Information (Name, Description, etc.), General Settings, Project Directory/File List
.dcf	A SQLite database (or connection xml for SQL Server)
.dcfx	Class Structure used in the Database

When project settings are being edited, a lock file (.plck) will also appear. As features are added, other xml files will appear in the project folder. Standard xml files are:

- DefaultConnectorsConfig.xml
- PIDTo3dClassMapping.xml
- PipeBendDefaultValues.txt
- SubstitutionPalettes.xml
- ProjectSetupSpecUpdateSettings.xml

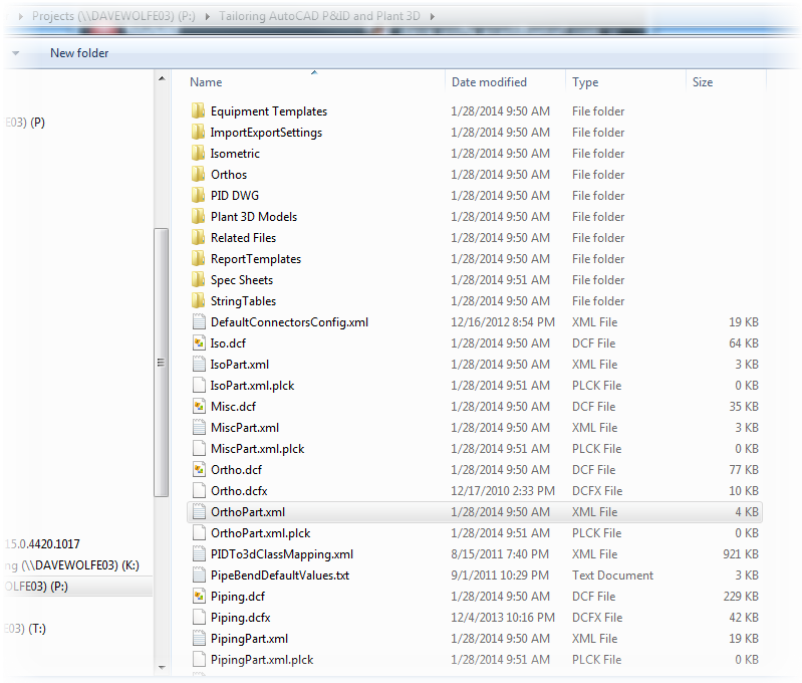


Figure 5: Files in Project Folder

Most of these files (except for PipeBendDefaulValues.txt) are generated from the project settings automatically. In general you will modify the files via the project setup or other customization dialogs.

The final key file in the project folder is the projsymbolstyle.dwg. This drawing file is a resource that contains the styles, layers, blocks, and other content that the program will use or insert.

Special Project Folders

Several folders simply store project related setup files. These are:

- EquipmentTemplates
- ImportExportSettings
- ReportTemplates
- StringTables

Equipment Templates path is customizable under the AutoCAD Plant 3D DWG Settings. However, this path doesn't affect new projects using the current project as a reference (wizard-based project creation).

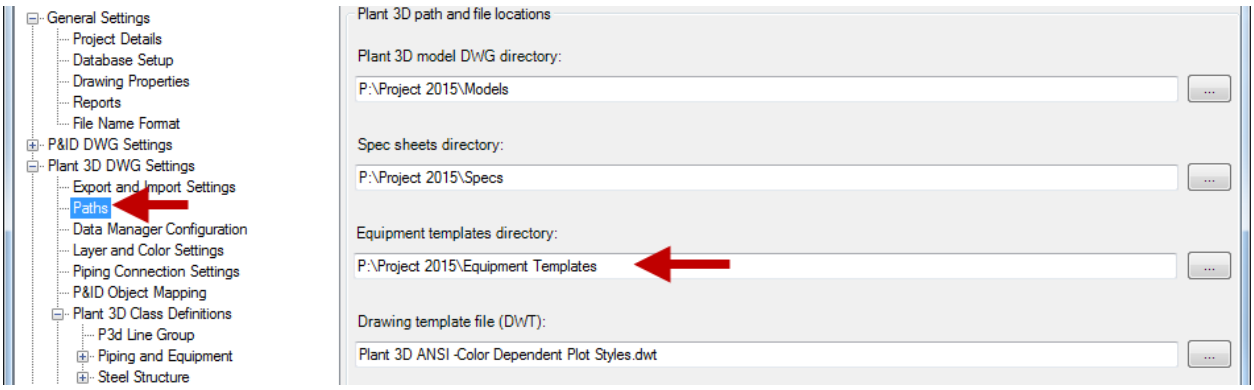


Figure 6: AutoCAD Plant 3D Paths

The rest of the special folder locations are not customizable. The settings stored within the folders are read into the project setup dialog.

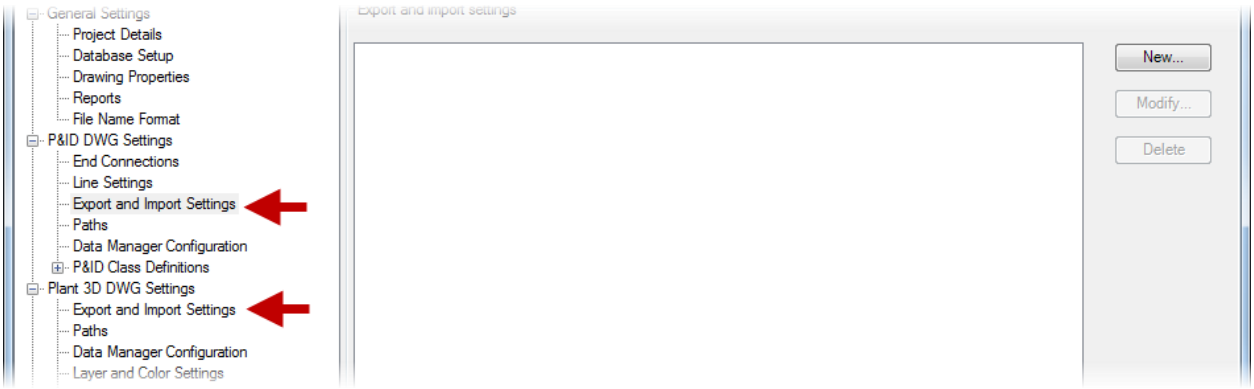


Figure 7: AutoCAD P&ID Import/Export Settings

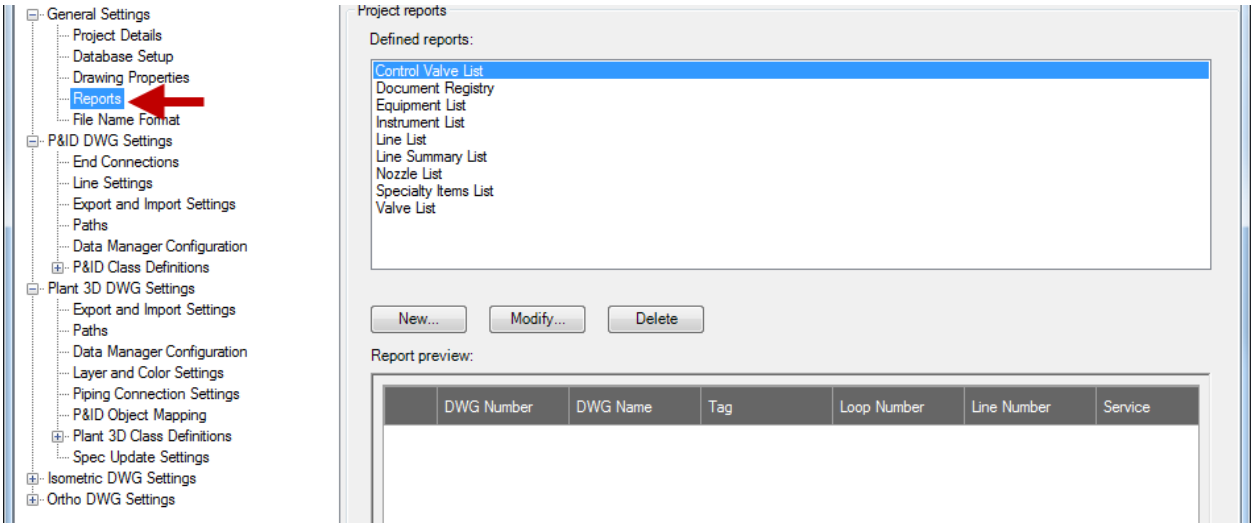


Figure 8: Predefined Reports

Orthographic Folder

The orthographic folder location is set under Ortho DWG Settings.

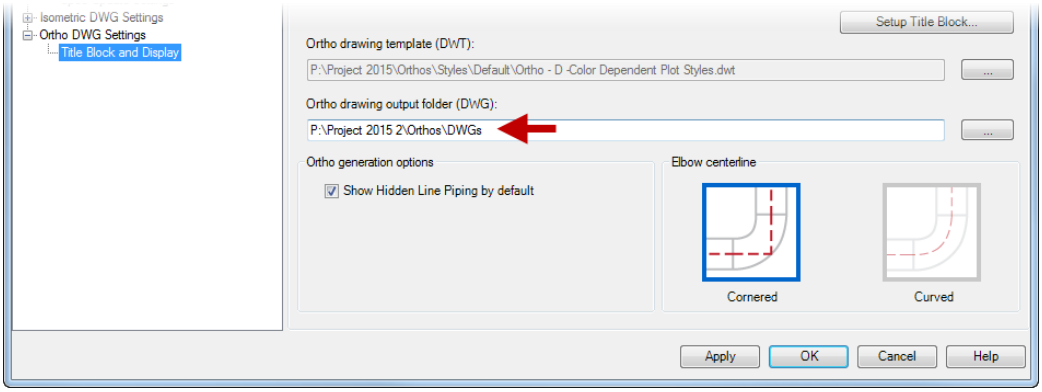


Figure 9: Orthographic Folder

This folder is the location where you want orthographics that you will issue. The drawings are linked to the project for title block information and can update from the model directly.

While the location for orthos can change, the program creates and uses an Orthos folder within the project folder to save Orthocubes, styles, and other settings.

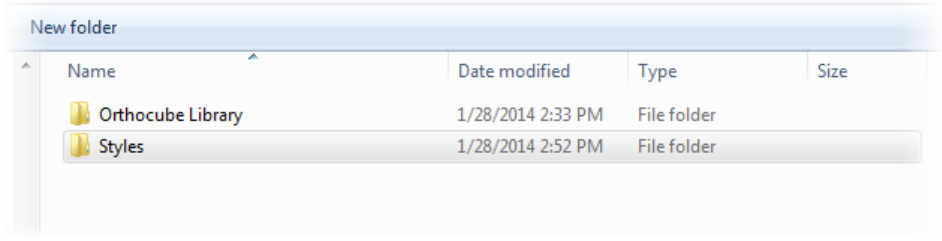


Figure 10: Ortho Settings Folder

The Styles folder contains the default style used to create orthos, while the orthocube library contains saved views created by the users.

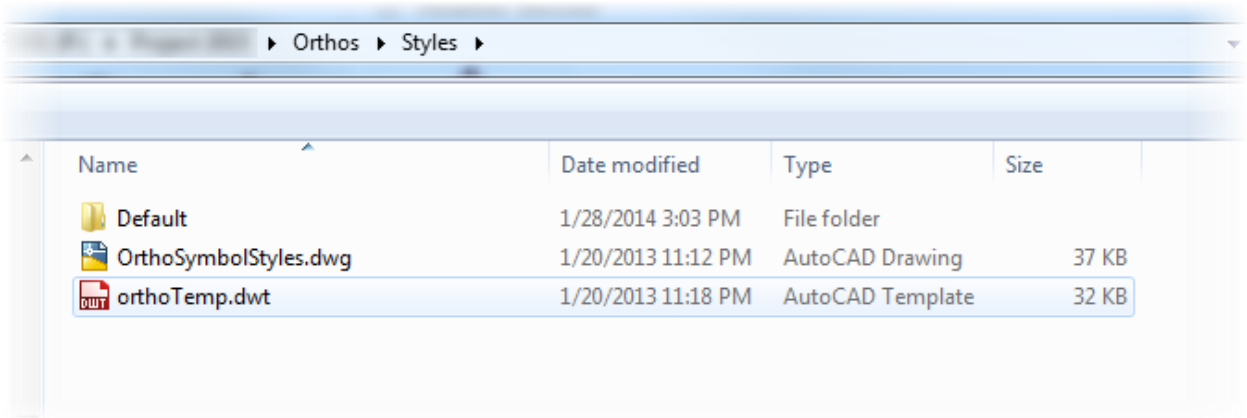


Figure 11: Ortho Styles Folder

Isometric Folders

While the ortho drawings are linked to the model, isometrics are a snapshot of the model. With a snapshot workflow, typically drawings are generated at a standard location and then moved to a separate folder to track the isometrics that are actually issued. The folder location specified in settings then, should not be a folder used to issue drawings. The folders will have to be regularly cleaned out to remove old or invalid isometrics.

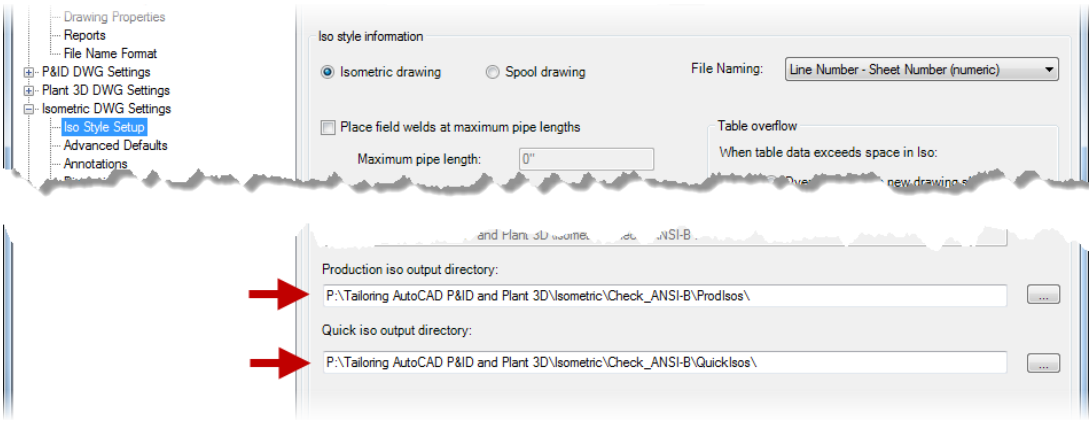


Figure 12: Isometric Folders

The Isometric folder contains Isometric Styles and some overall setting files.

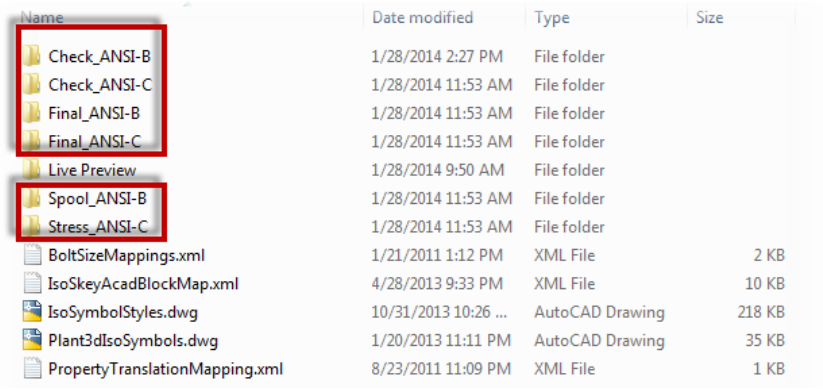


Figure 13: Default Isometric Styles

By default the Live Preview folder holds sample pcf files which can be used to preview changes in isometric settings for a style.

BoltSizeMappings.xml	Imperial –Metric Size Conversion & Operator Direction
IsoSkeyAcadBlockMap.xml	Symbol Keys and the Blocks associated with them.
IsoSymbolStyles.dwg	File contains blocks for symbol keys
Plant3dIsoSymbols.dwg	Symbols inserted into the model to call out iso information
PropertyTranslationMapping.xml	Substitute model values with pre-defined values on iso

The default styles are copied into each project. You will have to add your styles to the project template. Each style folder contains files that define isometric output. When working with isometrics, you may remove files from the PCFs, ProdIsos, and QuickIsos folders.

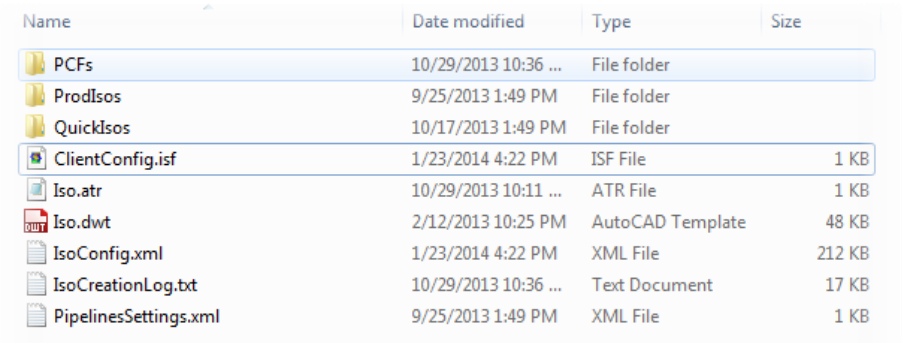


Figure 14: Default Isometric Style Files

PCFs	Text files that list the piping components used in an isometric
ClientConfig.isf	Contains properties related to generating an isometric
Iso.atr	Properties used on Isometrics and how they relate to Plant 3D properties
Iso.dwt	Isometric Title block template (doesn't have to be in the style folder)
IsoConfig.xml	Configuration file that controls the look of the isometric
IsoCreationLog.txt	Log file for the iso creation process
PipelinesSettings.xml	PCF header settings

Knowing the structure of the project folders gives you a framework to look at creating a project template. Setting up a solid project template folder structure will drive your success with AutoCAD Plant 3D.

Creation

You can create a project in one of two ways: you can use the New Project wizard, which is located on the Home Tab, or you can copy a project folder using Windows.

If you want to include standard project drawings (lead sheet, detail sheets), or want to have a default set of drawings included based on the project type (skid, plant, etc.), you should use a template folder.

To pull the settings from an existing project (or even a project template), you can use the project wizard.

For non-Vault projects, folders may be located outside of the project folder. You can use this option when you want to hide the actual project settings from the normal folder navigation process.

http://www.pdoteam.com/2012/07/adding-external-drawings-to-autocad-plant-3d/

Companies not using Vault often want to continue using their own project folder structure. To do so, they should use the steps above to use project files outside of the AutoCAD Plant 3D project structure.

Vault folder structure will be discussed later.

Create a Project using the New Project Wizard

The New Project wizard is available under Home > Project > Project Manager > New Project.

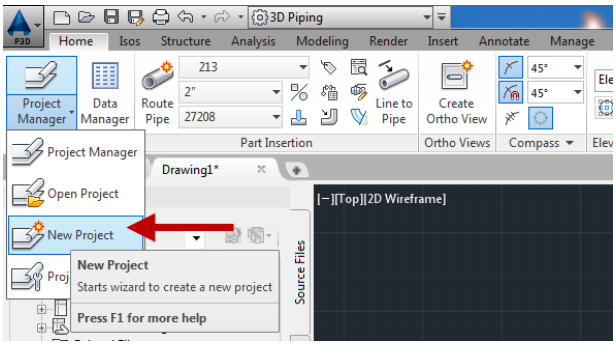


Figure 15: New Project

Notes on Wizard Options

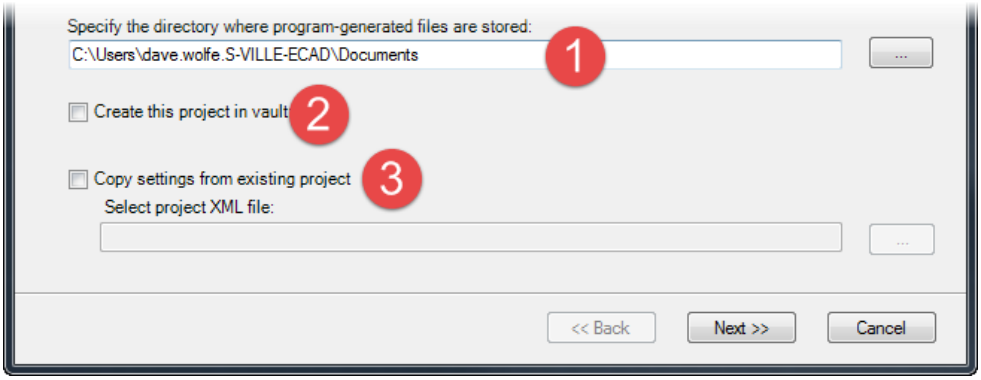


Figure 16: New Project Page 1

1. On page 1, the path specified should be the directory or drive that contains your project folders. For example, if your projects are stored on the P:\ root, you should pick the P:\ root, not P:\Project 1234. The program will create a folder matching the project name, and then put the related files inside of it.
2. Options for your project structure are more limited when using Vault. Check the Create Project in Vault to use Vault as your document management system.
3. You may choose this option to have your new project use the settings from an existing (often called a template) project.

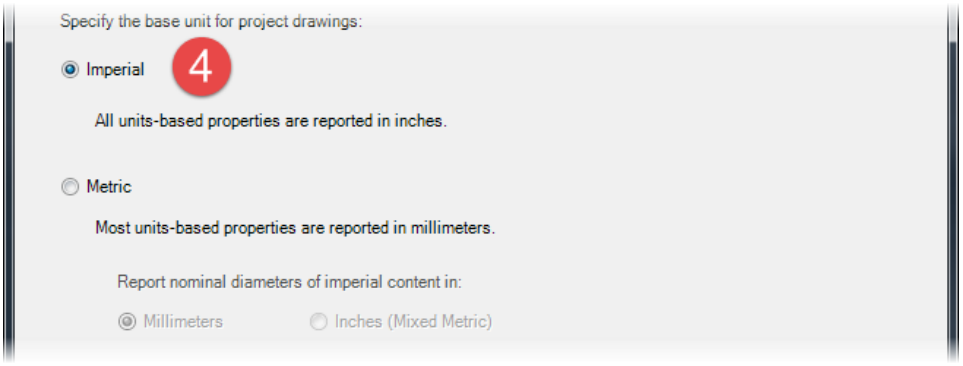


Figure 17: Choose Units

4. Units are a project-wide setting. In addition, you cannot copy drawings from one project unit type to another.

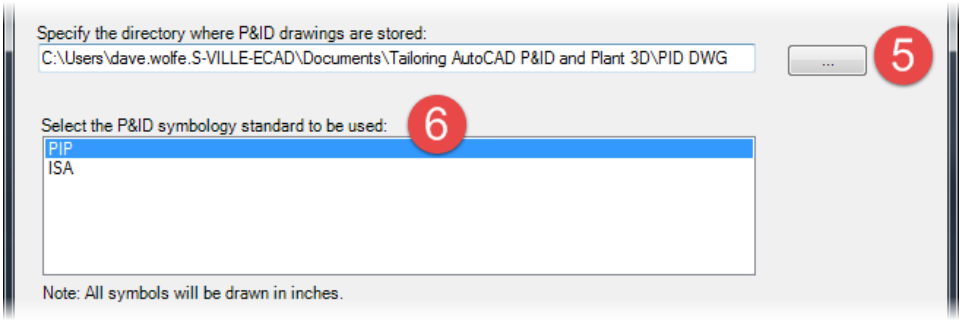


Figure 18: Choose Symbol Standard

5. P&IDs may be located at a custom location.
6. Symbol standards are not transferrable inside of AutoCAD P&ID. PIP has the most robust library, so typically you should start with that standard and add other symbols as needed, unless a different library addresses specific needs.

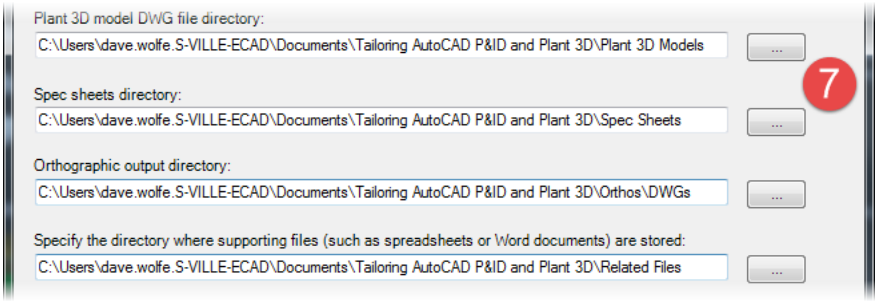


Figure 19: Set Plant 3D folder paths

7. AutoCAD Plant 3D folders whose location can be customized.

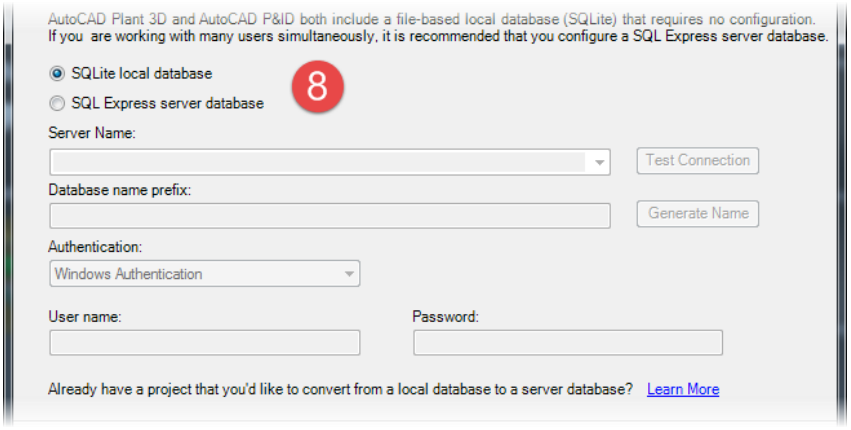


Figure 20: Select Database type

8. Select the database type. Database selection options are discussed in the Databases section.

Create a Project Using a Project Template

To create a project template from an existing project, you should copy it, and then remove unnecessary drawing files from the project manager. Afterwards, delete the files from the project template folder.

Even for project templates that will use the wizard, clear the drawing folders before you begin, as old isometrics may be left inside the style folders.

New projects can also be created from a template by using the Copy Settings option on page 1 of the wizard.

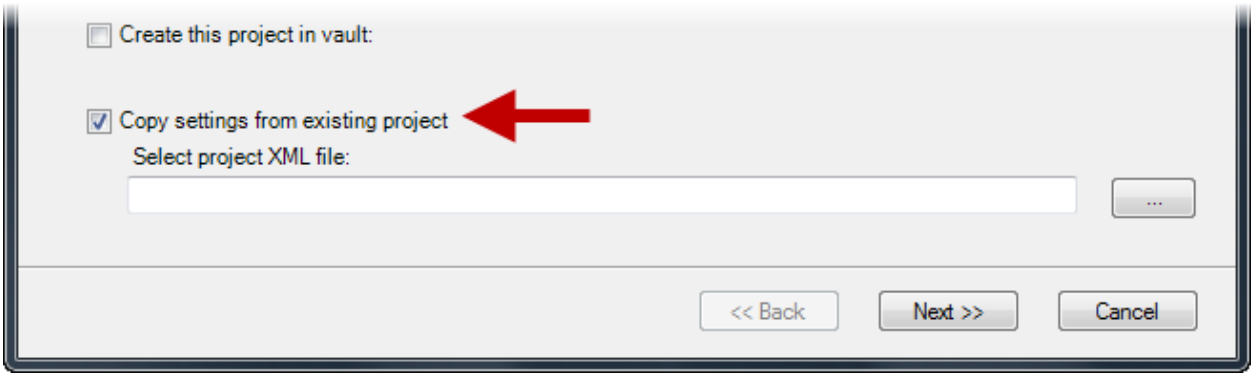


Figure 21: Copy Settings from Existing Project

Database

AutoCAD P&ID and AutoCAD Plant 3D take advantage of project databases to store information about the drawings and models within the project. While drawings are connected to a project database, they also remain independent as each drawing also contains a snapshot of its own database information. The drawings are portable and may be easily shared independent of the project.

Some information, like off-page connectors, which are specifically tied to multiple drawings, would require all the drawings to be shared together.

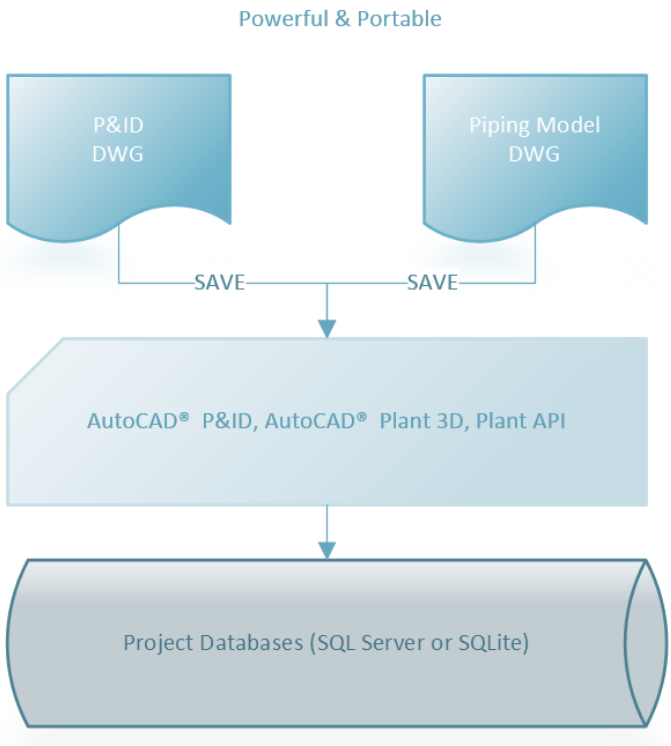


Figure 22: Database Interaction Outline

Types of Databases

Projects use either a local database or a server database. A local or file-based database is one that does not require services running to access the database. A server database requires a Windows service to run in order to access data. The database formats available are SQLite and SQL Server.

SQLite

For the default project setup, the SQLite format (http://www.sqlite.org) is used. SQLite is a file-based format, meaning that an application does not have to be running or launched to maintain a connection to the database. Because of the open format, developers, using many types of technology, have created methods for interacting with a SQLite database (for example, C++, .Net, http://www.sqlite.org/cvstrac/wiki?p=SqliteWrappers). SQLite provides excellent performance for small-scale projects.

SQL Server and SQL Server Express

SQL Server comes in two flavors: SQL Server Express and SQL Server. SQL Server Express is a free version of the Microsoft enterprise database server, Microsoft SQL Server. SQL Server Express has fewer features compared to SQL Server, but it is able to handle virtually any project that uses AutoCAD Plant 3D. Also, if you move from SQLite to SQL Server Express, no additional data migration is required to use SQL Server. SQL Server Express can be installed on desktop operating systems such as Windows 7, XP, or Vista. SQL Server Express can use databases up to 10 gigabytes (GB). For projects that are larger than 10 GB, using a full SQL Server instance is mandatory.

The server can be connected to the user's workstation via local area network (LAN), http://en.wikipedia.org/wiki/Local_area_network. While other connection types are possible, more complicated setups are better left to IT professionals experienced in domain permissions, wide area networks (WANs), and general network configuration protocol.

Database Viewers

For SQLite, the recommended viewer is SQLite Expert Personal which is a free tool:

- http://sqliteexpert.com/download.html

The program is robust and handles fields like guids well where other programs don't.

For SQL Server databases, you should use SQL Server Management Studio:

- Management Studio 2008
- Management Studio 2012 (click the download link and select only management tools)

Choosing a Database Type

Choose your database type based on the number of concurrent users on a project and how actively they are using the project. Because SQLite supports an unlimited number of simultaneous "readers" but only one "writer" at any instant in time (source: http://www.sqlite.org), it is recommended that use of SQLite be limited to small, non-production or single-user projects. If your team has multiple people doing similar tasks (i.e. modeling pipe, modifying P&IDs, etc.) they will probably need to be on SQL Server. Since projects can migrate from SQLite to SQL Server at any time in a project, starting with SQLite is not detrimental.

What signs tell me that I need to use SQL Server?

Most frequently, users who need to switch to SQL Server will get a prompt that says, "The drawing information has been changed, would you like to update the project database?" If you get this prompt on a regular basis, you should switch.

Other users may see property changes not staying. For example, they may set a line number, and then come back later and the value isn't changed. If you are using SQLite, this is an indication that you need to switch to SQL Server.

Schema

A schema for AutoCAD Plant 3D and P&ID will be released with the 2015 SDK (http://images.autodesk.com/adsk/files/PlantSDK_ERD.zip).

Cache

For databases on a network, whether SQL Server or SQLite, AutoCAD Plant 3D caches copies of the information locally. The cache may be purged by going to the installation folder (like C:\Program Files\Autodesk\AutoCAD 2015\PLNT3D\) and running the PnPLocalDataCachePurger.exe application.

You should run the purge application if you are seeing old data or if AutoCAD Plant 3D takes a long time to startup (or doesn't startup). Clearing the cache is typically the first step in troubleshooting.

Cache Links:

- http://www.pdoteam.com/2012/03/plant-3d-startup-hanging/
- http://lazcad.com/2013/11/automatic-cache-clearing

Switching Database Types

To switch database types, follow these steps:

1. Install and connect to SQL Server
2. Back up the project
3. Convert your project
4. Test the project and server connection

Install and Connect to SQL Server

Before you can switch database types, you need to set up SQL Server. For installation instructions, follow this blog series:

- http://www.pdoteam.com/series/plant3d-on-sql-express/

That series also contains an article on how to automate server backups.

Back up the Project

The most complete backup method is to copy the entire project structure. You may zip the project if space is a concern. By copying the entire project, you ensure that you have captured a complete snapshot with the drawings and the database and any other design information.

Once SQL Server or SQL Server Express is installed, your project can be converted.

Convert the project

For project conversion, navigate to the AutoCAD installation and then the AutoCAD Plant 3D folder (C:\Program Files\Autodesk\AutoCAD 2015\PLNT3D\) and locate the PnPProjectMaintenance.exe application. The application allows 3 procedures: converting from SQLite to SQL Server, moving a project, and copying a project.

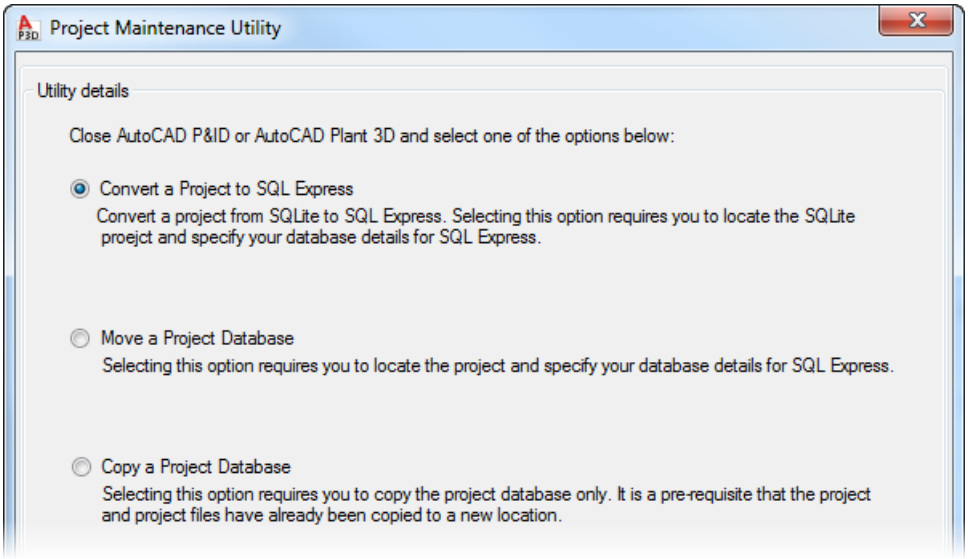


Figure 23: Convert to SQL Server

The conversion requires you to put in the existing project location (not the backup), and then server connection details. Test the connection before trying the conversion. Use the Project Number instead of a generated name, as the generated names will get confusing when looking at a long list of projects.

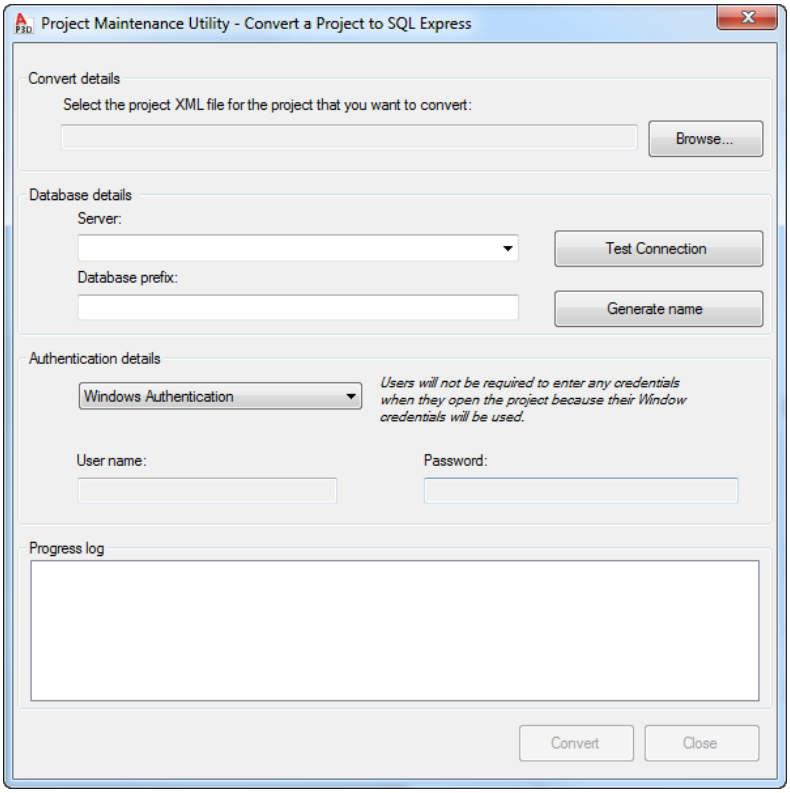


Figure 24: Select Project to Convert

The process will take several minutes depending on the size of the project.

Class Structure and Behavior

Classes are used to organize properties, symbols, and behaviors. On the P&ID side, each class can have symbols associated with it. On the AutoCAD Plant 3D side, each class has shapes drawn as defined by the specs or other model content.

Classes are organized in a hierarchy so that properties can be inherited. The four top level classes on the P&ID side are Engineering Items, Non Engineering Items, Pipe Line Group, and Signal Line Group. The purpose of the Engineering Items class is to be a container that represents any item that may be purchased in a project. Non engineering items are for non-purchasable objects (like off page connectors). The Signal Line and Pipe Line groups allow organizing line items into sets for reporting.

Inheritance

Having a hierarchal class structure means that child classes, like Equipment, get the properties from their parent class, as well as their own properties. For example, the Manufacturer property is defined on the Engineering class. If you select the Equipment class, you will see a Manufacturer* property. The * indicates the property is inherited from a parent class.

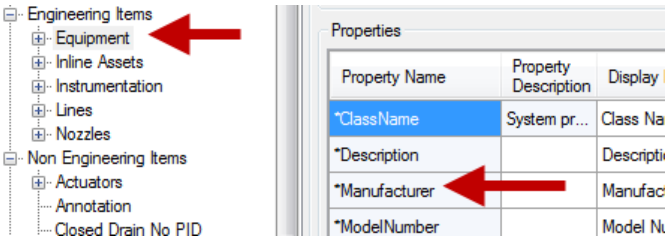


Figure 25: Inherited Properties

Notice also that the Non Engineering Items, Pipe Line, and Signal Line group classes do not have a Manufacturer property, as there is no parent common to all of them with that property.

Property Inheritance applies to AutoCAD Plant 3D classes as well. However, the default properties are not the same, and you cannot create custom classes on the AutoCAD Plant 3D side.

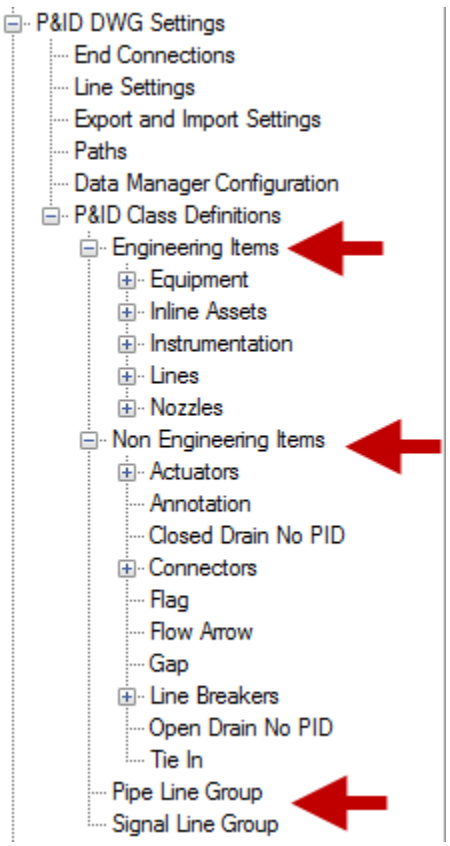


Figure 26: Top Level Classes

Property Flows

To understand where properties may be used, examine the property workflows in the images below.

Project and Drawing Properties

Project and drawing properties can be created and modified in the project setup.

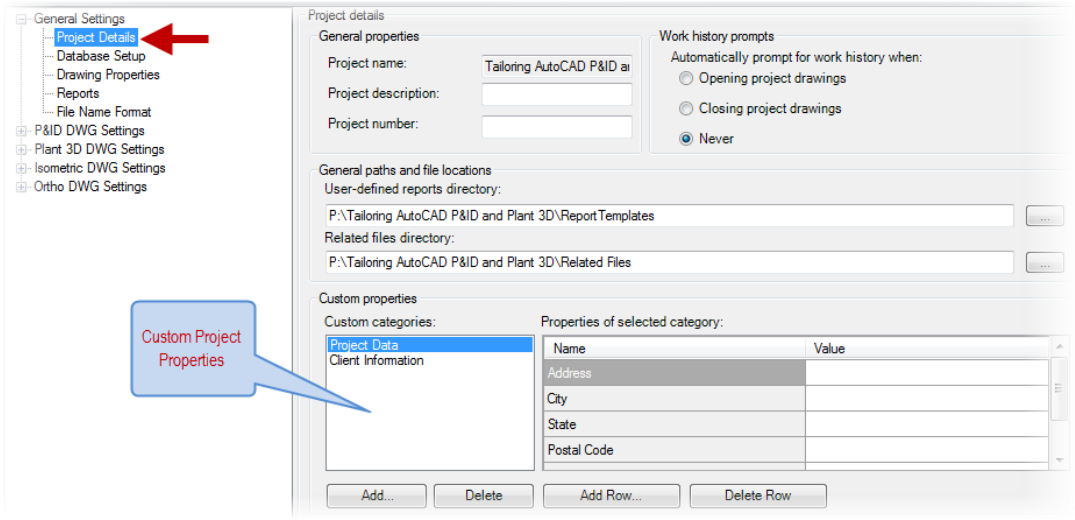


Figure 27: Custom Project Properties

Custom Drawing properties are on the Drawing Properties node.

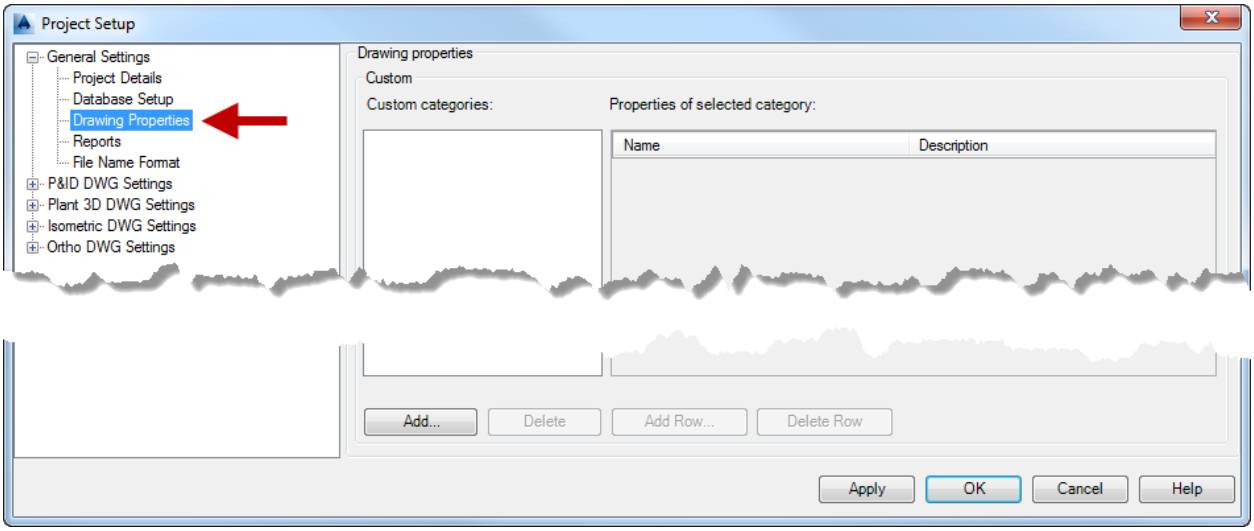


Figure 28: Custom Drawing Properties

Note that properties that will be used on isometrics should not have spaces in the category name or in the property name.

Project and drawing properties are available to use on reports, P&IDs, and orthographic drawings. Custom drawing properties are not available for use on Isometrics.

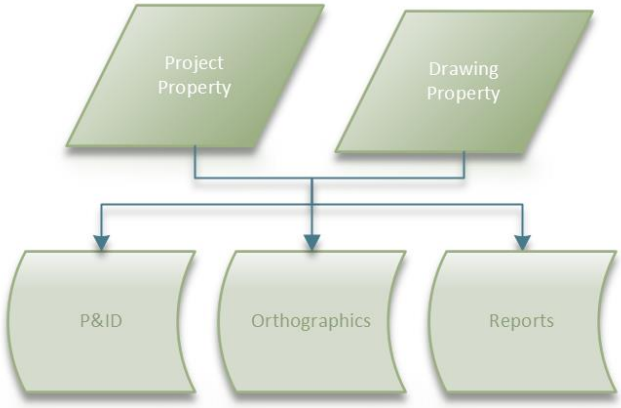


Figure 29: Property Usages

P&ID Class Properties

P&ID class properties are defined in the project setup, and available to use in P&IDs, reports, and push to AutoCAD Plant 3D parts through P&ID Object Mapping settings.

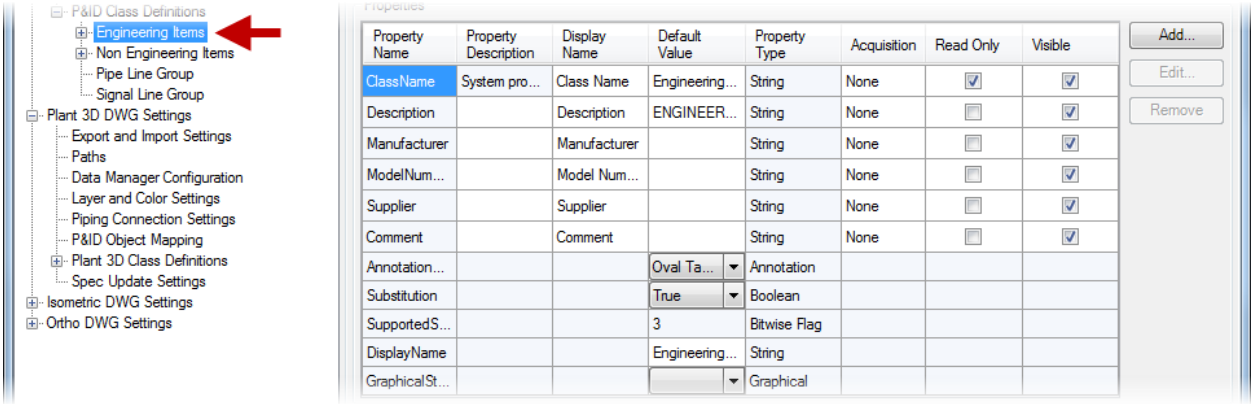


Figure 30: P&ID Class Properties

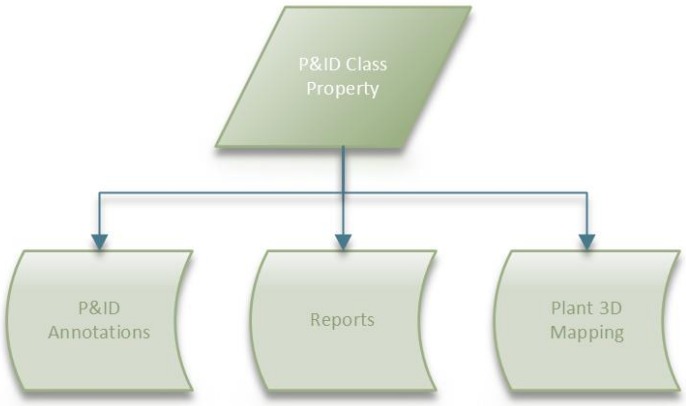


Figure 31: P&ID Class Property Usages

AutoCAD Plant 3D Class Properties

AutoCAD Plant 3D class properties are defined under the AutoCAD Plant 3D DWG Settings. These properties are available for use in Orthographic drawings, Reports, and Isometrics.

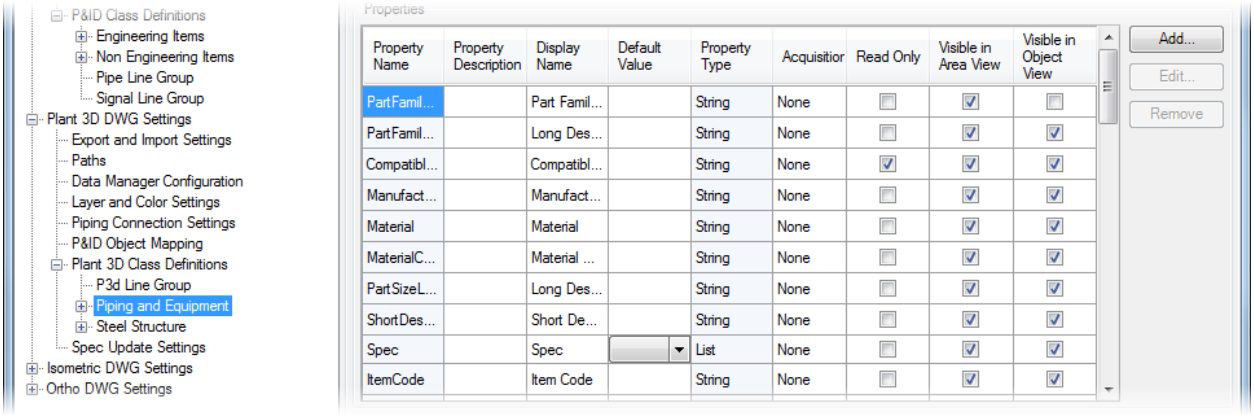


Figure 32: AutoCAD Plant 3D Class Properties

In addition to being able to define default property values, you can create custom values that read from the specs and catalogs. If a part has the same property defined in a spec as in the part's parent catalog, the spec will obtain the value from the catalog. Likewise, if a project class contains the same property as the spec part, the project will obtain the value from the part's spec. In both of these scenarios, the property names (not descriptions) must match.

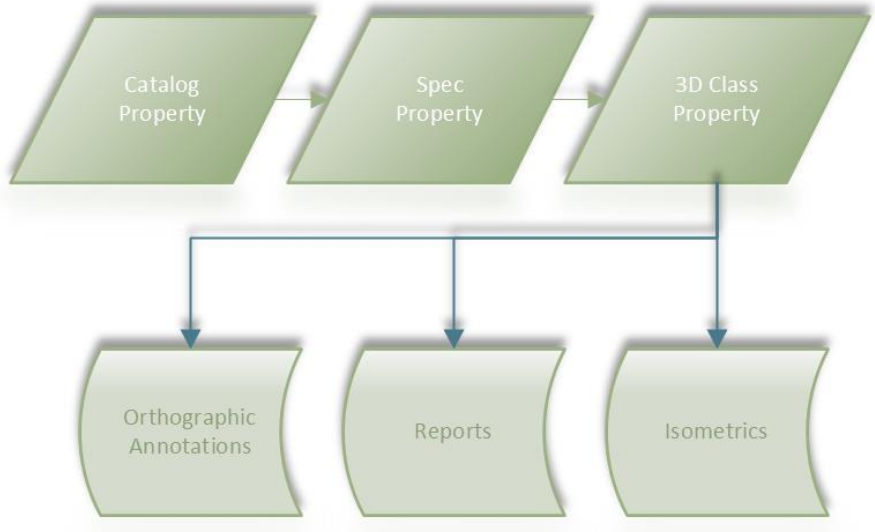


Figure 33: AutoCAD Plant 3D Class Property Usages

Custom Class Design (P&ID only)

While it may be tempting to create your own class structure within the Engineering Items class, you should only create a new class if there is not an existing class that utilizes the functionality you need. For example, if you create a new Hand Valves class (like “Your Company Hand Valves”) to hold custom valves classes, some of the functionality that the default Hand Valves uses may not be available to your custom class. Check these guidelines to decide if you should create a new class:

- Is there already a class for the category of item you want to insert?
- Does the existing class exhibit the behaviors you need?

If the answer to these two questions is yes, do not create a new class. There should not be a class per each symbol that you want to use, but a class for the category of symbol (i.e. Gate Valve). If you create extra classes, you may have to alter the class mappings for each user, and you may not get the proper object behavior when using the item on the P&ID.

In summary, before creating a class, you need to find the symbol that acts most like the item you need and has the most similar properties. One other caveat, if you are removing an existing class, you will have to remove any blocks that use that class from the project before you can delete the class.

Combination (PLANTDEFINECALCPROPERTIES)

AutoCAD P&ID and AutoCAD Plant 3D have a tremendous tool that allows you to combine property values.

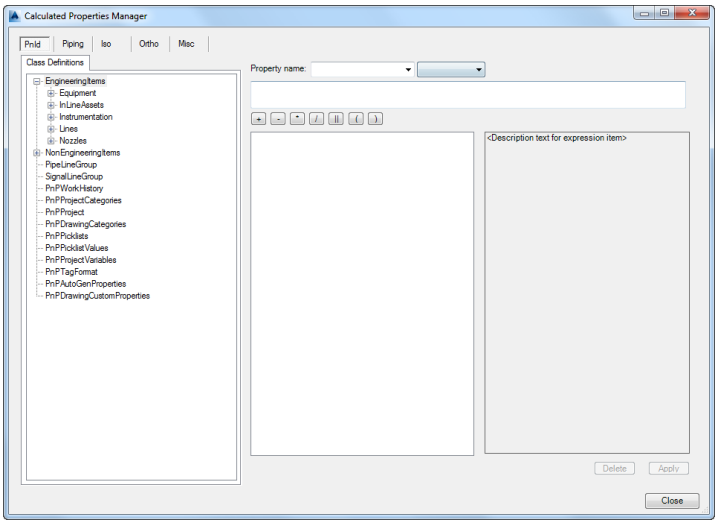


Figure 34: PLANTDEFINECALCPROPERTIES

You can view a knowledge base article on setting up an alternate tag format. You can also set up complex line number properties to use on your isometrics: http://www.pdoteam.com/2013/07/complex-line-numbers-in-isometrics/

Before using a calculated property, you should be aware of two issues:

First, you cannot include the Size property in your combined field. The Size property will not display. Secondly, calculated properties will not be displayed in reports as they do not get saved in the database, but are calculated as the program runs.

The expressions for the calculated properties are defined in the .dcfx file associated with the project part.

Folders

Within the project manager you can setup two different types of folder. You can create a standard folder that exists in the project manager and the Windows folder structure. You can also create a virtual folder which only resides in the project manager. Virtual folders are useful for categorizing the structure while maintaining the existing project folder structure.

By default in 2015, setting the folder name modifies the path. However, once you have set the name, you can click browse to change the modified path to any folder of your choosing.

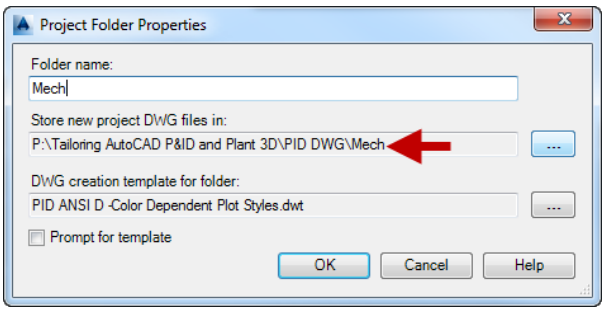


Figure 35: Create Project Folder

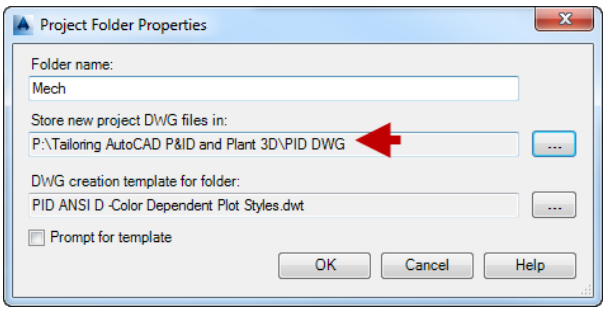


Figure 36: Create Virtual Folder

Using the Merge option will create the new folder in the project manager while keeping the contents in the selected folder.

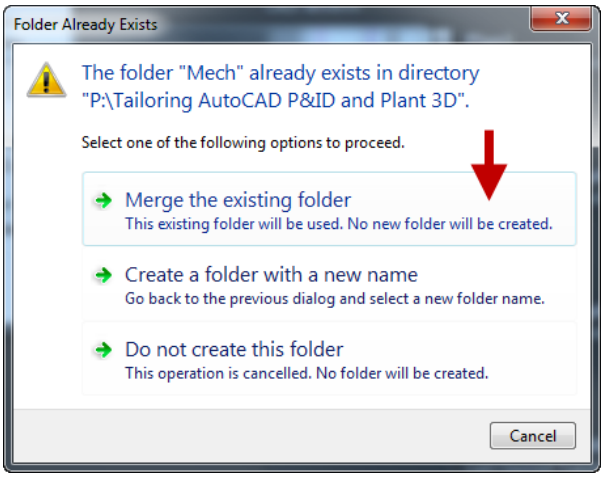


Figure 37: Merge to Existing Folder

With each created folder, you can supply a different template, or have it prompt for a template.

Filename Format

For each project, a file name format may be defined. The filename format will apply to each drawing in the project, but may be overridden. Only one filename format may be applied to a project. Within the types, selection lists may be used. When the drawing name is set, whether by using a file name format or not, the Drawing Title property uses the value given as the file name.

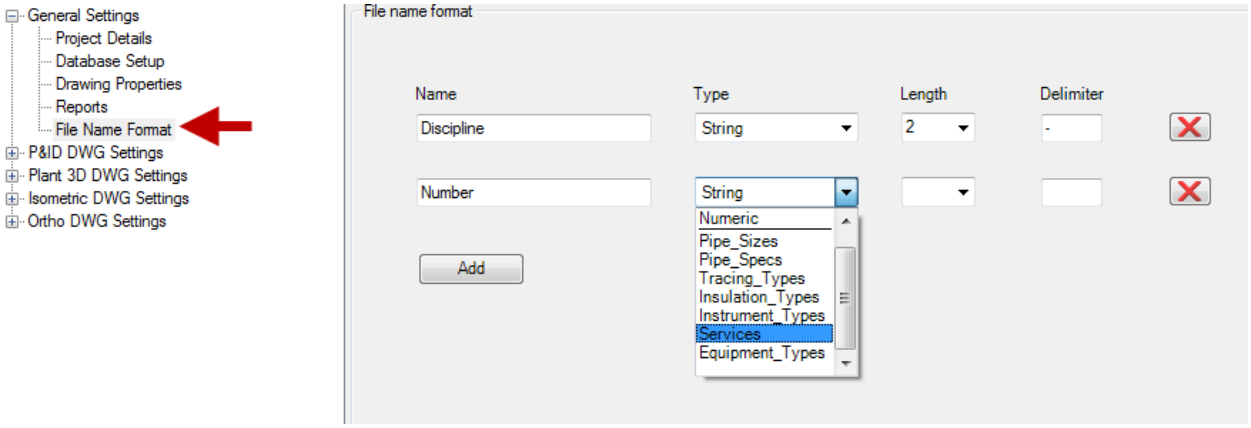


Figure 38: File Name Format Options

Title Blocks

By allowing the program to manage your title block information, you can greatly increase your productivity. You will have to setup title blocks for three areas of the program, P&ID, Orthos, and Isometrics. The P&ID and Ortho title blocks can be used from the same template, but the isometric title block must be setup through the Isometric Title block setup for your company style.

Configuring a title block allows us to define project and drawing properties and display those properties using fields.

AU Online has a link for a class with a walkthrough doc on creating drawings with title blocks:

- http://au.autodesk.com/au-online/classes-on-demand/class-catalog/2012/autocad-pid/customizing-autocad-pid

The main key is to modify your title block from within a project drawing. Use fields to include custom properties, and then put a space in for all the values. After that, you can use SaveAs to create a template. Some project properties will not hold a blank value (or a space), so you have to fill them in if you are going to use them in a title block.

Fields may be used by right-clicking and choosing Insert field while editing text or attributes.

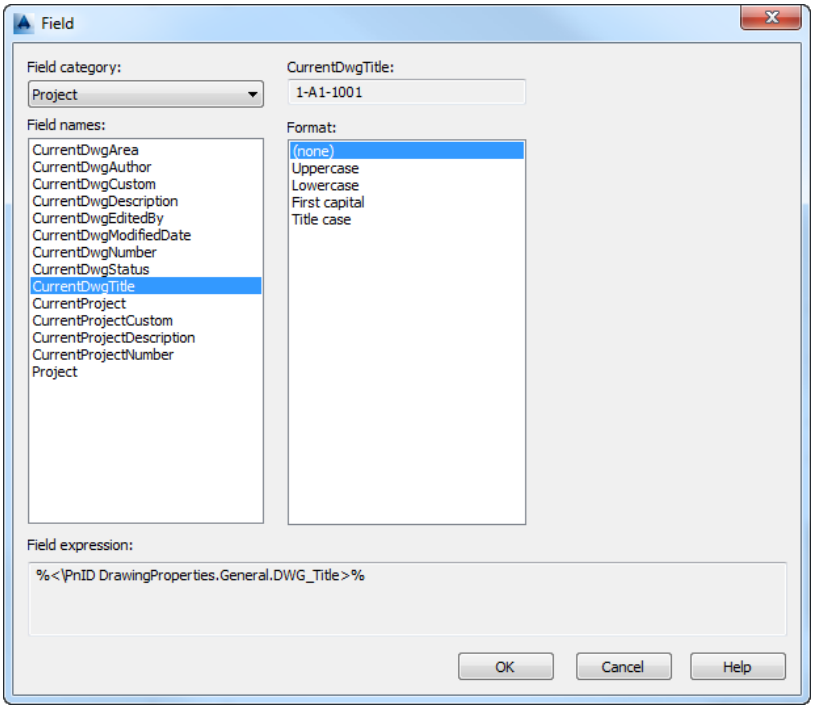


Figure 39: Choosing Field Properties

Sharing

Because of the powerful, portable database structure used with AutoCAD P&ID and AutoCAD Plant 3D drawings, sharing projects and drawings is simple and easy.

Sharing a Project

To share an entire project, create a zip of the top level folder, and send it via email or other service. If your drawings are outside of the project.xml folder, make sure you include a folder that contains the drawings and the project.xml folder.

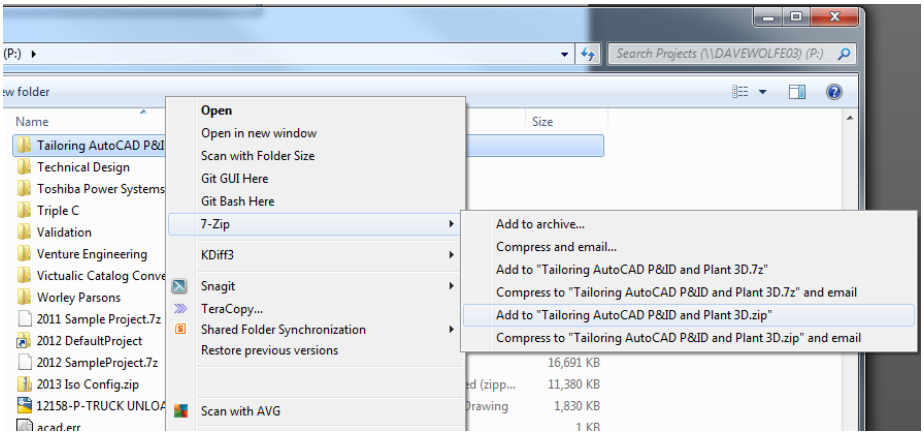


Figure 40: Zip a Project

Sharing a P&ID project

Because AutoCAD P&ID can be used separately from AutoCAD Plant 3D, you can separate out extra files that don't relate to the P&ID project. You can greatly decrease the size of the project and reduce the amount of customized information given out by including only the files shown below.

Name	Date modified	Type	Size
Equipment Templates	1/28/2014 9:50 AM	File folder	
ImportExportSettings	1/28/2014 9:50 AM	File folder	
PID DWG	2/11/2014 3:46 PM	File folder	
Related Files	1/28/2014 9:50 AM	File folder	
StringTables	2/10/2014 11:33 AM	File folder	
DefaultConnectorsConfig.xml	12/16/2012 8:54 PM	XML File	19 KB
Misc.dcf	1/28/2014 9:50 AM	DCF File	35 KB
MiscPart.xml	2/11/2014 9:28 AM	XML File	3 KB
MiscPart.xml.plck	2/11/2014 2:39 PM	PLCK File	0 KB
PIDTo3dClassMapping.xml	2/11/2014 9:28 AM	XML File	921 KB
PnIdPart.xml	2/11/2014 2:43 PM	XML File	130 KB
PnIdPart.xml.plck	2/11/2014 2:39 PM	PLCK File	0 KB
ProcessPower.dcf	2/11/2014 3:46 PM	DCF File	1,029 KB
ProcessPower.dcfx	9/22/2011 9:54 PM	DCFX File	151 KB
Project.xml	2/11/2014 9:28 AM	XML File	3 KB
Project.xml.lock	1/28/2014 11:53 AM	LOCK File	0 KB
projSymbolStyle.dwg	6/5/2013 5:00 AM	AutoCAD Drawing	409 KB
SubstitutionPalettes.xml	5/22/2010 10:54 AM	XML File	76 KB

Figure 41: Sharing P&ID Project Only

Sharing Individual Files

Because the data for each drawing is stored internally, you can send individual files. In the drawing annotation checker, you can specify whether to update the off page connector annotations or not.

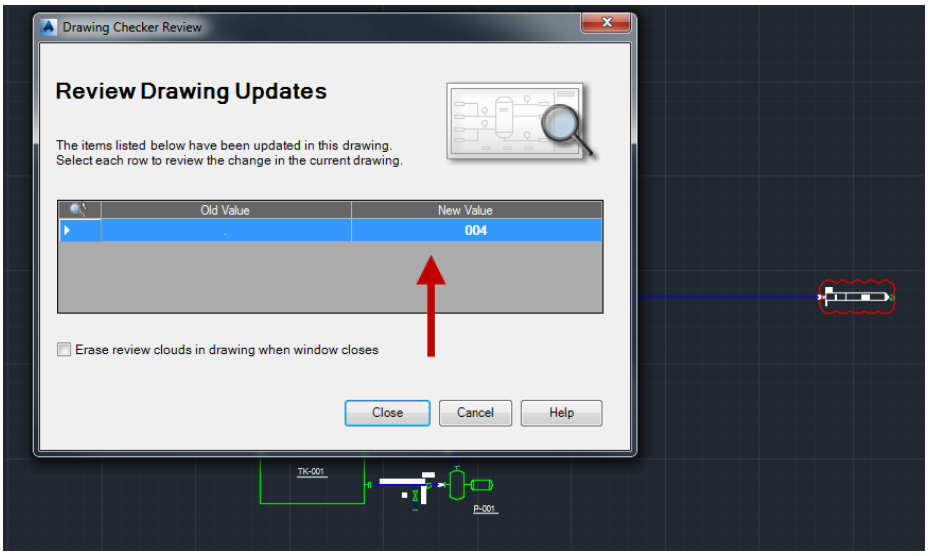


Figure 42: Drawing Review Checker

When an individual file is copied into a project, the program first checks to see if all of the classes from the drawings are in the project database. Then it checks to see if all of the properties from the drawing are in the project database. A log will be created of the classes and properties that do not exist.

All of the classes and properties common to the project and drawing will be updated using the drawing values.

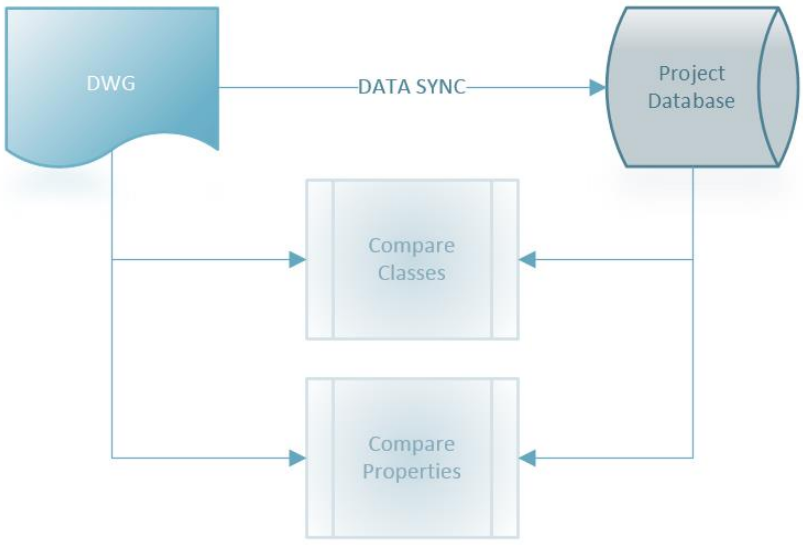


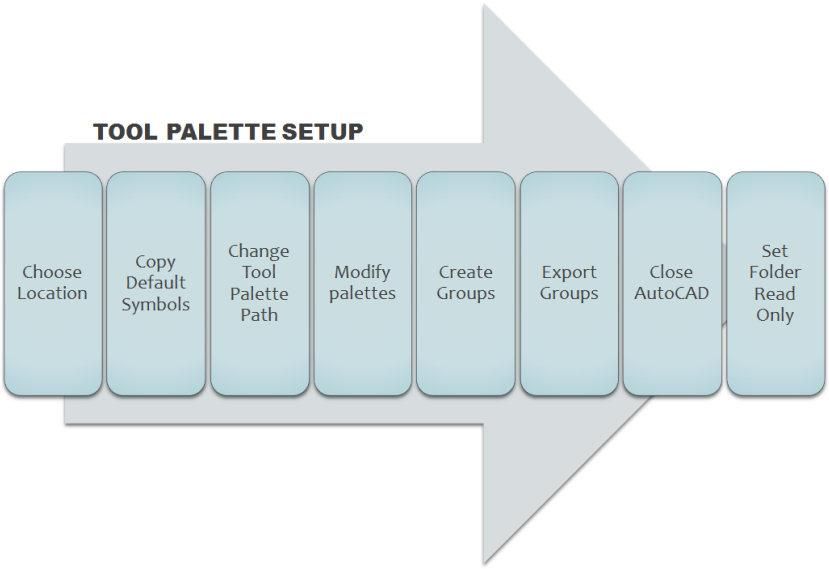
Figure 43: Data Sync Process

Project Symbols

In AutoCAD P&ID tool palettes are using for symbol insertion. Managing these palettes can be done through standard AutoCAD techniques.

View these articles for more in depth help on managing palettes:

- http://au.autodesk.com/au-online/classes-on-demand/search?full-text=tool+palettes
- http://au.autodesk.com/au-online/classes-on-demand/class-catalog/2011/autocad/managing-your-standards-and-reusable-content-with-autocad-tool-palettes-revealed
- http://au.autodesk.com/au-online/classes-on-demand/class-catalog/2012/autocad/the-guts-of-autocad-cad-management-simplified



In order for project symbols to be usable by everyone, they need to be put on a network location. To setup this up, we will follow eight steps.

Figure 44: Network Tool Palette Setup Process

Choose a Location

Create a location similar to K:\Plant Content\Tool Palettes, or choose one that fits your existing network folder structure. The location can be managed in a CAD admin folder (with permissions) or a regular folder. At this point, no one should be using this location for palettes. If they are they must exit AutoCAD before you can modify the palettes.

Copy Default Symbols

Decide whether you will start off with the OOTB (out of the box) symbols, or if you only want to create your own. If you are going to use the default ones, copy them from your local user data folder:

- C:\Users\%USERNAME%\AppData\Roaming\Autodesk\AutoCAD Plant 3D 2015\R20.0\enu\Support\ToolPalette

Change the Tool Palette Path

The tool palette path is stored in AutoCAD options. Set this location to the network folder.

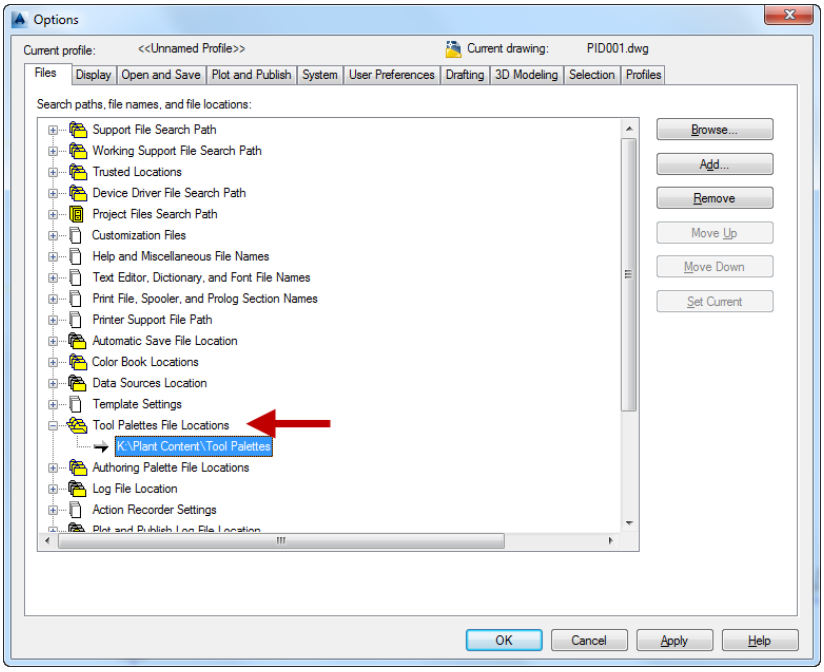


Figure 45: Set the Tool Palette Path

Modify your palette

Create your palettes and arrange them by using standard AutoCAD methods. Add objects by going into project setup and click add to tool palette.

For custom line types, refer to these articles:

- http://www.pdoteam.com/series/custom-pnid-lines/

Create Groups

If you are integrating different client symbols, or default symbols, you should organize these using groups. The user can switch groups to view a subset of the palettes.

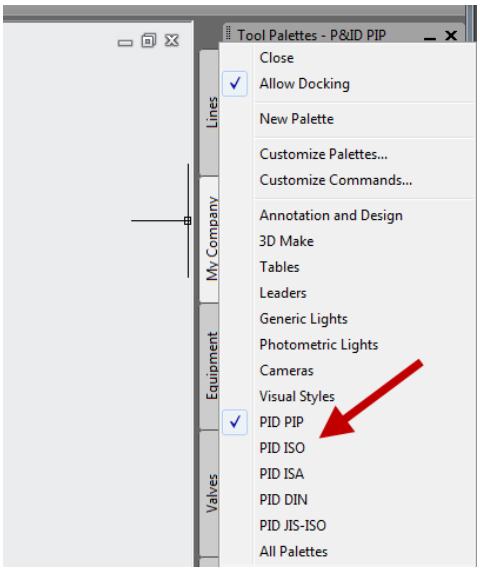


Figure 46: Selecting a Tool Palette Group

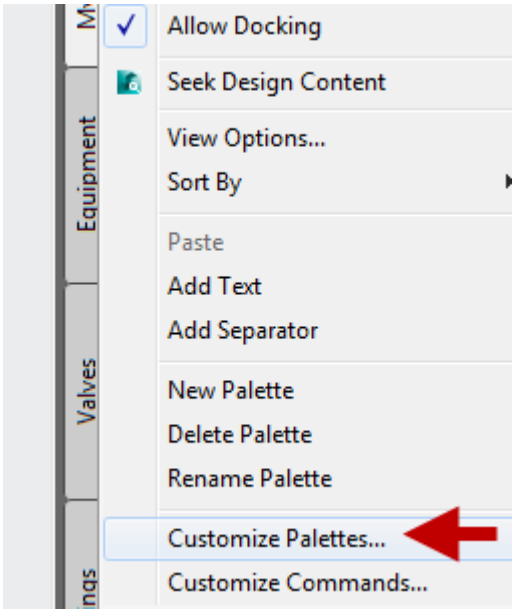


Figure 47: Customize Palettes

Unfortunately, groups cannot be managed from the network, so users will have to import them in order to organize their palettes.

To access groups, you can right-click on a palette, and go to customize palettes. Create a new group by right-clicking and then choosing new group. Then you can add a palette to the group by dragging and dropping it into the group folder.

Export Groups

To export a group, right-click on the folder and choose export, or you can choose the export all option.

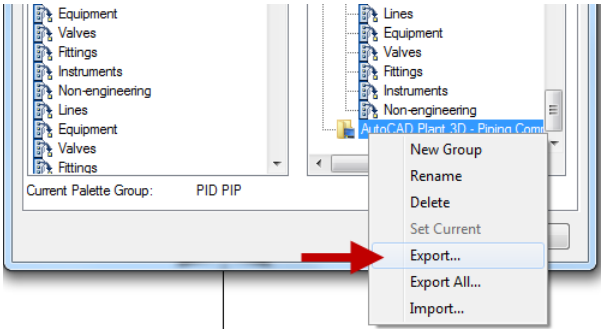


Figure 48: Exporting Default Tool Palette Groups

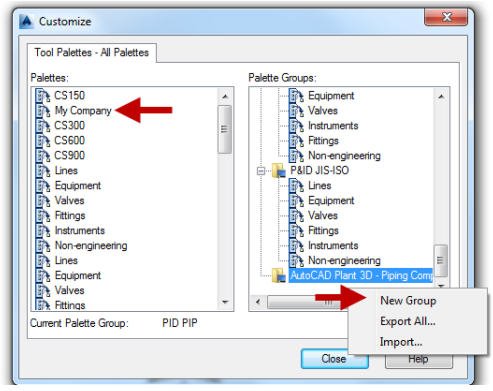


Figure 49: Create Palette Group

AutoCAD will export a .xpg file. Save this in the tool palette folder on the network. Use the same access methods to import the tool palette groups onto each of the user computers. As a backup, you should export all of the default groups before making modifications.

Close AutoCAD

AutoCAD doesn't save the tool palette changes permanently until the program closes. Once you have made your modifications and exported your palette groups, close AutoCAD.

Set Folder to Read-Only

In order to properly manage tool palettes, only one user can edit them at a time. To facilitate this, you should set the tool palette folder to read-only while palettes are not being edited. Select the folder in Windows Explorer, and click Properties.

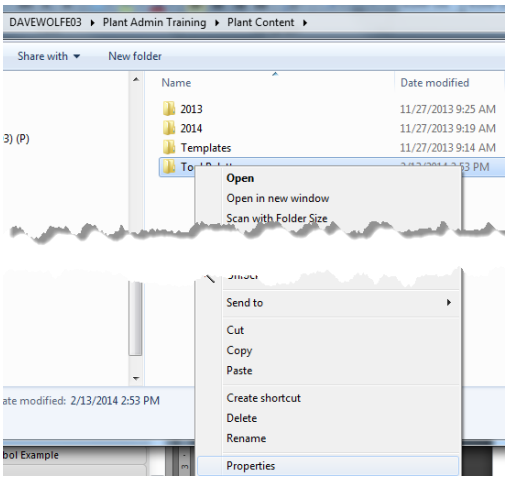


Figure 50: Set Palette folder to Read-Only

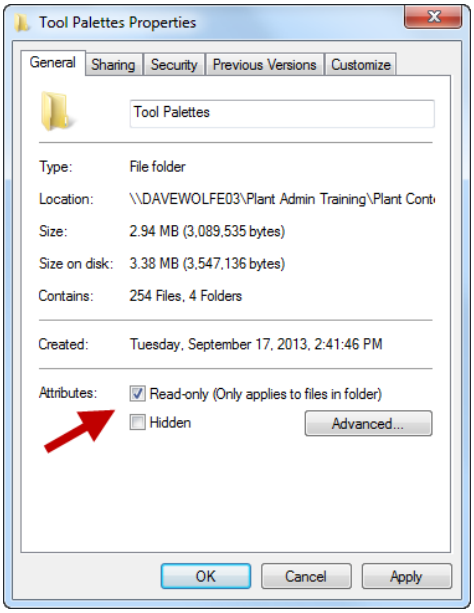


Figure 51: Check Read-Only Property

Even if you manage access to the folder via group policy, people with read/write access should use the read-only option to prevent admin users from accidentally changing the tool palettes.

Other Considerations

Once the initial tool palette setup is complete, you may customize your AutoCAD options further. If you only created company palettes on the network (did not copy default palettes), and you wish to make the default palettes available, you can add another tool palette path to user AutoCAD options.

Make sure to keep the network path below the user modifiable folder, as the top folder will be the tool palette read/write directory. On your own computer, you will have to move the network path to the top to modify the network symbols.

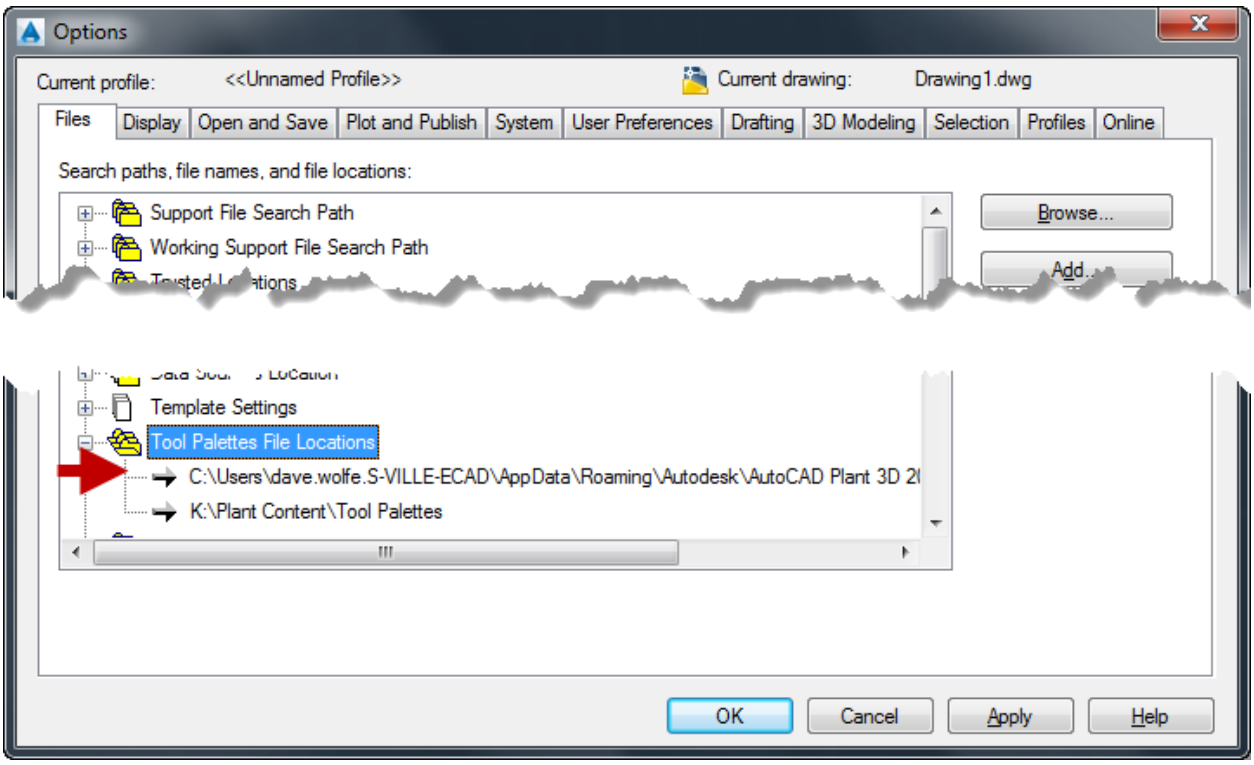


Figure 52: Multiple Palette Folders

Going Local

Some companies have a need to provide settings for mobile users. In those cases, use offline drives for CAD standards like tool palettes and templates, or use a program like Microsoft's SyncToy to copy the content to a local folder (whether on the network or not).

If using offline drives, they often do not get initialized before AutoCAD starts. To have them initialize properly, use the reply in this article.

- http://answers.microsoft.com/en-us/windows/forum/windows_7-networking/mapped-network-drive-is-connected-but-shows-as-not/4ac5ff92-863c-4921-8632-e38cbd136fb4

http://support.microsoft.com/default.aspx/kb/937624?p=1
To configure the EnableLinkedConnections registry value, follow these steps:

1. Click **Start** , type **regedit** in the **Start Search** box, and then press ENTER.
2. Locate and then right-click the following registry subkey:
   HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Policies\System
3. Point to **New** , and then click **DWORD Value** .
4. Type **EnableLinkedConnections** , and then press ENTER.
5. Right-click **EnableLinkedConnections** , and then click **Modify** .
6. In the **Value data** box, type **1** , and then click **OK** .
7. Exit Registry Editor, and then restart the computer.

SyncToy can be downloaded from here:

- http://www.microsoft.com/en-us/download/details.aspx?id=15155

P&ID Content

AutoCAD P&ID controls the symbols, layers, colors through the project setup dialog. Each class may have a setup of symbols associated with it, and the class will define the default properties for that symbol.

Layers

The project setup layers available for symbols are read from the projsymbolstyle.dwg. To have new layers show up in the Symbol Settings dialog, add your layer to the projsymbolstyle.dwg. The projsymbolstyle.dwg may be modified outside of the project setup command. While the projsymbolstyle.dwg controls the layers, it will not override existing layer settings in a drawing, but the template and/or current drawing will control the actual layer settings like color, line type, and line weight. Because usually the layer defines the necessary object properties, typically the Color, Linetype, and Line weight will all be assigned by layer.

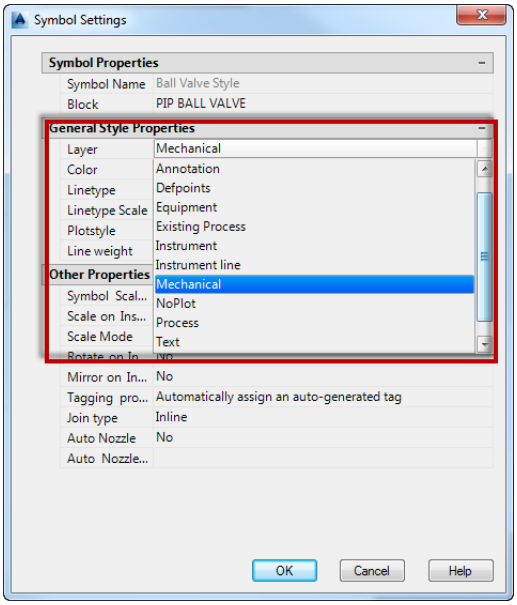


Figure 53: Symbol Settings

Selection Lists

Selection lists allow you to pre-define a list of commonly used values without limiting the values entered in the cell. To view selection lists, go to P&ID DWG Settings > P&ID Class Definitions > Engineering Items > Equipment. Select Type and click Edit.

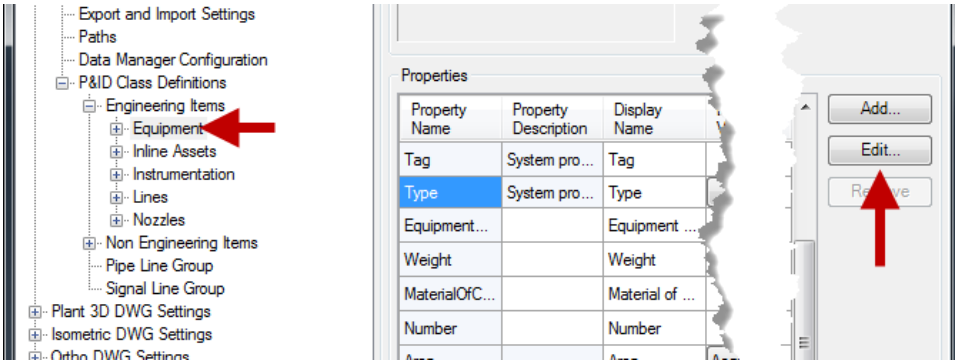


Figure 54: Modify Selection Lists

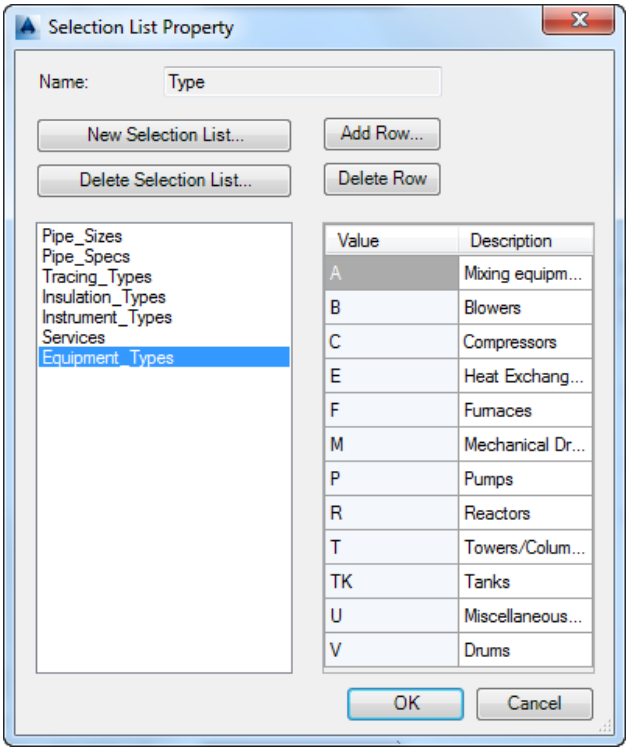


Figure 55: Selection Lists Dialog

You may edit, remove, or add lists as needed when you create properties.

The selection lists for AutoCAD P&ID are separate from AutoCAD Plant 3D. For example, you would have to modify the Services for AutoCAD P&ID and the Services under AutoCAD Plant 3D Classes to make a matching services list.

Before creating a selection lists, make sure to note the order of the items as you want them to appear. You cannot re-order items in the list without removing items.

The Selection Lists plugin in the App Exchange store can help manage list entries by importing/exporting to Excel.

Symbol Lists

Symbol lists are similar to selection lists in that you can select a property value. The difference is that with a symbol list the selection is tied to a symbol style. For example, the Normally property is defined at the Hand Valves class level. If you change the Normally property on Ball Valves, you can see how the symbols are linked to the property value. A class may only have one Symbol list property.

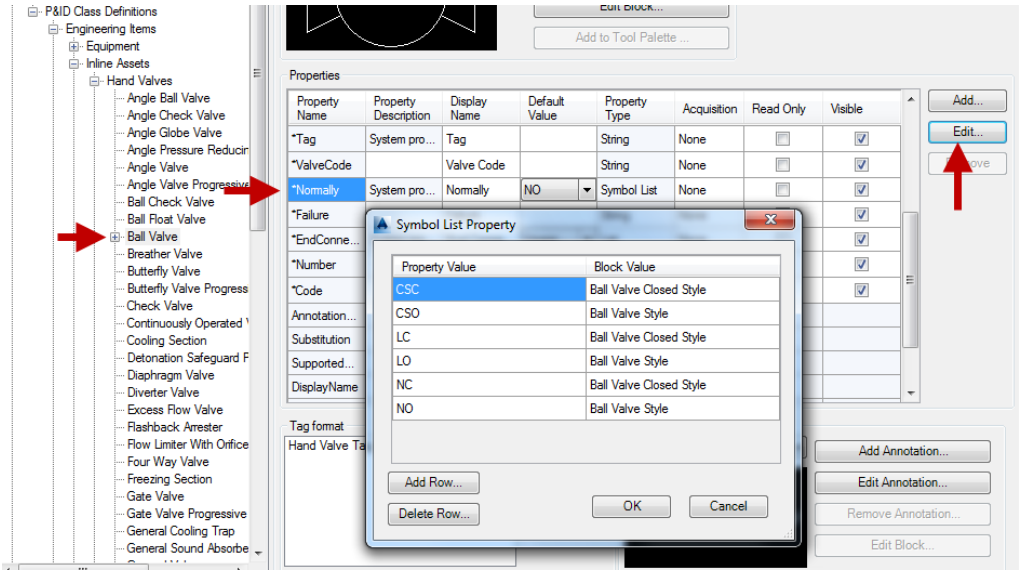


Figure 56: Modifying Symbol Lists

Symbols

There is a difference between how Block Definitions are managed for Annotations vs. Graphical Styles. By default, each Annotation Style is always associated with a unique Block Definition. When creating a new Annotation Style, if you select a Block Definition for an Annotation Style that already exists, the product will automatically make a copy of that Block Definition for you to modify.

In contrast, Graphical Styles are designed to share Block Definitions. If you create a new Graphical Style and select the Block Definition for an existing Graphical Style, both the old and new Graphical Style will use that one Block Definition, and changes to that Block Definition will affect both Graphical Styles.

The reason that an Annotation Style automatically creates a new Block Definition is that usually, each Annotation Style will define different Attribute Definitions with different Format Strings, thus each Annotation Style must have its own Block Definition.

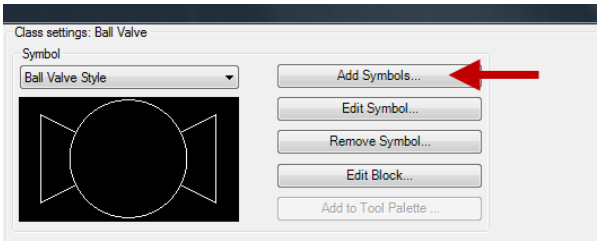


Figure 57: Add Symbol

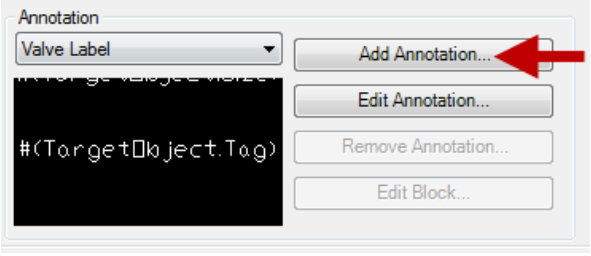


Figure 58: Add Annotation

A block definition is the way AutoCAD keeps track of a block in a drawing. You can view the block definitions by using the BEDIT command. A block reference is a selectable representation of the block definition within the drawing area.

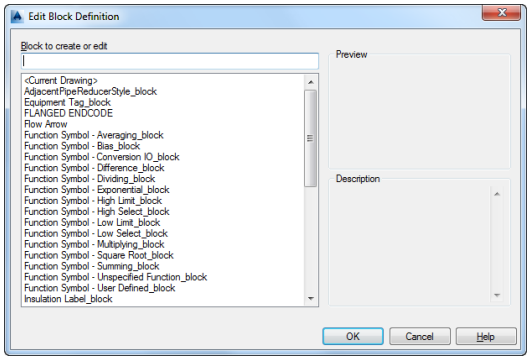


Figure 59: View Block Definitions

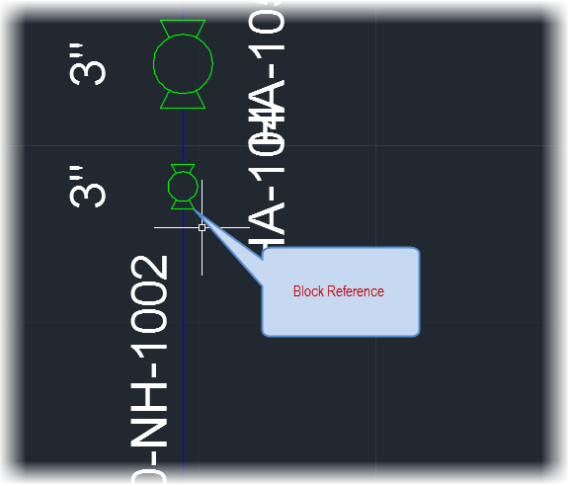


Figure 60: Block Reference Example

The block definitions for the project symbols are stored in the projsymbolstyle.dwg. To create a new symbol you can use any standard AutoCAD method for creating a block. Typically the block already exists in a drawing, so the Create Symbol dialog asks you to select a drawing that has the block within it. To select a block that is part of the default AutoCAD P&ID package, select the current project's projsymbolstyle.dwg.

Create new Screw Pump Symbol Example

Use the methodology of your choice to create the new Block Definition. In the following example, the Block Definition for the PIP Screw Pump has been copied and the copy has been renamed and modified. The new Block Definition has several lines added to the pump body. Both versions are shown.

Remember to save the drawing in which you have defined the new block definition.

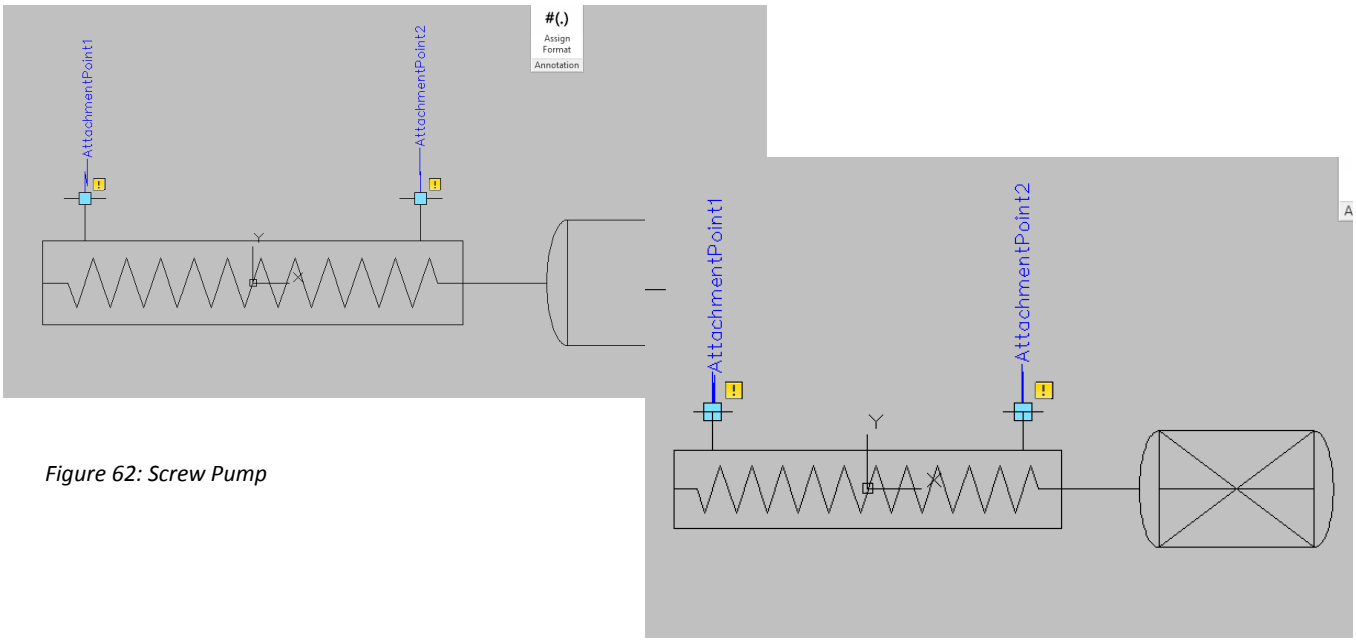


Figure 62: Screw Pump

Figure 61: Reversible Screw Pump

The next step is to add a new Graphical Style for the Screw Pump class based on this Block Definition. Open the Project Setup dialog and navigate to the SCREW PUMP node under P&ID Object Definitions.

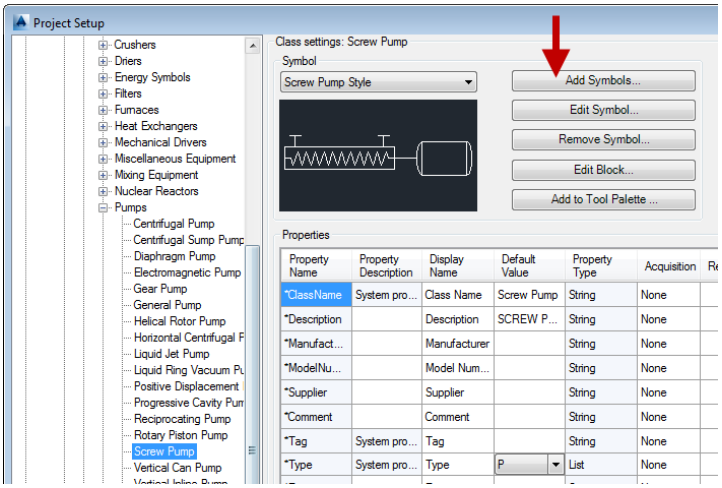


Figure 63: Add Symbol

Pressing the Add Symbols button raises the Browse dialog. Navigate to the current project's projsymbolstyle.dwg.

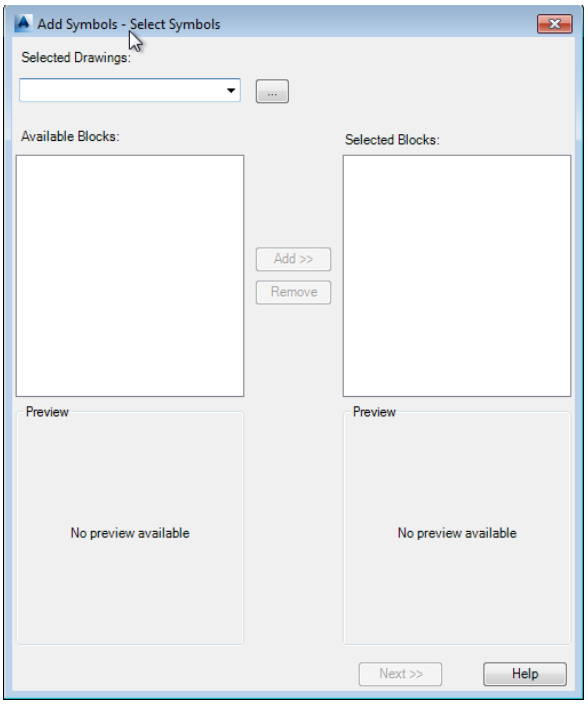


Figure 64: Add Symbols Dialog

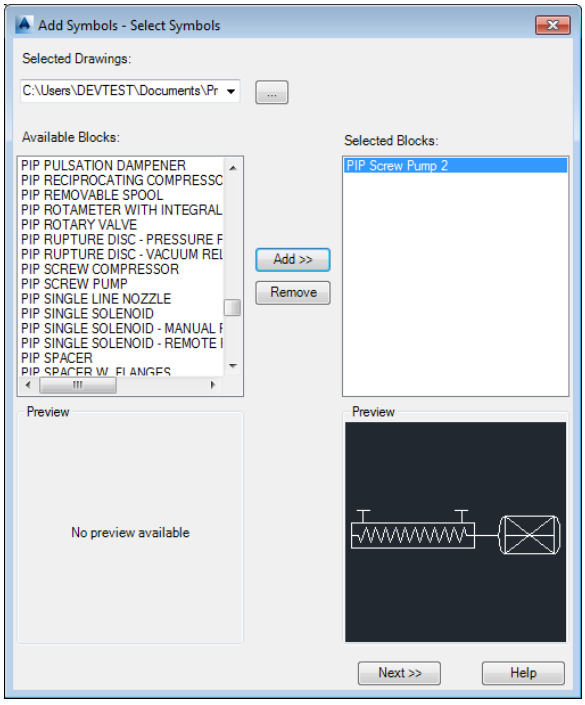


Figure 65: Select Symbols

Type in a name (Reversible Screw Pump). You also can specify AutoCAD object properties like Layer, Color, Linetype, and more here.

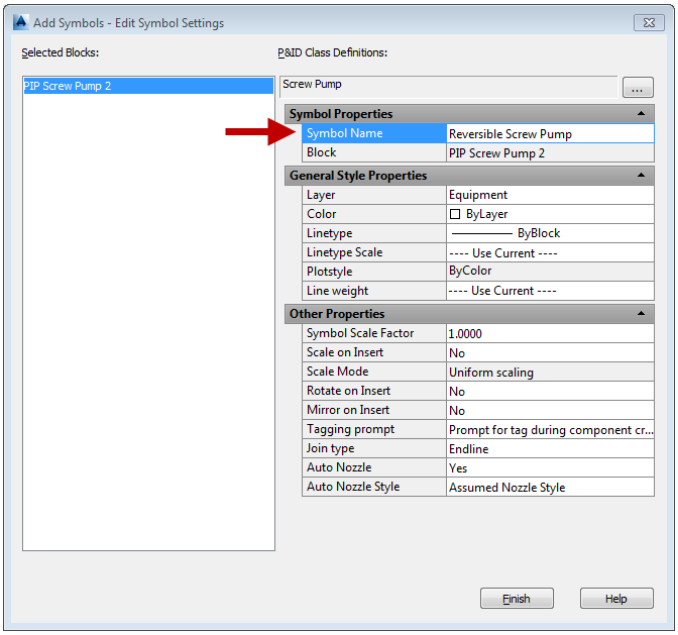


Figure 66: Edit Symbol

The new style is has now been created in the Project Setup and is available.

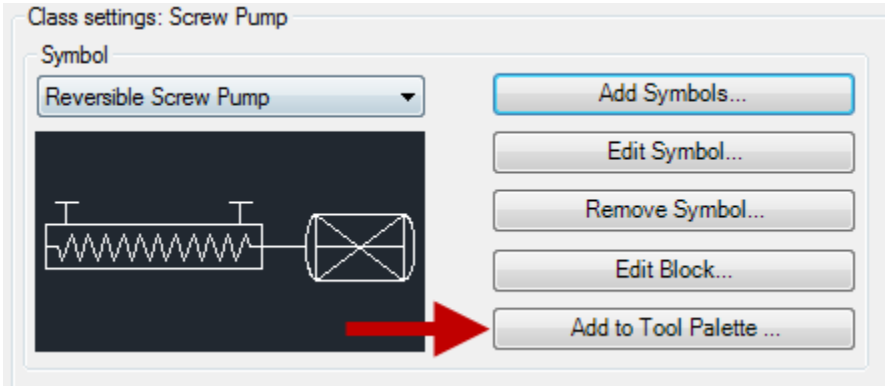


Figure 67: Add Symbol to Tool Palette

For the final step, press the “Add To Tool Palette…” button to place a button on the Tool Palette which will create a screw pump using this new style.

The following dialog indicates that the tool has been successfully added, but be aware that it will be added to the bottom of the palette group currently active, and you may need to reposition in using drag/drop into the desired location on the palette.

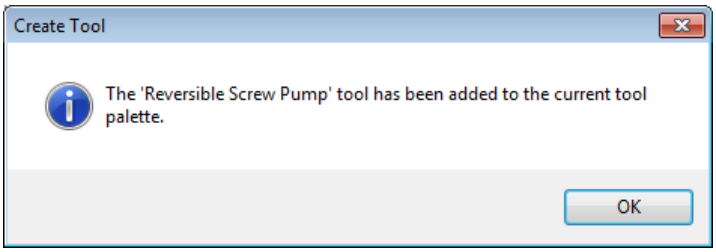


Figure 68: Symbol Added Dialog

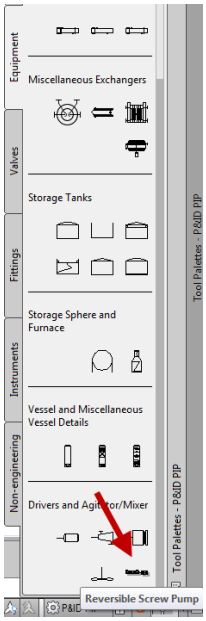


Figure 69: Initial Position of new symbol

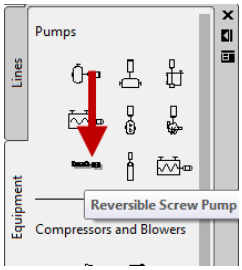


Figure 70: Symbol Moved to new location

Batch Adding Symbols

To create several symbols at once, select them from a drawing file. Then you may assign a class and different properties to each symbol in the list. By using this method, you can significantly decrease the time it takes to make new symbols.

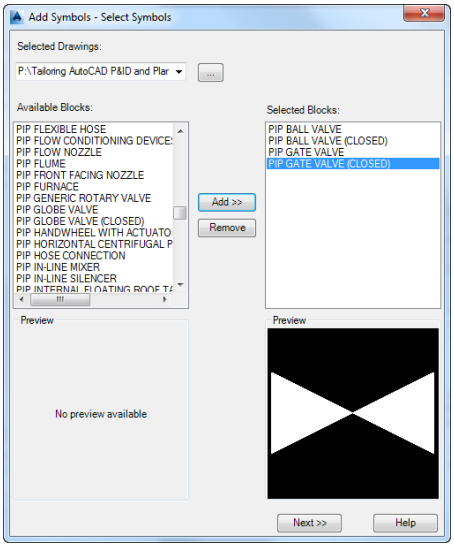


Figure 71: Select Multiple Symbols

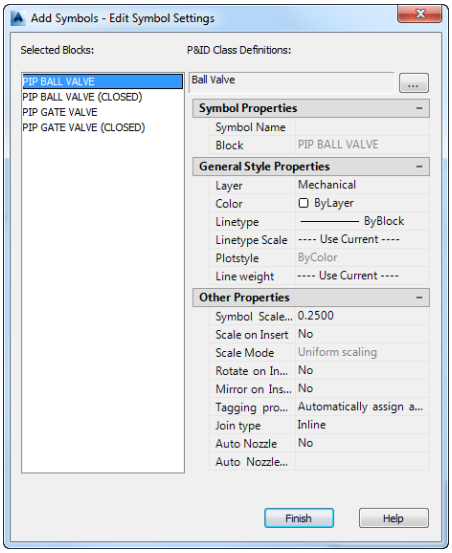


Figure 72: Set Symbol Properties

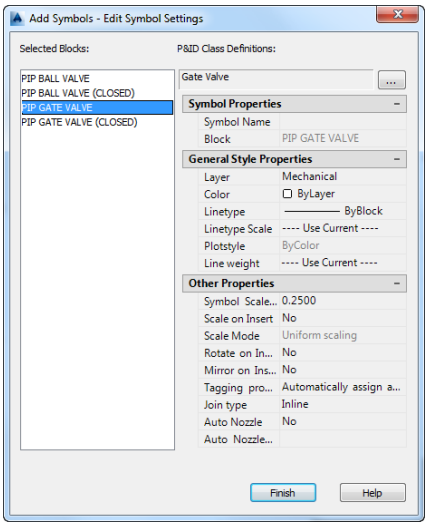


Figure 73: Modify all symbol properties

Connection Point Requirements

In order for your blocks to function correctly within the AutoCAD P&ID environment, you need to be aware of connection point requirements. You can use blocks without attachment points, and the program will break the line at the extents of the block. However, you will not be able to take advantage of any of the features mentioned below.

Attachment Points in the block definition of a graphical style have two purposes.

The first purpose is to provide snap points. When you begin drawing a PipeLine from a piece of equipment, any Attachment Point in the equipment's block definition functions as a snap point.

Additionally, for assets that have flow direction, the naming of the Attachment Points is important. The validation feature, when completed, will interpret the flow direction of an asset as going from AttachmentPoint1 to AttachmentPoint2, and will use this to recognize situations in which an inline asset's flow direction is in conflict with the flow direction of the line upon which it sits. Validation will also check for line connections at the attachment point to determine if a line is fully connected to an asset.

Finally, some assets, such as Hand Valves, support the "End Connections" property. This property allows for various end connection graphics to be drawn on the valve. These graphics also rely on the attachment points in order to determine where to draw. In addition, they rely on the label of the attachment point to determine at what angle to draw. The intention to support endcodes is indicated by adding a colon, the word EndCode, and an angle, in counterclockwise degrees, to the attachment point. Thus, an attachment point label could be: AttachmentPoint1:EndCode180.

The following example shows attachment points with endcodes defined for both a 2 port and a 4 port valve.

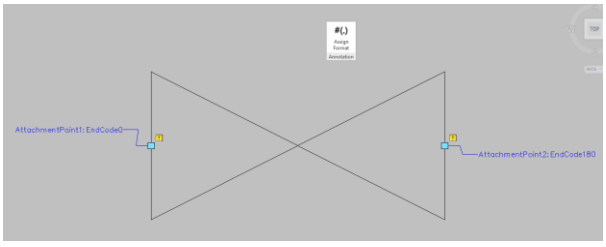


Figure 74 Two port Attachment Point Block

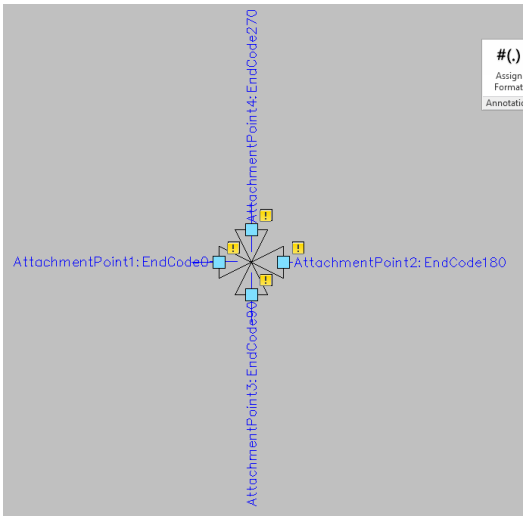


Figure 75 Four Port Attachment Point Block

Guidelines for Orienting Attachment Points

Most inline components, such as those found on the Fittings and Valves tabs of the PIP Tool Palette, are horizontally oriented by default. Their attachment points fall on the X axis, following the flow of a horizontal line either to the left or right. Some components, however, are oriented by default along the Y axis. If you place such a component, for example, a flame arrestor, on a horizontal line, it automatically rotates to align with the line's direction.

When configuring symbols for components, keep in mind the following guidelines:

- The imaginary vector drawn from Attachment Point 1 (AP1) to Attachment Point 2 (AP2) defines the direction of the component.
- Most components are horizontally oriented, with the vector along the X axis.
- Some components, such as Flame Arrestors, have AP1 to AP2 along the positive Y axis. These are vertically-oriented components.
- If the vector is non-orthogonal, it is ignored.

NOTE These guidelines do not apply to items on the Equipment tab of the tool palette or to General Instruments on the Instruments tab. Equipment (endline components such as tanks and pumps, for example) cannot be inserted inline. Similarly, General Instruments, which contain text inside graphics, cannot be inserted inline.

Symbol Settings

Once your block has been defined properly, you will want to test its behavior to see which symbol settings are appropriate.

The items under General Style Properties are standard AutoCAD settings.

Scale, Rotate, and Mirror on Insert are self-explanatory.

Tagging Prompt

Available tagging options are automatically insert, prompt, and not a tagged component. Note that while this setting may be applied to a symbol, only classes that are registered in the tag registry will actually allow tags.

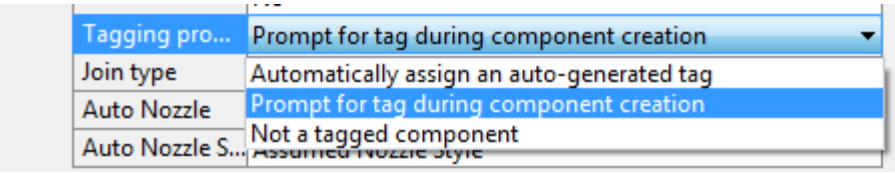


Figure 76: Tagging Prompt Types

The default classes registered for tagging are:

- Equipment
- Nozzles
- HandValves
- Instrumentation
- PipeLineGroup

If you are creating a new class that requires a tag, you can register it for tagging as shown in this article:

- http://lazcad.com/2014/01/how-to-enable-unique-tag-for-p-id-specialty-items-manual

Join Type

The second symbol feature is the join type which specifies how the asset is expected to join with the line. In recent release more emphases has been given to 'Join Type' property of any P&ID symbol to control the behavior of symbols especially the graphical representation with respect to line.

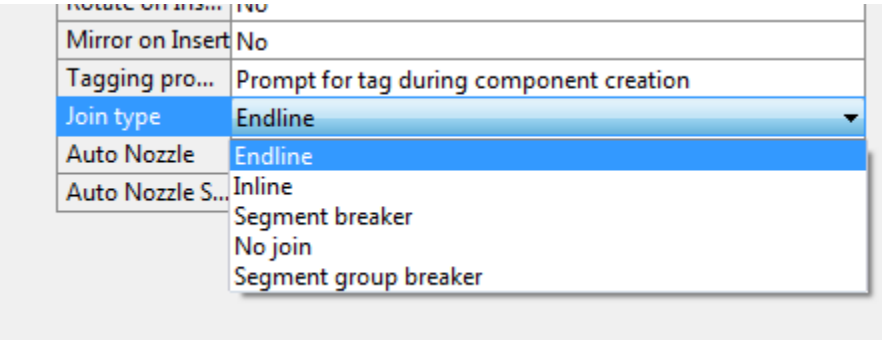


Figure 77: Join Types

- Endline: Asset remains stationary when line is moved. Moving the asset or the line causes the line to elbow. Erasing the line does not erase the asset.
- Inline: Asset cleans the line without breaking it. Moving the line causes the asset to move as well. Erasing the line erases the asset.
- Segment Breaker: Inserts into a line and breaks the line into two segments in the same group. Retains the line group and line number for both line segments.
- Segment Group Breaker: Inserts into lines and breaks line into two segments and two line groups.
- No Join: Does not connect with lines.

This table demonstrates the typical join type for several classes.

Symbol	Endline	Inline	Segment Breaker	No Join	Segment Group Breaker
Gate valve, some ISO style pumps		X			
Horizontal Centrifugal Pump, Tank	X				
Vertical Can Pump					X
Spec Break, Reducer			X		

Nozzles

The Auto nozzle settings control whether a nozzle is placed by the program when a line connects. The symbol chosen will be the nozzle placed. The default pump nozzle type is assumed which places a node at the connection point.

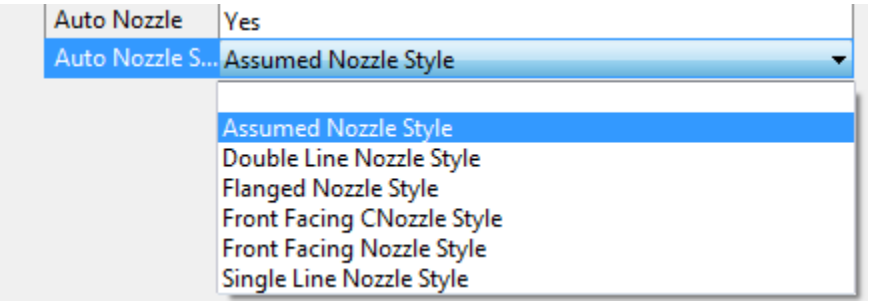
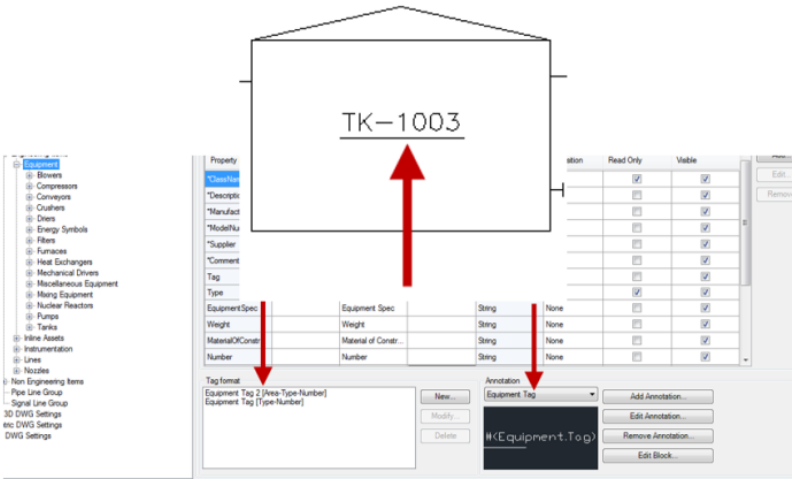


Figure 78: Nozzle Styles

Tagging

When tags are place on a P&ID, you expect to see a unique value that represents a piece of equipment.



The program accomplishes tagging by separating it into two parts, the tag and the annotation.

Here the Equipment class has two tag formats defined, and an Equipment Tag annotation displayed.

Only one tag format can be active for a class at a time within a project.

Figure 79: Equipment Tag and Annotation

Like properties, tags and annotations are inherited from the parent classes.

Tag Formats

Tag Formats are defined at a given level in the P&ID Class Definitions hierarchy, and once defined, are available to be used by all P&ID classes below that node in the Class Definitions tree. The subparts consist of Properties and/or Expressions, and may be automatically generated. The subparts are concatenated together into a single value, and this completed tag value is validated for uniqueness and stored as a tag property in the Data Cache. We'll examine each of these concepts, and discuss a few pitfalls that should be avoided.

Defining a Tag Format for a family of P&ID Class Definitions

Tag Formats are accessed from the Project Settings dialog, under the P&ID Class Definitions portion of the tree.

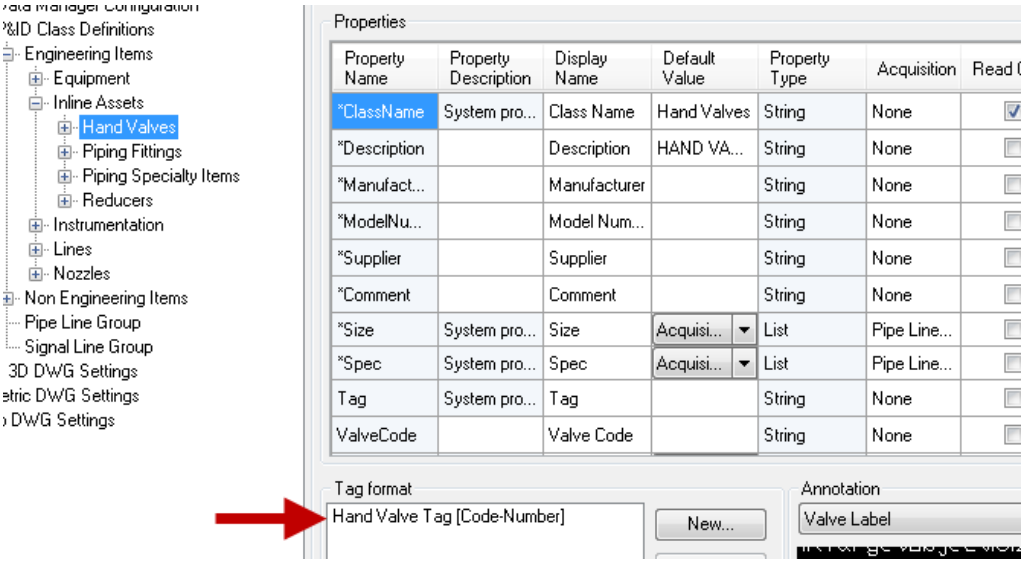


Figure 80: Hand Valve Tag Format

Pressing the "New…" button raises the Tag Format Setup dialog.

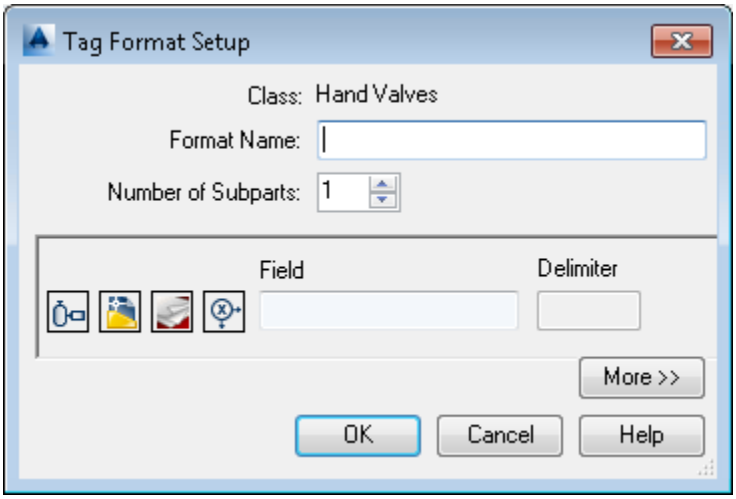


Figure 81: Tag Format Dialog

Subparts can be added and removed using the spin control next to "Number of Subparts." A Tag Format consists of maximum 25 subparts.

- Each subpart consists of a Field and an optional Delimiter.
- The Field is built up from Properties and/or an Expression.

Note that this Tag Format is being defined for the class "HandValves", because the Hand Valves node was selected in the P&ID Class Definitions tree when we pressed "New…"

This Tag Format will be available to all child P&ID classes defined at a more deeply nested level in the P&ID Class Definitions hierarchy; in other words, this Tag Format will be available for all Hand Valves.

The tag value stored in the project is calculated when the tag is assigned, rather than being a live link. If you alter the tag format, you have to re-assign the tag in order for the tag value to update. Existing tags are not changed on drawings or in the database until their tag is re-assigned.

Properties

A Tag Format may specify properties of the object to be tagged, properties of the Drawing, or properties of the Project. The first three buttons next to each sub part provide a means to select a property of each type for the subpart, and are mutually exclusive. We will examine each in turn.

Drawing and Project properties are read from the Drawing and Project respectively, and will become part of the overall tag property, but assignments made to these properties via the tagging operation will NOT be stored back to the Drawing and/or Project. These properties are used for initialization only.

Class Properties

Pressing the leftmost button raises the Select Class Property dialog.

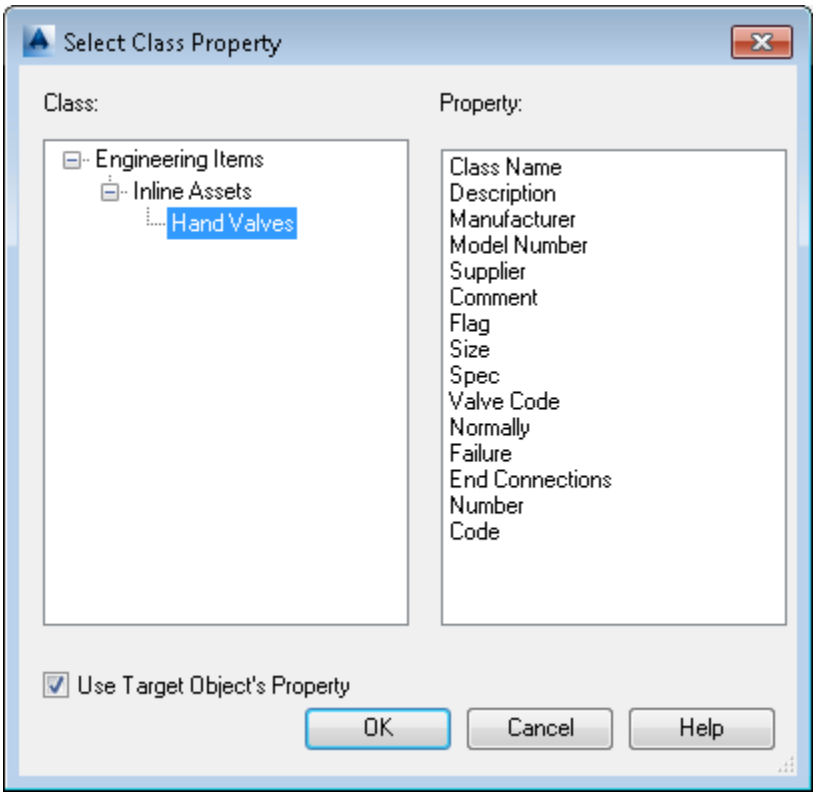


Figure 82: Class Property Selection

In most situations it is necessary to specify a property that actually exists on the class for which the Tag Format is being designed. Therefore, this example navigates to the Hand Valves node in the Class tree, and selects one of the properties, Valve Code, defined for Hand Valves.

The “Use Target Object’s Property” checkbox is also selected. Pressing OK yields the following result for the subpart in the Tag Format Setup.

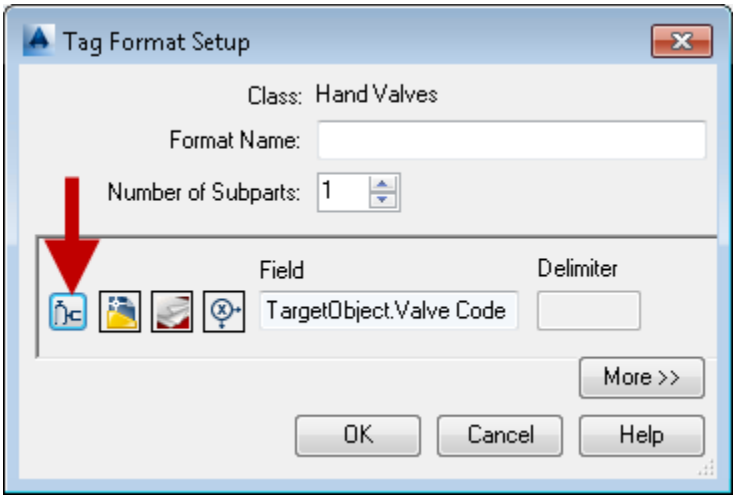


Figure 83: Insert Class Property

As can be seen, the Property has been specified as TargetObject.ValveCode. Also, notice that the graphics for the first button now appear raised. This indicates that a class property is in use.

If the “Use Target Object’s Property” checkbox were deselected, the result would have looked like this:

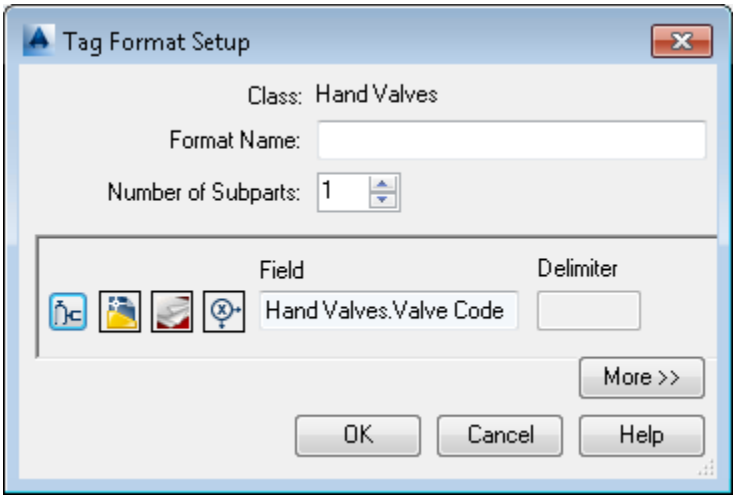


Figure 84: Property Displayed

Notice that the Property is now defined using the actual class name: HandValves.ValveCode.

TargetObject indicates to the program that the requested property is expected to be defined for the P&ID class for which this Tag Format is being defined. It is a shortcut in the nomenclature, and if you are careful in selecting the property, is generally appropriate to use.

Any property available at a given level in the hierarchy is always available at the child level. That is, since "ValveCode" is a property of HandValves, you can be certain that it is a property of Gate Valve, Ball Valve, etc.

There is an exception to the requirement to specify a property defined on the P&ID class being targeted.

In some special situations, you may want to include a property from a P&ID class directly related to the item being tagged, as part of the tag value. For example, the Tag Format for PipeLines specifies a property from the PipeLineGroup as two of the subparts.

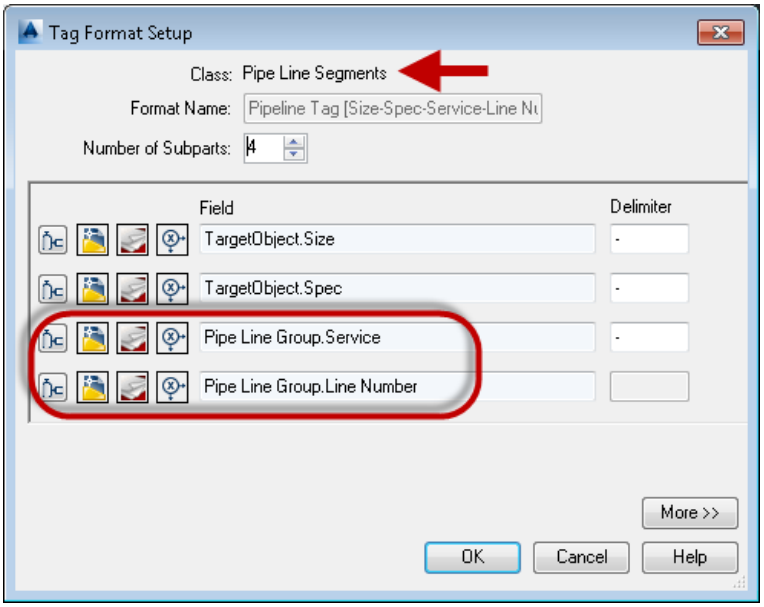


Figure 85: Set Related Class Property

To utilize a class property from a class other than that for which the Tag Format is defined, the following must be true:

- The property must be from a class that shares a direct relationship to the target class.
- The actual class must be specified --the TargetObject nomenclature cannot be used.

Drawing Properties

Pressing the second button beside the subpart raises the Select Drawing Property dialog.

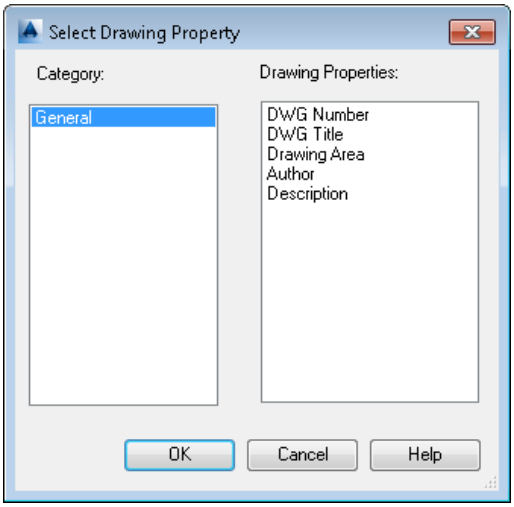


Figure 86: Drawing Properties

A drawing property may be specified as one subpart of a Tag Format. Selecting the properties above yields the following in the subpart.

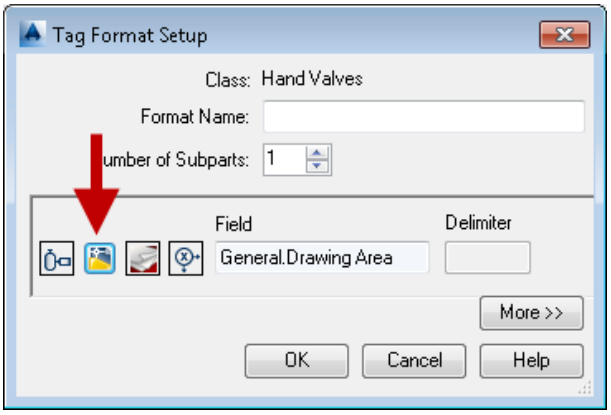


Figure 87: Insert Drawing Property

The second button now shows the raised graphics, indicating that a Drawing property is selected, and the Drawing property is specified in the Field by Category and Property.

Users may add custom Drawing categories and properties in the Project Setup dialog.

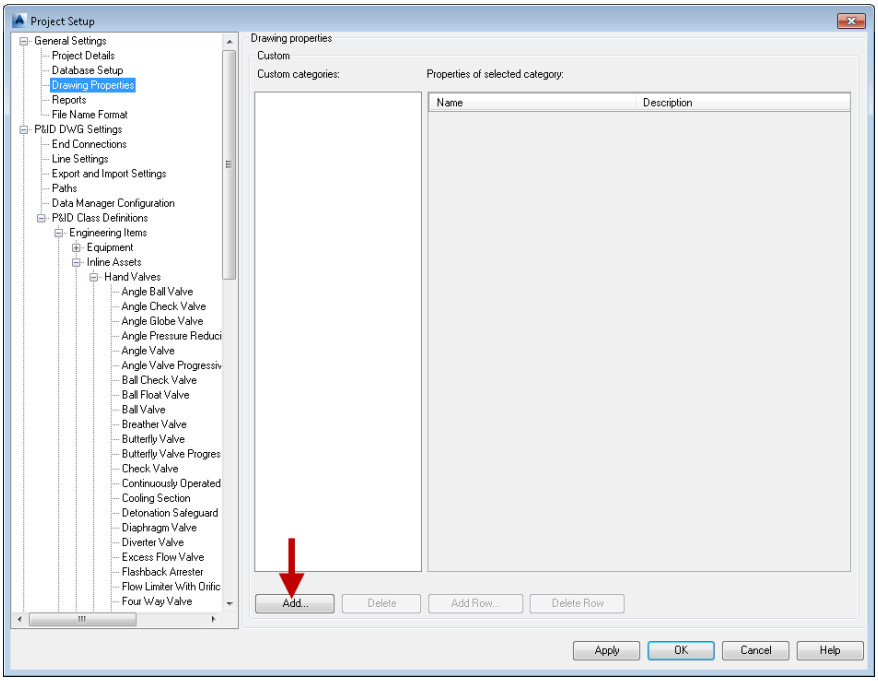


Figure 88: Custom Drawing Properties

Any custom Drawing properties added here will be available to select from the Select Drawing Property dialog.

Project Properties

Pressing the third button next to the subpart raises the Select Project Property dialog.

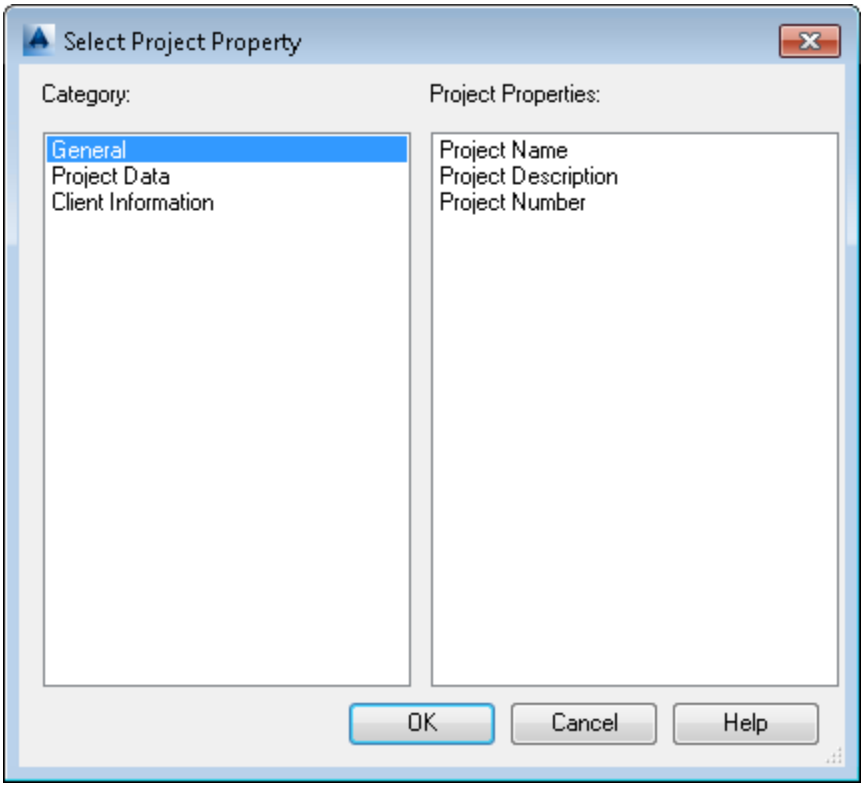


Figure 89: Project Properties

Any project property may be selected as one subpart of the Tag Format. Selecting the properties above yields the following subpart:

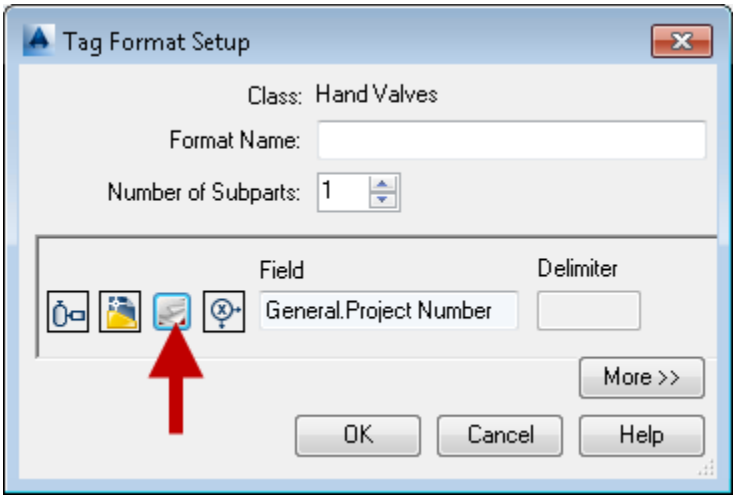


Figure 90: Insert Project Properties

The third button now shows the raised graphics, indicating that a Project property is selected, and the Project property is specified in the Field by Category and Property.

Users may add custom Project categories and properties in the Project Setup dialog.

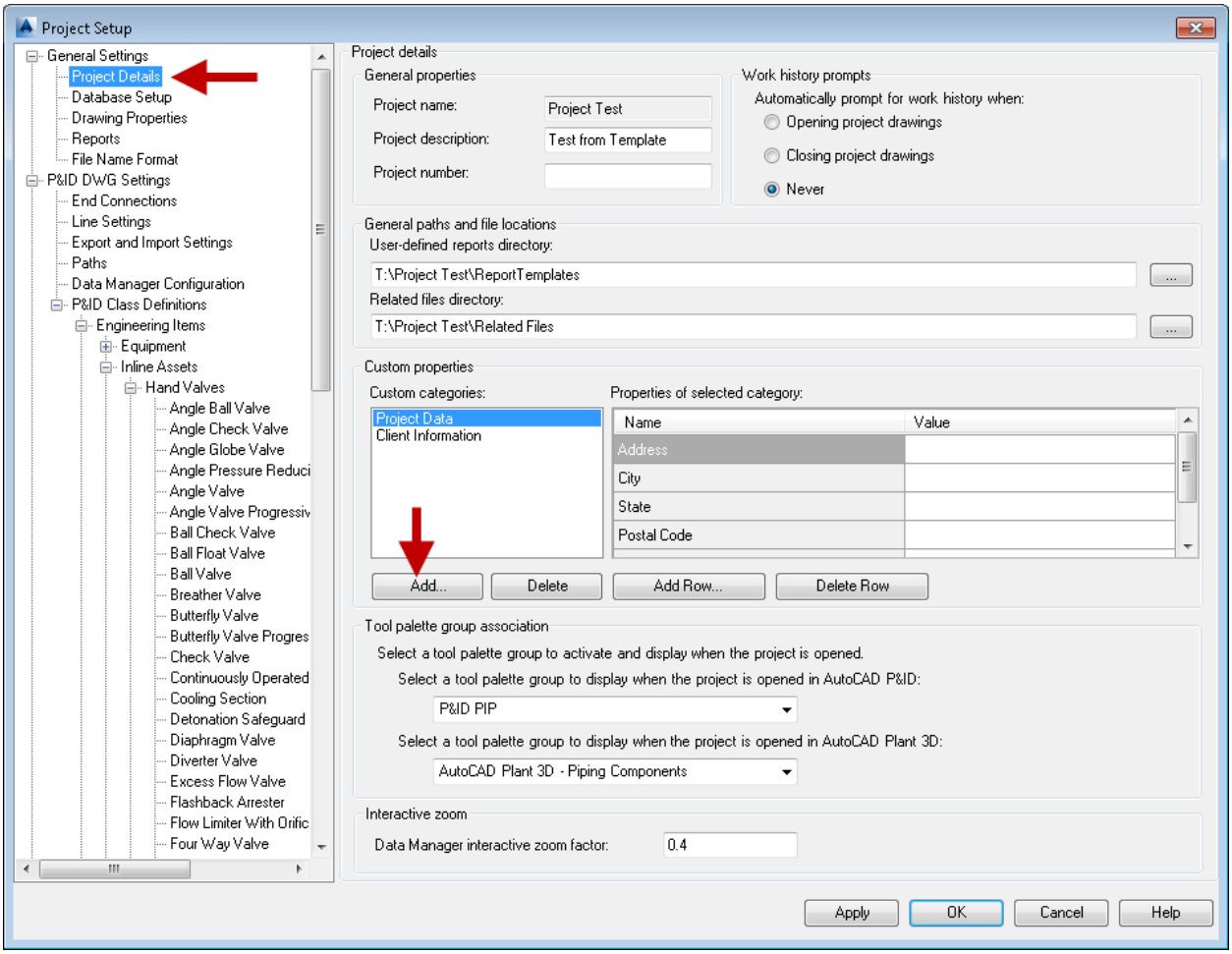


Figure 91: Create Custom Project Property

Any custom Project properties added here will be available to select from the Select Project Property dialog.

Expressions

The fourth button in the Tag Format Setup dialog defines expressions for the tag.

Syntax

Expressions define a simple syntax for character matching.

- The character ‘A’ specifies an upper case alphabetic character.
- The character ‘a’ specifies any alphabetic character.
- The character ‘N’ specifies a digit.
- The character ‘*’ specifies that any number of the previous type of character (A, a, or N) may follow.

When expressions are used in conjunction with a property, they enforce the type of input that is accepted when the property value is being assigned, via the Assign Tag dialog. When an expression is used independently, not in conjunction with a property, it is called a “pure” expression, and also enforces the type of input.

Enforcement is “weak”, meaning that mismatched input will result in user feedback (red text) in the Assign Tag dialog, and an attempt to correct the input automatically via padding, truncation, or uppercasing, if possible. However mismatched input that cannot be corrected is still accepted.

The upper two sections of the Define Expression dialog illustrate this syntax in action.

The Expression portion of the dialog provides a means to define the most common expressions.

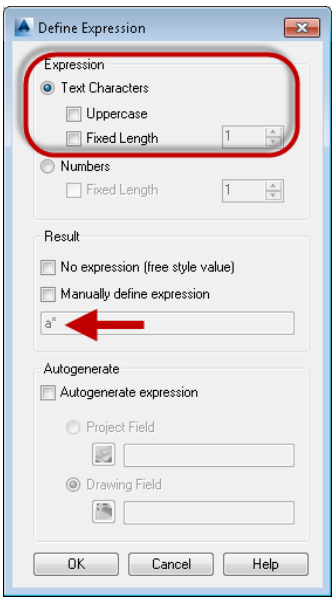
The Result portion of the dialog shows what the expression looks like as it is constructed, and also allows direct modification of the expression, or disabling of the expression.

Any alphabetic characters, any length

The Expression specifies Text Characters. It does not specify uppercase or a length. Thus the expression built in the Result box is: **a***

a --requires any upper or lower case character

* after the **a** --means any number of upper or lower case characters.

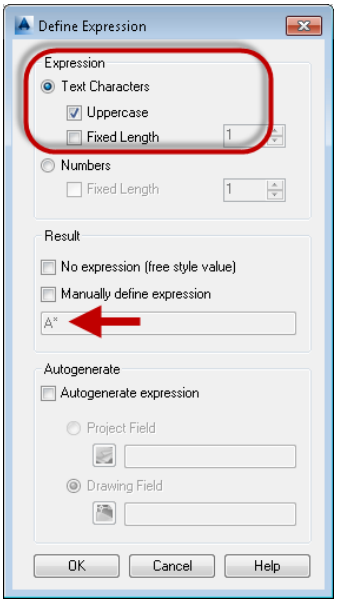


Upper case characters, any length

Here, checking the "Uppercase" box in the Expression area of the dialog has changed the Result to: **A***

A --requires an upper case character

* after the **A** --means any number of upper case characters.



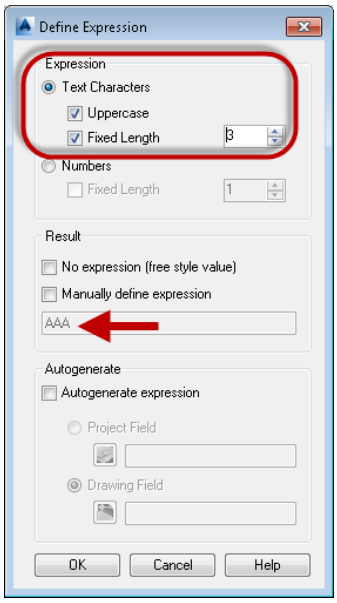
Upper case characters, fixed length

In this example, selecting the "Fixed Length" checkbox and incrementing the spin control to 3 has changed the result to: **AAA**

Each **A** requires an upper case character.

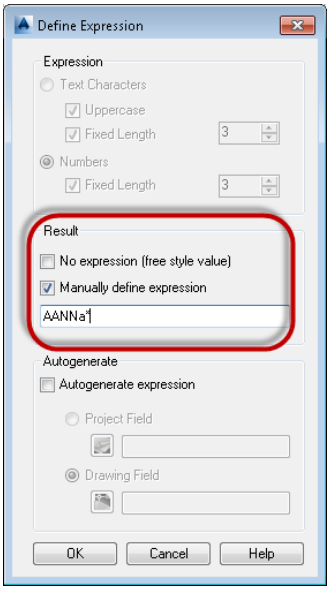
The asterisk (*****) is no longer shown.

This expression requires *exactly* 3 upper case character as input.



Manually Defined Expression

In this example, the “Manually Define Expression” checkbox is selected, and the edit box showing the expression is no longer disabled.



This option is required to construct more complex expressions involving combinations of characters and numerals.

The expression above, **AANNa***, would require the user to input:

Two upper case characters followed by Two digits followed by Any amount (including zero) of text characters.

The following would be acceptable input:

`AB10`
`AB10s`
`MM12qwE`

The following input would fail:

`ab10` -- initial characters should be upper case
`AB10s10` -- final characters must be alphabetic, this ends with digits
`AB5ss` -- only one digit, exactly two are required
`AB345xy` -- three digits, exactly two are required.

Using Expressions

The first three buttons, specifying the property source, are mutually exclusive. The fourth button, which raises the Define Expression dialog, may be used either alone, or in conjunction with one of the Property buttons.

When an expression is used in conjunction with one of the Property buttons, it acts to *enforce* a specific type of input for that Property.

If the expression from the third example above, requiring three uppercase alphabetic characters, were used in conjunction with the ValveCode property, the resulting subpart would appear as follows:

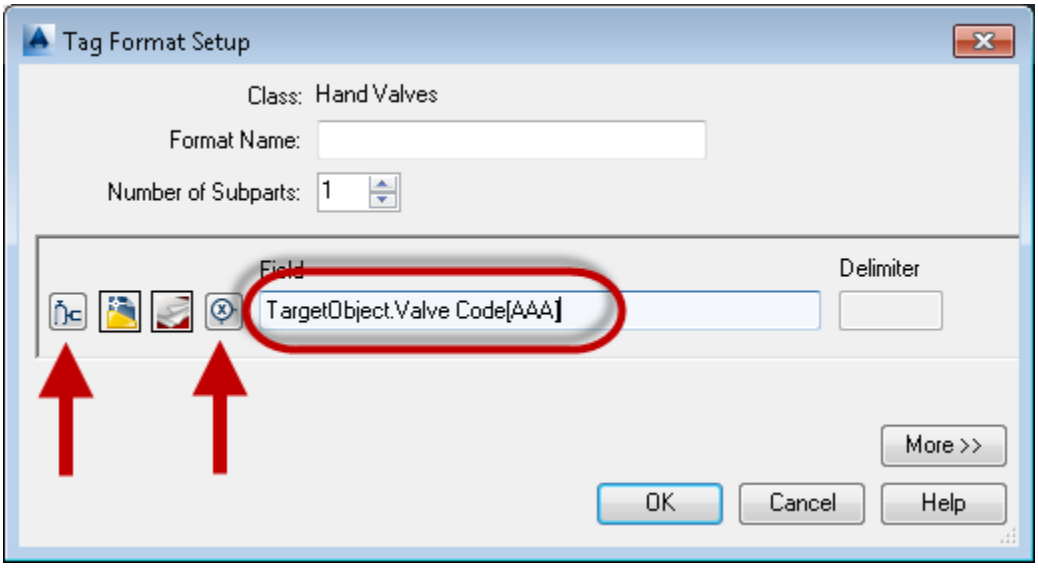


Figure 92: Tag with Expression

Notice that both the Class Property and Expression buttons show the raised graphics, indicating that both are in use. In addition, notice that the expression is being shown in square brackets after the property.

When an expression is used alone, it acts to require the user to input a pure alphabetic or numeric value which is not associated with any property.

If an expression requiring 3 numbers were used by itself, the resulting subpart would appear as shown below. Notice that the Expression button shows the raised graphics, indicating an expression is in use, but none of the Property buttons show the raised graphics. In addition, the Field shows ONLY the expression, NNN, and does not show any property. This is a “pure” expression.

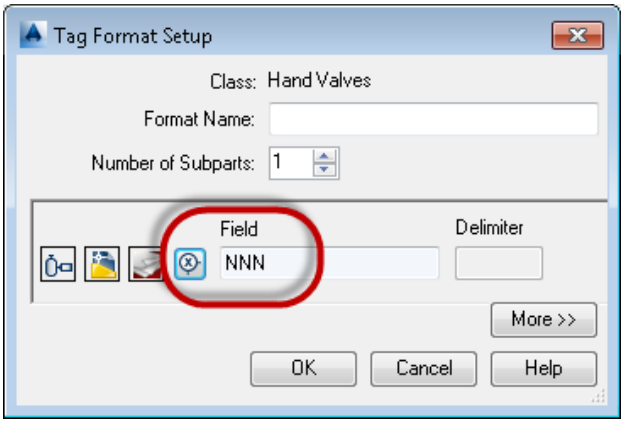


Figure 93: Tag with Input Value

Autogeneration

Some values can be Autogenerated. There are two approaches to Autogeneration.

Simple Autogeneration

Simple Autogeneration has the following requirements:

- Every subpart but one is a property
- Every property subpart has a known value
- The only non-property subpart is a pure numeric expression

If these requirements are met, it will calculate a numeric value that results in a unique Tag.

The Tag Format for Nozzles provides an example of a Tag Format that has been authored to support simple Autogeneration:

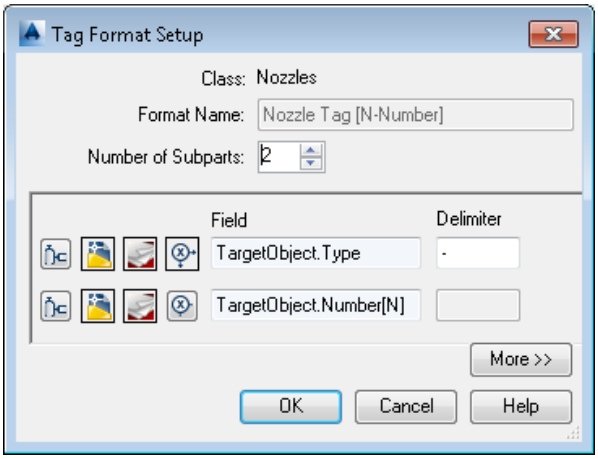


Figure 94: Nozzle Tag

Property based Autogeneration

Property based Autogenerated values can be per-Drawing or per-Project.

Example 1: Assume that the Tag for all pumps in a drawing must reflect which drawing contains the pump. For example, all pumps in drawing 1 should contain a numeric subpart starting from 100, all pumps in drawing 2 should contain a numeric subparts starting from 200, etc.

Example 2: Assume that all Tanks in an entire project should be numbered incrementally across the entire project.

Both of these examples can be supported with Property based Autogeneration. There are two fundamental requirements to use property-based autogeneration:

1) You must define and initialize the Autogen properties
2) You must indicate in the tag format that you are using autogeneration, and which property is the autogeneration basis

Autogeneration based on Drawing Properties

You can access the drawing Autogen properties by right clicking on the drawing in the Project Workspace, and selecting “Drawing Autogen Properties.”

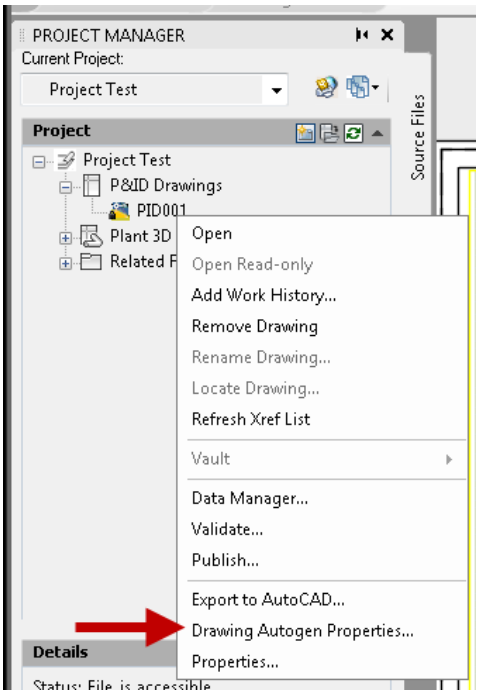


Figure 95: Drawing Autogen Properties

Selecting Drawing Autogen Properties will raise the Drawing Autogen Properties dialog.

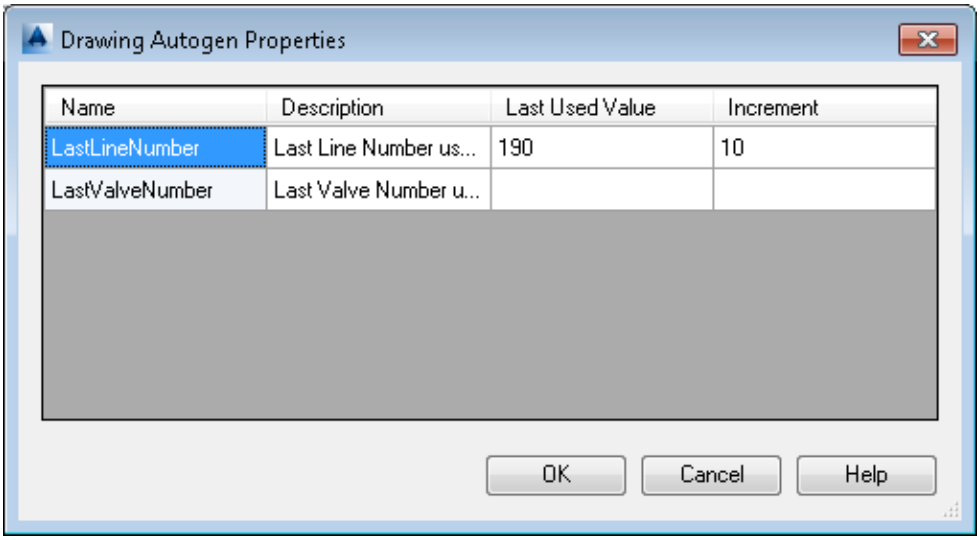


Figure 96: Select Autogen Property

In this example, the “Last Used Value” for Last Line Number is initialized to 190, and the Increment is initialized to 10. The autogeneration code looks at the “Last Used Value” and adds the increment

amount, so the *first* autogenerated value will be 200, *not* 190. Each time an autogenerated value is used, the program will update the value stored in "Last Used Value."

Note that you MUST enter values in "Last Used Value" and "Increment," otherwise the code cannot calculate an autogenerated value. In this example, any attempt to use the "Last Valve Number" will not succeed. For Drawing Autogeneration Properties, you must do this for EACH drawing.

To use these values, you must select the Expression button on the Tag Format Setup dialog, to raise the Define Expression dialog.

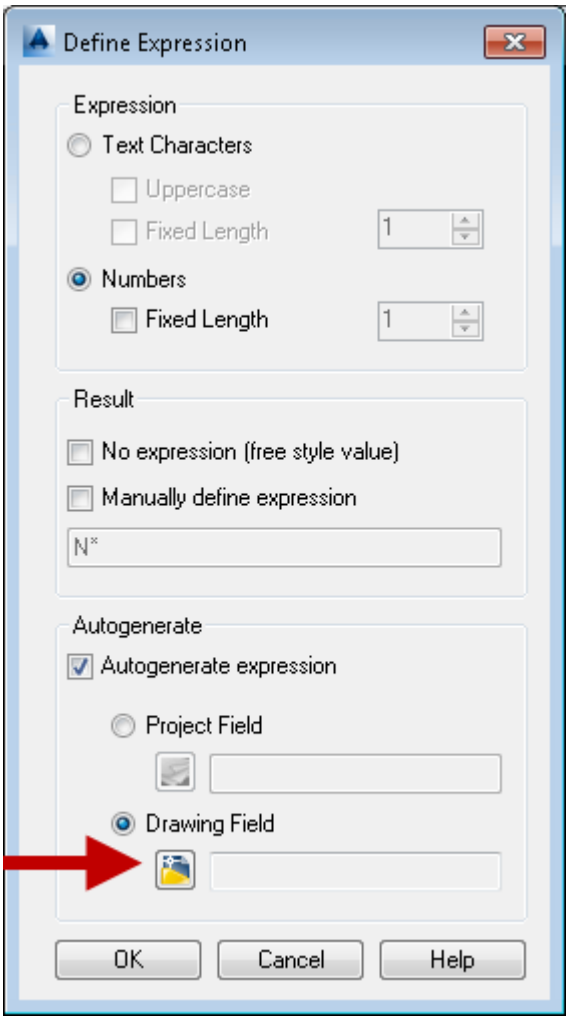


Figure 97: Autogen from Drawing Property

In this example, we intend to autogenerate a numeric value based on a Drawing Field. To do so, we select the Drawing Field radio button, and press the Drawing Field icon. Pressing this icon raises the "Select Drawing Autogen Property" dialog. This dialog presents all the defined Drawing Autogen Properties already defined, and allows you to select one. Here we can select "Last Line Number", which we initialized in the steps above.

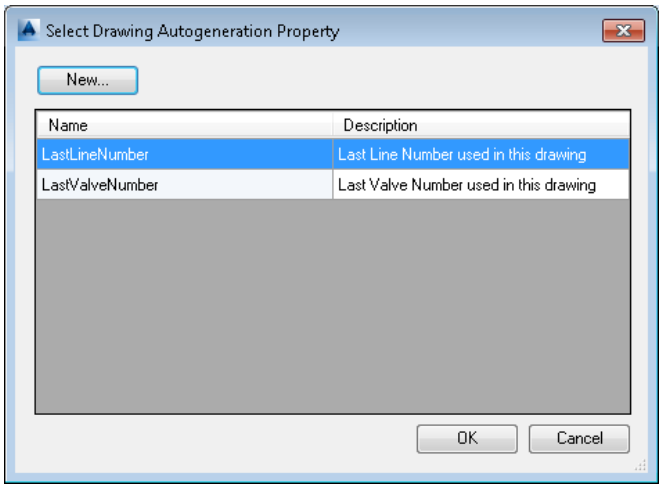


Figure 98: Select Line Number

In addition, new Drawing Autogen Properties can be added by pressing the “New…” button. This button allows the creation of a new Drawing Autogen Property, but it is necessary to return to the “Drawing Autogen Properties” dialog to assign their initial values.

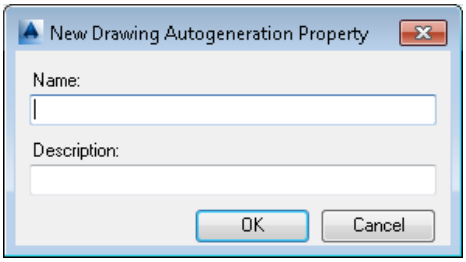


Figure 99: New Autogen Property

After selecting “Last Line Number”, the Define Expression dialog shows the selection.

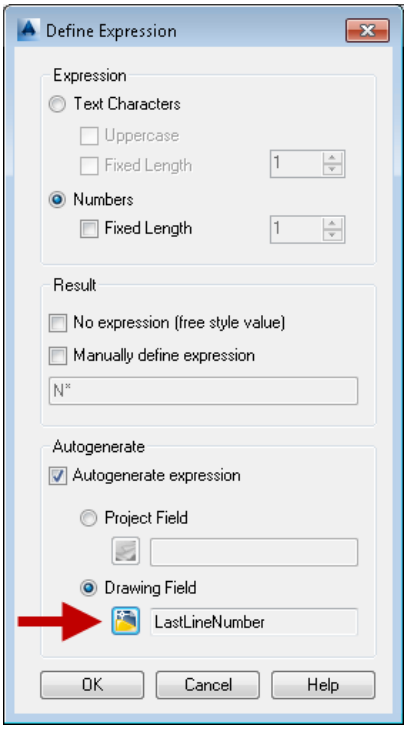


Figure 100: Picking Last Line Number

The default view of “Define Tag Format” dialog doesn’t provide much feedback to indicate that an autogen property is in use. It shows the expression N*, but it does not indicate that auto-generation is in effect or upon what value it is based. Pressing the “More>>” button does show the additional information.

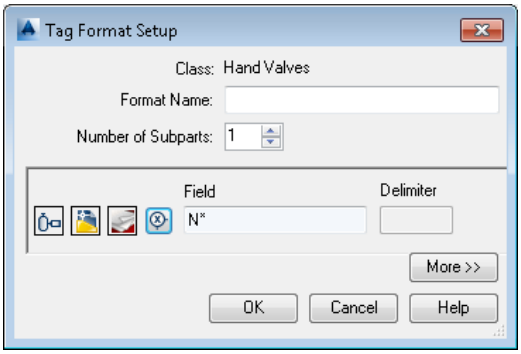


Figure 101: Default Tag Format View

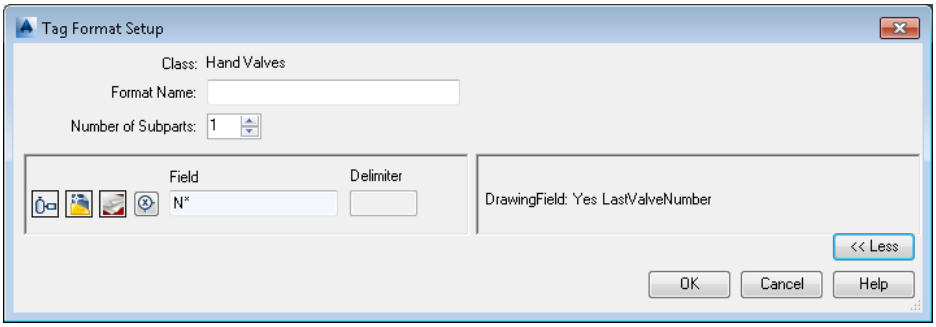


Figure 102: Expanded Tag Format View

Auto-generation based on Project Properties

Setting up Project Autogeneration Properties is similar to setting up Drawing Autogeneration Properties, except that there is no project-level UI giving direct access. Selecting a Project Autogen property starts with the "Define Expression Dialog" raised from the "Tag Format Setup" dialog, just as above, except that the "Project Field" radio button must be selected.

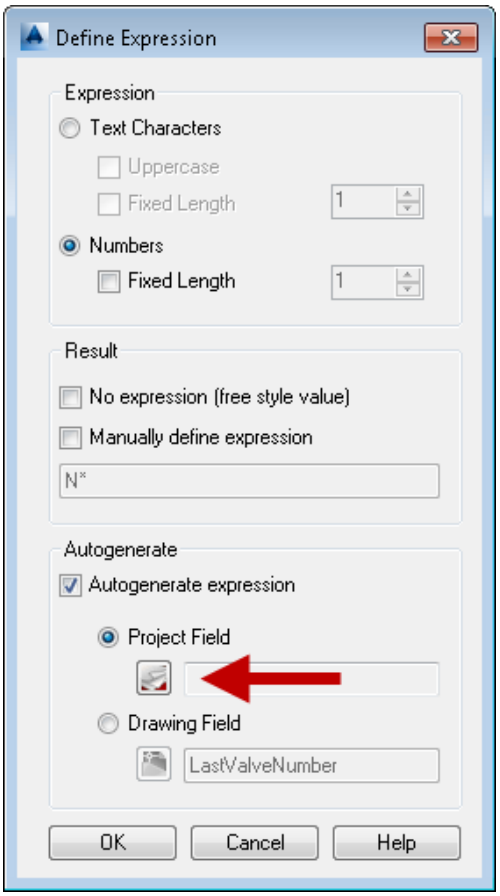


Figure 103: Project Autogen Property

Pressing the Project Field icon raises the "Select Project Autogeneration Property" dialog. This dialog differs slightly from the "Select Drawing Autogeneration Property" dialog in that the property AND the initial values are shown in the same dialog. You may also create new Project Autogeneration Properties from this dialog by pressing the "New…" button. Since Project Autogen properties exist once for the entire project, initialization is done from this centralized location, in contrast to the way per-drawing initialization for each drawing in the project workspace.

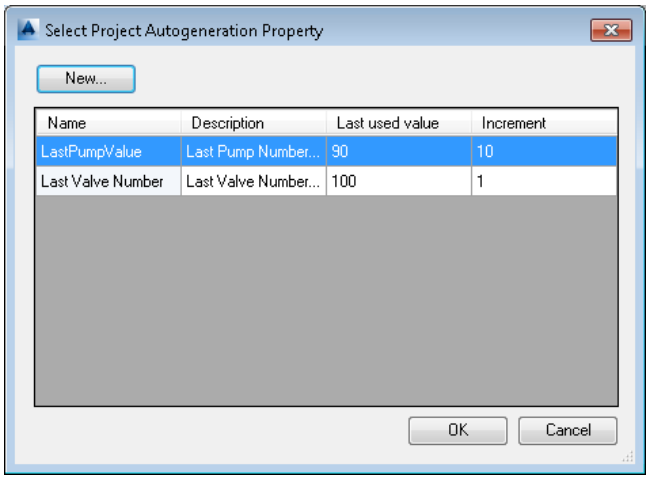


Figure 104: Last Pump Number

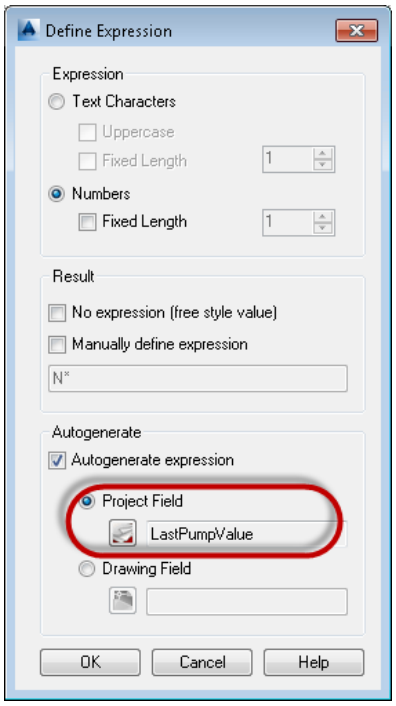


Figure 105: Last Pump Value Selected

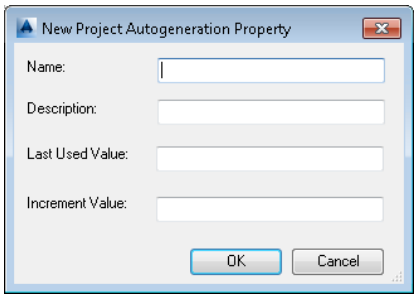


Figure 106: New Project Autogen Property

As with the Drawing Autogen properties, selecting the "Last Pump Value" autogeneration property results in that property being shown in the "Define Expression" dialog, but not in the "Tag Format Setup" dialog, unless you press the "More>>" button.

Delimiters

Each subpart of the "Tag Format Setup" dialog has space for a delimiter. Most commonly, a delimiter is a single character, such as a dash or colon, but there are no restrictions on the type or number of characters that may be used as a delimiter. The delimiter is placed in between each subpart when the tag is built up and stored.

It is also allowable to leave the delimiter blank, however there can be situations in which this is inadvisable. Several other features in AutoCAD P&ID may need to parse the completed tag value back down to its constituent parts, and it is possible to create Tag Formats where this is impossible. This will be covered further in the "Pitfalls and Workarounds" section.

Assigning a Tag Format

After a Tag Format has been set up, it must be selected as the Tag Format to use for a given P&ID class.

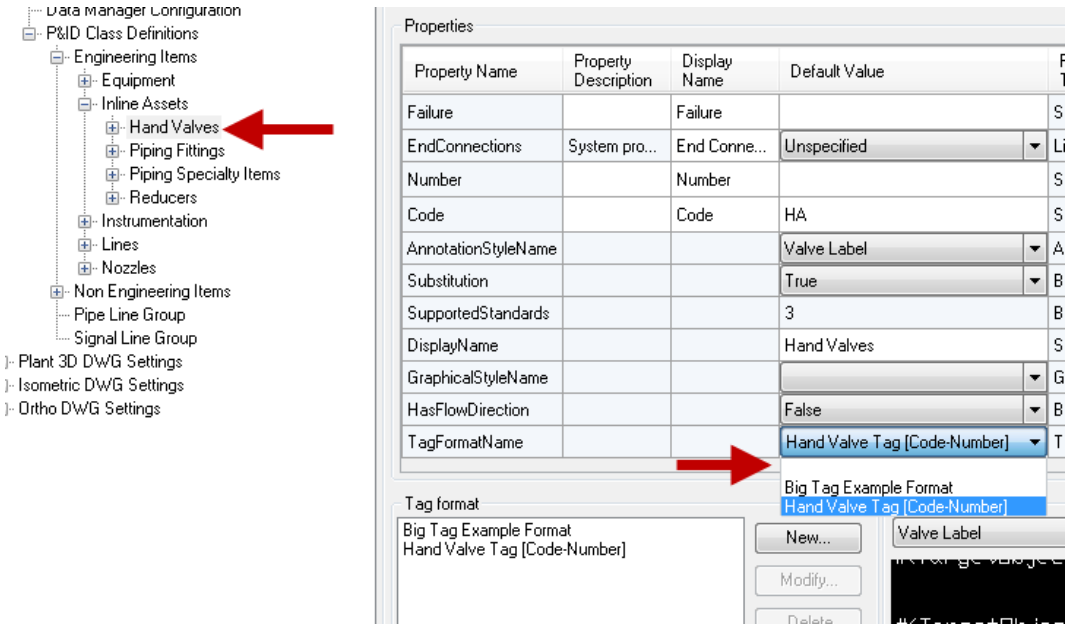


Figure 107: Assign the Tag Format

Here we see that two Tag Formats are now defined at the Hand Valves level, and that the default Tag Format for Hand Vales is “Hand Valve Tag [Code-Number]. Since “Big Example Tag Format” was defined at the Hand Valves level, it is available to be used by any Hand Valve. In the following example, the new Tag Format is assigned as the Tag Format for Ball Valves, only, by selecting it from the picklist for the TagFormatName property.

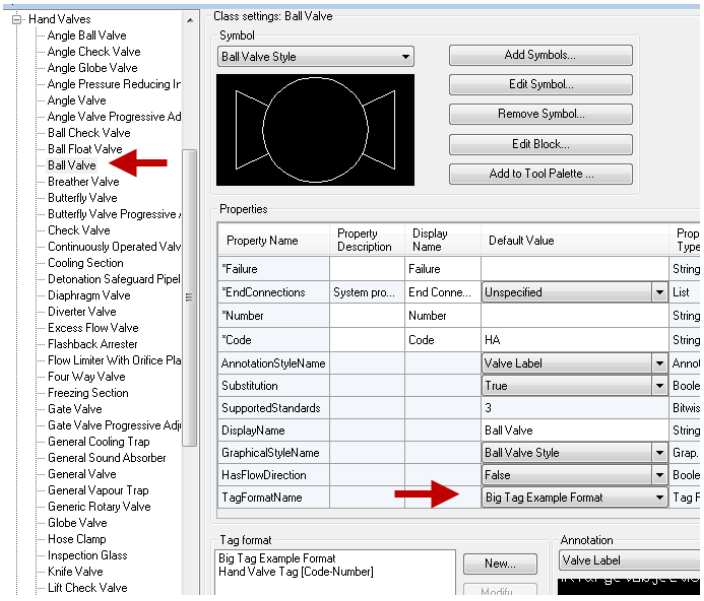


Figure 108: Set Class Tag Format

Line Tagging

One class that is somewhat different is line tagging. In order to decide properly how to setup tagging, you need to understand what is affected by the line tag. In P&ID the line segment tag matches what most people would use as an annotation (Size-Service-Spec-Number). The line group tag enforces a unique number for each line group (Line Number), but the line segments may have duplicate items. Some companies would use the line group to enforce a unique Service + Line Number which works as well. The issue comes when thinking about the AutoCAD Plant 3D line group tagging. In general, you should use the same line group tag format between the P&ID Line group, and the AutoCAD Plant 3D Line group. Then in the P&ID Line list palette, the objects are all still stored under the line group tag (Line Number) (but displayed by the Line Segment Tag).

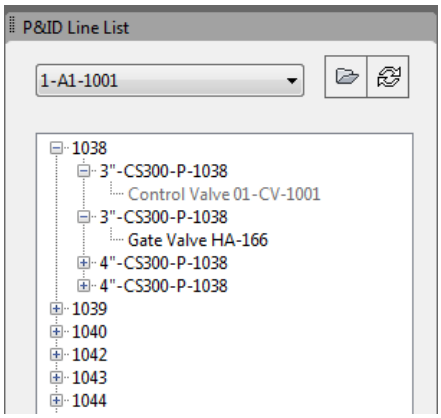


Figure 109: P&ID Line List Palette Tags

The other issue is how to get the line information on the iso. As shown below, AutoCAD Plant 3D pushes the P3D Line Group tag to the PIPE-LINE-REFERENCE property which displays the line number on the iso. To get the same line number as the Line Segment tag displayed on the P&ID, you will need to configure your isometric setup. See the link below for configuring complex line numbers on the isometric.

- http://www.pdoteam.com/2013/07/complex-line-numbers-in-isometrics/

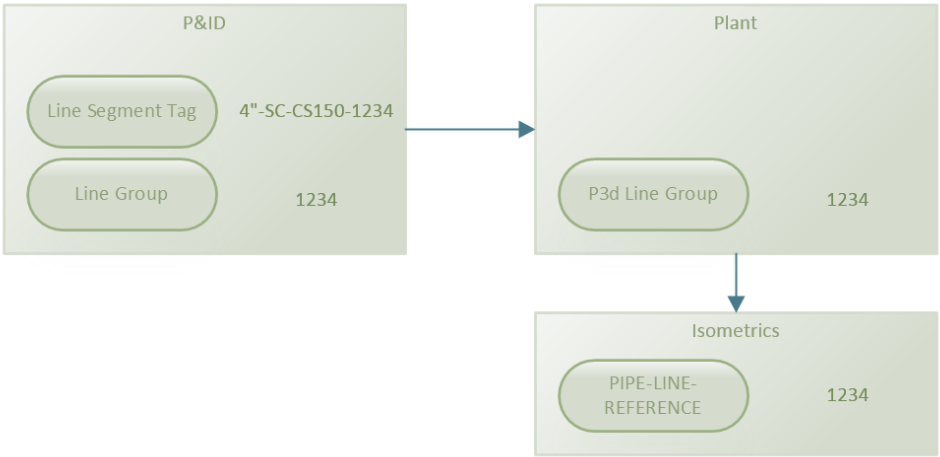
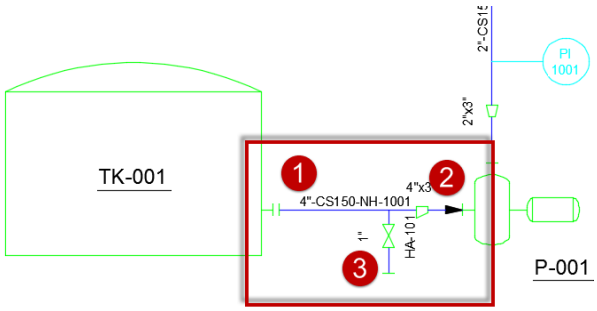


Figure 110: Line Tagging Flow

Line Groups vs. Line Segments

Line groups are used to organize segments into a cohesive, reportable item. Your line list report should be built using line groups so that you get one row per group (vs many segments for one group). In the example below, 4”-CS150-NH-1001 is a single line group that contains 3 segments (4”, 3” and 1”). By default the service and line number values are stored on the group, while the size and spec are stored on the segment.



Tagging Prompt Behaviors and the Graphical Style

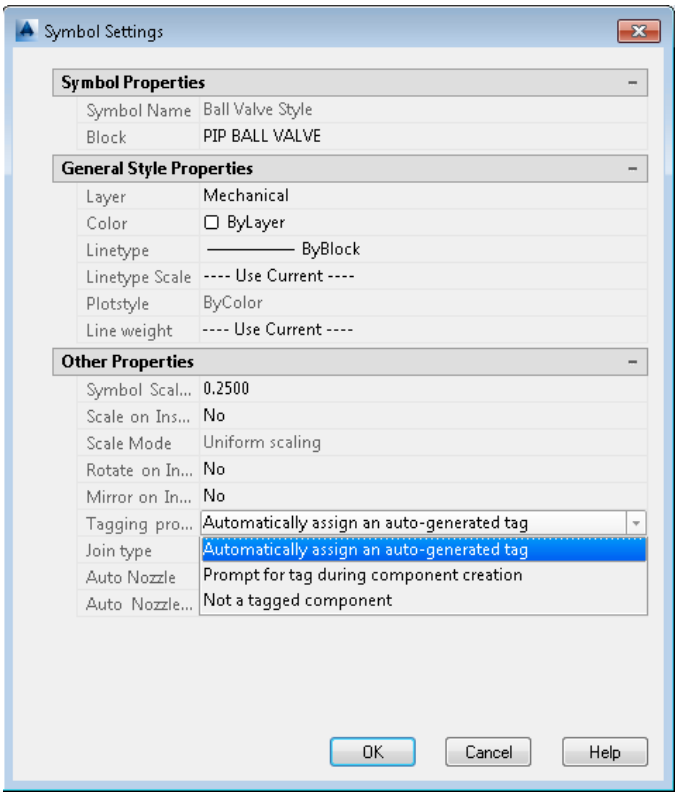


Figure 111: Tagging Behavior

Each P&ID Class Definition specifies one or more Graphical Styles. (These will be covered in more depth in the section on custom symbols.) In addition to the symbolic properties defined by the Graphical Style, several behavioral properties are also defined. One of these involves the prompting behavior for Tags for that specific P&ID class.

While most aspects of the Tag are defined using the Tag Format as described so far, if the Tagging Prompt setting in a P&ID class's Graphical Style is set to "Not a Tagged Component", then the "Assign Tag" option will not be available in the right click menu for assets of that type.

The value "Prompt for tag during component creation" will cause the "Assign Tag" dialog to be raised each time an asset of this type is placed in the drawing.

The value "Automatically assign an autogenerated tag" will not raise the "Assign Tag" dialog, but will attempt to fill in the Tag property by filling in all known and autogeneratable subparts. Depending upon how the Tag Format is defined, this may or may not produce a complete Tag.

How Tags are Stored

Once the Assign button on the Tag Assign dialog has been pressed, there are two types of storage which occur. For each sub part which refers to a class property, the individual value is assigned back to that property, as seen in the previous section. In addition, all the subparts and delimiters are concatenated together and stored as a single string in the Tag PROPERTY of the asset. This is shown in the data grid image below.

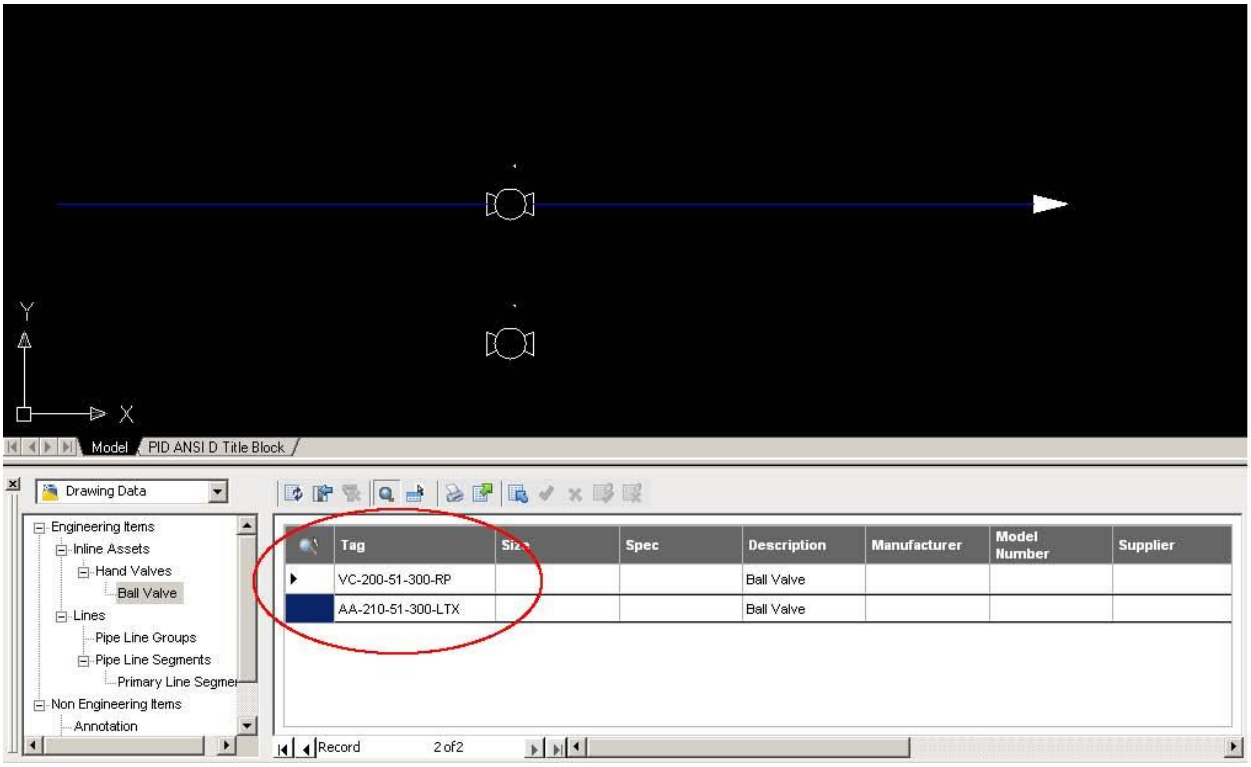


Figure 112: Tag Storage

Our second subpart was a pure expression, assigned 200 for the first Ball Valve and 210 for the second Ball Valve. Since these values are not based on any property, the ONLY place where they are stored is as part of this fully combined string, in the Tag property.

In order to keep the Tag property synchronized with the underlying properties, many activities will cause the Tag to be rebuilt from its constituent subparts. These include:

- Changing any class property of the tagged asset
- Changing any class property of a related item that is used by the tagged asset

Changes to Drawing and Project properties used in the Tag Format do not automatically rebuild the Tag. Drawing and Project values, once assigned, are retained in the completed tag, even if those properties are later changed at the Drawing or Project level. In other words, when the tag property is rebuilt from subparts, only changes to class-property subparts are incorporated.

Examples:

In this example, the second Ball Valve is modified by assigning GE as the manufacturer. Notice that the PaintCode subpart value changes to a '?'. Since this Ball Valve is not situated on a PipeLine, and the Tag Format specified PipeLine.PaintCode, this value is actually blank. In point of fact, it shouldn't have shown the LTX value even after the initial assignment, but the fully formed string from the Tag Assign dialog is always assigned at first.

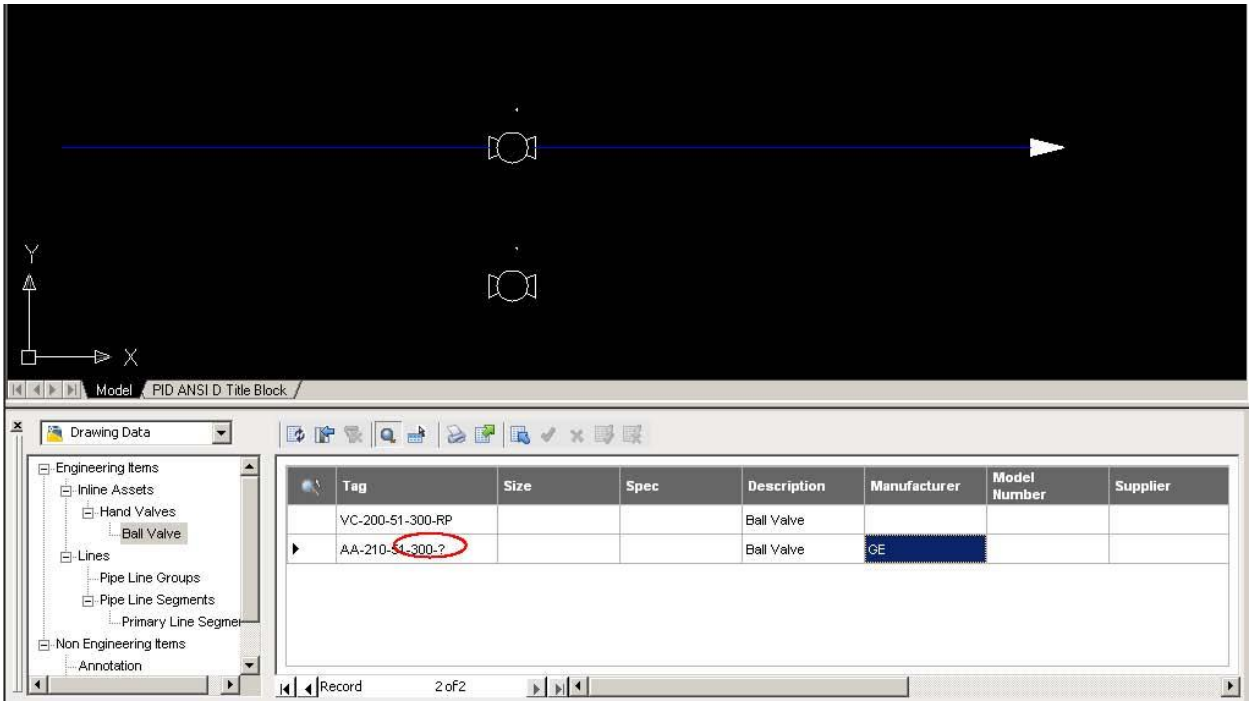


Figure 113: Tag Parts from Pipe Line

In the next example, the PaintCode property of the PipeLine upon which the first Ball Valve sits is changed to OIL. Notice that the Tag property of that Ball Valve has updated to reflect the change.

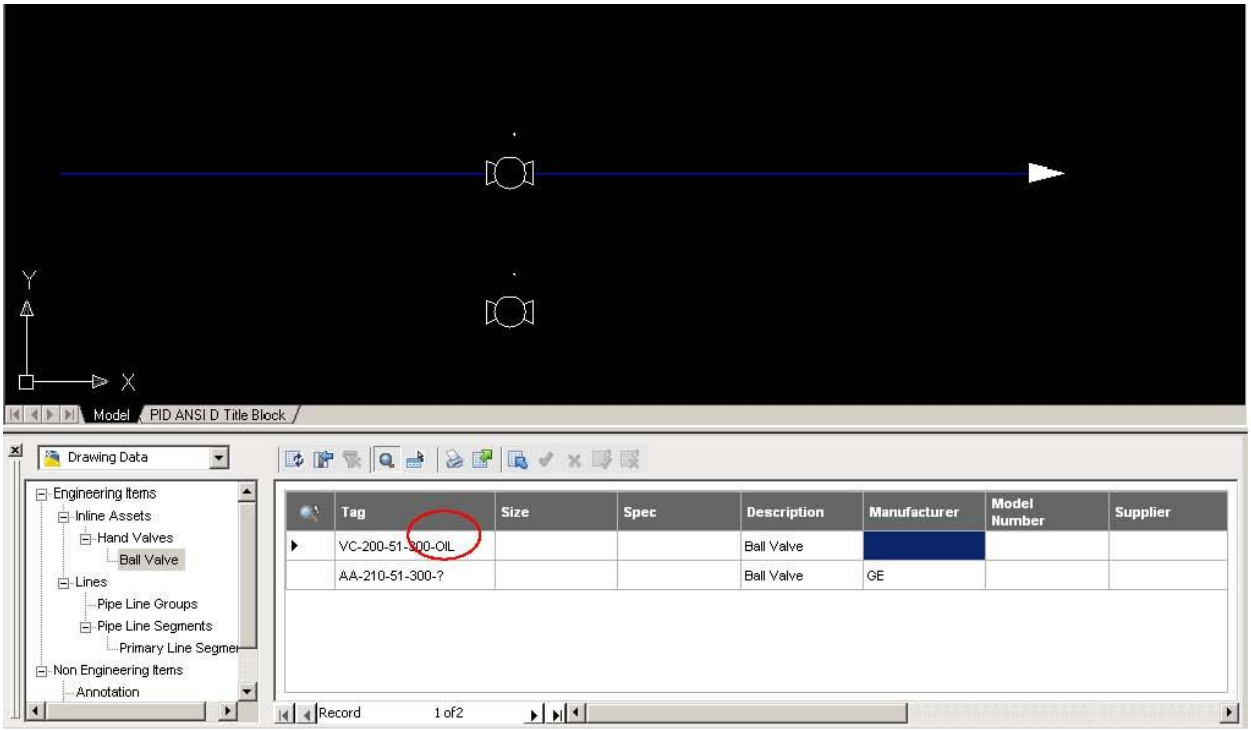


Figure 114: Paint Code Change

You may notice that the PaintCode property of the line doesn't show up in the BallValve's Tag immediately. An operation must occur that triggers the rebuilding of the Tag, such as changing the property in another cell. *(This delay will be addressed in the future.)*

Notes on Uniqueness

The tag property, in its entirety, is compared for uniqueness with all other tags in the project.

Duplicates are not allowed in the same drawing. Duplicates in different drawings of a single project are assumed to be different parts of the same item, for example, a line which spans drawings using an Off Page Connector could have the same tag in each drawing. It would be represented by a single row in the data grid.

Note that there is no allowance for items of different types to have the same tag. In other words, if a pump were tagged A-123, and a subsequent attempt was made to assign A-123 as the tag of a Tank, the Tank tag would be rejected as a duplicate.

For nozzles, the Tag value of the owning asset is considered along with the nozzle tag itself when determining uniqueness. In other words, two tanks, tagged TK-100 and TK-200 can each have a nozzle tagged N-1,and N-1 will not be treated as a duplicate. This is because the program considers all of TK-100N1 and TK-200N1 when comparing nozzle tag uniqueness.

Tag Behavior during Cloning

"Cloning" refers to any operation that places a copy of an AutoCAD entity into a drawing. Examples of cloning include the COPY, ARRAY, and MIRROR commands, Clipboard Copy/Paste (Ctrl-C/Ctrl-V), BLOCK, WBLOCK, INSERT and EXPLODE.

Since AutoCAD P&ID is a project-based product, copying a tagged asset within a single drawing or between drawings belonging to the same project would tend to create duplicate tags, which would be rejected.

To avoid this duplication and rejection, the product will blank out both the Tag property, and all the class property subparts which contribute to the tag. Blanking out the subparts is necessary because automatic tag regeneration would simply rebuild the duplicate tag if the subparts were left intact.

Avoiding Data Loss

There are a number of operations which will parse the tag back down into its constituent sub parts. One example is any operation that presents the sub parts to you for editing. For example, raising the Assign Tag dialog on an asset that is already tagged (for example to modify the Tag property) presents all the sub parts in each edit box. Another example is the operation which rebuilds the tag from its subparts whenever one of those subparts has been changed.

To do this, it has to parse the combined Tag property back down into its constituent parts. It has several ways of distinguishing between the subparts. One is to compare against the delimiters. Another is to re-retrieve the property from its original storage and compare the strings.

However, if pure expressions are placed without any delimiter, then it cannot determine where one sub part ended and the next begins.

If your Tag Format requires that there be subparts which are not delimited, you may need to add a new property to hold each subpart.

This is important as any refresh of the Tag could result in the loss of the pure expression portions of the Tag.

In the following example, the Tag Format is defined with a Property and a pure numeric expression, but no delimiter.

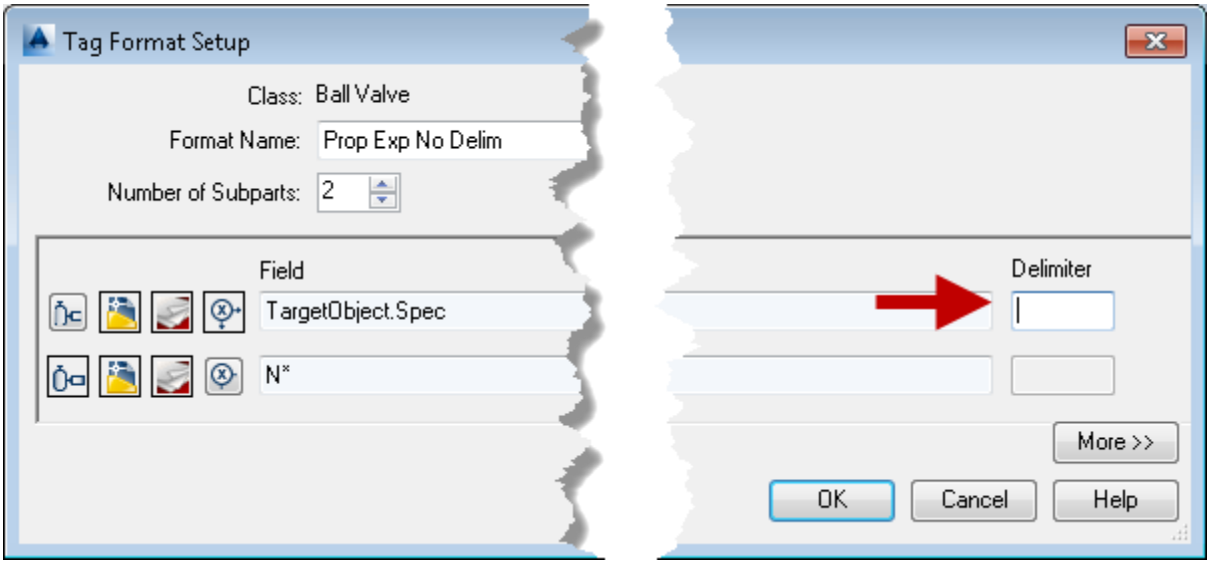


Figure 115: Tag with Expression, No Delimiter

Here, a Tag based on this Tag Format is being assigned. The concatenated, un-delimited Tag can be seen in the Data Grid. “CS150333”

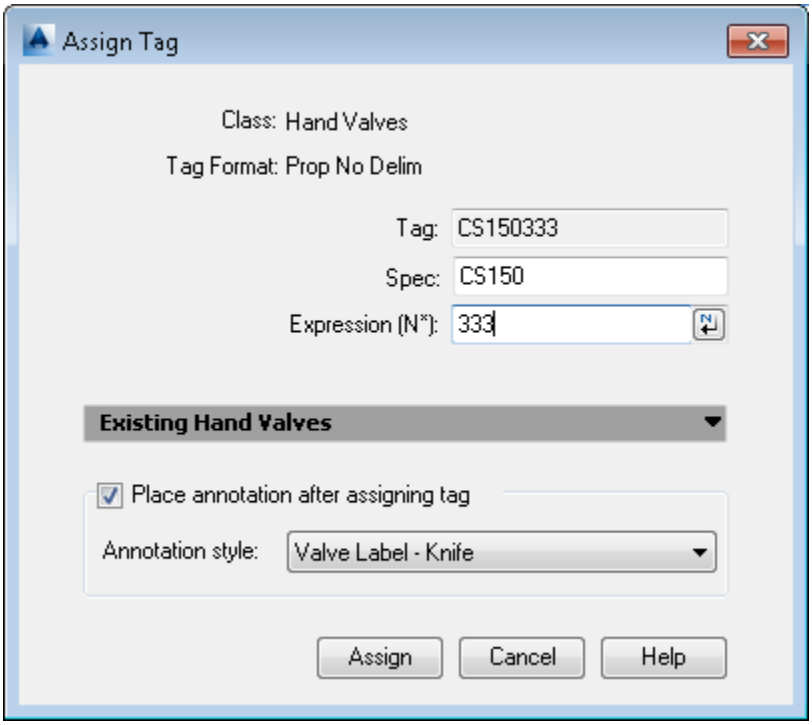


Figure 116: Input Tag Value

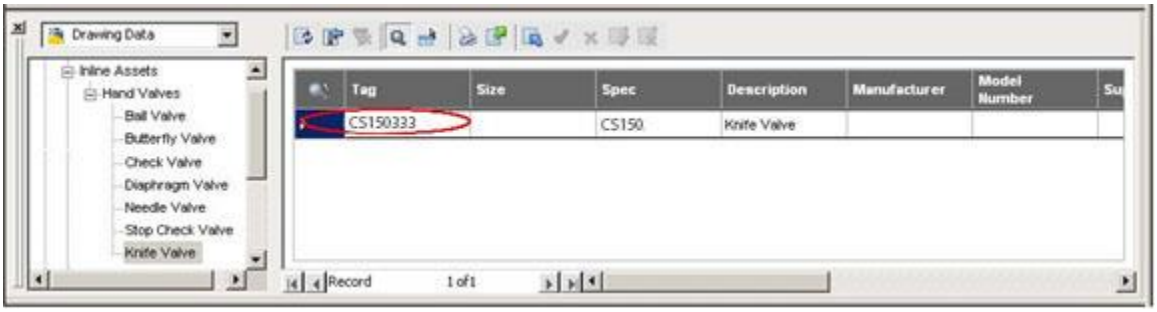


Figure 117: View Tag in Data Manager

Re-raising the Tag Assign dialog illustrates that the pure-expression subpart has not been successfully parsed.

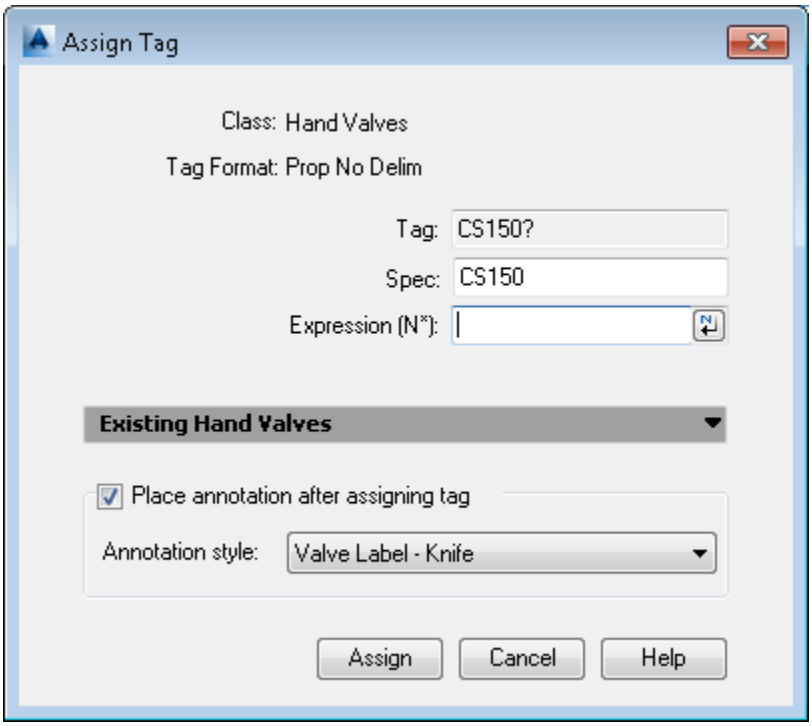


Figure 118: Lost Expression Value

In the example below, assigning a new property to one of the asset's fields has caused the Tag to be recalculated. In doing so, the pure expression portion of the Tag has been lost.

Figure 119: Data Manager Lost Expression Value

If you must have un-delimited subparts, then those subparts should always be based on a property. Adding properties in AutoCAD P&ID is straightforward, and can be accomplished from the Project Settings dialog.

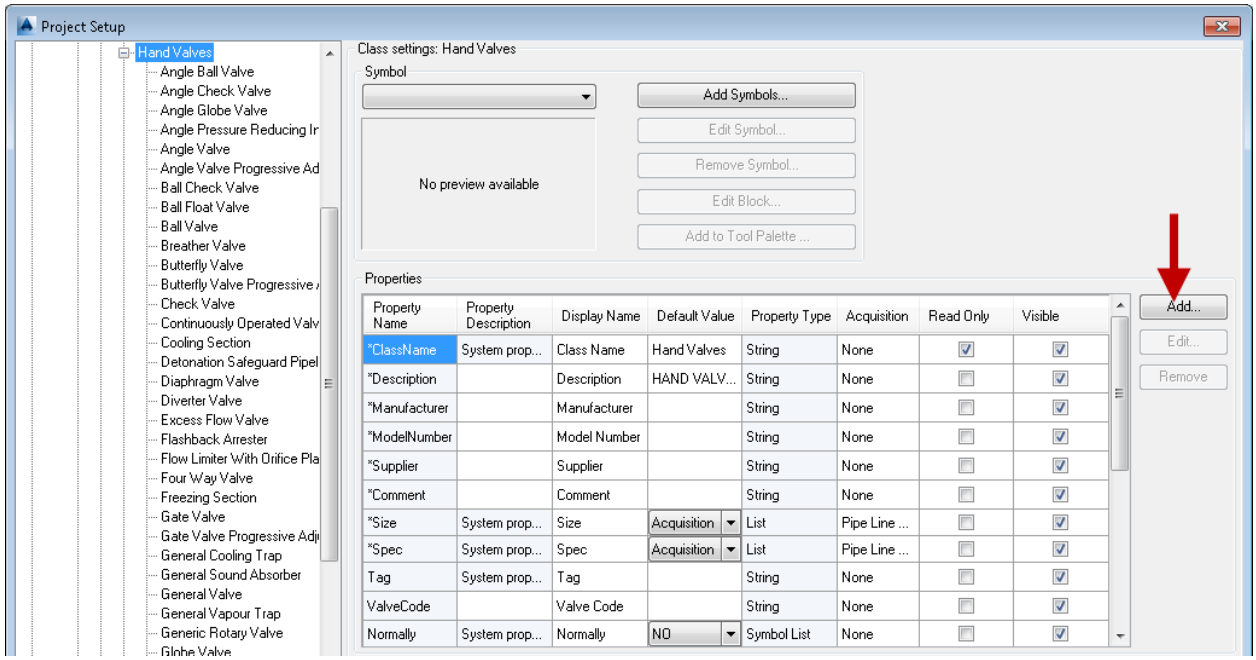


Figure 120: Add Property to Hold Expression Value

Avoiding Tag Conflicts

The tag registry tracks used tag values across all classes. Therefore two classes use separate but matching formats, an error will display similar to this.

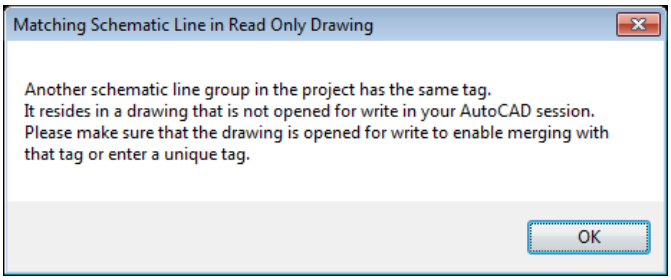


Figure 121: Class Tag Conflict

To avoid this error, make sure that your tag format uses unique values for each type of item. For example, using a tag format for valves based on only the Valve Number creates a conflict with Line Groups. You will only see the issue when one value exists and the same value is attempted to be used in the other class.

While tagging has to create values unique to classes, annotations may be modified to show only the unique property like TargetObject.Number instead of TargetObject.Tag.

Annotations

While the tagging feature prevents item duplication in the project, annotations allow us to display the tag (and any other properties) on a drawing. Remember when creating a new annotation, a new block is always created based on the selected annotation.

Annotations are implemented as regular AutoCAD Block References and Text entities, with some additional behaviors that allow the actual textual values shown to be dynamically evaluated based on properties of the item being annotated. We call the item being annotated the “target”.

Annotations utilize the same mechanism as the Tag Formats, with a few minor differences, to specify the property that should be shown.

Simple Annotations

Simple Annotations are created by dragging and dropping a Data Grid cell onto the drawing pane. These annotations are not based on any annotation style, are created using AutoCAD text entities, and the text properties (font, size, etc.) are based on the current AutoCAD settings for text.

Behind the scenes, a simple annotation Text entity carries the Format String “TargetObject.Property,” where “Property” is the property specified by the Data Grid column from which the cell was selected.

Compound Annotations

Compound Annotations provide a great deal more flexibility and customizability than Simple Annotations. Creating a new Annotation Style has similarities to both Tag Format creation and creation of a Symbol for a P&ID object.

Annotation Styles are defined at a given class level in the P&ID Class Definition hierarchy. Once defined, that style can be used to annotate any target P&ID entity defined at or below that node in the hierarchy.

Annotation Styles are also associated with an AutoCAD Block Definition, which provides for their appearance and behavior.

Annotation Styles are manipulated from the Project Settings dialog. Pressing “Edit Annotation” raises the Symbol Settings dialog for Annotation Styles.

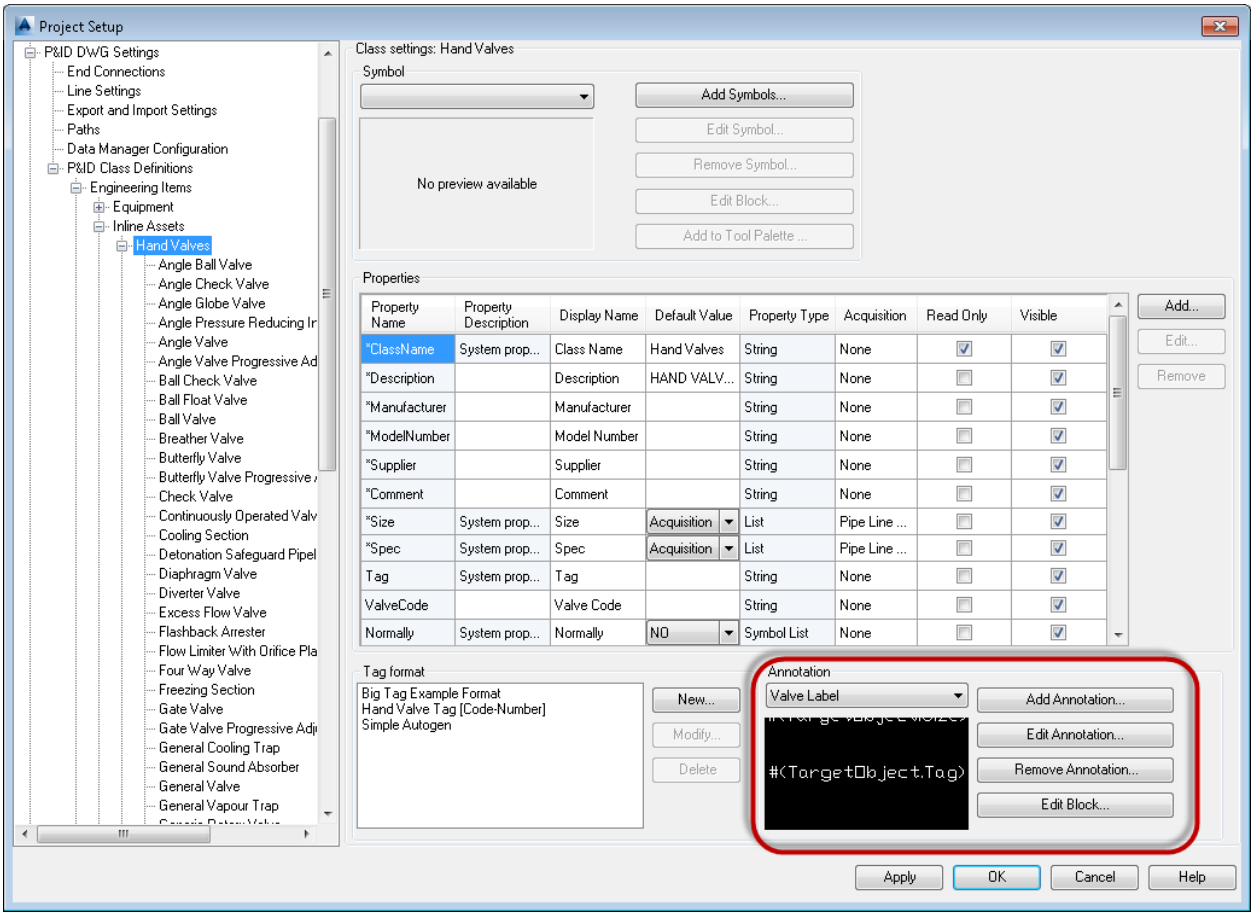


Figure 122: Annotations

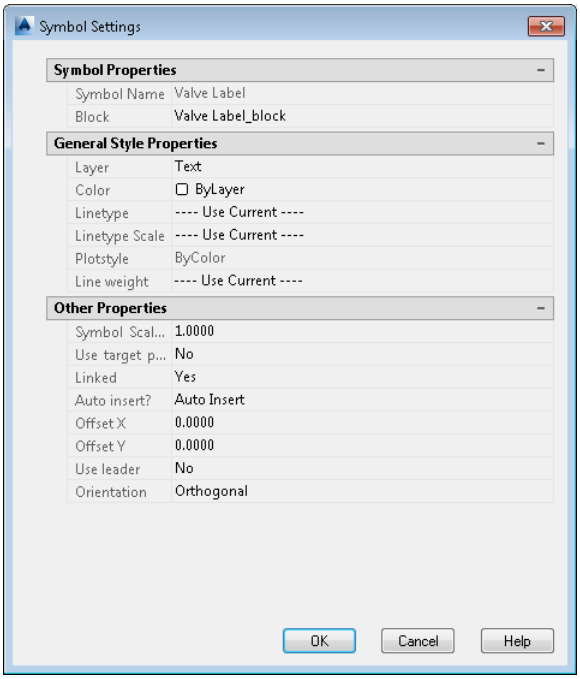


Figure 123: Annotation Settings

"Symbol Name" is the name of the Annotation Style. This is the name that appears in the submenu when the right-click/Annotate… context menu item is selected.

"Block" is the name of the block definition upon which this Annotation Style is based. The name is always the Symbol Name post-pended with "_block." To select a different block definition, select the Block edit box, and press the more button.

Figure 124: Annotation Select Block

Doing so will raise a three-step navigation dialog allowing you to choose a drawing, and a block within that drawing, and alter settings for the annotation to use for the Annotation Style. Once you have selected the block definition, the contents of that block definition will be cloned into your ProjSymbolStyle.dwg and renamed to SymbolName_block.

AutoCAD P&ID ships with a file called AnnotationTemplates.dwg. This drawing file contains a number of block definitions that may be useful as templates for a variety of standard annotation styles, and for which the more complex settings (the dynamic sizing settings) are already in place.

The “General Style Properties” are the standard AutoCAD entity properties, such as color, layer, and linetype that will be used when the annotation (an AutoCAD block reference) is created and inserted into the drawing.

The “Other Properties” area of the dialog offer settings that are specific to Annotations.

- “Symbol Scale” is a scale factor at which the annotation will be inserted into the drawing.
- “Use Target Properties” will cause the annotation to be inserted with the same entity properties (color, layer, linetype) as the target entity, rather than using the settings from “General Style Properties.”
- “Linked” controls whether the annotation moves along with the target entity.
- “Auto Insert?” controls whether the annotation will be automatically inserted when the target entity is inserted into the drawing.

“Auto Insert?” has three possible states, worthy of further discussion.

“No Auto Insert” is self-explanatory. Annotations based on this style will never be automatically inserted into the drawing.

The other two settings are primarily meaningful for the default Annotation Style for a given P&ID class, and take effect when placing an asset of that class into the drawing from the tool palette.

“Auto Insert With Prompt” --If an asset is placed in the drawing whose default Annotation Style specifies Auto Insert With Prompt, the user will be automatically prompted to select a position for the annotation.

“Auto Insert” --If an asset is placed in the drawing whose default Annotation Style specified Auto Insert, then the annotation will automatically be placed in the drawing at the positions specified by “Offset X” and “Offset Y” from the asset’s insertion point.

The Offset X and Offset Y settings are only enabled when the Auto Insert state is Auto Insert.

Note that the Auto Insert option, relying on the Offset X and Offset Y settings, does not work when the target is a pipe or signal line. The reason is that the software cannot determine a single insertion point from which to calculate the offset from a line, as it can do from an asset.

“Auto Insert” using an offset will also have an effect when placing an annotation via the right click “Annotate” menu. Normally, when using the Annotate… context menu, you would be prompted to select a position using the mouse. However, when annotating an asset using an annotation style which specifies “Auto Insert” with an X,Y offset, the annotation will be placed at that offset.

Use leader indicates whether a leader will be included with the annotation.

Orientation changes how the annotation is rotated.

Creating a New Annotation Style

In this example, a new Annotation Style at the Hand Valves level will be created by pressing the “Add Annotation” button, and then pressing the more button beside the Block property, as described above. In the navigation dialog, select “AnnotationTemplates.dwg” and press “Open.”

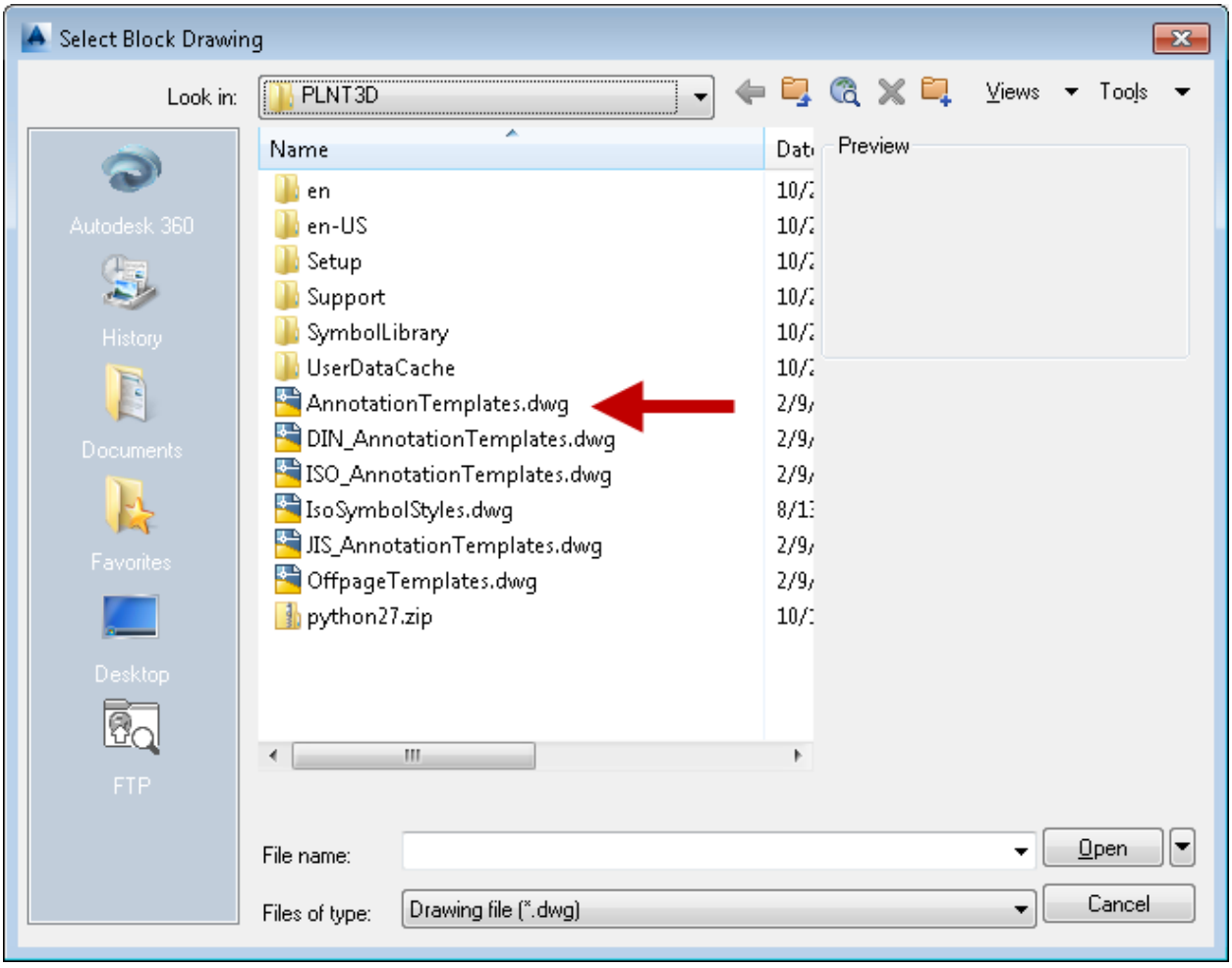


Figure 125: Annotation Templates

In the “Select Block” dialog, select “InfoTag2LineBox” as the block definition upon which we’ll base the new Annotation Style.

Note: See the section ‘Linear Parameters in Annotation Block Definitions’ for subtle improvements to the as-shipped “InfoTag2LineBox” block definition that you will need to make to get the best possible results when this annotation is placed.

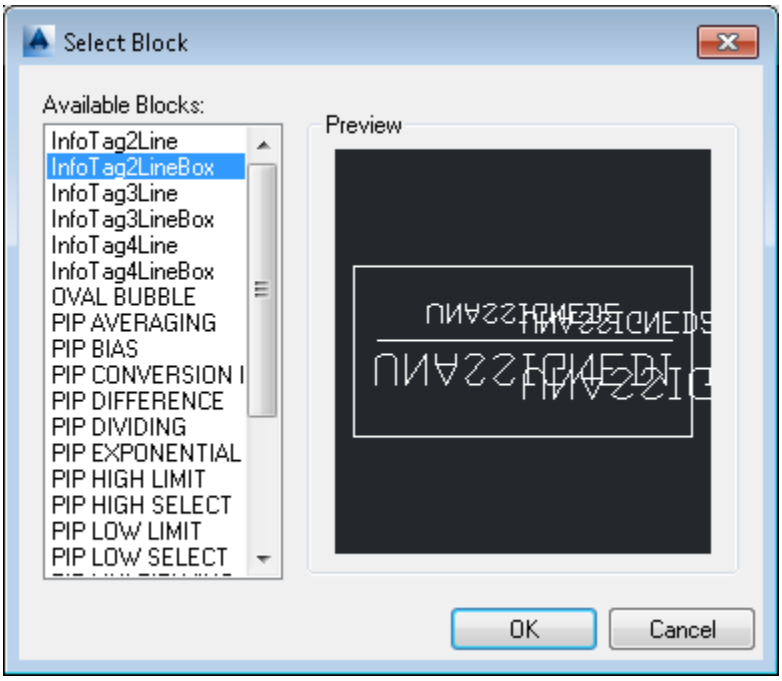


Figure 126: Block Editor Select Blocks

Here, the annotation is given the name “Valve InfoTag” and the desired settings are made to the Style properties.

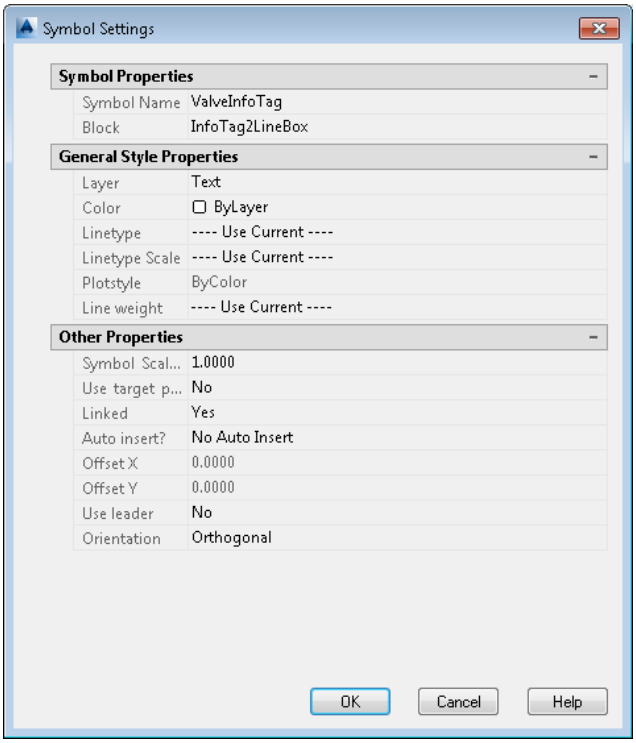


Figure 127: Annotation Properties

Press OK. This will copy the block definition into ProjectSymbolStyle.dwg and rename it to “ValveInfoTag_block”.

Every Annotation Style points to a unique block definition. Graphical Styles (“Symbols”) may share block definitions, but Annotation Styles do not, because the Attribute Definitions for each Annotation Style need to be set up so as to be unique to that Annotation Style.

Assigning Format Strings to Attribute Definitions

Pressing the “Edit Block” button allows editing of the Block Definition upon which the Annotation Style is based. To set up for this section, select our new Annotation Style, “Valve InfoTag” from the picklist and press “Edit Block…”

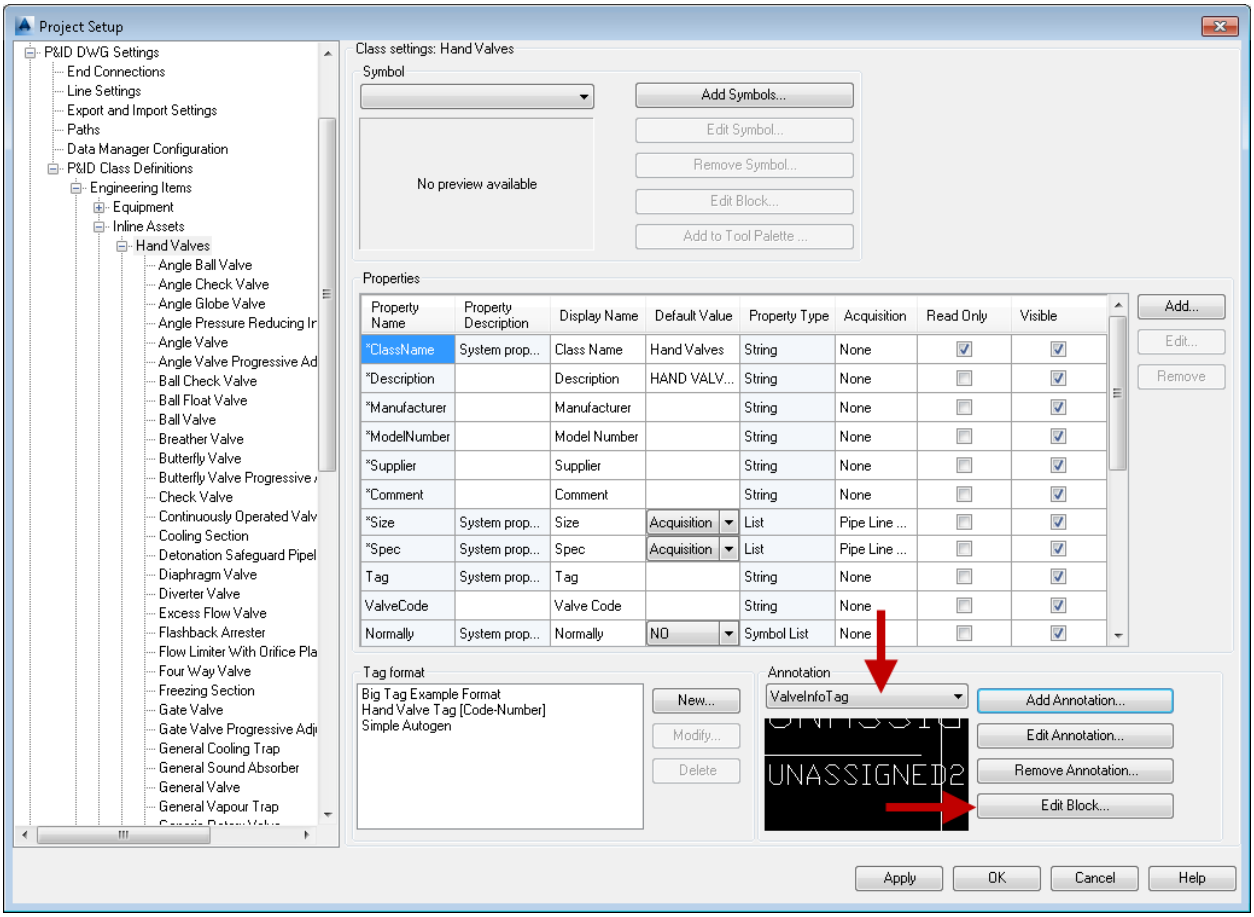


Figure 128: Edit Annotation Block

This operation will drop you into the AutoCAD block editor for the Annotation Style's block definition.

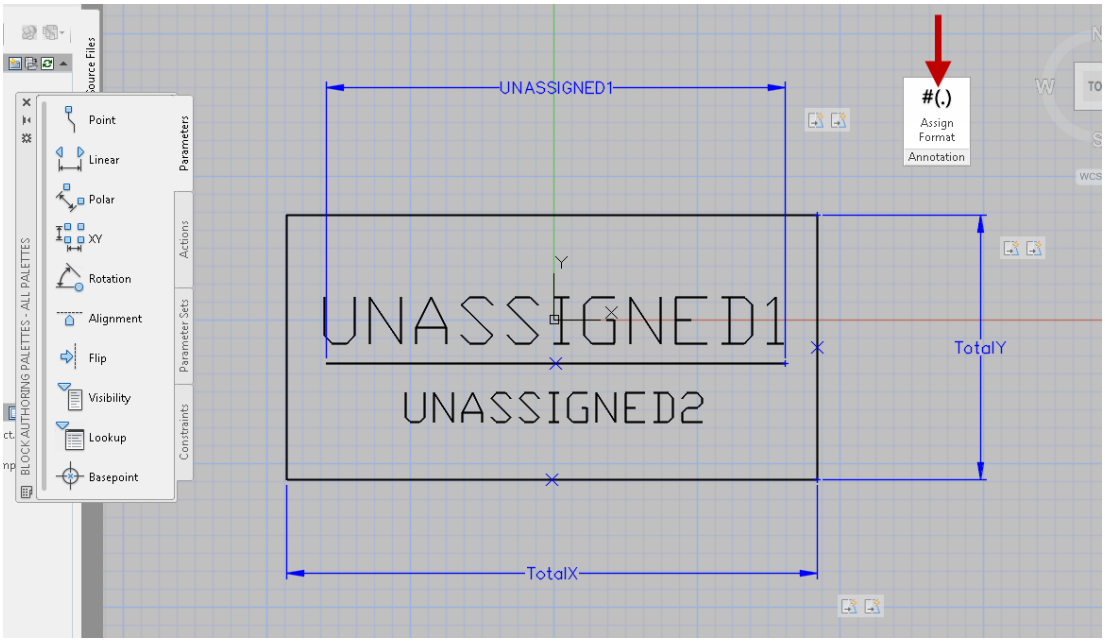


Figure 129: Assign Format

This template contains two Attribute Definitions, named UNASSIGNED1 and UNASSIGNED2. It also contains some geometry; an underline beneath UNASSIGNED1, and a box around both Attribute Definitions. Finally, it contains 3 linear parameters, each associated with a pair of stretch actions. We'll discuss the geometry and parameters in the next section. The block may have several duplicates directly on top of each other (4 duplicates should be erased for the line, each of the attributes and the rectangle).

To fix an action, right click on the action icon, select Action Selection Set, and then Modify Selection Set.

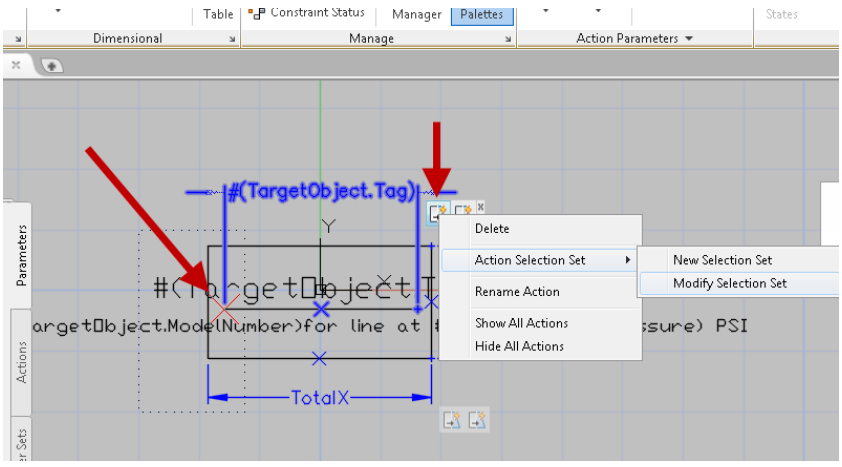


Figure 130: Modify Selection Set

Notice that the action as a dotted window associated with it. Re-create the stretch window, and make sure the object that needs to stretch is selected.

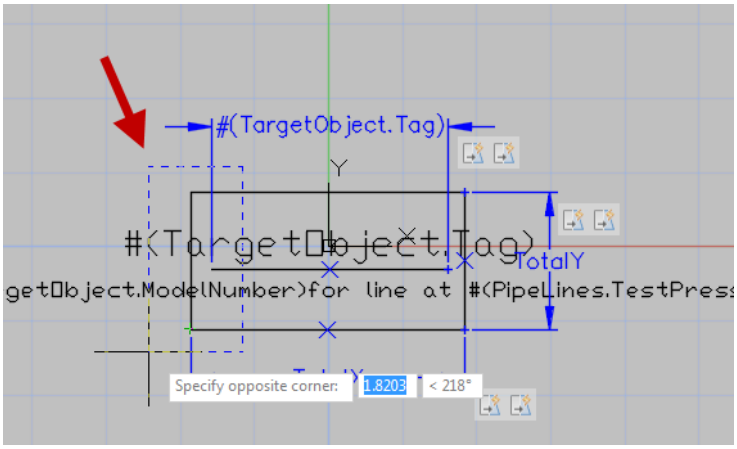


Figure 131: Selection Window

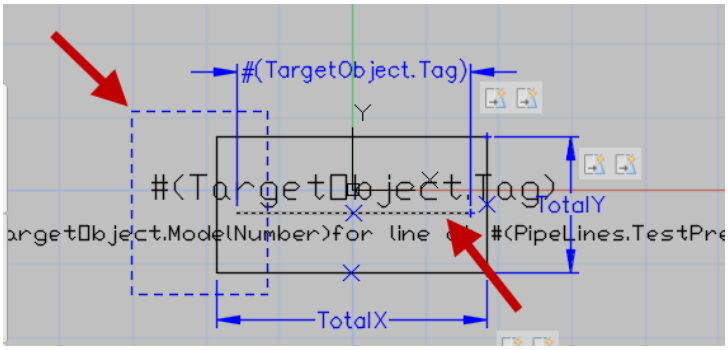


Figure 132: Selection Window with UnderLine

Repeat the process for all of the actions.

These Attribute Definitions currently are not set up to display any particular properties of an Annotation's target entity. Pressing the icon in the small toolbar will prompt you to select an Attribute Definition. Making that selection will then raise the Assign Annotation Format dialog. This dialog is extremely similar to the Assign Tag Format dialog, and lets you set up the Attribute Definition to display properties of the target entity, properties of the Drawing, or properties of the Project.

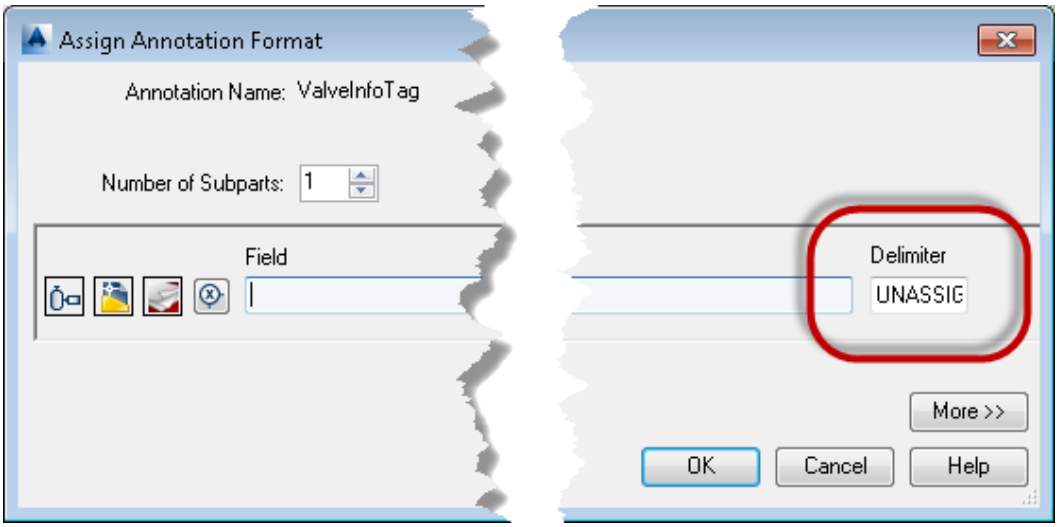


Figure 133: Clear Delimiter

When this dialog is first raised for an UNASSIGNED Attribute Definition, the word "UNASSIGNED1" appears as the first delimiter. In fact, for Annotation Formats, the area between subparts needn't be thought of as a delimiter at all. Any kind of free text is allowable. Remember to remove or change this delimiter value if you don't want any additional text appearing when the annotation is placed in the drawing.

Using the same skills discussed in defining a Tag Format, we'll assign the target object's Tag property to the first Attribute Definition, and remove the word UNASSIGNED1 from the delimiter field.

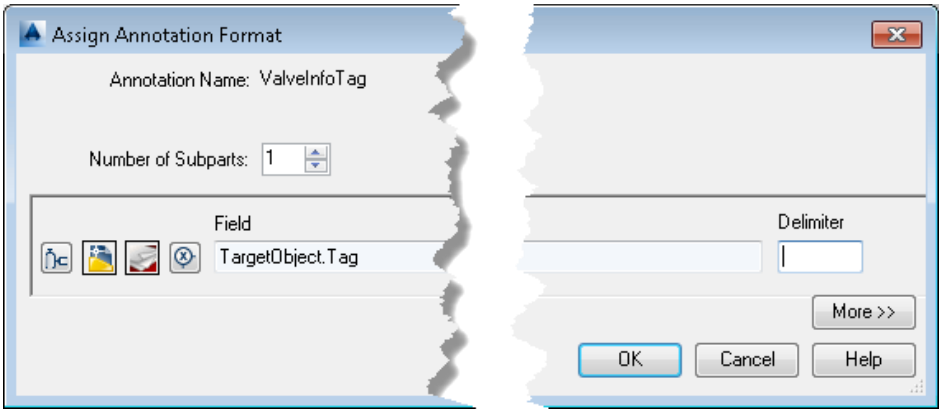


Figure 134: Assign Annotation Property

After pressing OK, the value of the UNASSIGNED1 Attribute Definition has been changed to #(TargetObject.Tag). Notice that the linear parameter named UNASSIGNED1 has been edited in the

same way. We'll discuss why this is important in the next section. *You may have to execute BSAVE in order to see the linear parameter label change. This is an AutoCAD artifact.*

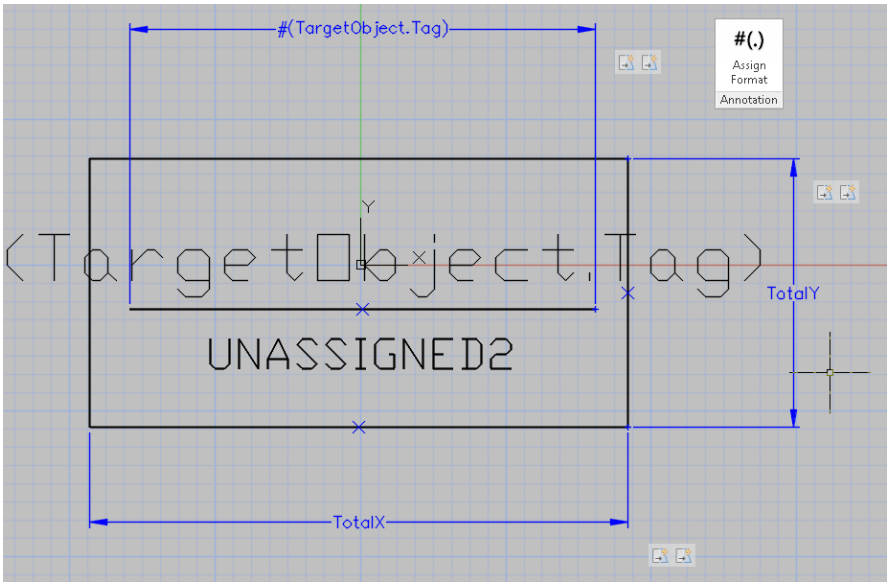


Figure 135: Modified Annotation Attribute

Each Attribute Definition can be used to place a single line of text (you can use multi-line attributes too), but each line of text may present more than one property. We'll define the second subpart to show two properties, one from the Valve, and one from the Line upon which the Valve sits. The following properties and delimiters yield the format string as shown below.

Just as with Tag Formats, it is allowable to specify a property from another P&ID class, so long as that class is directly related to the target object's class. The second subpart illustrates using a property from a related class.

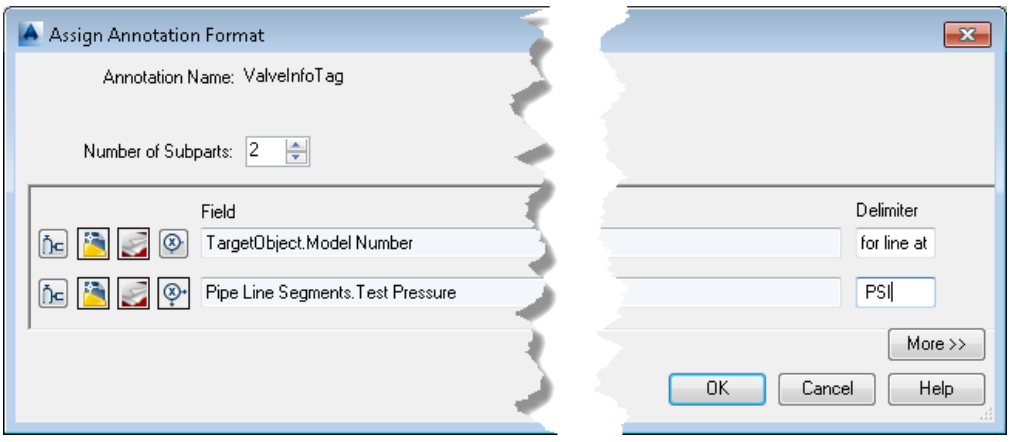


Figure 136: Annotation with Related Property

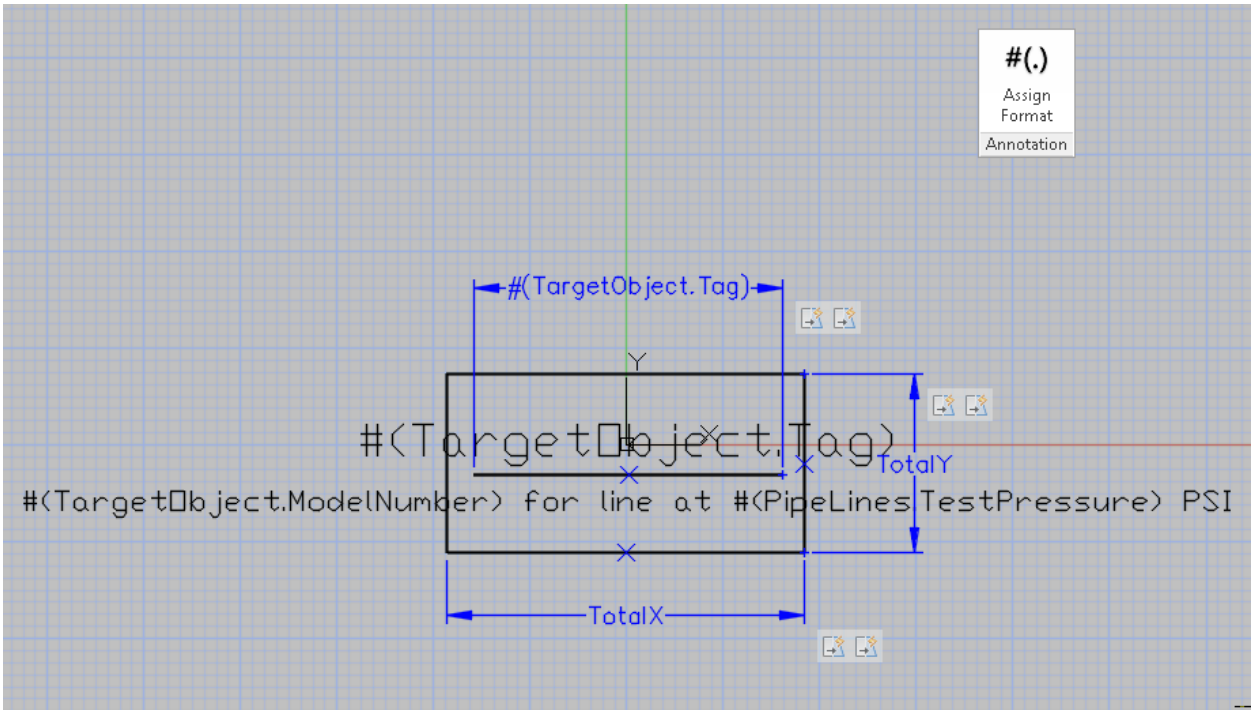


Figure 137: Updated Annotation Attribute

Notice that the properties to be retrieved are enclosed inside the field identifiers #(), while the delimiters appear as free text in between the fields.

It is also possible to have leading text. You can add leading text by one of two methods. Start off with an additional Field, and set the delimiter on the first blank item. The rest of the fields get filled out normally. You can also select the Attribute Definition in the AutoCAD properties palette, and prefix the free text you want. In this example, the free text “Model No.” is prefixed.

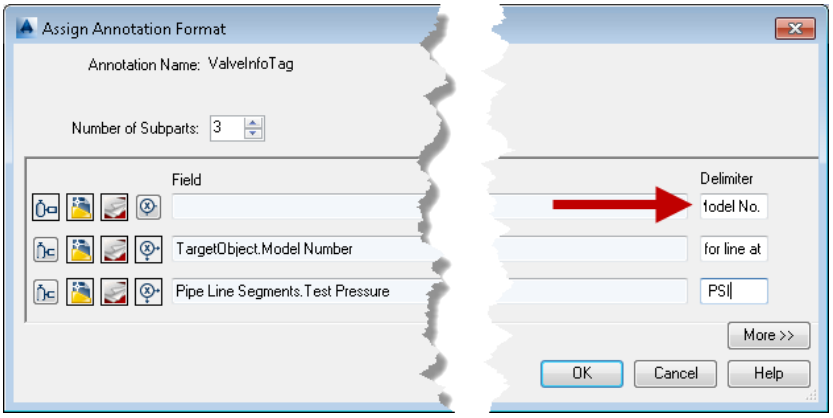


Figure 138: Leading Text in Annotation

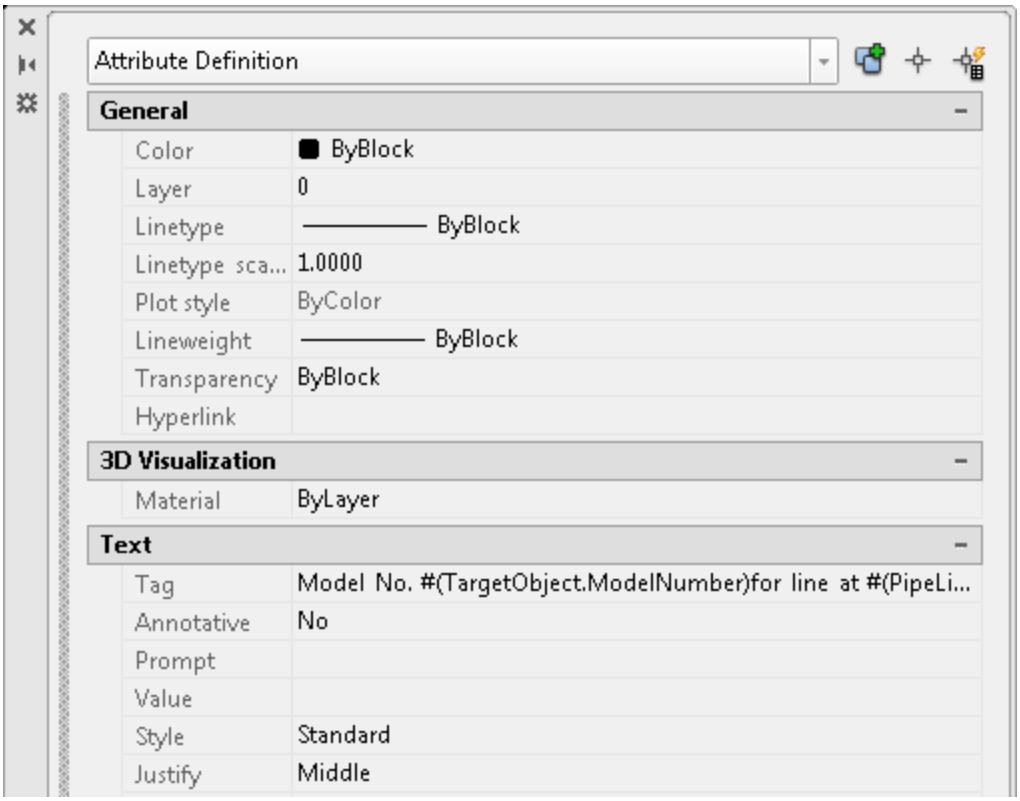


Figure 139: Attribute Properties

Close and save the changes to the block, and click OK to close the Project Settings dialog.

Underlining in Annotations

While the default annotations used dynamic properties to underline, you can use the previous technique to underline values as well. For the prefix, add %%U to the delimiter, and then add your properties as normal.

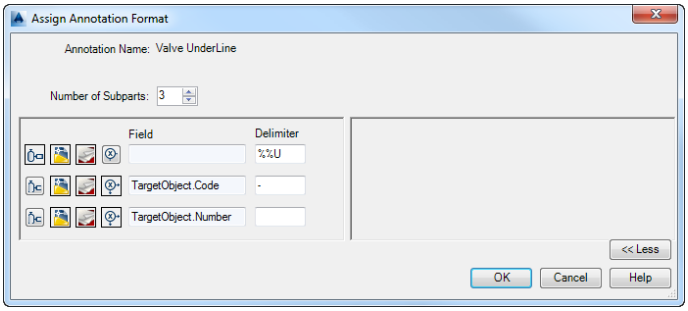


Figure 140: Annotation with Built-in underline

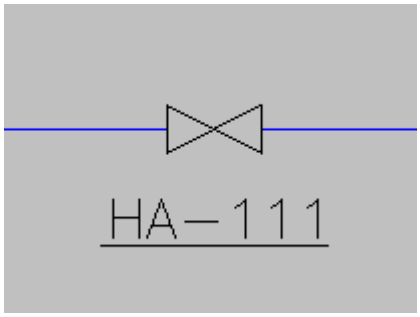


Figure 141: Valve with Underlined Tag

Now, let's use this annotation. Create a Pipe Line and assign a Test Pressure of 250 using the Data Grid. Place a Gate Valve on that line and assign the tag value of GV-001, and a model number of 810.

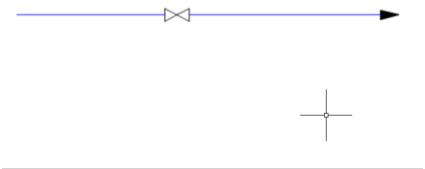


Figure 142: Placed Gate Valve

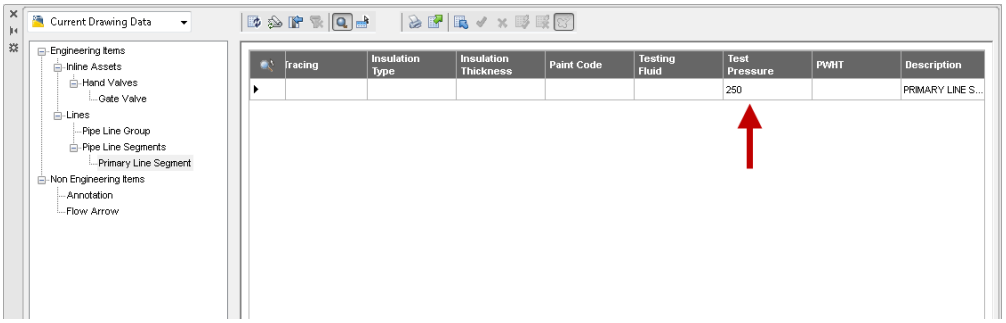


Figure 143: Edit Values

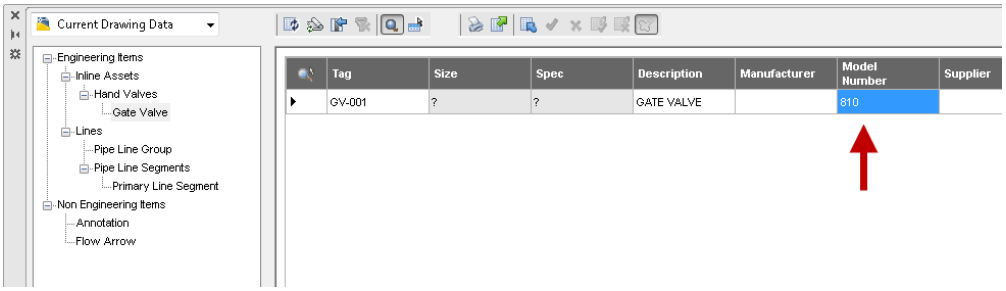


Figure 144: Edit Model Number

Select the Gate Valve, right click and select Annotate… from the context menu. From the submenu, select the “Valve InfoTag” style that we have just defined.

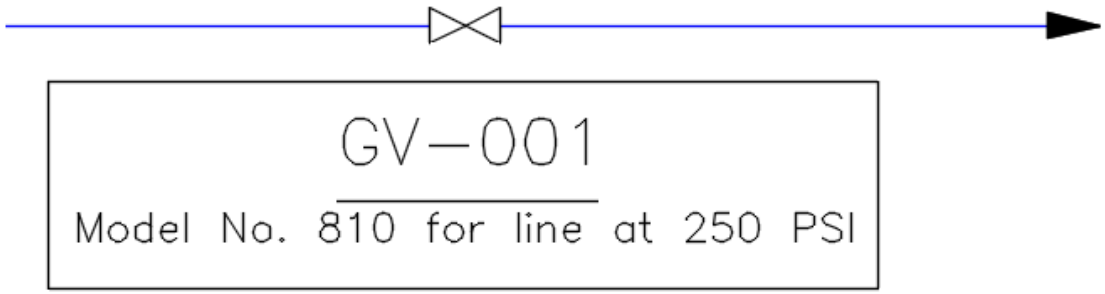


Figure 145: Updated Annotation

The Attribute Definition for #(TargetObject.Tag) collected the target object’s tag value. The second Attribute Definition has replaced #(TargetObject.ModelNumber) with 810 and #(PipeLines.TestPressure) with 250. The rest of the text, both the leading text and those placed using the delimiter boxes, appears unchanged.

Using Expressions in Annotation Format Strings

It is possible to use the Expressions portion of the Assign Annotation Format dialog to enforce formatting. With tags, expression enforce the way the data is input, while with annotations, expression can enforce the way properties display. Let's define another, simple Annotation Style.

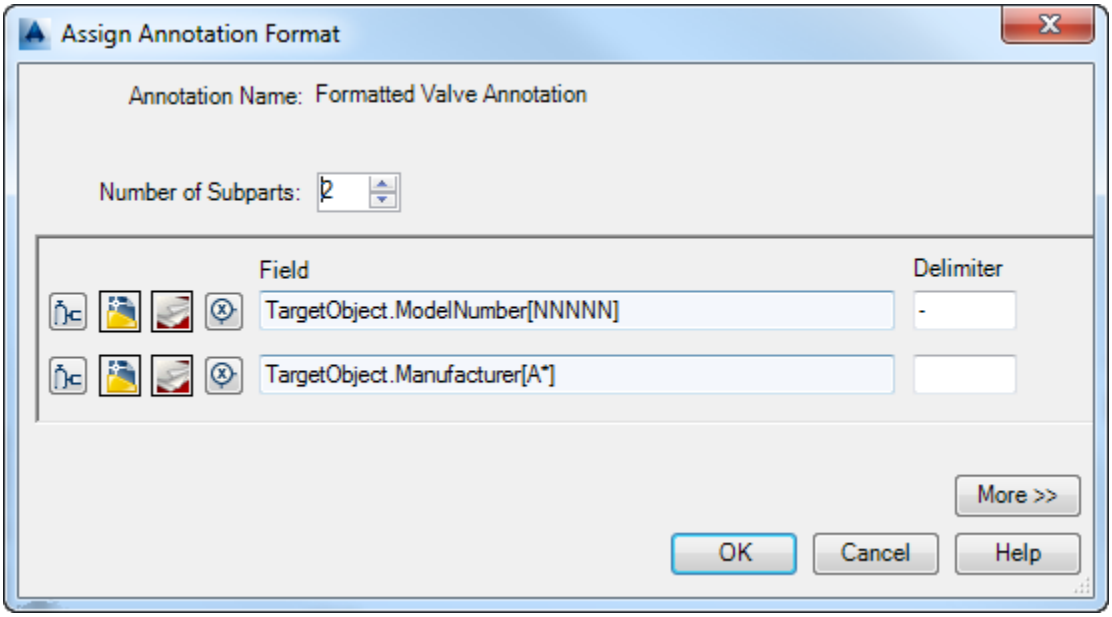


Figure 146: Expressions in Annotations

Using the same techniques as for create Tag Formats, expressions have been applied to both the subparts of this Annotation Format. The first subpart specifies 5 numerals, and the second subpart specifies that the property should be upper cased, any amount of characters.

The resulting format string appears as follows:

```
#(TargetObject.ModelNumber%NNNNN)-#(TargetObject.Manufacturer%A*)
```

The main difference between using an expression in a Tag Format vs. an Annotation Format is that when the expression is used for Tagging (which is the act of creating and storing data), the expression is used to validate and enforce input. When the expression is used as part of Annotation (which reads and displays existing data) the expression is used to format output. The percent sign appears as a symbol after the property, but inside the format string field, to separate the property being read from the formatting being requested.

Our Gate Valve already has a model number of 810. We will add “Bosch” in mixed-case as the manufacturer, and place the annotation. Notice how the number has been pre-pended with zeros to make it 5 digits long, and the manufacturer has been upper cased.

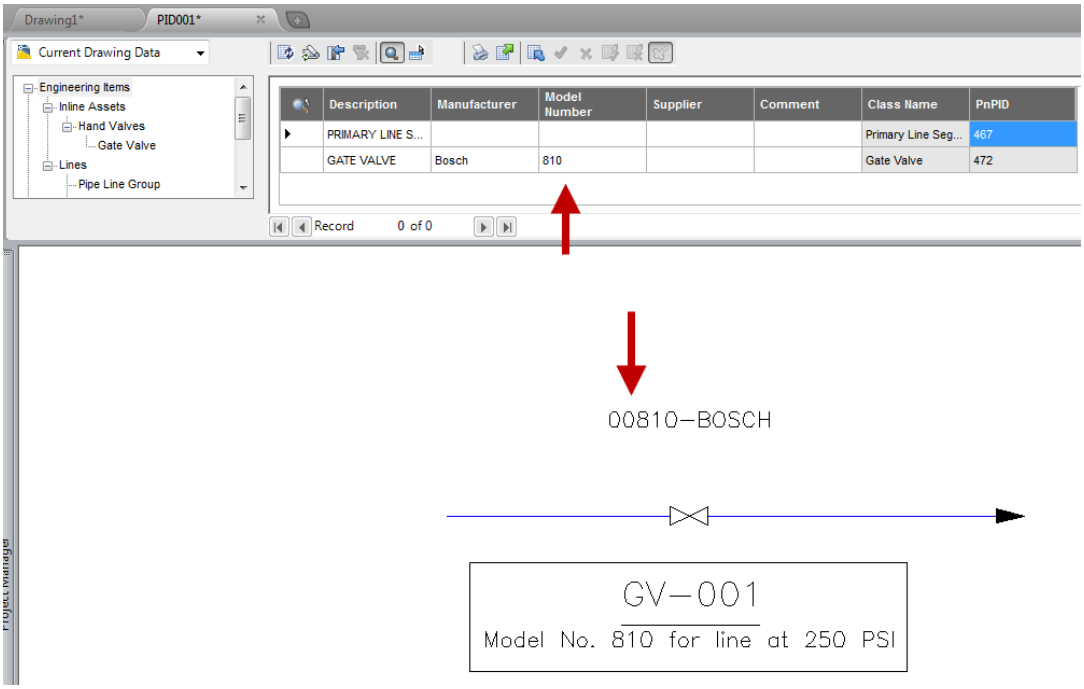


Figure 147: Annotations with Different Formats

It is possible to use the Autogeneration portion of the Define Expression dialog when creating an Annotation Format String so long as you do so in combination with a pure expression, and not with a property. Remember that Annotations READ data, they do not create data, so using an autogenerated expression as part of an Annotation will store the result ONLY in the annotation itself, and not in any underlying storage. AutoCAD P&ID will pre-evaluate any autogenerated expressions so that they do not continue to increment with every refresh of the Annotation. While possible, this usage is not recommended.

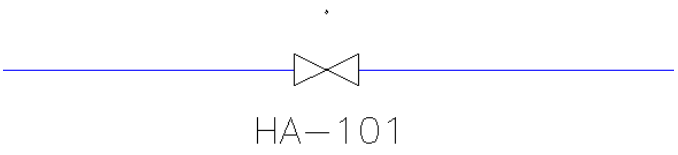


Figure 148: Unpopulated Property

In the image above, notice the small dot above the Gate Valve.

The default Annotation Style for Hand Valves is “Valve Label”, which is defined as a single Attribute Definition showing #(TargetObject.Size). Because there is no size assigned to the Gate Valve, the annotation shows up as a dot. Whenever the annotation cannot retrieve a specific property, it shows

that property as a dot. As soon as we assign a size value to the Gate Valve, the dot will update to show that size.

Any modification to a class property of the target or the class property of a related asset will cause the annotations to refresh automatically. The refresh button on the Data Grid will also refresh all the annotations in the drawing.

If an annotation becomes disconnected from its target entity, all the properties show the string “No Target”.

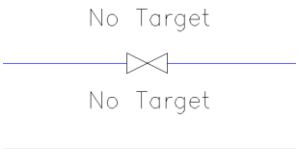


Figure 149: Disconnected Annotation

The most common way to cause such an occurrence is to copy an annotation by itself, and then paste it into a different drawing. To re-associate the annotation, select it, right-click and then choose Assign Annotation Target. Then choose the object the annotation should use to read properties.

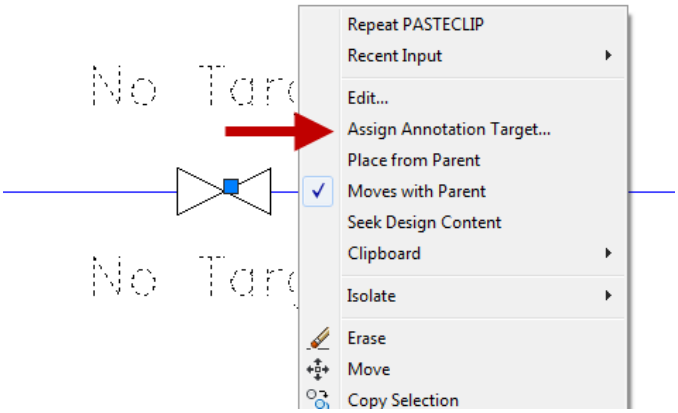


Figure 150: Assigning a Target

The shortcut nomenclature TargetObject is particularly useful for Annotations, even more so than for Tag Formats. Many P&ID classes in the same family have the same properties. For example, all Hand Valves have the Size property.

AutoCAD P&ID provides a very simple method to substitute related P&ID objects; the substitution palette. Using this tool, one can easily convert a Gate Valve into a Ball Valve. Now consider if the Annotation for the Gate Valve was #(GateValve.Size). Upon substitution to a Ball Valve, your annotation would cease showing the size and would instead show the dot. The reason for this is that it is trying to retrieve a GateValve property, specifically, from a Ball Valve. It recognizes the class mismatch, and fails

to return any data. However, by using #(TargetObject.Size), data will be retrieved for any target entity that has a Size property. Thus, the TargetObject nomenclature provides the best possible support for maintaining annotation integrity in conjunction with the substitution palette.

The main thing to remember about Annotations is that the Format Strings in an Annotation Style's block definition are always used to retrieve and display existing data, never to assign data to storage.

Using Annotations to Edit Underlying Data

Once an annotation has been placed, double clicking on that annotation brings up a special editor.

Double clicking our Valve InfoTag annotation yields the following dialog.

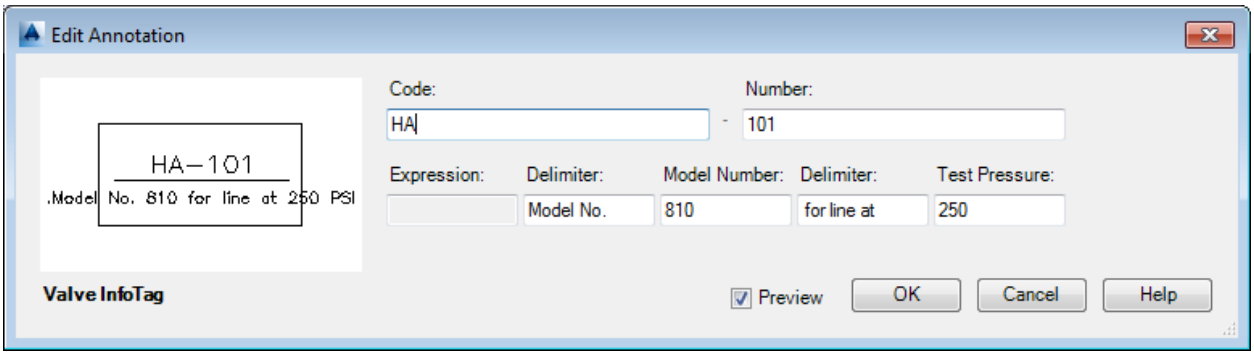


Figure 151: Edit Properties through Annotations

Recall that the first line of this annotation style was based on the format string #(TargetObject.Tag), and that Tags are themselves built up of subparts as defined in the Tag Format.

The first line of this editor utilizes the Tag Format to break TargetObject.Tag back into its constituent parts and present those parts for editing.

The second line of this editor directly uses the Annotation Format String to present each part for modification.

Modifications made in this UI are stored in different locations. All PROPERTY changes to the target object's properties are actually stored directly back to the underlying storage in the datacache. (Drawing and Project properties cannot be modified from this dialog.) The changes seen in the annotation are a result of the annotation refreshing after these values have been changed in their original location.

Modifications to the delimiters are stored in the Annotation itself, and affect ONLY the annotation being edited, not ALL annotations.

Although it would be odd to use pure expressions in an Annotation Format String, nothing prevents you from doing so. Be aware, however, that if you use pure expressions without any delimiters in an

Annotation Format String, the program will have the same problem reparsing the annotation into its subparts as it has when parsing Tags, and you may get unexpected results in the Edit Annotation dialog.

Periodically, users have reported that editing the annotation by double-clicking it doesn't always update the underlying properties. The best practice is to modify through the properties palette.

Special Behaviors --Reducer Annotation Styles

AutoCAD P&ID ships with 4 standard annotation styles for Reducers:

- Fixed Reducer Style
- Large by Small Reducer Style
- Flow Based Reducer Style
- Adjacent Pipe Reducer Style

The first of these is based on the properties of the reducer itself. The Annotation Format String for Fixed Reducer Style is: #(TargetObject.Size1)x#(TargetObject.Size2).

The other three styles are "Pipe Based" reducer styles, and AutoCAD P&ID applies special rules when evaluating the format string, which is the same in all three styles: #(PipeLines.Size)x#(PipeLines.Size).

For these special Annotation Styles, even though the same property appears twice in the format string, the program will collect the sizes from the two separate pipelines attached on each side of the reducer. It will then calculate, based on the name of the reducer style, which size value should be shown first.

- Large by Small Reducer Style --Shows larger size value first always.
- Flow Based Reducer Style --Shows the size value of the incoming pipe first.
- Adjacent Pipe Reducer Style --Shows the size values on each side for the pipe on that side.

Users may create other pipe based reducer styles that show additional data. In order for it to recognize such styles and apply the special processing, the following must be true:

- The Annotation Style must be defined at the Reducers level in the class hierarchy
- The Annotation's target must be a Reducer.
- The Annotation Style's name must begin with:
    - Large By Small Reducer
    - Flow Based Reducer
    - Adjacent Pipe Reducer
- The Format string must contain exactly one delimiter
- The Format string must contain a property #(PipeLines.Property) on each side of the delimiter.
- The Format string may contain other properties, as long as the properties on each side of the delimiter are identical and in the same order.

It is *not* recommended to use the Edit Annotation dialog to modify the values of Pipe Based Reducer annotations, as this dialog cannot distinguish to which pipeline the new size value should be assigned.

Principles of Dynamic Sizing

In the Annotation examples above, several geometric items dynamically changed size to match the evaluated values of the attributes. When the “Valve InfoTag” Annotation Style was placed in the drawing, the underline under GV-100 matched the width of the text, and the box fit nicely around all of the evaluated text.

AutoCAD P&ID leverages the AutoCAD “Dynamic Block” feature in custom ways to enable this behavior. Dynamic block definitions are defined in such a way that each block reference (aka insert) of that block can be independently manipulated in ways that change its appearance, for specific sub portions of the block.

The AutoCAD Dynamic Block feature is extensive and will not be covered in depth here, but it is recommended that interested users spend some time studying the documentation and/or taking an AU course on this feature. Authoring a dynamic block has nearly as much power and potential complexity as writing computer code!

A brief overview of Dynamic Block Principles

Creating a Dynamic Block Definition in AutoCAD involves placing and defining several items in the block definition which are not geometry. These items are known as “parameters” and “actions”.

There are a variety of different types of parameters, each of which track some set of properties, for example:

- Linear parameters have a length
- Rotational parameters have a rotation.
- Point Parameters have a position.

Each parameter can be associated with one or several types of actions. Typical actions include:

- Stretch
- Move
- Rotate
- Flip

Each action is associated with an “action frame”, which indicates which geometry within the block definition participates in the action, and in what fashion. The available actions may be presented to the user as a special grip on the block insert, or the grips may be hidden from the user, depending upon whether the block author intends.

For each block insert, the properties tracked by the parameter may have different values. A single block definition with a single Linear Parameter may be inserted into an AutoCAD drawing multiple times, and the length of the linear parameter may be different in each block insert. Depending upon that length, actions associated with the linear parameter will yield different results.

The list above is by no means complete. The AutoCAD P&ID Annotation feature extends the behavior of Linear Parameters, so the remainder of the discussion will focus on those.

Linear Parameters in Annotation Block Definitions

In truth, AutoCAD P&ID makes only a slight tweak to the behavior of Linear Parameters. It recognizes when an Attribute Definition caries an annotation Format String. If the block definition also carries a linear parameter whose label is identical to the Attribute Definition, then in each block *reference*, *after* the format string has been evaluated, it will set the linear parameter to be the same width as the evaluated value.

In other words:

In the block definition you may have a format string #(TargetObject.Tag), 19 characters long, but which becomes GV-001 in the block reference, 6 characters long. In the block *reference*, the linear parameter will be set to the length equal to the six characters.

Let's look at a simple example, the "Oval Tag Style" Annotation Style. Using the skills discussed above, navigate to the Engineering Items node in the P&ID Objects Definitions class hierarchy, select the Oval Annotation Style, and press the Edit Block button.

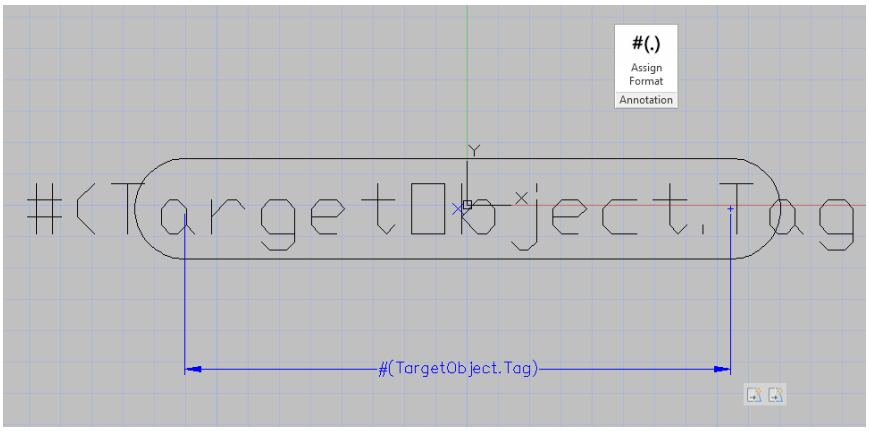


Figure 152: Oval Tag Dynamic Sizing

The block editor shows an oval shaped piece of geometry consisting of line and arc segments, an Attribute Definition carrying the #(TargetObject.Tag) format string, and a Linear Parameter also carrying the #(TargetObject.Tag) format string. The Linear Parameter is associated with two stretch actions.

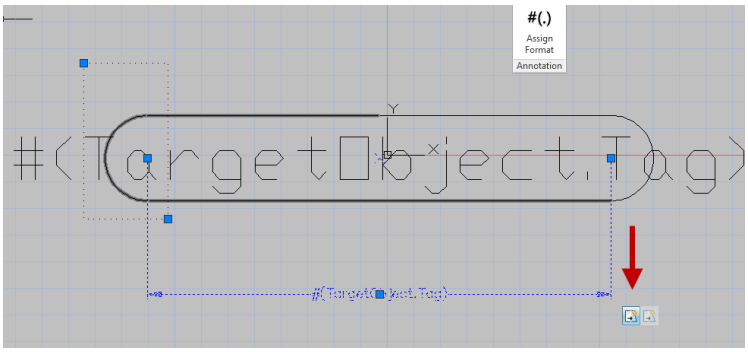


Figure 153: Stretch Association

Selecting the stretch action on the left highlights the stretch action itself, the parameter with which it is associated, and the geometry which is participating in the action. A small rectangle shows the "action frame" for this action.

For stretch actions, the rules are as follows:

- Entities which are not selected by the action are not modified
- Entities which are selected by the action and are completely inside or completely outside the frame are moved.
- Entities which are selected by the action and which cross the frame are stretched.

In the example above, the arc on the left side of the oval is completely inside the frame, so when the left side of the linear parameter moves right or left (depending upon how the overall linear parameter is sized) the arc will move. The line segments at the top and bottom of the oval cross the linear parameter, so they will stretch.

For completeness, we highlight the other stretch action as well.

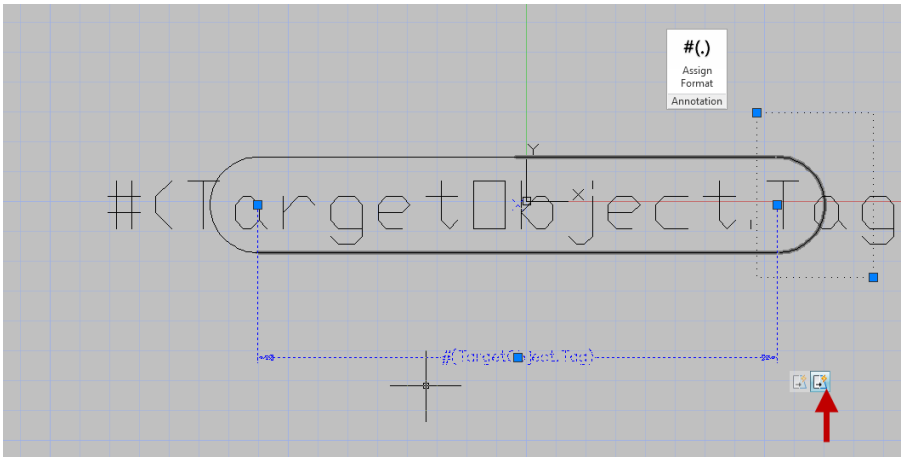


Figure 154: Right Hand Stretch Association

The right hand stretch action also selects the top and bottom lines, but selects the arc on the right hand side of the oval.

Selecting the linear parameter itself and raising the properties palette shows some special settings specific to the linear parameter.

Notice that the endpoints of the linear parameter are actually slightly inside the edges of the geometry, rather than directly on the edges. Doing this ensures that there will be some space around the text when the stretch actions are applied. If the linear parameter were exactly on the edges of the geometry, then when the parameter is sized to match the attribute's width, the geometry will exactly enclose the attribute, which usually doesn't look very good.

Note: The InfoTag2LineBox block definition in AnnotationTemplates.dwg used in the previous section, "Creating a New Annotation Style," actually shipped with this very problem. To get the best results, you

may need to adjust the end points of the linear parameters to be inside the box, and assure that the Base Location property on each is "midpoint", as discussed below.

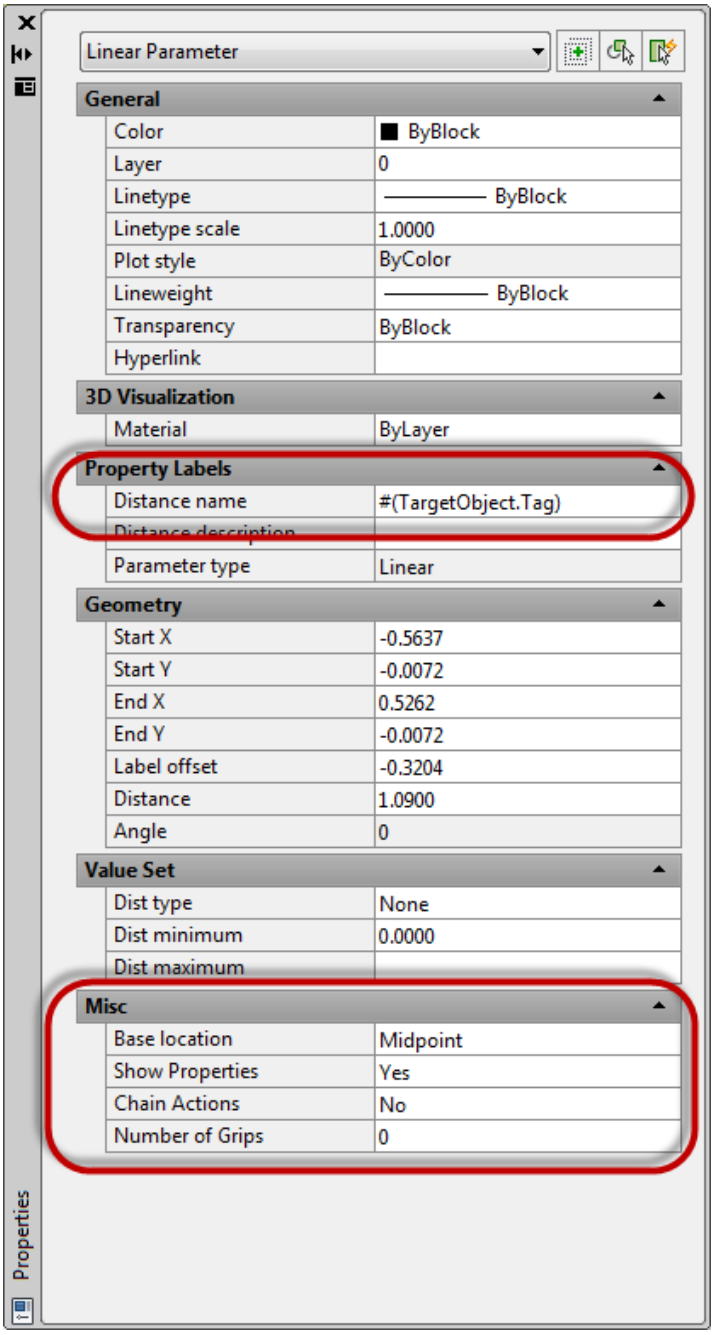


Figure 155: Parameter Properties

The property called "Distance Label" is the property that must match the Attribute Definition's format string. If these values ever fail to match, then the linear parameter will not be associated with the Attribute Definition. You can use the property palette to fix such errors, should they occur.

The linear parameter's **Base Location** property can be start point or midpoint, and indicates how the linear parameter will be anchored as its size is changed. In most of our shipping annotation templates, we use a midpoint anchor for the linear parameter, and center-justify the Attribute Definition at the same X Value. This combination results in the geometry controlled by the linear parameter staying centered on the attribute.

The linear parameter's **Show Properties** property controls whether the linear parameters properties will be shown in the properties palette of the block *reference*. Generally, the annotation block definitions we ship have this value set to No, because the block reference's parameter's properties are being controlled programmatically, and user modifications made via the property palette would not "stick". However, setting this property to Yes can be extremely helpful when you are first authoring, and testing, your dynamic blocks.

The linear parameter's **Chain Actions** property is an advanced property, and indicates whether the parameter can be controlled by another parameter in the same block definition.

The linear parameter's **Number of Grips** property indicates how many grips should be shown for each block *reference* placed in the drawing. The number and type of grips differs depending upon the type of parameter and the number of associated actions. Because the linear parameter is going to be programmatically controlled, we ship our annotation block definitions with the number of grips set to zero for our linear parameters.

Here are several examples of this annotation style inserted into the drawing.

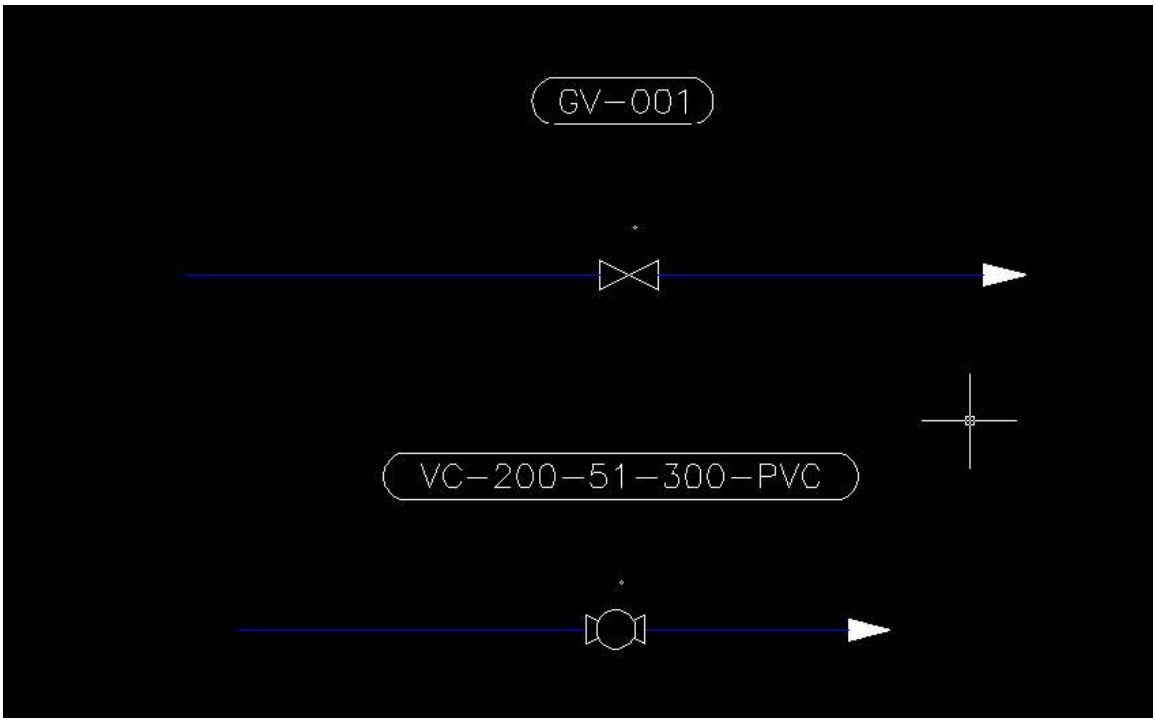


Figure 156: Inserted Oval Annotation

The Gate Valve has a short tag value, and the Ball Valve has a very long tag value (based on our “Big Example Tag Format” above.) As can be seen, the oval has stretched to fit around the text, with a bit of padding.

For illustrative purposes, I enabled the “Show Properties” setting in the block definition, and show the properties for the two block references below.

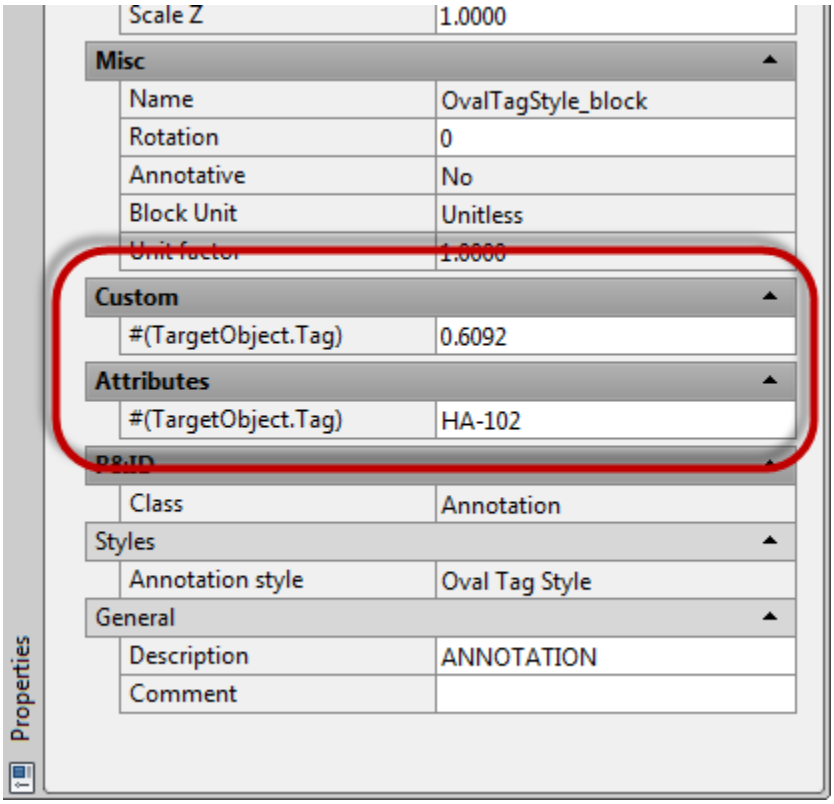


Figure 157: Insertion Annotation Properties

Notice the property under the “Custom” heading called #(TargetObject.Tag). In this case, you are being shown the “Distance Label” of your linear parameter as the property name, and the actual width of the linear parameter as the value. When you first begin authoring dynamic blocks, you may find it helpful to have this enabled, and to make manual changes to the width of the linear parameter, in order to preview how your actions will behave at various widths.

Directly below, under the Attributes heading, you see the attribute for #(TargetObject.Tag), and its current text value. Selecting this item and clicking the ellipsis button will raise the Edit Annotation dialog for that attribute.

TotalX and TotalY

There are two special names that AutoCAD P&ID understands for linear parameters, in addition to those named to match Attribute Definition's format strings. Let's review the block definition for our Valve InfoTag annotation style.

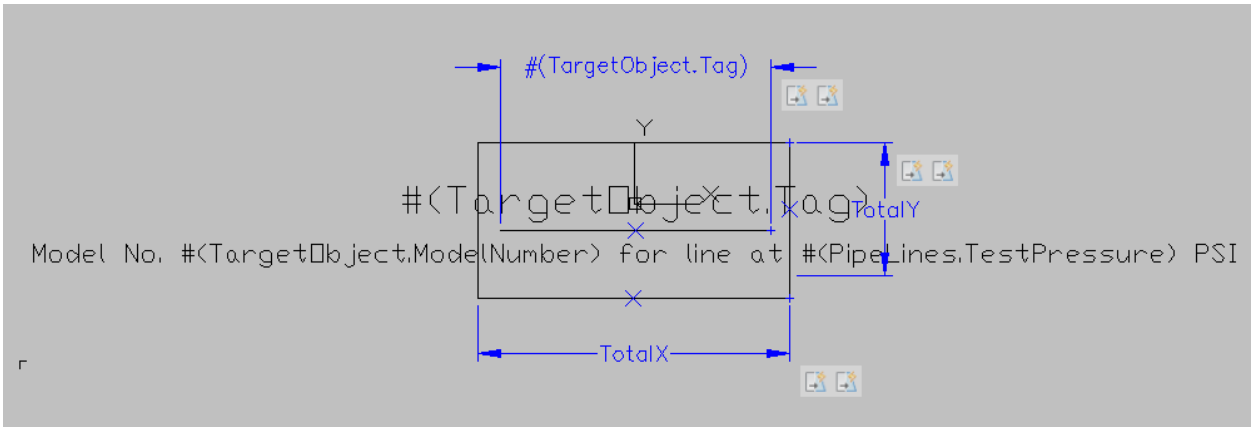


Figure 158: TotalX and TotalY

Notice the two linear parameters named TotalX and TotalY.

In addition to using the width of an evaluated format string to set the width of an associated linear parameter, AutoCAD P&ID also calculates the combined extents of all the evaluated format strings in the entire block reference. If it encounters the TotalX linear parameter, it sets its size equal to the width of those combined extents. If it encounters the TotalY linear parameter, it sets its size equal to the height of those combined extents.

This can be useful, as in this example, for creating a box that surrounds several format strings.

Note that TotalX and TotalY will not collect the extents of other textual entities in the block; only those associated with format strings.

AutoCAD Plant 3D Content

Class Structure

Class structure functions the same way in the piping projects as in the P&ID project. See the Class Structure and Behavior section under P&ID.

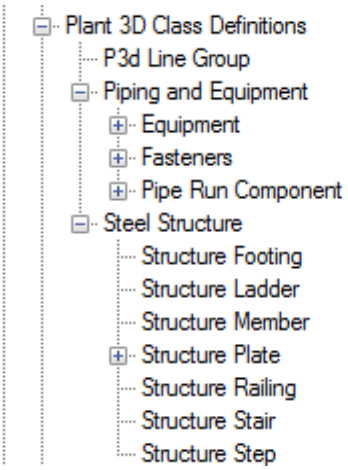


Figure 159: Plant 3D Class Structure

The default structure in AutoCAD Plant 3D content is organized according to modeling functionality. The AutoCAD Plant 3D class structure is not modifiable, as behaviors are strictly defined by the program. The P3d Line Group functions similarly to the Pipe Line Group class in AutoCAD P&ID. However where the Pipe Line Group class contained Line Segments, P3d Line Groups contain Fasteners and Pipe Run Components. Also, in AutoCAD P&ID, every valve has its own class, while in AutoCAD Plant 3D, every valve is in a single class. Often Isometrics are generated according to the P3d Line Group. See the discussion of Line Tagging for information on how to build a tag for P&ID and Isometrics.

The other main difference is class structure is how data can be populated. In AutoCAD Plant 3D, properties from a spec will populate a matching project field, so custom properties are easily defined and used from catalogs all the way through to drawing production (see Property Flows).

Selection Lists

Selection lists function as they do in AutoCAD P&ID, but are stored in the piping database. See the selection lists section in P&ID. The selections lists between the P&IDs and AutoCAD Plant 3D piping are completely separate, and have to be setup independently of each other.

Equipment Templates

Equipment templates can be used from two locations, the project equipment templates location defined here, and the shared content folder location: C:\AutoCAD Plant 3D 2015 Content\CPak Common\equipment.

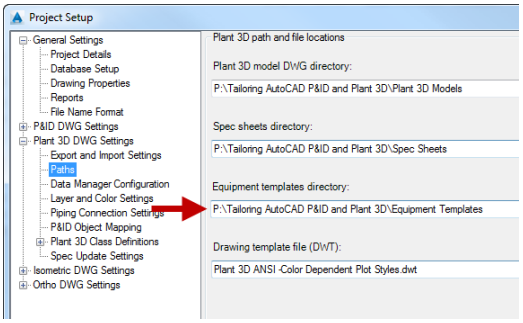


Figure 160: Equipment Template Path

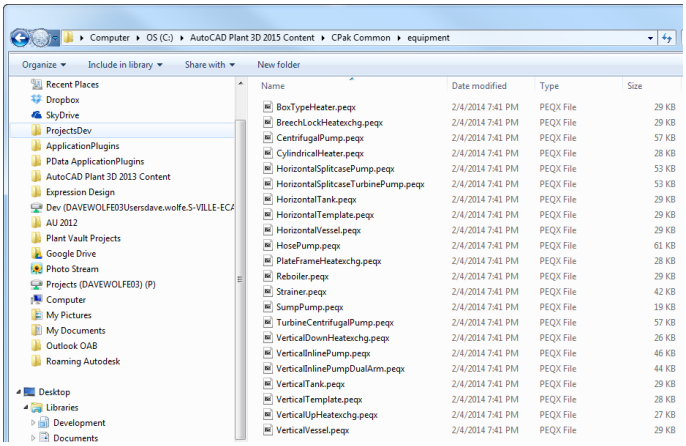


Figure 161: Default Equipment Path

Templates stored in the project's equipment templates location will show up at the bottom of the equipment dialog, while templates stored in the content folder will show up in the top list.

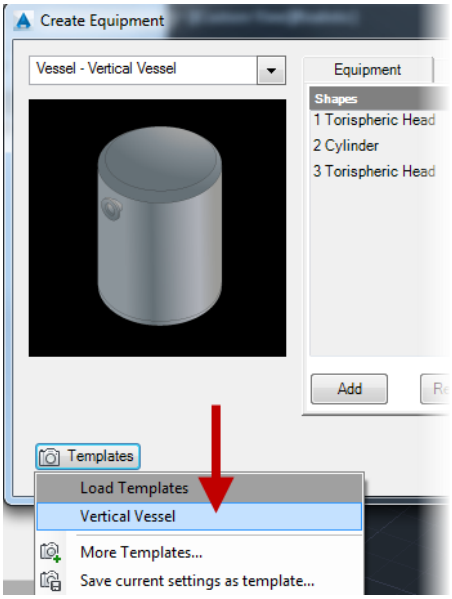


Figure 162: Project Templates

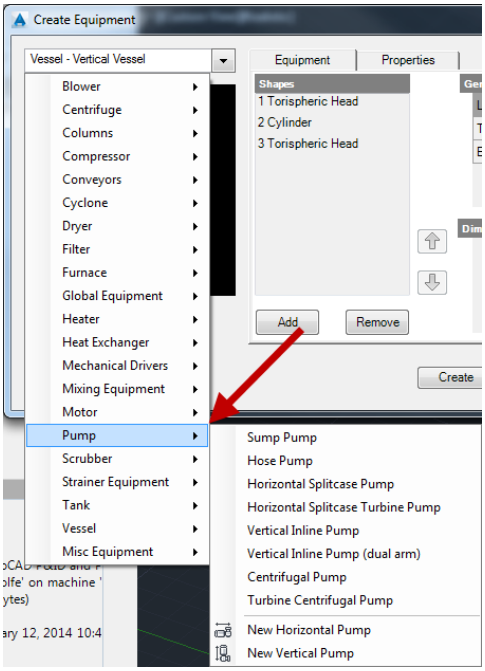


Figure 163: Default Equipment

This lecture contains information on modifying equipment scripts for setting their category:

- http://au.autodesk.com/au-online/classes-on-demand/class-catalog/2012/autocad-plant-3d/scripting-components-for-autocad-plant-3d

Model Organization

For small projects, like skids, or individual line or piping runs, model organization doesn't play a large role. However, when you have to work with multiple users on the same project, or designing a large facility, you need to set some time aside up front to organize your file and model structure.

Before you start to create the project in AutoCAD Plant 3D, take a big step backwards and look at the project as a whole. Think about how to logically split it up. You can have more than one model per area, so keep in mind that there is a degree of flexibility.

- If it is a big site, mostly all on one level, like a refinery, then look at a site plan and divide it up into areas. Each area may be a process unit, or it may be a logical sub-division of a process unit, but consider it as an AREA.
- In the case of a multi-story plant, you may want to treat each floor as an area and then subdivide into physical areas.
- For complex areas, giving every service its own model may make sense

In this example the project is a refinery. The site plan that is broken into 14 areas:

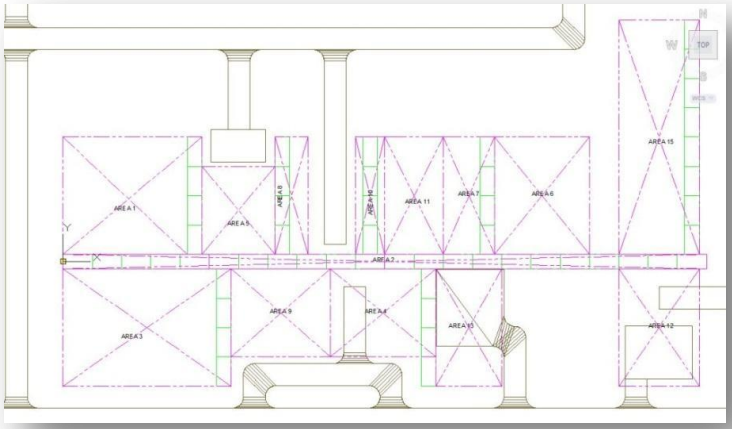


Figure 164: Area Layout

In the middle is a pipe rack which joins up the separate areas (Area 2). This is considered a separate area since one designer will be responsible for considering the layout of only the pipes in the rack. Area 13 is a servicing area and contains no 'plant' items (i.e. the models in this area are AEC models, not AutoCAD Plant 3D models) and so there are no P&IDs and no AutoCAD Plant 3D models in this area.

Within each area you have to consider Equipment, Structures and Piping. And if we have different designers focusing on each discipline, you may want to divide up areas into the disciplines.

Setting Up Your Project

Using the example above, 14 areas were created. Each area can be managed by a lead designer, but each area has a design team comprising equipment layout designers, Structural layout designers, and piping designers.

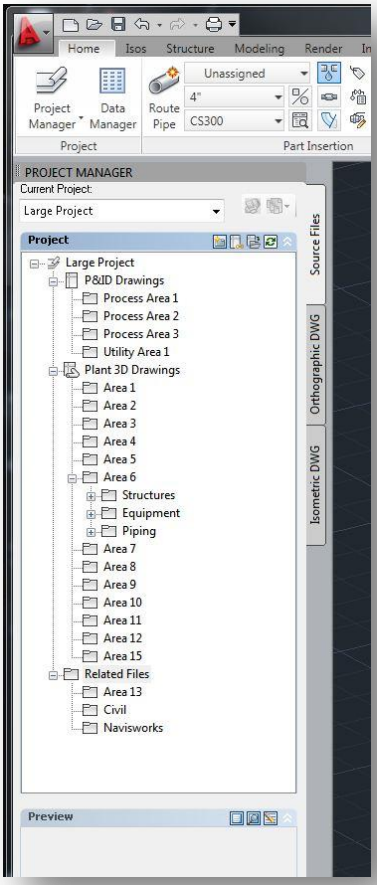


Figure 165: Large Project

In the case of P&ID's, the project splits up into three process areas plus one utilities area which will contain all the Utility P&ID's (or ULD's). The fact that the P&ID area will span several of the Piping areas is not important as AutoCAD Plant 3D will track and manage all the lines and equipment in the project.

So the project will look like Figure 165 in AutoCAD Plant 3D Project Manager.

Note the 4 P&ID Area's and 13 Piping Areas. The project also has allocated folders in 'Related Files'. Related files may store or reference the site plan drawings in the 'Civil' folder. Since Area 13 does not contain any AutoCAD Plant 3D models (it is AEC) it has been placed in the Related Files section of the project. A Navisworks folder was created to store the associated .nwf files which will allows assigning materials to – or managing layer visibility within – the model files in order to create realistic renderings of the model while in design. Published .nwd files can be placed here for project review sessions. This allows project reviews to take place while design work proceeds, without interrupting design work.

Now look at Area 6 as an example:

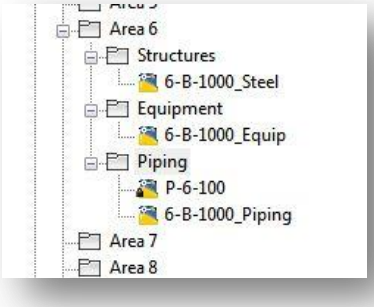


Figure 166 Area Organization

The Structures folder holds a structural model, Equipment Modeling goes in the Equipment folder, and 2 piping files are in the Piping folder.

This structure keeps the file content small and manageable and yet allows maximum flexibility in working in the project. By using Xref's, the designer can focus on the part of the model he is particularly interested in at any moment by unloading the Xref's he doesn't need. Then when he needs more information, simply reload the Xref's for full model realization.

The example above is one possible solution to organizing models. You may find that it doesn't suit your needs, but with the flexibility in AutoCAD Plant 3D, you can find a solution that enables good design and ease of use.

Other factors for deciding model boundaries, in addition to process flow, are model density, file size, and number of users in the project.

For example, an energy center typically would not have the amount of piping that a process building would contain, so it would contain fewer areas. Also, you would never want to have fewer models than the peak number of projected model designers, since multiple people cannot edit the same file simultaneously.

Your structural model might be an entire building, which may not lend itself to being broken down into smaller areas. Typically structural models are not complex, so retaining an entire building in one file does would not hurt performance or accessibility.

Consider also whether you need to issue general arrangements without piping. Some companies do not issue model-based general arrangements, and may keep the equipment within the piping model. Those that do need to issue live GA's should have a model that loads the equipment separately from the pipe to avoid the work involved in having to separate out the individual pieces of equipment into separate models.

Controlling Access

In some organizations, drawing access needs to be controlled between disciplines. Typically this can be accomplished by setting up folder permissions with group policy. Make sure to structure your project so that drawing access can be controlled if necessary. You can use virtual folders as discussed in the Folders section. Permissions would need to be set after the project is created (or on the template folder if you are going to copy the project folder to create a new project).

Coordinate Systems

Project locations are rarely in a position where north is at 90 degrees. AutoCAD Plant 3D includes the ability to generate isometrics with coordinates oriented to plant north. In Advanced Defaults under Isometric DWG settings, you can set up an offset and orientation for your isometrics. Note that the elevation doesn't have an offset, so your model should be at the true elevation you need to display. You can also set the location overrides in the Advanced Settings Dialog when creating an isometric.

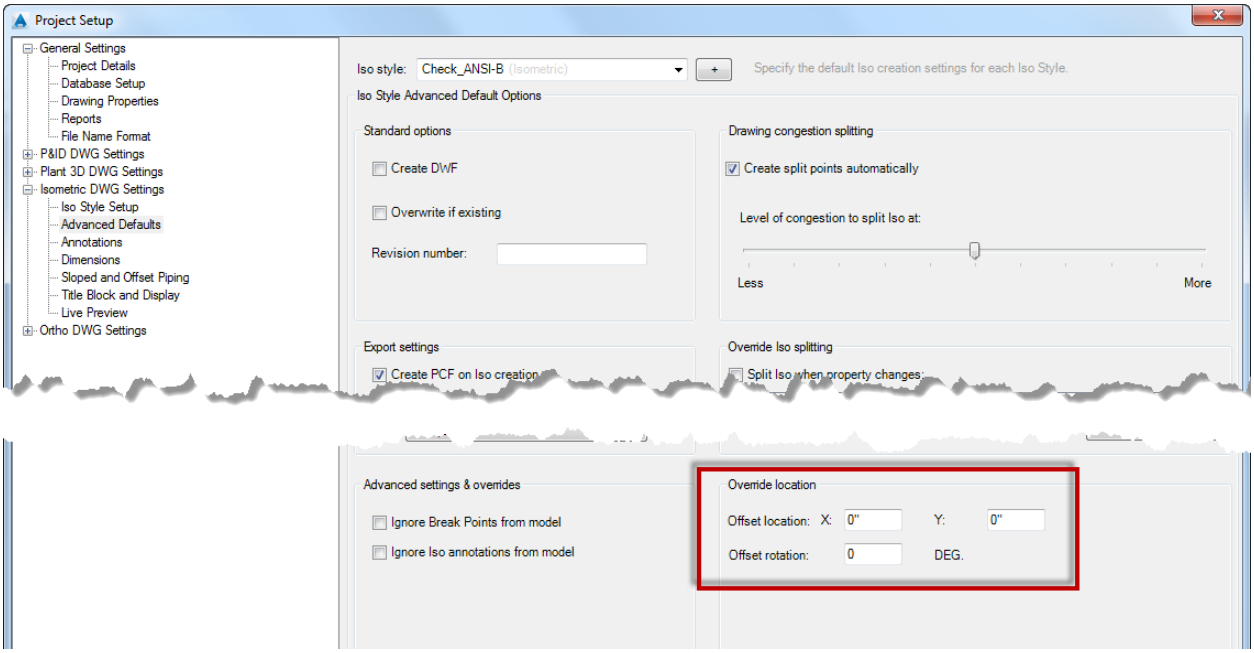


Figure 167: Coordinate Overrides

External References

AutoCAD Plant 3D is designed to work ***intelligently*** with Xref's. Pipes and Piping Components will connect to pipes and piping components, and even equipment nozzles, in Xref models. For example, when using the Route Pipe command, by clicking on the equipment nozzle in the Xref, the connection will be made and the correct components will be placed as shown here:

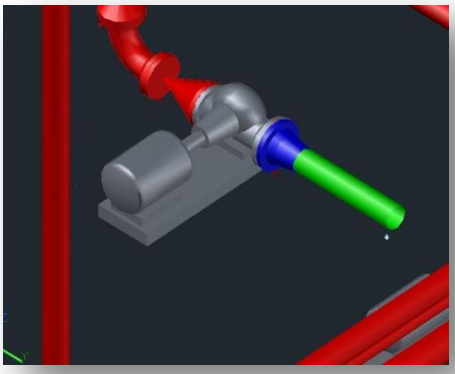


Figure 168: Pump Routing

Note that the pump is in the equipment Xref and that an 8" line is being routed and connecting to a 10" nozzle, a 10" mating flange with a 10x8 reducer was placed in the piping model. Also note the connection was also placed which will give the correct bolts and gasket.

Simplifying Working With External References

One useful way of managing the Area Xref's is to have a master area drawing. This is essentially an **empty** model file with **all** the area models loaded as Xref's. In this example, Area 6 in the master area drawing is Area6.dwg:

It is important that these Xref's be loaded as **Attachments**. You should use a relative path as shown below. (As the files are part of a project using a **relative path** will make moving a project easier):

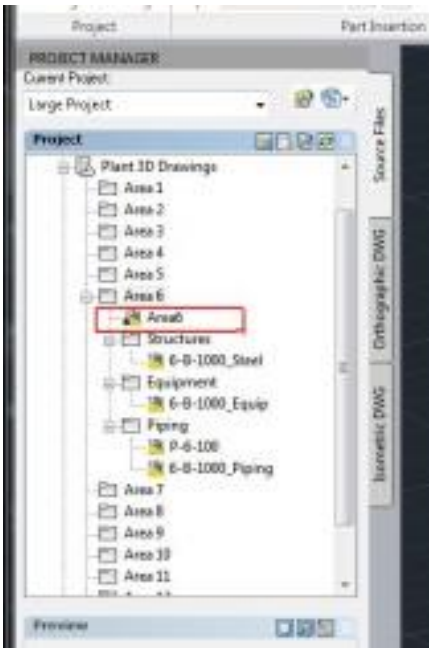


Figure 169: Master Model

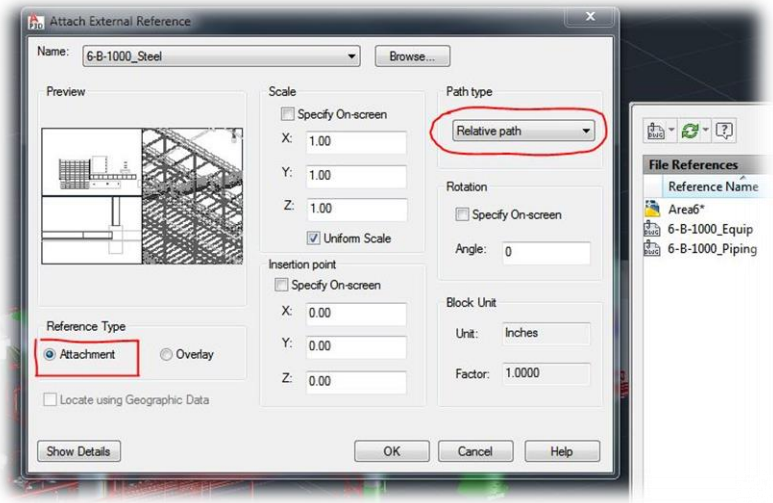


Figure 170: Attaching Xref

The Area Master file is maintained by a project lead that will ensure that all the model files in that area are attached. The reason to use attachment reference type, rather than the usual overlay, is that these Area Master files will be Xref's in the Master Model (see later) and the nested Xref's must display.

By creating these Area Master models, managing model files in the area is simpler. Now any designer working in that area simply attaches this one file as an Xref, but in this case in Overlay mode**,** and he will be sure that all the models in that area are attached. (A warning message detecting circular references may be displayed; simply click Continue.) If working across areas, for example connecting pipes to a pipe rack that may be in a separate area (for example attaching to pipes in Pipe Racks), then attach that Area Master drawing and then detach (or unload) when no longer needed.

Word of warning regarding Circular References**:** If you follow the guidelines above, then circular references (file referring back to itself as a lower nested Xref file) will be taken care of by AutoCAD. The danger arises when you need to Xref a file in another area. In this case if you attach the external area file to your model file as an attachment, you will create circular references in the Master Model file (see below) or in the Orthographic Drawing master model files. To avoid this problem, best practice is to use the overlay reference type for these external files.

Designers can turn on or off any layers in the reference files or unload the whole reference file as needed.

Modeling Using Xref's

Xref's make it very easy to work in large models. Here the piping designer is concentrating on piping around a pump:

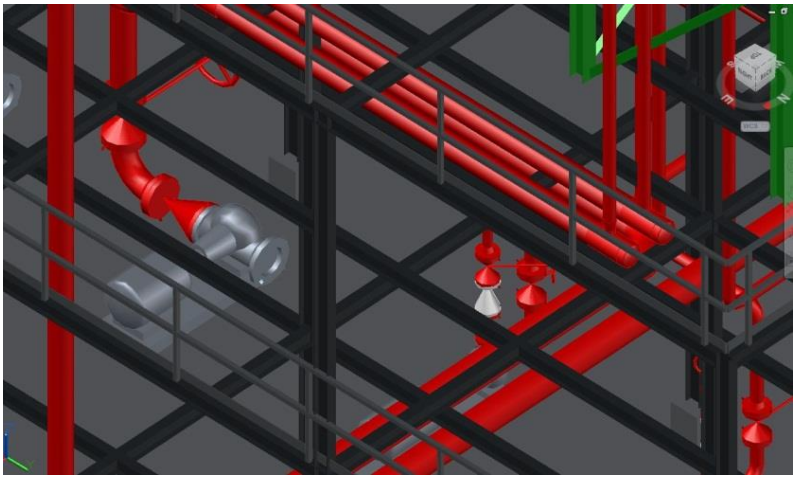


Figure 171: Crowded model

By unloading the structure and turning off some of the piping layers, a better view is obtained making piping easier:

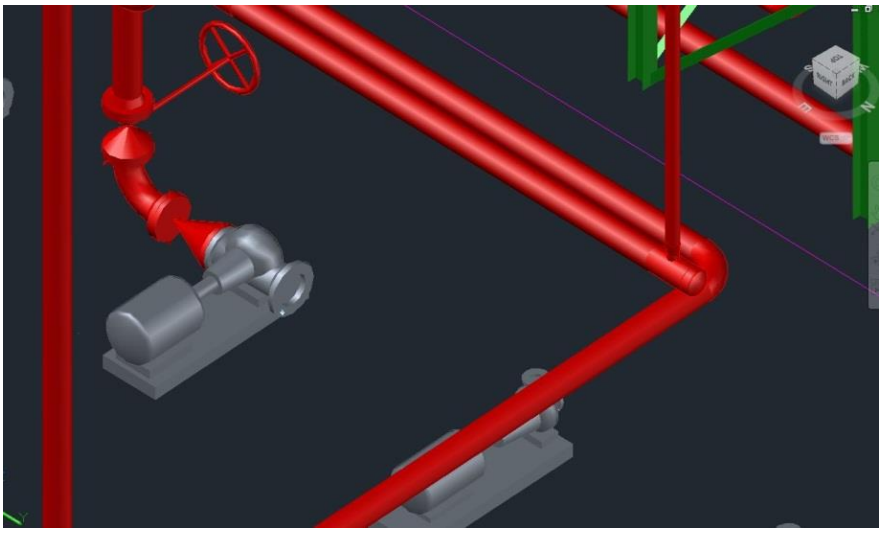


Figure 172: Unloaded Xref

Managing Drawing Sizes With Xref's

The optimum size for a model file is 5-10MB. Depending on the hardware you are using and the type of plant model, it may be better to keep the model size smaller. This can be especially true if you have objects that are not AutoCAD Plant 3D objects (e.g imported from Inventor or Revit). This will result in

more dwg files, but good model management will maintain good performance. It doesn't matter how many files you have in the project, AutoCAD Plant 3D project manager can manage large numbers of drawings.

With experience, you'll be able to estimate how many lines or how many equipment items will be in each model file in order to keep the sizes down, but there is no problem adding more model files as the project develops. This is especially true for piping files.

Some users like to put all equipment in an area into one dwg. Others like to split them up into sub-areas where others may split the equipment up by levels on a structure.

In the case of structures, many users will place a 'single' structure into a single model file. This is so the structure can be exported for analysis and then replaced with a detailed structural model after the detailed engineering has been done. In some cases, the structure may actually extend across an area boundary. This is not a problem, simply place the structural model in one area and then have the master area model of the other area(s) xref this structural model.

Pipe Racks are typically handled separately too. This is to allow the piping designer to optimize the space available on the rack and this is its own layout problem. Thus the Pipe Rack area may also contain sub-racks which will encroach into process areas. The best practices approach discussed so far handles this approach easily.

Managing piping models can be done in many different ways since AutoCAD Plant 3D allows intelligent connections across Xref's. Thus, piping models can be divided into sub-areas or by line numbers, with each designer allocated a certain number of lines. There are no constraints on how to divide up the area.

Catalogs and Specs

Beginning in 2015, catalogs and specs created in AutoCAD Plant 3D 2014 and higher do not have to be migrated and are version independent. While the format isn't guaranteed for the future, through the next couple of releases at least it should stay consistent. The organization of your specs and catalogs become dependent on one main decision, whether you will need to display or use weights on parts. The decision to use weights general relies on whether you are going to display an approximate weight on the iso, or whether you will be interfacing with a stress analysis program. In both of those cases, you will have to make certain your catalogs include the appropriate weight information.

For some customers, the out-of-the-box catalogs and specs contain enough parts to address their modeling needs. However, for most people they need to add some additional parts, especially for instrumentation, control valves, or other non-standard items.

To set up our catalogs and specs in an extensible, manageable manner, you need to manage new and existing catalogs to make the necessary parts available for any project.

Workflow

Everyone must leverage the installed content in some manner. First, you will create a network folder structure to hold content. Then you will create a new catalog to hold new and modified content. Third, you will manage how your specs are used. Finally, you will see how either point users to a common network location, or transfer custom content for use on their local computer so they can use deployed content.

Network folders

In order to locate and manage catalog content, you need to move the default content folders to a network location. Since you no longer have to make the content version specific, you can setup a folder like K:\Plant Content\Catalogs.

To move content to the network, you must open the spec editor and click Tools > Modify Shared Content folder.

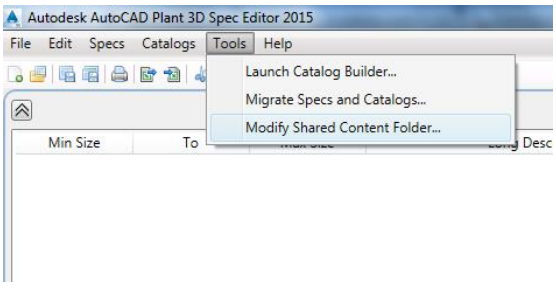


Figure 173: Modify Shared Content Folder

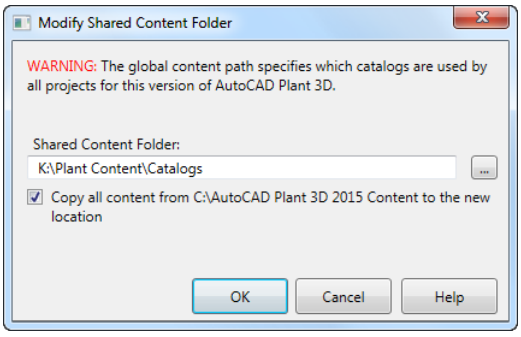


Figure 174: Copy Default Catalogs

Check the Copy all content to the new location. This copy function runs some code in addition to the copy method, so this is the recommended workflow. After all the content is setup, you must modify each user's shared content folder by using the same method except for the copy option.

One potential network structure could be like this:

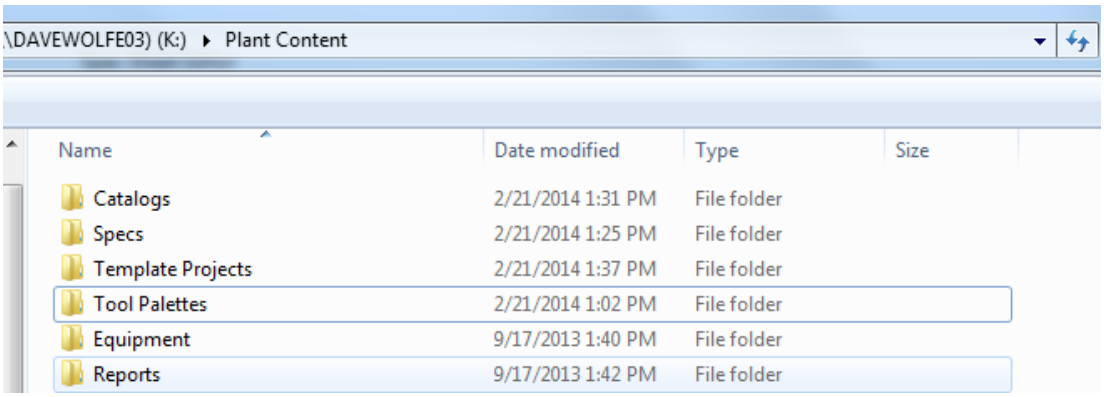


Figure 175: Network Folder Structure

In this example, the Catalogs folder becomes the Shared Content folder. Specs are kept separate so they can be managed using client folders. Other folders are included as a way of organizing other types of templates and project content.

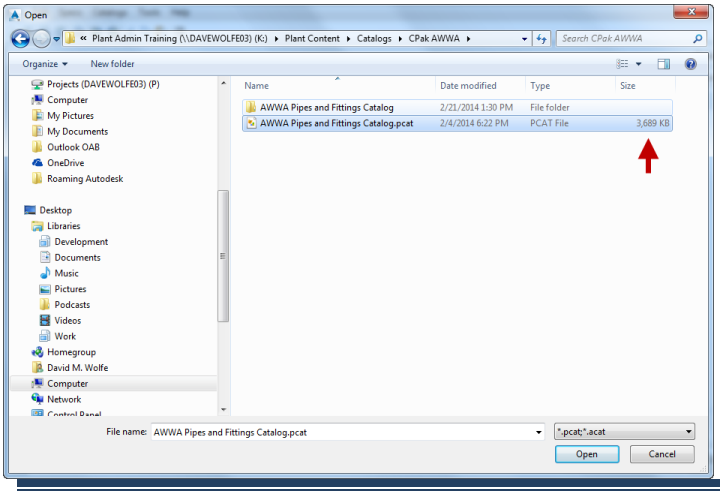
When using a content folder on the network the program loads core drawing information from the network. For example, in the previous example, the location K:\Plant Content\Catalogs\CPak Common\ contains drawing and shape information required for the program to run. Information in the CPak Common folder includes steel shapes, equipment templates and nozzles.

Note: If you created custom block content while using the local content folder, C:\AutoCAD Plant 3D 2015, you need to make sure the CatalogSupportFolder gets copied up to the network content folder, or your custom blocks will not be found. Creating block-based content is not recommended as it becomes harder to manage and will increase the file size

Create Catalog

Next you need to create a new blank catalog. There are a few reasons for this:

- Separates out of the box content
- Allows you to replace default catalogs with updates without losing customization
- Better performance opening, filtering, and importing/exporting



Remember that if you plan to include valves, you should also create actuators in the same catalog.

To create a blank catalog, open up one of the smaller default catalogs like AWWA Pipe and Fittings in the catalog editor.

Figure 176: Copy small Catalog

Then go to File > Save As and give the catalog a name. Make sure to save it within the network catalog folder. The select the first item in the catalog, scroll down to the last item in the catalog, hold down Shift and click on the last item. To remove all of the selected items, click Delete Component. This may take a few seconds, but it will remove all the components.

Manage Specs

After placing the content on the network, you need to organize your specs. Specs should typically be organized outside of the catalog folder. Parts in the spec are loosely linked to the catalogs they are built from. Meaning that the spec knows where the item came from, but it won't update unless you tell it to. Before updating specs, you should check the update settings. Typically users modify the long description family after placing an item in the spec.

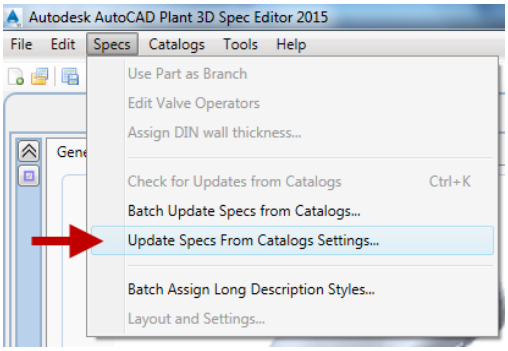


Figure 177: Update from Catalog Settings

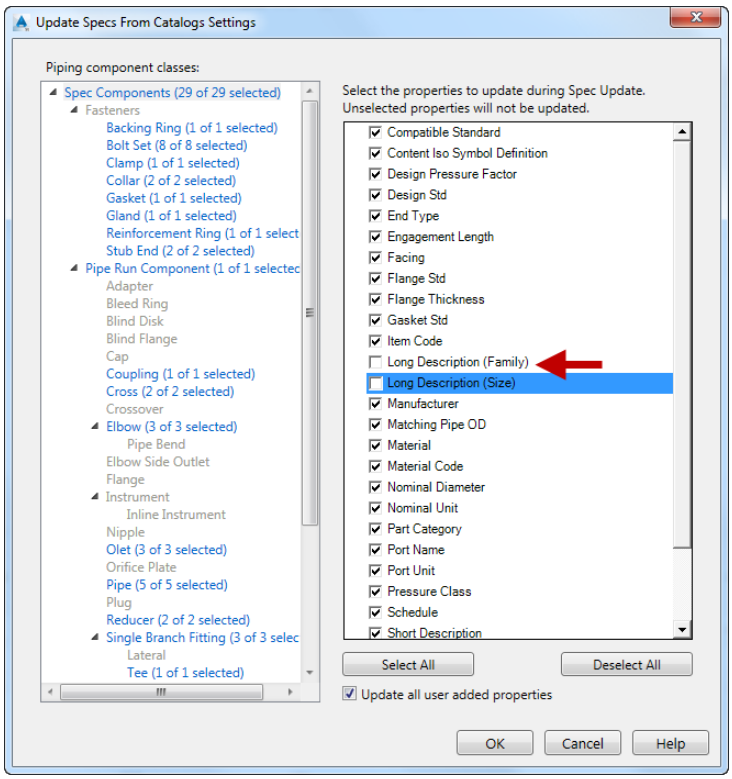


Figure 178: No Long Description Update

In order to avoid losing the description modifications, you should uncheck the properties in the update settings. A similar dialog is also available on the project which allows you to modify the descriptions in the model, and update without losing description changes.

Once your update settings are modified, you can make catalog changes, and then use the Check for Updates from Catalogs option:

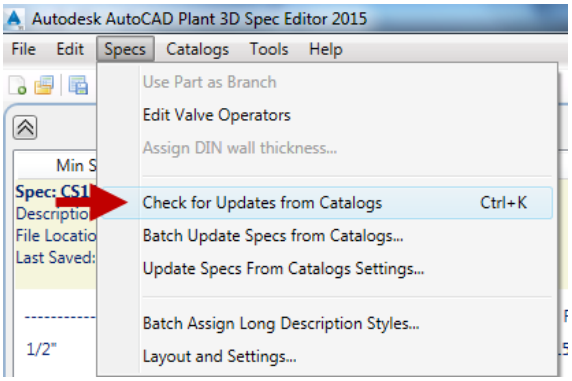


Figure 179: Check for Catalog Updates

Long Description Styles

Another feature in the spec editor that needs to be evaluated for use with managing specs are long description styles. Long description styles provide a way to build a part family long description or part size long description based on existing property values. To access the default long description style, open a spec and click the Layout and Settings button.

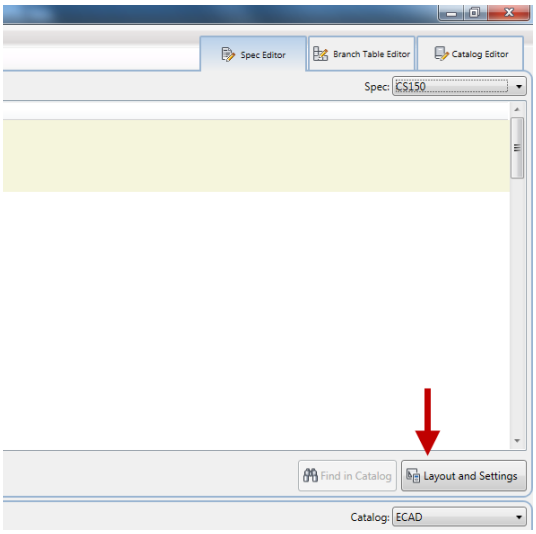


Figure 180: Layout and Settings

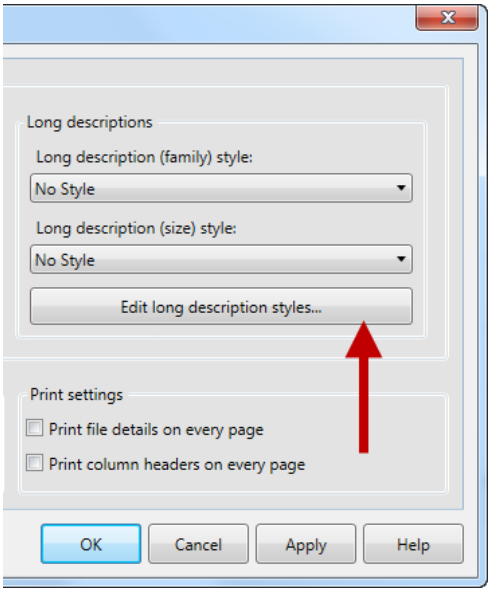


Figure 181: Edit Long Description Styles

Note: In order to modify long description styles, you must be running the spec editor with administrator privileges. The styles are saved to C:\ProgramData\Autodesk\AutoCAD Plant 3D 2015\R20.0\enu\SpecEditor.

To create a new long description style, click Save As and provide a name for your style. You may want to choose the name of a client, or your company if that is who the style will be used for.

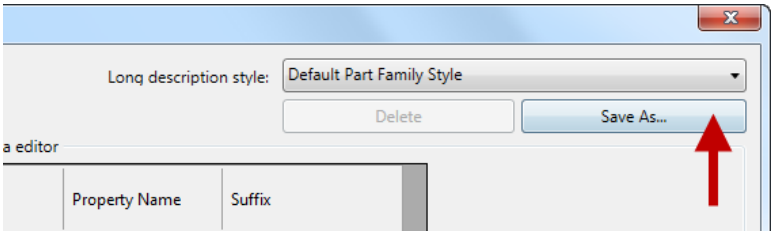


Figure 182: Save Long Description Style

Once your long description style is saved, you need to modify the properties for each class. One common modification is to use only spaces (no commas) to decrease the number of characters in the description. After modifying your long description style, click OK to return to the Layout and Settings dialog. Here you can choose a long description style to use for the Long Description Family or Size. Doing so and clicking OK will update the current spec description information. You can update a selection of specs by going to Specs > Batch Assign Long Description Styles.

Because the long description style is built from catalog properties, when creating and managing part families you will have to track which properties are used in your style, and make sure they are filled out. Establishing long description styles lets you quickly modify and generate specs for clients.

Custom Part vs. Script

In some cases, AutoCAD Plant 3D may not contain a shape or script similar enough to what you need modeled. This article shows steps to making a custom component.

- http://www.pdoteam.com/2011/10/custom-components-in-autocad-plant-3d/

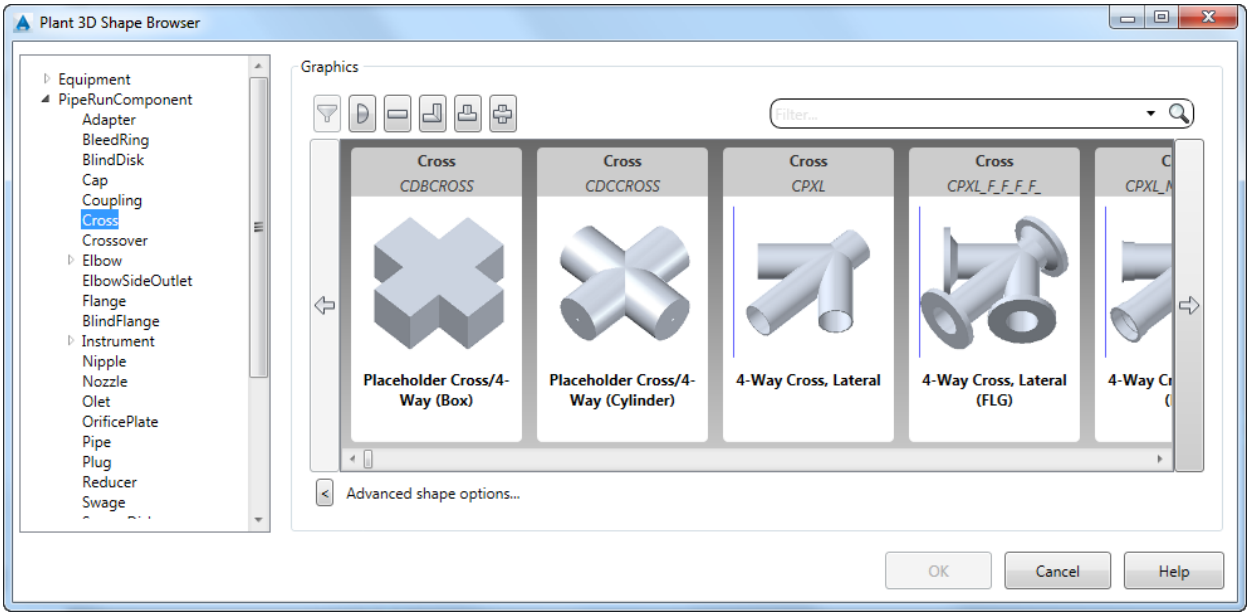


Figure 183: Scripts

Weights

If you need to use weights, make sure the items coming from the default catalogs have weights assigned to them.

Deploy Content

Many users want to keep all the content on the network. In this case, the content is considered deployed by setting the shared content path on the user's computer.

In other cases, users need to be able to disconnect from the network. In others if network performance is poor, users should be able to run the content off of their local computer.

- For these scenarios, you can use batch scripts or a program like Microsoft SyncToy: http://www.microsoft.com/en-us/download/details.aspx?id=15155

Custom Properties

Previously you learned how properties can be defined in the catalog and spec, and have the values populate a matching project property. In order to create those properties, you will have to modify the catalog and spec. Note that if you are modifying properties after a project has been started, you should export the data to Excel, so that you can bring it back in without losing it. Removing properties from a catalog, spec, or project removes the value, so the Excel export is a good way to make a backup.

To add or modify a custom catalog property, open the catalog in the catalog editor tab, go to the Catalog menu and choose Modify Catalog properties. Make sure to note that the first field is the Display Name which may include spaces.

The fourth field is the Field name which should not be specific to a single part group or size. Generally, creating the property so that it applies to all part groups is best. Fill out the fields, and click Add.

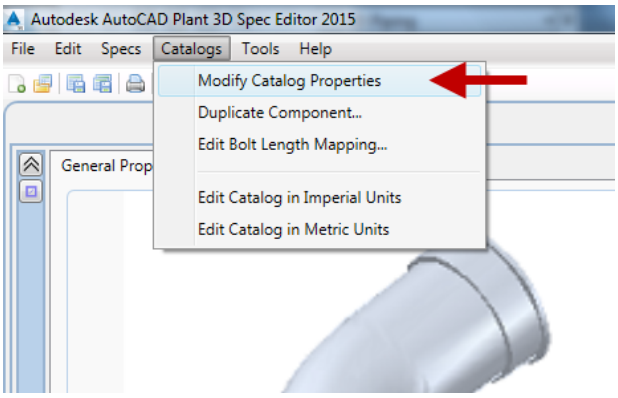


Figure 184: Add Catalog Property

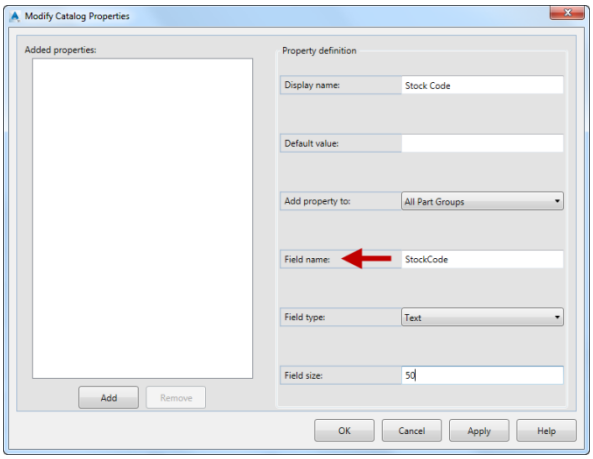


Figure 185: Set Field Name

After you have added your properties, click OK. Then in the catalog editor under the far right panel, Piping Component properties, you should be able to locate your custom property for editing.

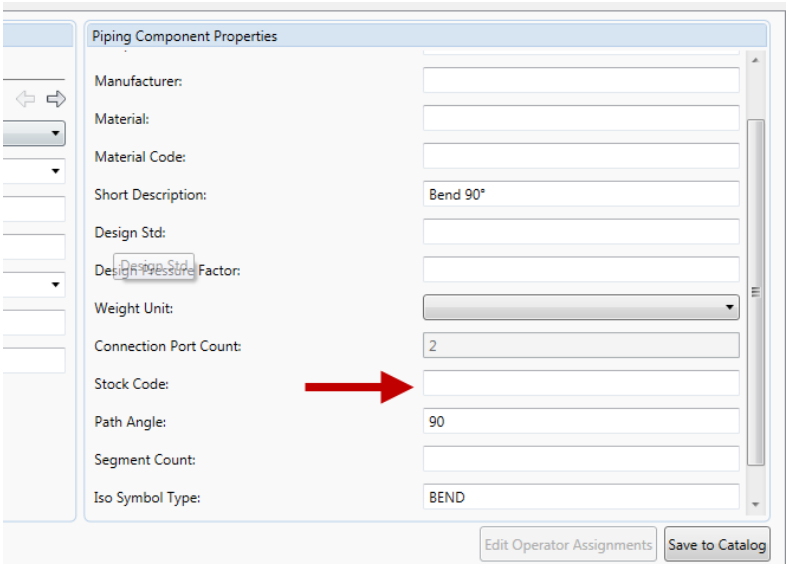


Figure 186: Catalog Property in Editor

To add a custom spec property, open the spec that needs to have the property. After the spec has opened, right click in the spec part list, and choose Edit Properties.

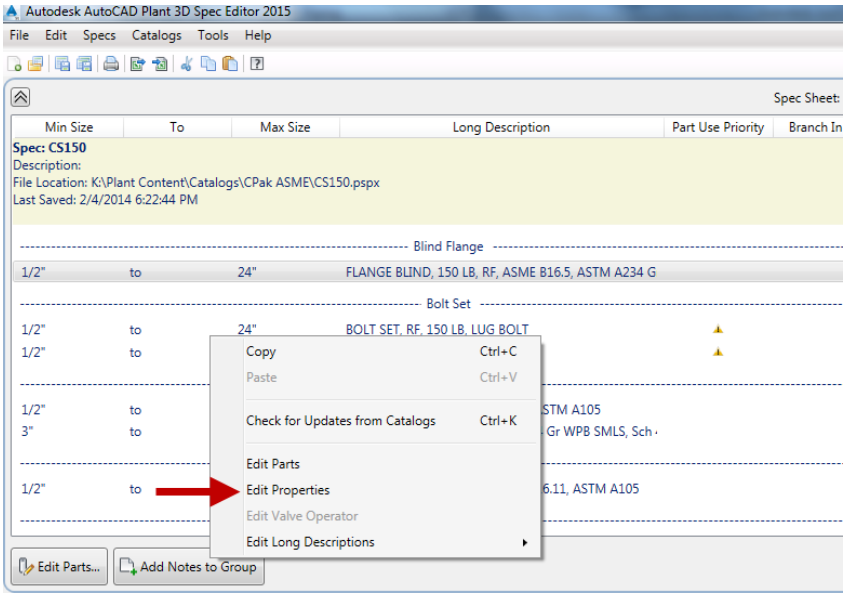


Figure 187: Create Custom Spec Property

Here you follow the same steps as in the catalog to create the custom property.

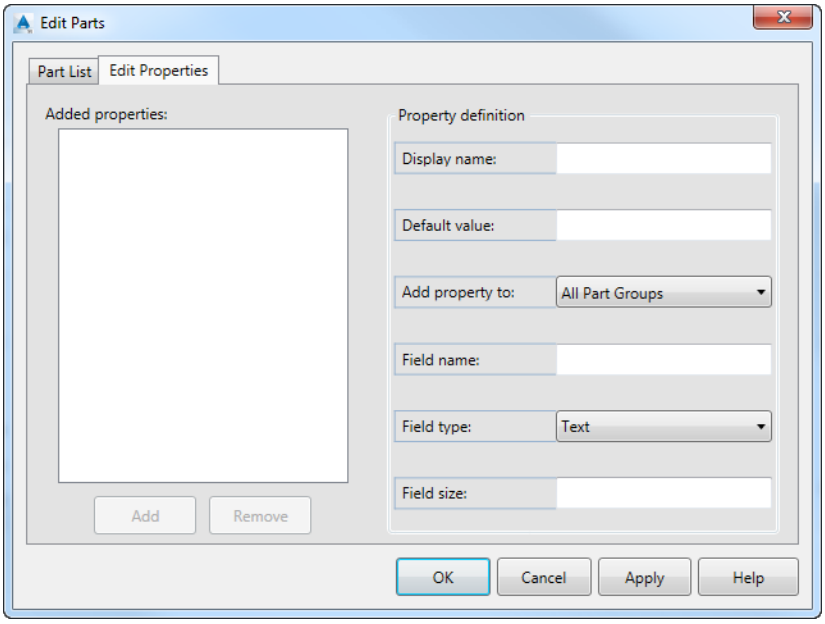


Figure 188: Add Property Dialog

After adding the property, you can edit it like any other spec property. Also, there is an easy filter to use to show only user properties.

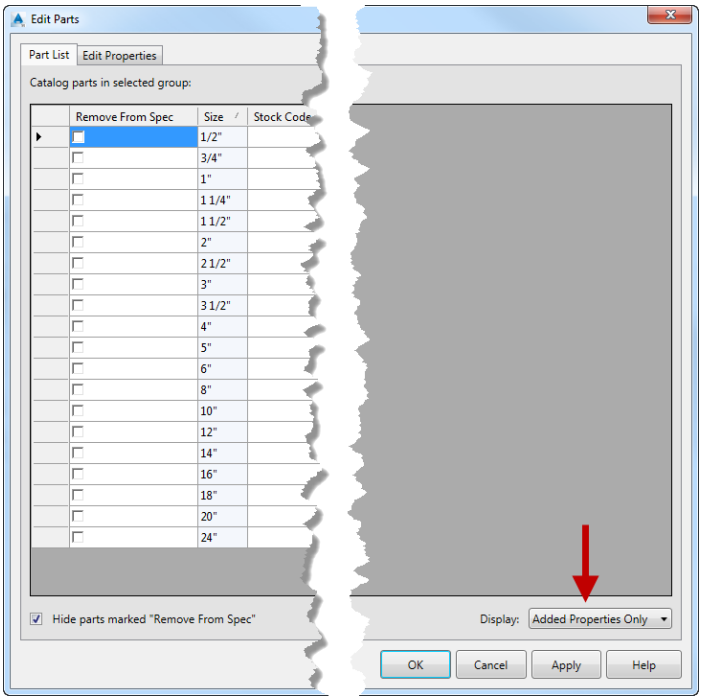


Figure 189: Filter to View Custom Properties

The next step is to make sure you have the property included in your project setup. You can create the property on the Piping and Equipment class, or you will need to create it for both the Fasteners and Pipe Run Component classes.

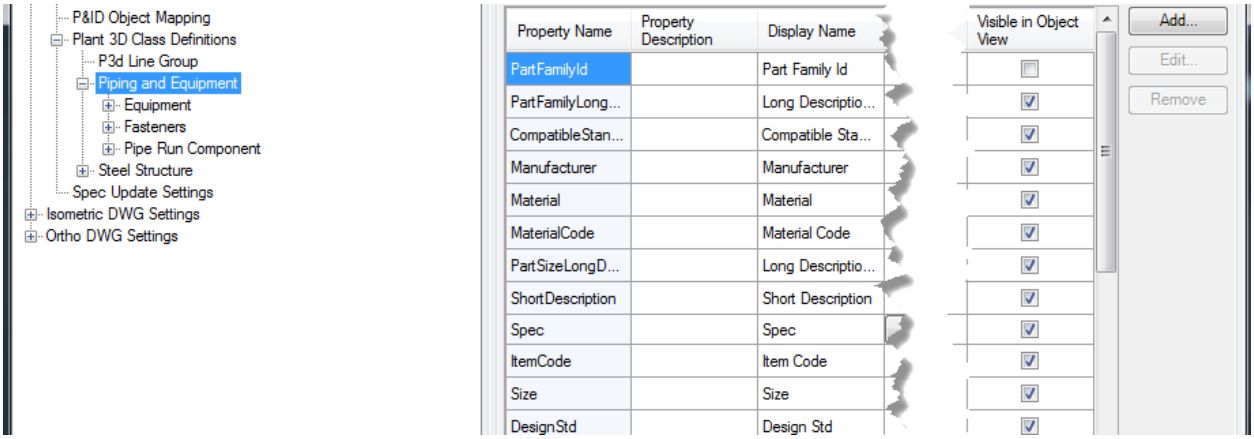


Figure 190: Custom Project Property

In this example the stock code was used like a part number. A new property was created because it will apply to the entire part family. The item code property can be used for part numbers that differ for every item with in a part family (unique for every size).

Sharing Tool Palettes

As with P&ID palettes, the plant palettes may be shared using standard AutoCAD methods. However plant also includes the ability to generate palettes automatically.

Currently, if you do use the dynamic palettes, they should only be stored on the local computer. The program will generate images for the palettes which may end up requiring a lot of disk space.

For best performance, turn off the dynamic palettes using the command PLANTDYNAMICTOOLPALETTE.

If you modify the support spec, you can enable the dynamic palette to get the supports palette generated so you can export it to the network if desired, but then turn it off to improve performance.

Joints

The other main area that integrates with the project settings is the joint configuration. Joints are stored in the DefaultConnectorsConfig.xml within each project. Joints are used to control how parts connect to each other by defining rules for the connections. The program uses two types of joints, simple and compound. Like equipment and nozzles, joints (or connectors) use nested parts (or subparts) to control several objects in one group. To select individual parts of a joint, hold down the Ctrl key and left-click on an object. You can also select multiple joints by holding the Ctrl key and using window selection. Before setting up a joint connection, you need to think through how the connection will work. In the examples below, you will see how some of the default joints are defined, and how to identify the critical aspects of a joint.

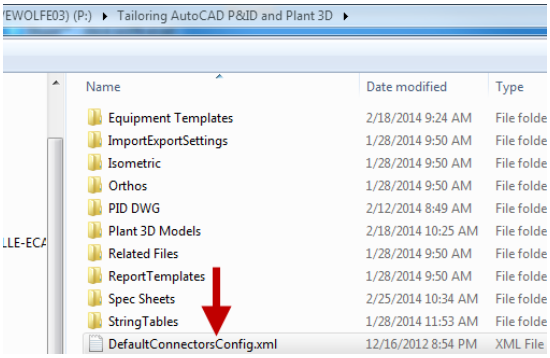


Figure 191 Joint definitions

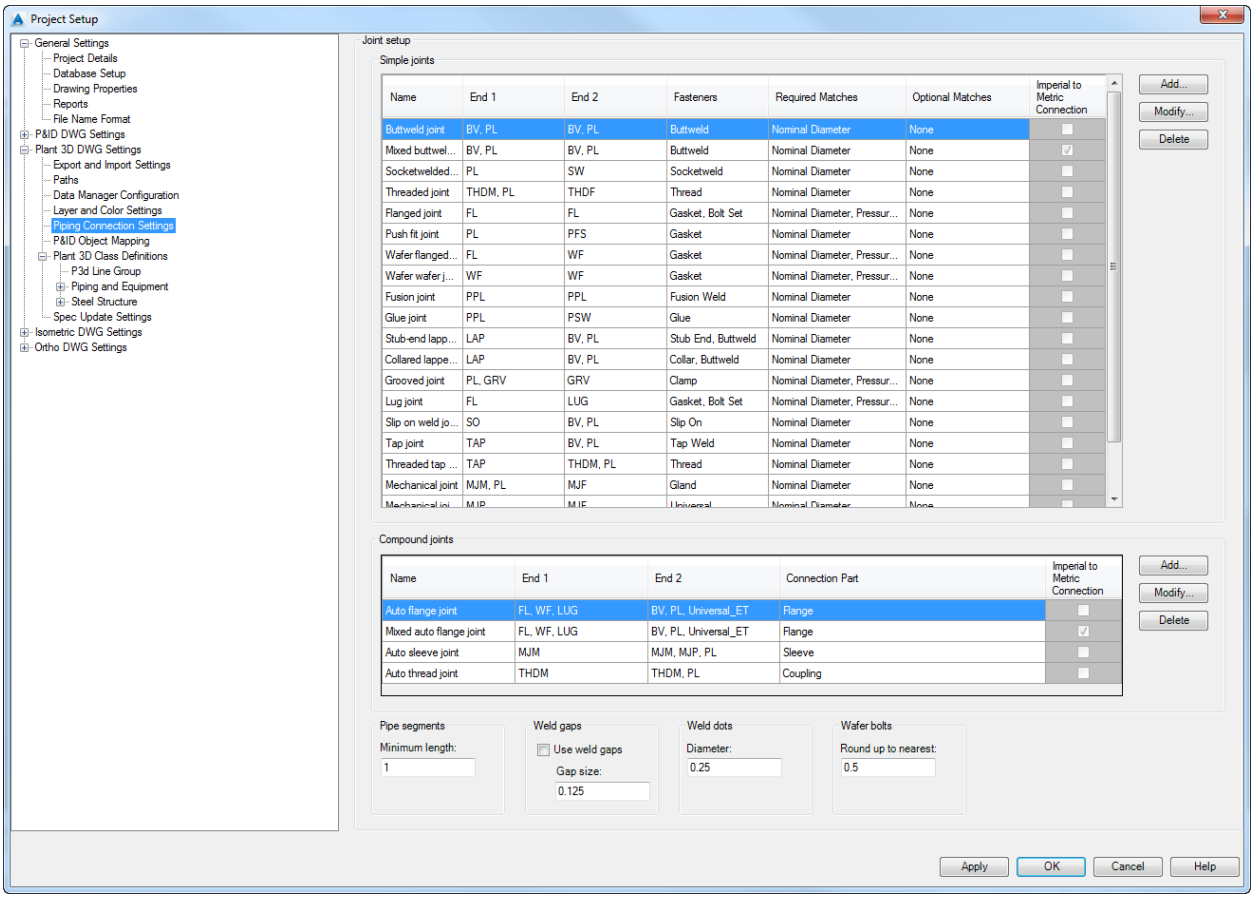
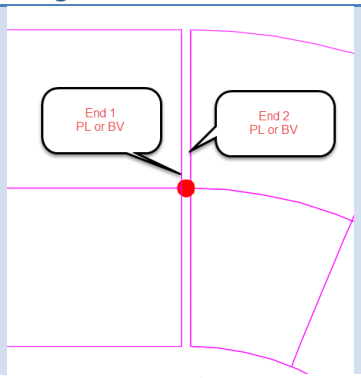
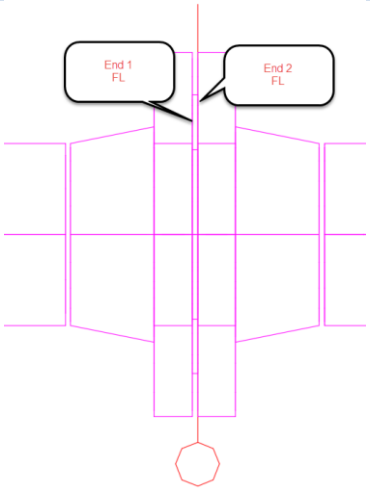


Figure 192: Joint Editor

Simple Joints

Examine the table below for information on how joints are configured. For simple joints, the ends of two parts are compared to the simple joint list. Then the program checks the required matches to see if the properties of the parts match. If the criteria are met, the program inserts the specified fastener.

Joint Name	Settings	Image	Description
Buttweld	End1 = BV,PL End2 = BV,PL Fastener = Buttweld Matches = Nom. Dia		A PL end pipe is near a BV elbow. Because the nominal diameters match (Required Matches), the program inserts the fastener specified (Buttweld).
Flanged Joint	End 1 = FL End 2 = FL Fastener = Gasket, Bolt Matches = Press. Class, Nom. Dia, Facing		To flanges are near each other. The nom. Dia., pressure class, and facing must match for the program to insert the bolt and gasket.

When creating new joints, you should draw up the end types and compare the ports from each of the parts that will be involved in the joint (fasteners too).

Matching Values

Matching Values are only valid when inserting an object (like a valve). They are not used while routing piping. For example, if you click the + grip on a flange that doesn't match the current spec settings, you may get an error like this, even with matching values set.

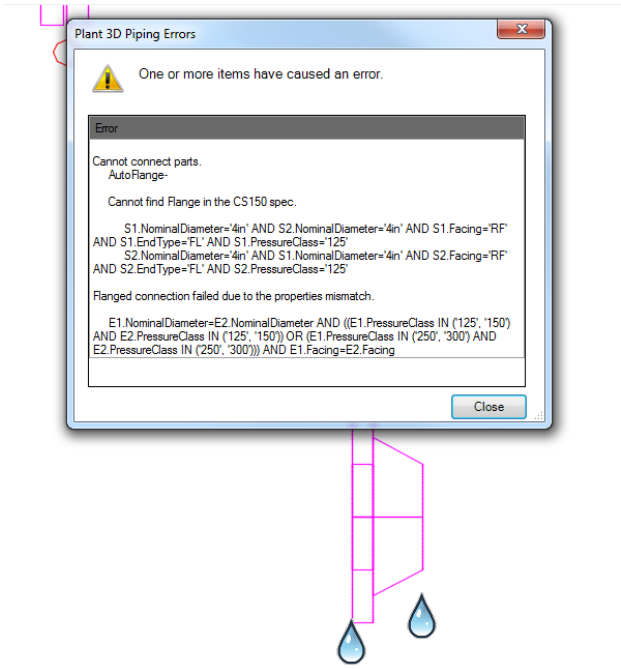


Figure 193: Joint Error

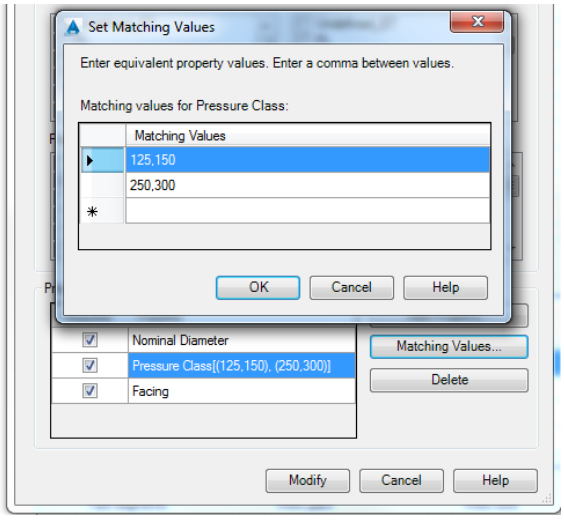


Figure 194: Matching Value Editor

However, if you place a valve with 125 lb flanges in a 150 lb line, the valve will connect using the right flanges.

Precision

Even though the Connectors config xml has a setting for offset tolerance, all joints require absolute precision. Meaning that to correctly verify the connection, you have to change your UNITS to decimal (not fractional), and set the precision to the hundred millionth (0.00000000). Then you can use the DIST command to measure the distance between two ports to make sure they are at the correct distance for the joint. The other most common issue is that users try to connect joints while routing pipe from BOP. Make sure that pipe is being routed via COP to make connections properly.

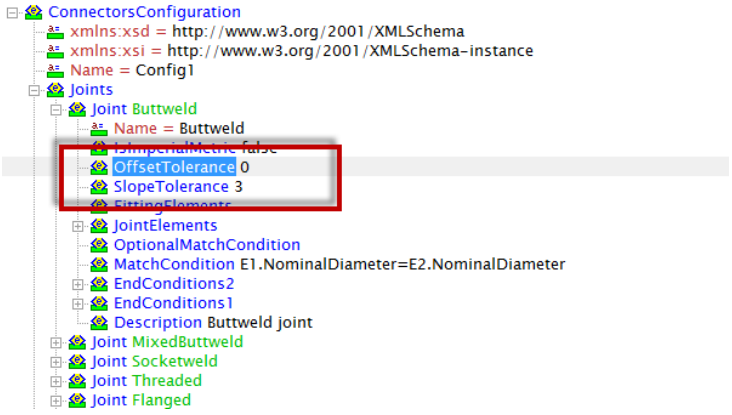


Figure 195: Tolerance

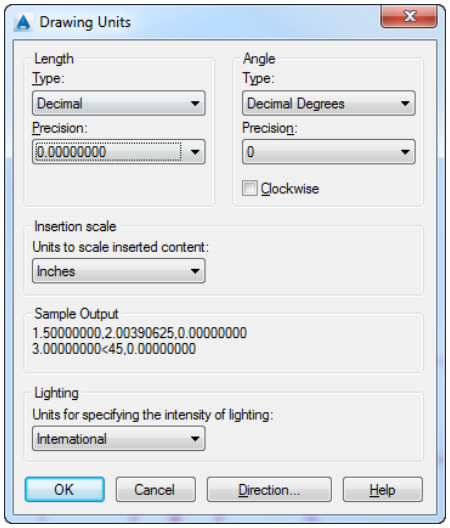


Figure 196: Model Precision

Compound Joints

Compound joints are used to insert a component to bridge gaps or connection differences. For example, when you place a flanged valve in a pipe, the program inserts a flange. The rules defined in the compound joints section indicate the part that should be inserted.

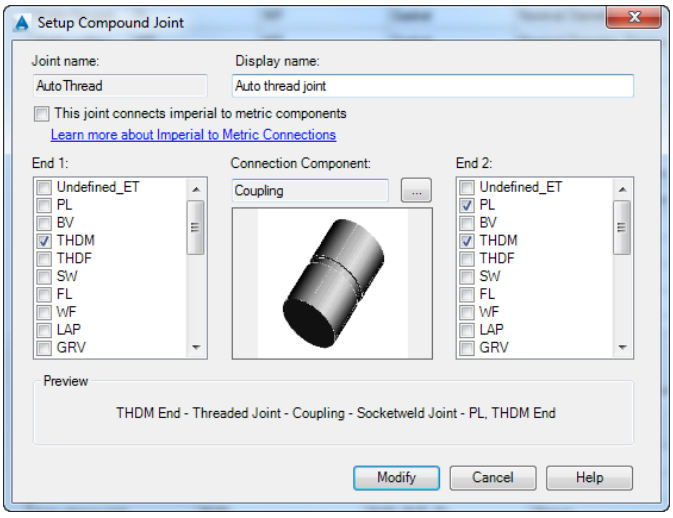
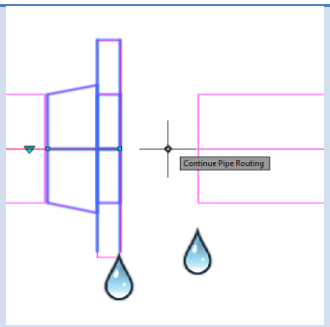
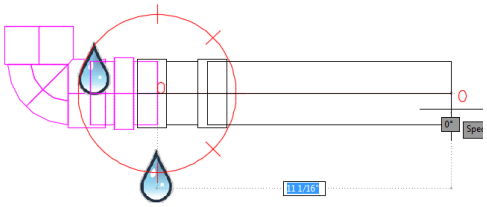


Figure 197: Compound Joint

Joint Name	Settings	Image	Description
Auto Flange	End 1 = FL, WF, LUG End 2 = BV, PL, Universal_ET Part = Flange		On the left side, we have a flange with an end type of FL. If the + is click, and routed to the pipe, the program tries to put in a pipe or elbow, [PL or BV] (depending on the turn) There is no BV to FL simple joint, so the compound joint list is checked. It locates the part to be inserted (Flange), and then creates the simple joints between all the parts.
Auto Thread	End 1 = THDM End 2 = THDM, PL Connection = Coupling		On the left is a nipple (THDM) and the + grip was clicked. The program puts in a pipe (PL). In order to connect a coupling is placed. The parts are then connected using the simple joint settings.

Compound joints may only insert 1 intermediate component, so you cannot combine multiple parts together in the joint (like you can fasteners).

Deliverables

The main goal of AutoCAD P&ID and AutoCAD Plant 3D is to create drawings that you can submit to your clients. This section will discuss how to tailor your deliverables to meet client needs. The information gathered during the first section will come into play here. By being able to answer questions like what layers should I use, or what format should my reports be in, you can decide which parts of the program need to be modified to match the current client needs.

P&IDs

For the most clients, the deliverable of the P&ID is a pdf, dwf, or dwg. In some cases however, you will have to provide and/or customize more information.

If clients use AutoCAD P&ID, you may be asked to send the project.

If a client doesn't use AutoCAD P&ID, typically they just want the drawings. You can send AutoCAD P&ID drawings which require an object enabler (drawings are read-only), or you can export the drawings to AutoCAD which will make all the objects standard AutoCAD block references.

To export a drawing, open it. In the project manager, right-click on the drawing and select Export to AutoCAD.

DWFs

To allow clients without AutoCAD P&ID access to the intelligence, you can publish the P&IDs to a DWF or DWFx format. Click the publish options dialog to make sure the P&ID information is included, and specify the type of file that will be created.

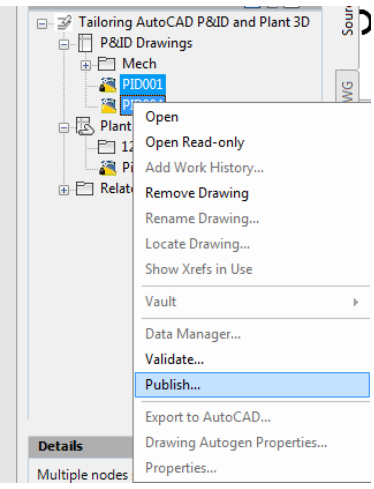


Figure 198: Project Manager Publish

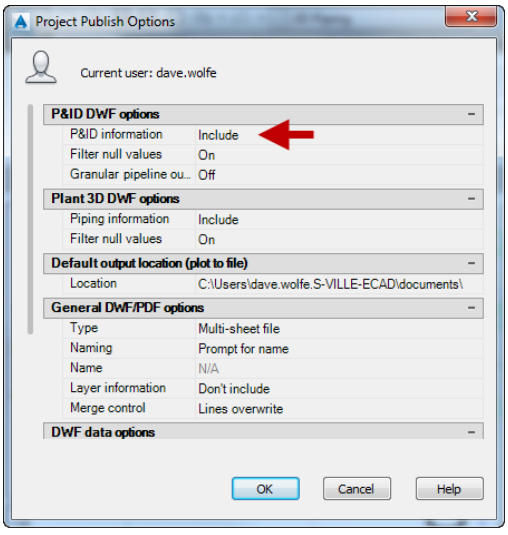


Figure 199: Include Data

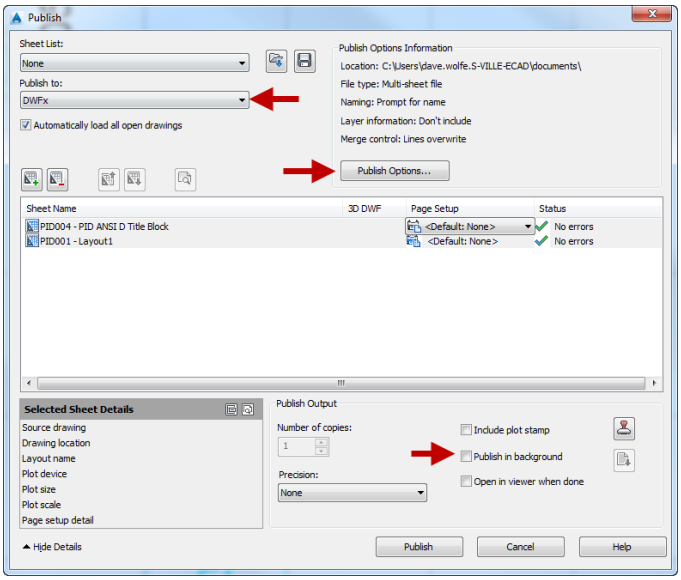


Figure 200: Publish

The published file can be opened in Autodesk Design Review (2014-2010), Autodesk 360, and Navisworks.

Title Blocks

In the P&IDs, title blocks can display project and drawing information whether using a default field or custom properties (see Project and Drawing Properties).

To set up a P&ID or orthographic title block follow the following steps:

- Setup project and drawing properties that match your title block needs
- Modify your title block by inserting fields into attributes or text.
- Insert the title block into a current project drawing.
- Fill out all of the drawing and project properties with a space, " ", for the attribute values, and save the drawing.
- Use Save As to save the drawing as a .dwt file.

Fields may be inserted into text or attributes by right-clicking and choosing insert field.

Orthographics

Orthographic drawings are another major deliverable in plant design. The orthographics generated from AutoCAD Plant 3D consist of flat shots of the model stored as block references in model space, and annotations, and dimension in paper space. These drawings do not require an object enabler for viewing inside of AutoCAD, but the objects in them can be annotated intelligently and dimensioned.

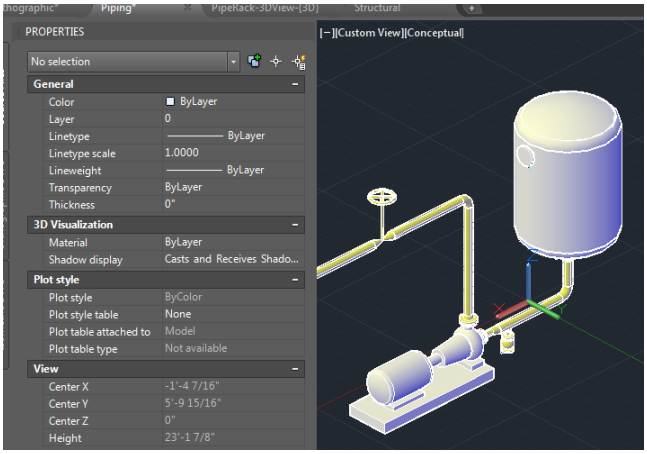


Figure 201: 3D Model

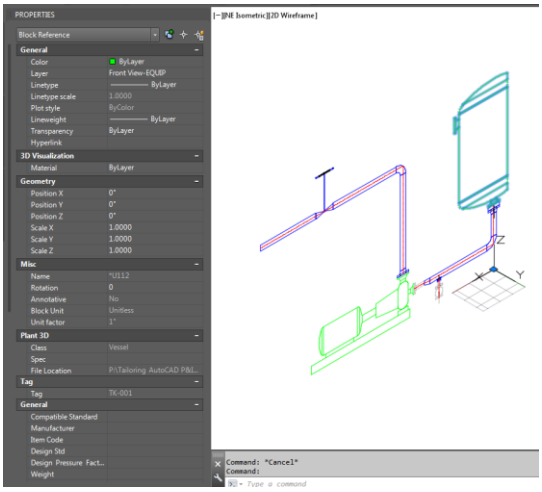


Figure 202: Ortho block in model space

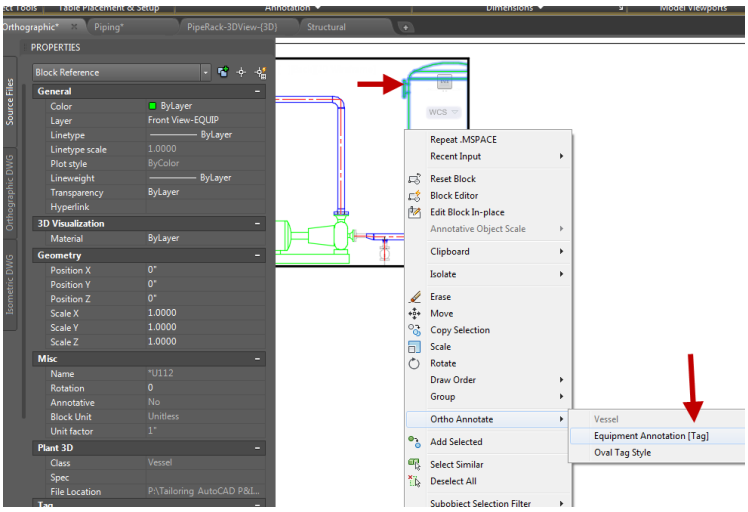


Figure 203: Annotating from Paper Space

Title blocks for orthographic drawings are set up using the same methods as P&ID title blocks.

To define how your orthographic will look you need to customize the orthographic settings.

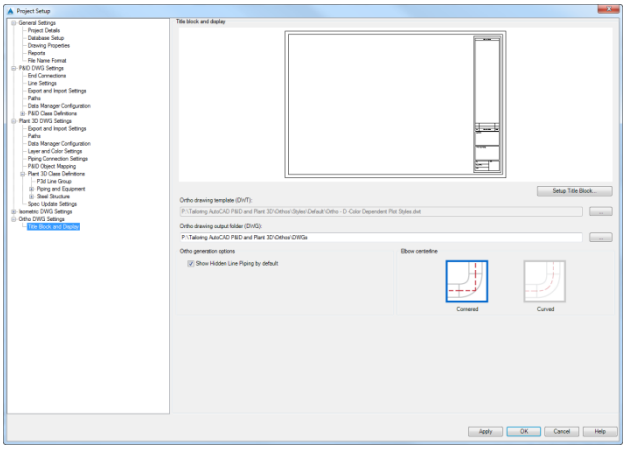


Figure 204: Ortho Settings

Templates

Your ortho settings are managed in the Ortho DWG Settings node in project setup. By clicking the Setup Title Block button, you will launch an editor that allows you to visually manage your drawing creation settings. In the main tab, you have an option of choosing a template, specifying how the elbows display as well as whether hidden lines are shown by default.

Using the ortho settings, you can define the overall appearance of your layout.

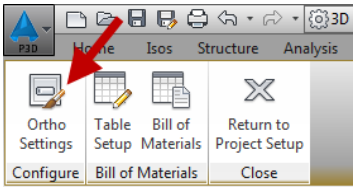


Figure 205: Ortho Ribbon Settings

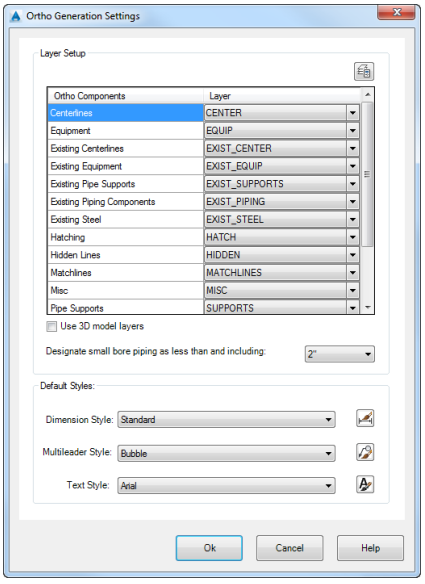


Figure 206: Ortho Layers and Settings

The layers, dimension styles, multi-leader styles, and text styles are defined in the template chosen. You can choose whether to have objects placed in pre-defined layers by object type, or use the layers used in

the model. You also have the option to place a BOM on the drawing. This BOM setup is similar to the isometric BOM setup, but you do have different options for the table type.

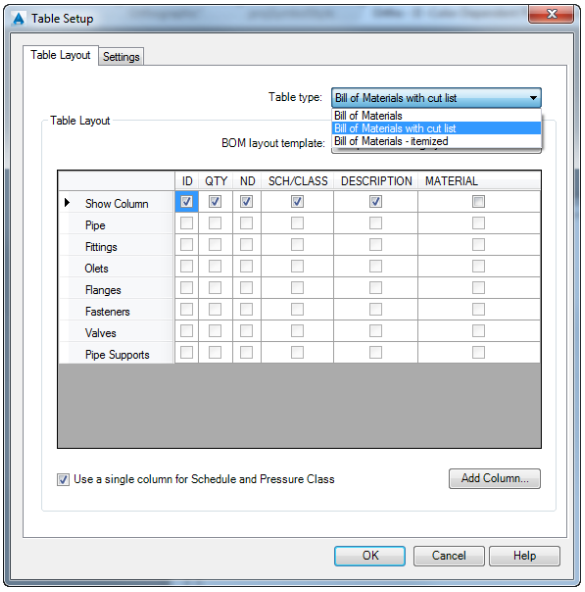


Figure 207: Ortho Tables

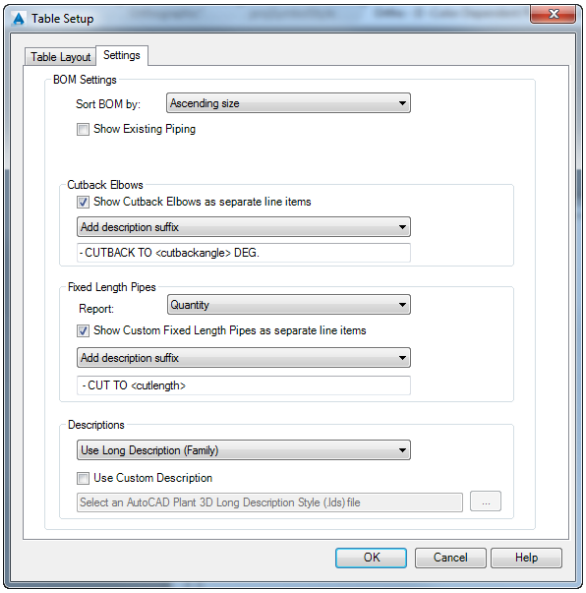


Figure 208: Ortho Table Settings

After selecting your layer options, and placing a BOM if necessary, the customization is complete for the template. The rest of the customizations are done when the views are generated, though you still have options for customizing some of the blocks used in the views.

- http://www.pdoteam.com/2013/04/modifying-the-pipe-gap-in-plant-3d-2014/

Annotations

Annotations for orthographic drawings are defined on the AutoCAD Plant 3D class, and created like the P&ID annotations.

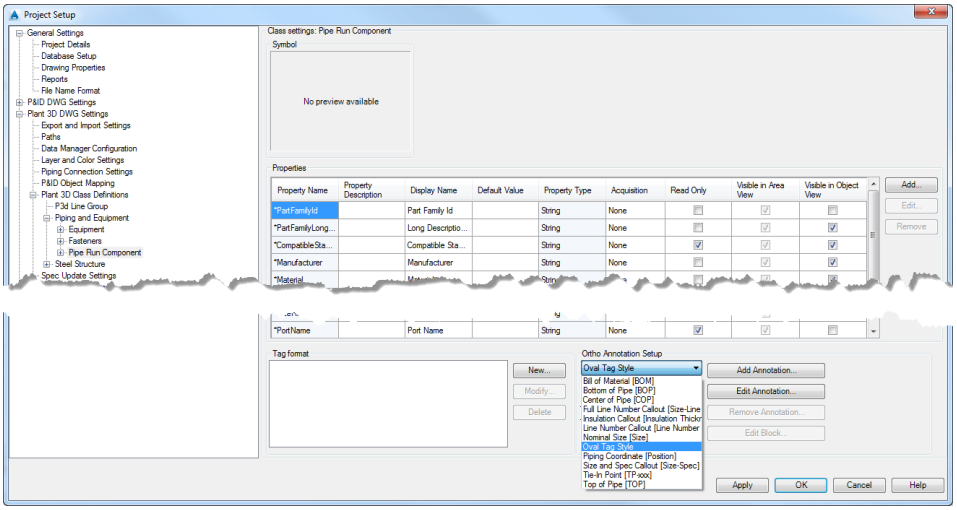


Figure 209: Ortho Annotations

Isometrics

Isometrics are another type of deliverable within a plant design package. AutoCAD Plant 3D isometrics are snapshots of the models and are created using styles. The style controls the template, layering, colors, appearance and content of the isometric.

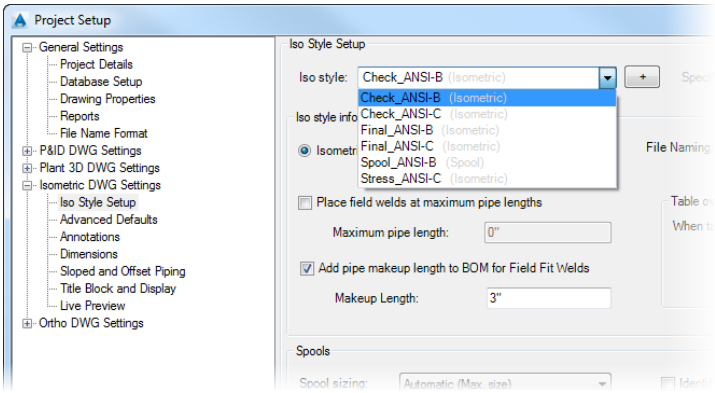


Figure 210: Isometric Styles

The Isometric folder in the project contains settings for creating Isometric drawings.

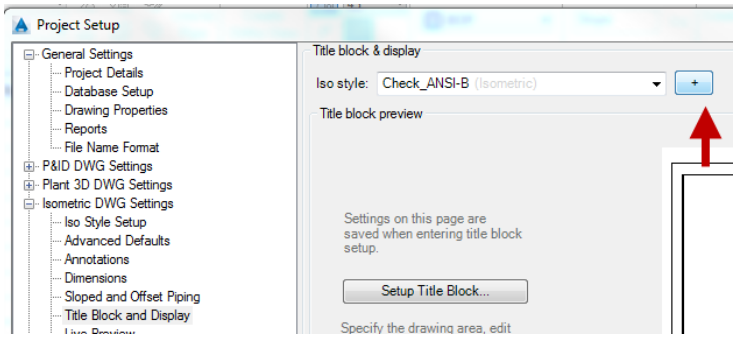
Style Management

The default styles are always included in a project as a sample from which you can build new styles. Styles may be copied between projects and the folder name is used for the style name in the project setup dialog.

To start customizing your isometric output, you should follow the screens in this order:

- Setup the Isometric Title Block
- Advanced Defaults
- Annotations
- Dimensions
- Sloped and Offset Piping

Add a New Style



You can add a new style by copying an existing one or using the project setup dialog. Enter the name of the style you want to create and select a style to base it on. A new folder with the default files will be created in the Isometrics directory.

Figure 211: Add Style

Title Block Setup

To modify the Title Block, go to Isometric DWG Settings > Title Block and Display. Click Setup Title Block.

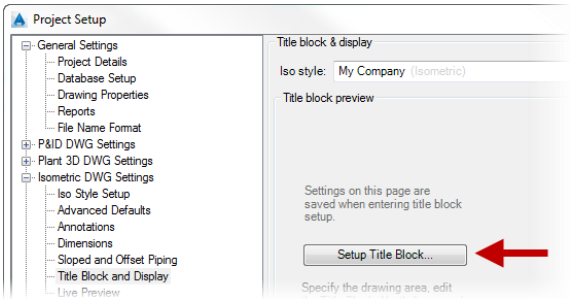


Figure 212: Setup Title Block

The program switches to edit the template defined in the settings. Along the ribbon menu are options for customizing your title block.

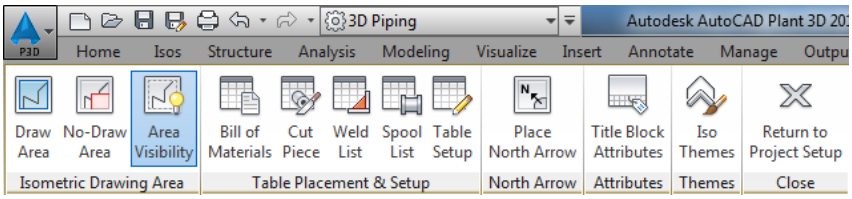


Figure 213: Isometric Customization Buttons

The first step is to replace the default title block with your own.

- Erase the existing title block
- Purge ONLY the block named “Title Block”
- Insert your own title block
- Use the RENAME command to change your title block to have the name “Title Block”.

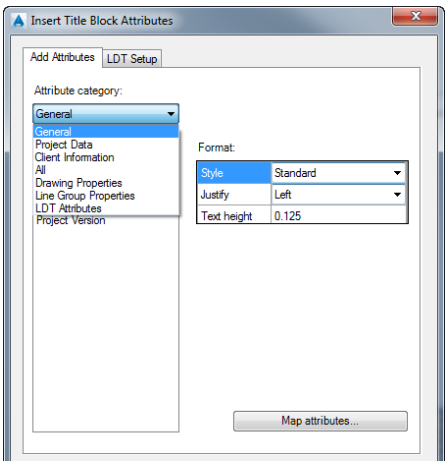


Figure 214 Attributes

To use the Title Block Attributes command, your title block must include attributes. Some companies insert an attributes block in addition to the title block. For those title blocks, the attributes should be copied into the title block. Once the title block is inserted, you can modify the title block attributes.

The AutoCAD Plant 3D Line Group class is the best place to add information to place in the title block. Isometrics are somewhat disconnected from the standard drawing properties, so the custom drawing properties are not available to isometrics. All project categories and custom project categories are available. See where to create custom properties under Project and Drawing Properties.

Advanced Defaults

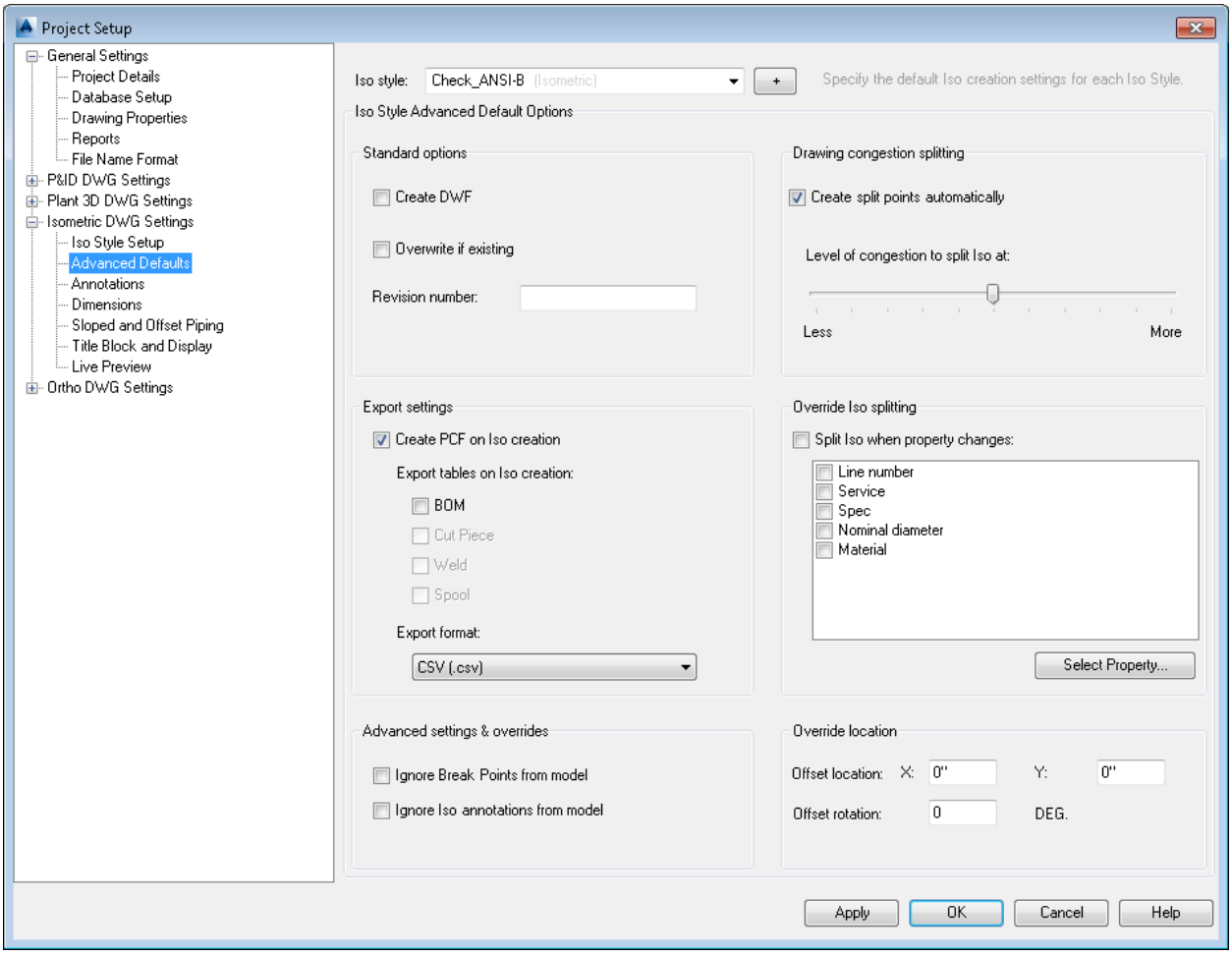


Figure 215: Advanced Defaults

In the Advanced Defaults screen you can define settings that previously could only be configured when creating an iso. With these settings, you can default a style that produces isometrics with coordinates that reflect a Plant or real-world coordinate system without modifying your model.

Please read this document for further information on styles and configuring them.

- De-mystifying AutoCAD Plant 3D Isometrics: a configuration reference

Text Styles

AutoCAD Isometrics can be customized to use your own text style. However, three different styles must be used in order for text to display correctly.

- Normal AutoCAD objects
- Annotations
- Skewed Annotations

Like dimension styles, annotations are stored in the isometric template.

Examining the bill of materials shows you that that the table uses the AdskIsoImperial text style.

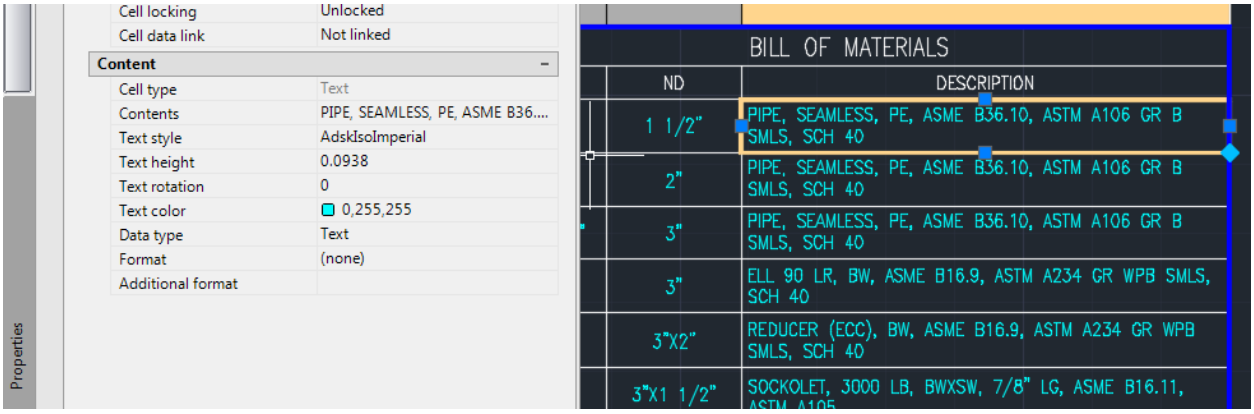


Figure 216: Default BOM Text Style

The text on the drawing uses the AdskIsoAnnotation text style.

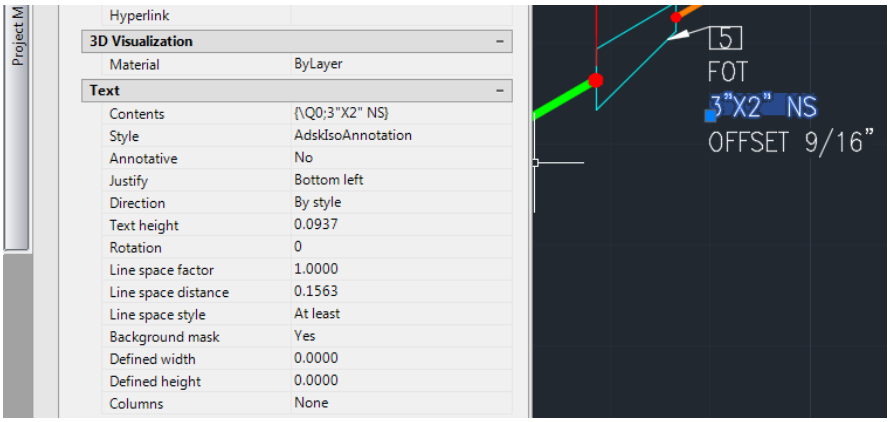


Figure 217: Annotation Text Style

However, the skewed text displays smaller than it should.

Figure 218: Skewed Text Style

The AdskIsoImperial text style's height may be set to any value desired. However, you should set the value in the text style. The AdskIsoAnnotation text style is set to 1.

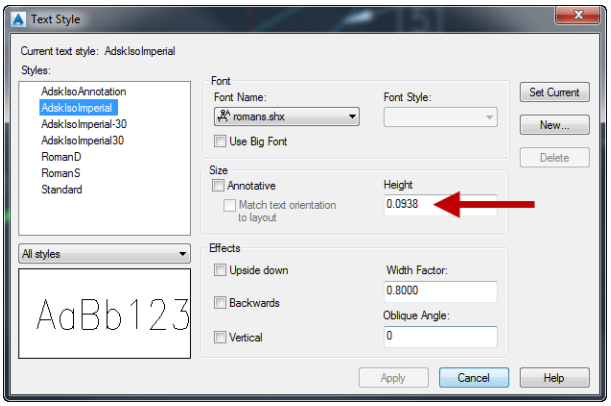


Figure 219: Default Text Style

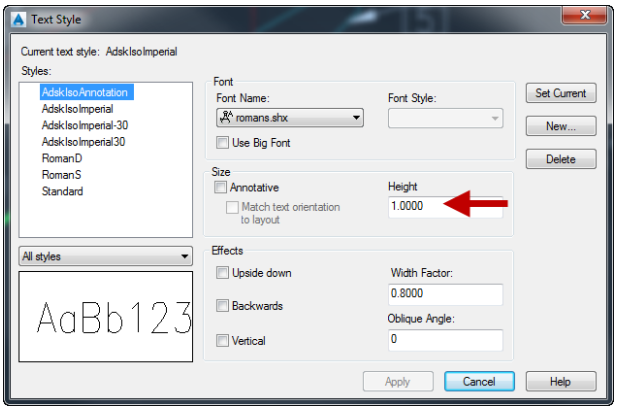


Figure 220: Annotation Text Style

The isoconfig.xml file contains a variable that is used to scale the text in annotations. The ScaleFactor setting indicates the size of the text height, and whether or not the size of the annotation is imperial or metric. For example, if the Scale factor is set to 1, all the symbols are also scaled up 25.4 units.

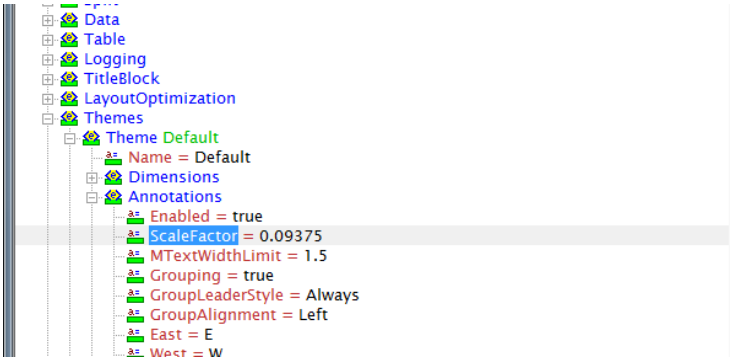


Figure 221: Annotation Scale Factor

Using a little math, you can find the ratio needed to output the skewed text at the correct size.

Create a text style called AdskIsoSkewed and set the text height to 1.4140 to have your text drawn at 3/32”.

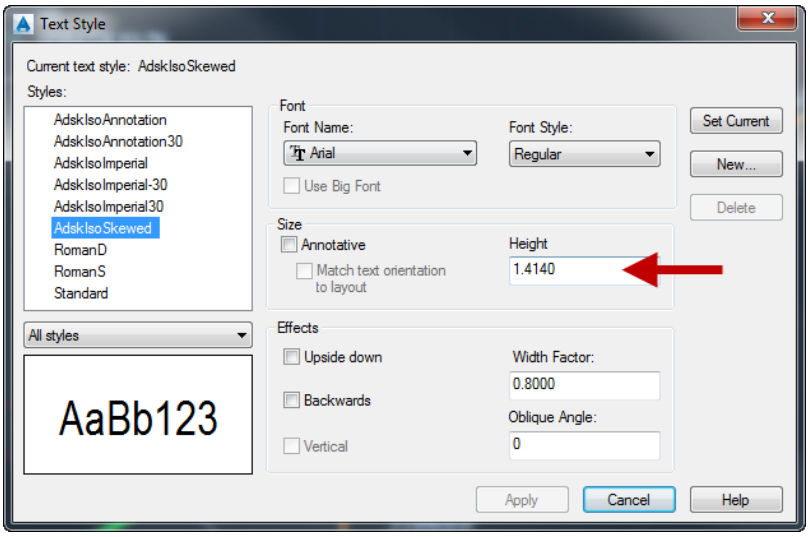


Figure 222: Skewed Text Style

The input value necessary for the text height is given by the formula:

X = T / 0.0663 where T is the desired text height, and X is the input value required.

In order to implement the skewed text style, you have to create an annotation style that utilizes it, and then reference the text style where you used skew alignment.

Locate the Standard Annotation Style in isoconfig.xml, copy it and paste a new one below it. Call the copied version Skewed. Set the Text Style to AdskIsoSkewed

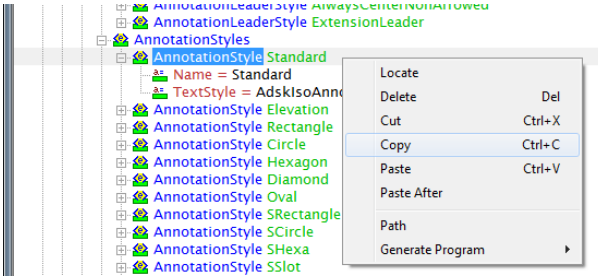


Figure 223: Copy Annotation Style

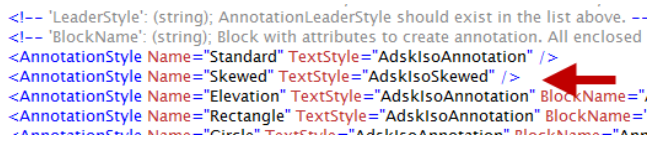


Figure 224: Create Skewed Annotation Style

The final step is modifying any annotation scheme that uses skew alignment to use the Skewed Annotation Style.

By default these annotation schemes use the Standard Annotation Style and should be change to use the Skewed annotation style:

- Line Number Scheme
- Field Weld
- Field Fit Weld

You may want to evaluate how these items appear to see if you want to change the text height for them:

- Elevation Extended
- Insulation
- Insulation With Tracing

In the screen shot below, a value of 1.8853 was entered so that a text height of .125 would be used for the line number. Also, the text styles were set to use an Arial font (True-Type fonts are searchable in pdfs).

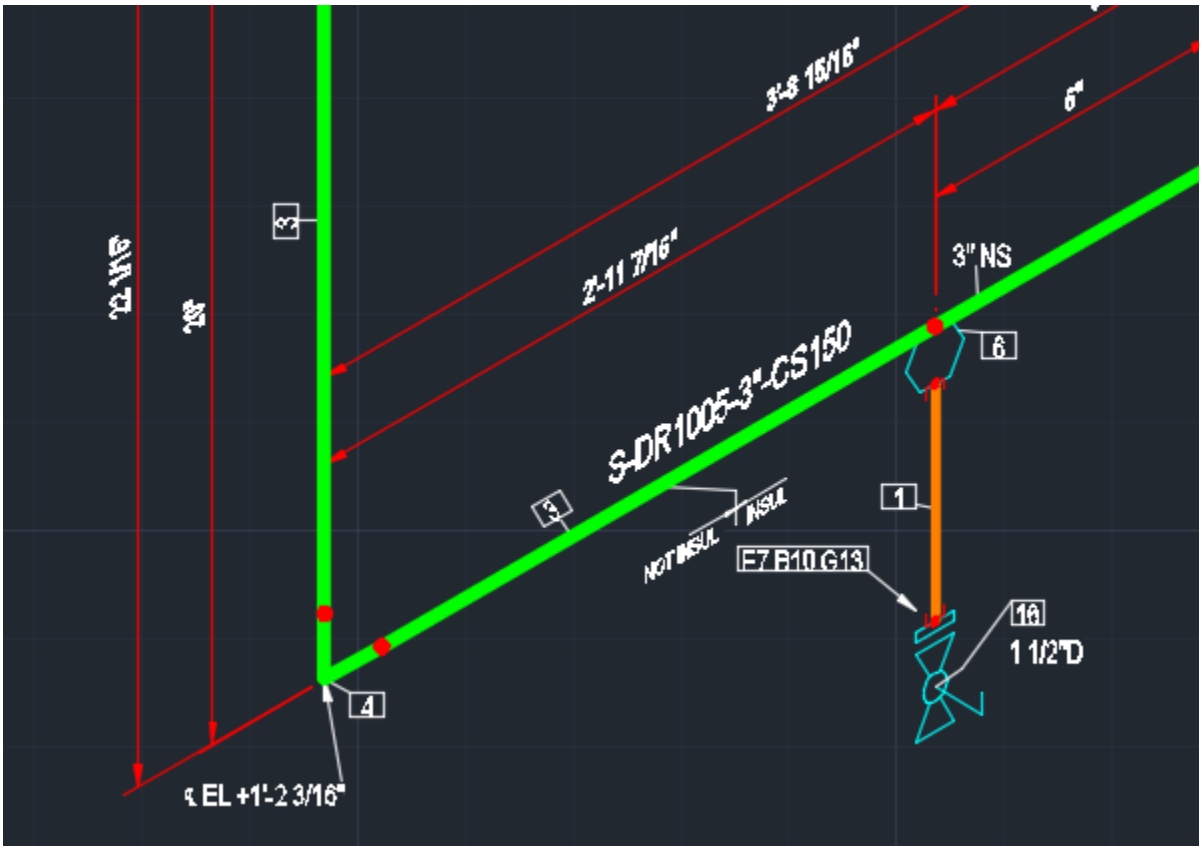


Figure 225: Skewed Text Style of 1/8"

Dimension Styles

You may customize the existing AdskIsoImperial dimension style or use your own. If you use your own dimension style, you need to bring it into the isometric template. One easy way to do this is to use Design Center (Ctrl+2) to import it.

Inserting a Dim Style using Design Center

To import the dimension style, navigate to a drawing that contains it in Design Center.

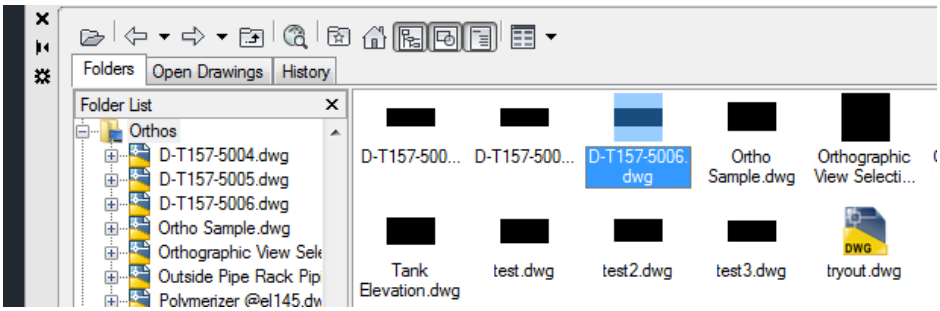


Figure 226: Design Center

Double-clicking on the drawing file will take you to a view of the object types within the drawing.

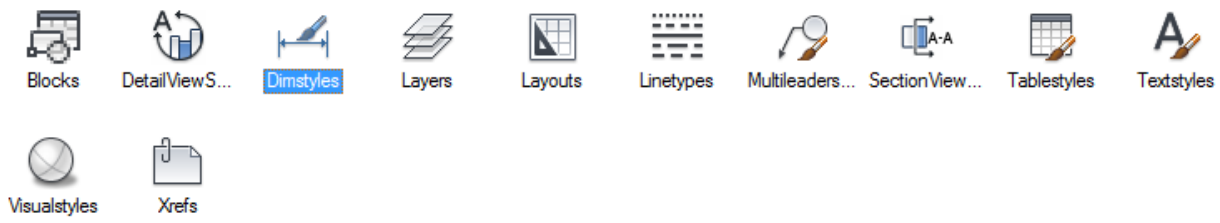


Figure 227: Dim Styles

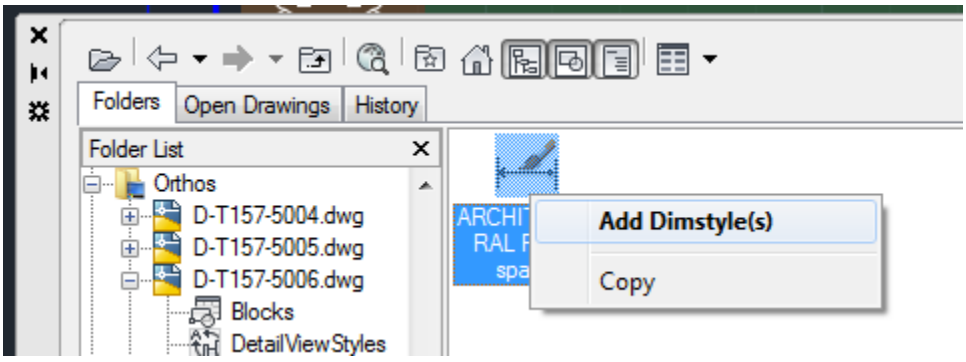


Figure 228: Add Dimstyle

Double-click on DimStyles and then right-click and choose Add Dimstyles.

Using a custom Dimension Style

To use the dimension style when creating isometrics, open the isoconfig.xml and navigate to Units > DimensionStyle. Enter the name of the dimension style to use in either the Imperial or Metric values.

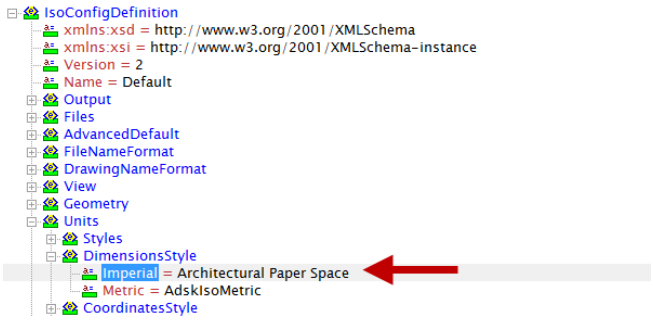


Figure 229: Set Custom Dimstyle

The final step is to set the Fraction Format to Not Stacked in your dimension style. Otherwise when the isometrics are generated, the fractions will not have a space between the whole number (inches) and the numerator (fraction).

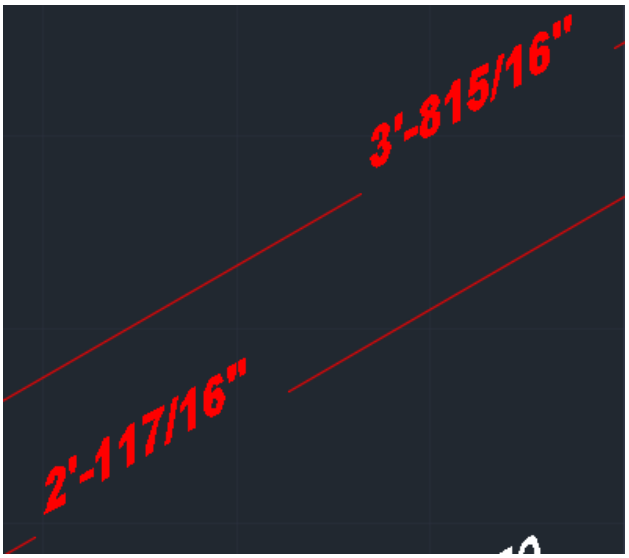


Figure 230: Wrong Fraction Format

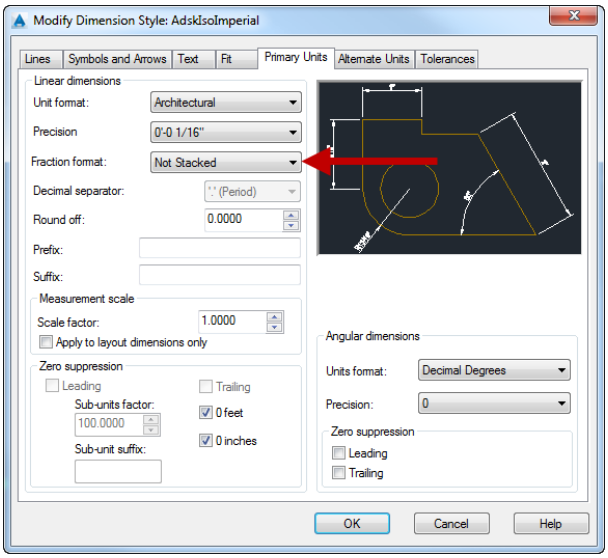


Figure 231: Not Stacked

Reports

AutoCAD P&ID and AutoCAD Plant 3D have two main ways to report properties and objects: the Data Manager and the Report Creator.

The Data Manager is a window into the project or current drawing data that lets you modify the properties without requiring object selection.

From any display in the data manager you can import or export data to Excel by using the Import/Export Settings, Reports, or Custom Views. The Report Creator however is used to export data only. If you want to issue a format that can be used to modify data, you should choose one of the Data Manager options. If you are issuing information that will not need to be imported, or needs to use sum or totaling functions (BOM), you should use the Report Creator.

The Report Creator is geared to creating polished reports with custom formatting, headers, fonts, totals, subtotals, and more. While powerful, by using report templates, it allows non-CAD users a way to view and generate reports.

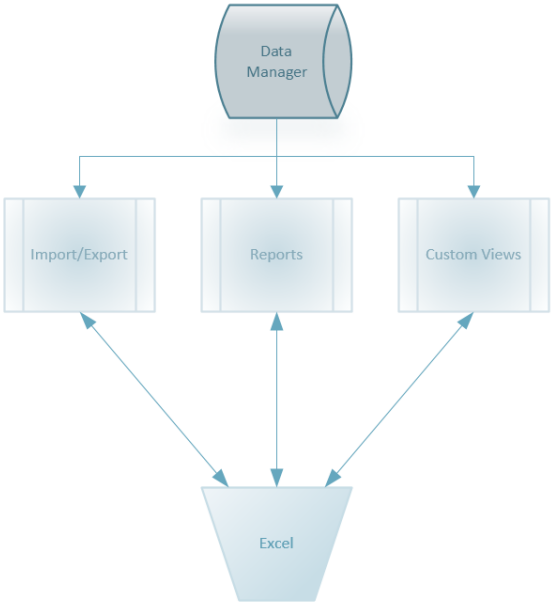


Figure 232: Data Manager Workflow

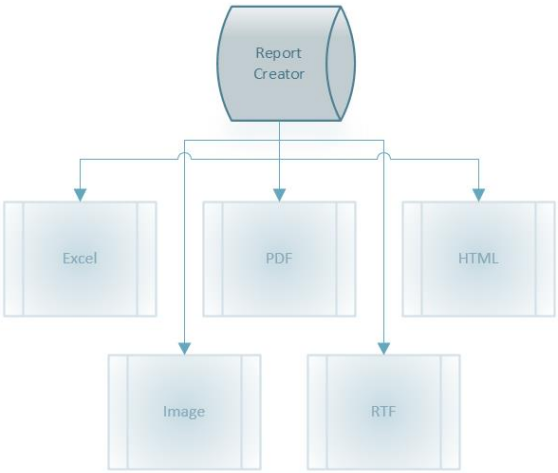


Figure 233: Report Creator Workflow

Data Manager Views

In addition to the standard object or Area views (AutoCAD Plant 3D), you can use Custom Views to change the item navigation tree.

You can setup a view to display items based on a common property.

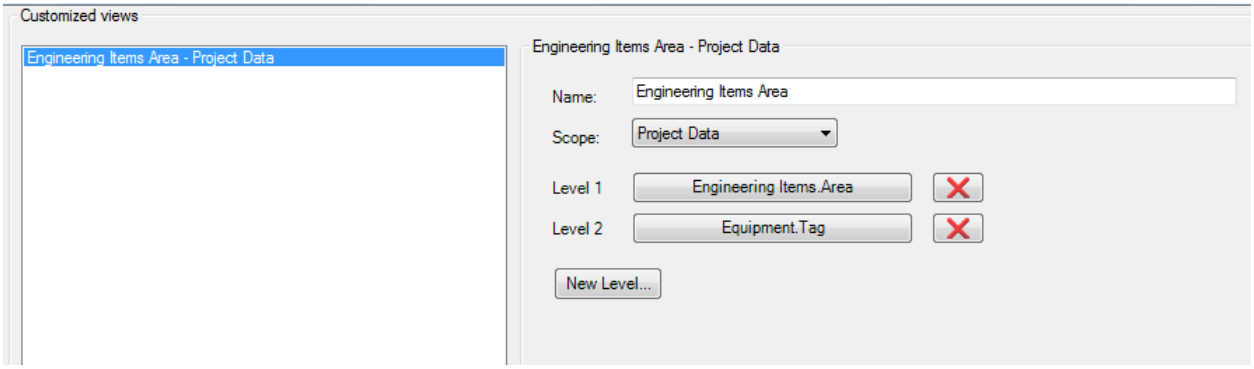


Figure 234: Custom View

Items will be grouped by the common property.

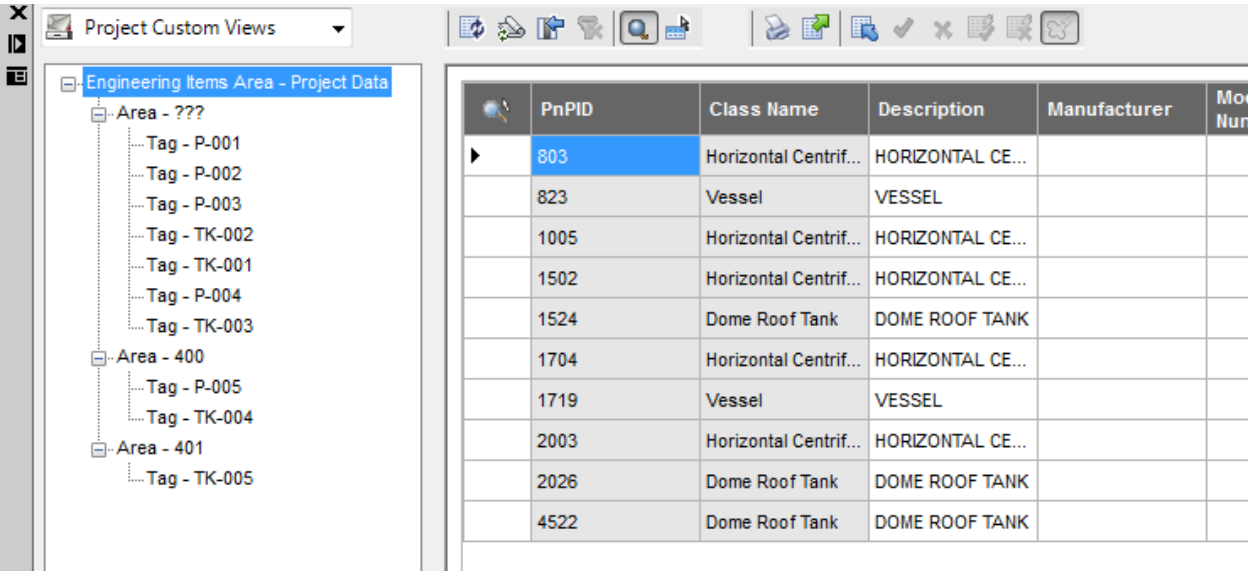


Figure 235: Grouping

Additional views will be added as nodes below the first one.

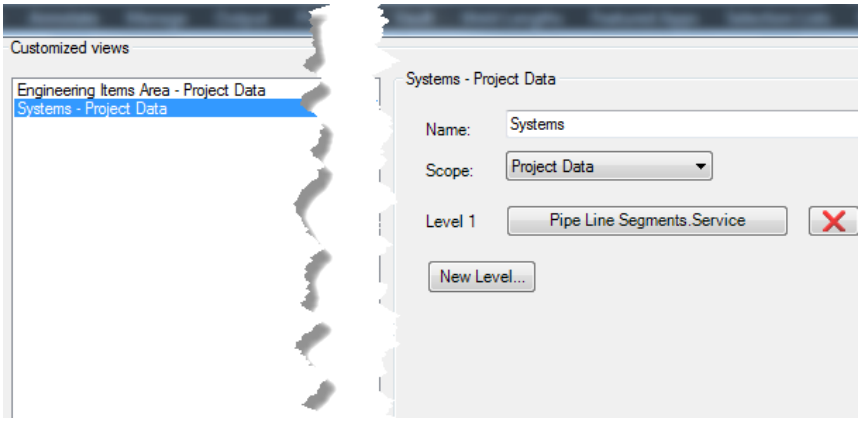


Figure 236: Multiple Views

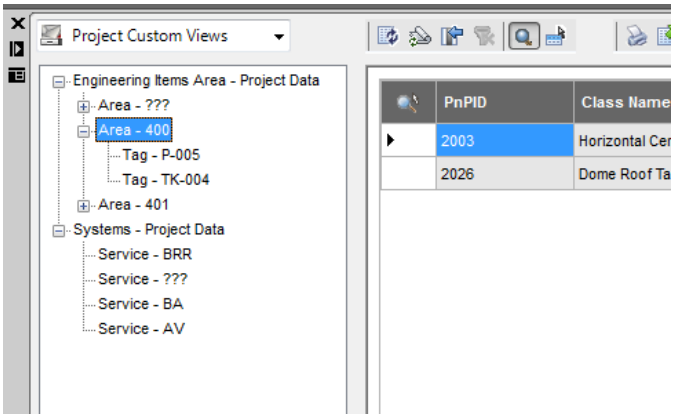


Figure 237: Areas

To group items by a Pipe Line Group property, create a property on the Pipe Line Segment class that acquires its value from the line group and use that property as the basis for the View.

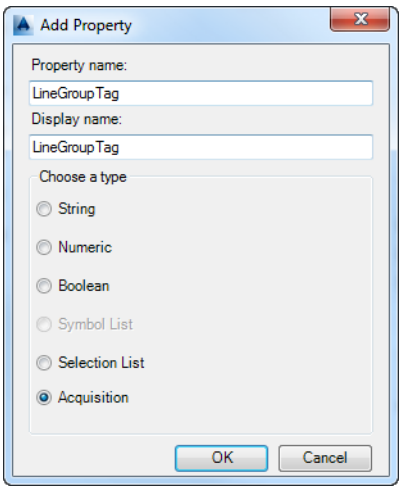


Figure 238: Items by Line Group

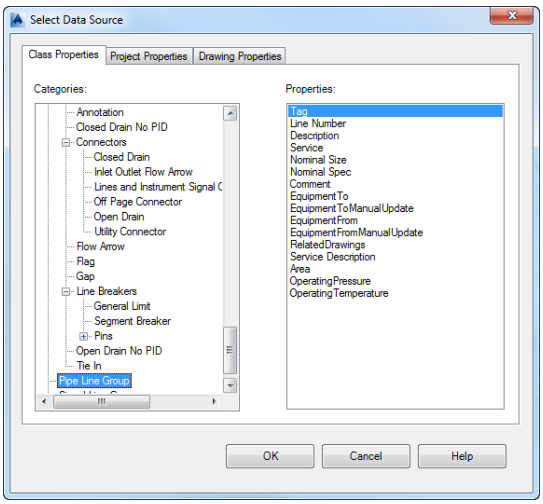


Figure 239: Acquire Line Group

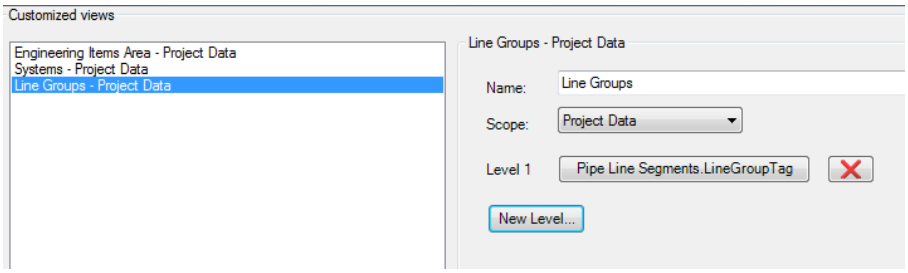


Figure 240: Line Group Items View

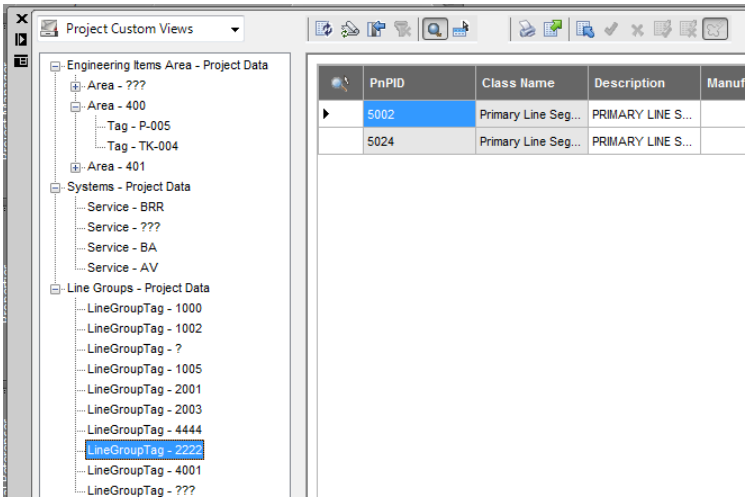


Figure 241: Data Manager View

Note that this still only displays the line segments. Line Groups themselves are not usable within custom views.

Custom views export the displayed data only using the Export/Import buttons in the data manager.

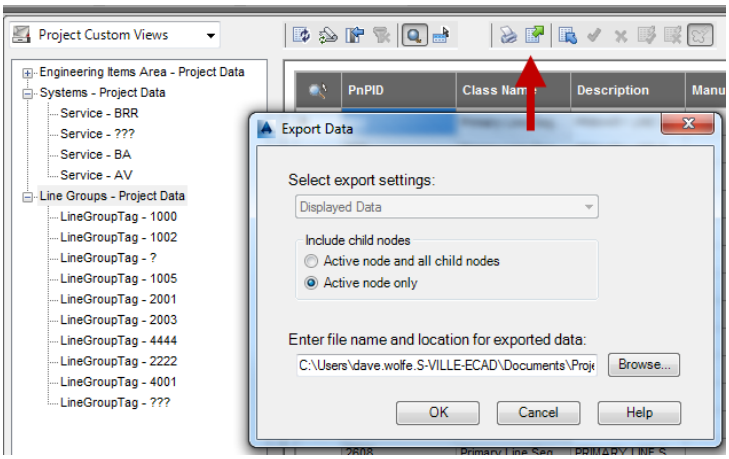


Figure 242: Export/Import Custom View

Import/Export

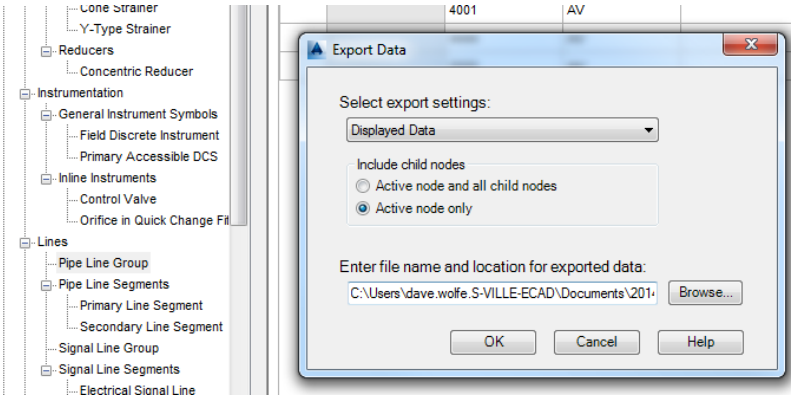


Figure 243 Import/Export Dialog

Both AutoCAD P&ID and AutoCAD Plant 3D can have custom import and export settings. These settings are used in conjunction with the normal import/export routine and allow customizing which fields are included and what property is used as a unique id. You can right click on a class, or click the Export or Import button to get to this dialog.

By default, Displayed Data is used for a filter. When exporting using displayed data, every property is exported for the current view and the PnPID is used as a unique id. Typically, you should export only the active node. Choosing the Include Active node and all child nodes option creates a single workbook with a worksheet for each child class. You should export the minimum amount of information necessary especially when working with large projects.

Setting up an Import Export Setting

You can access import and export settings in the project manager.

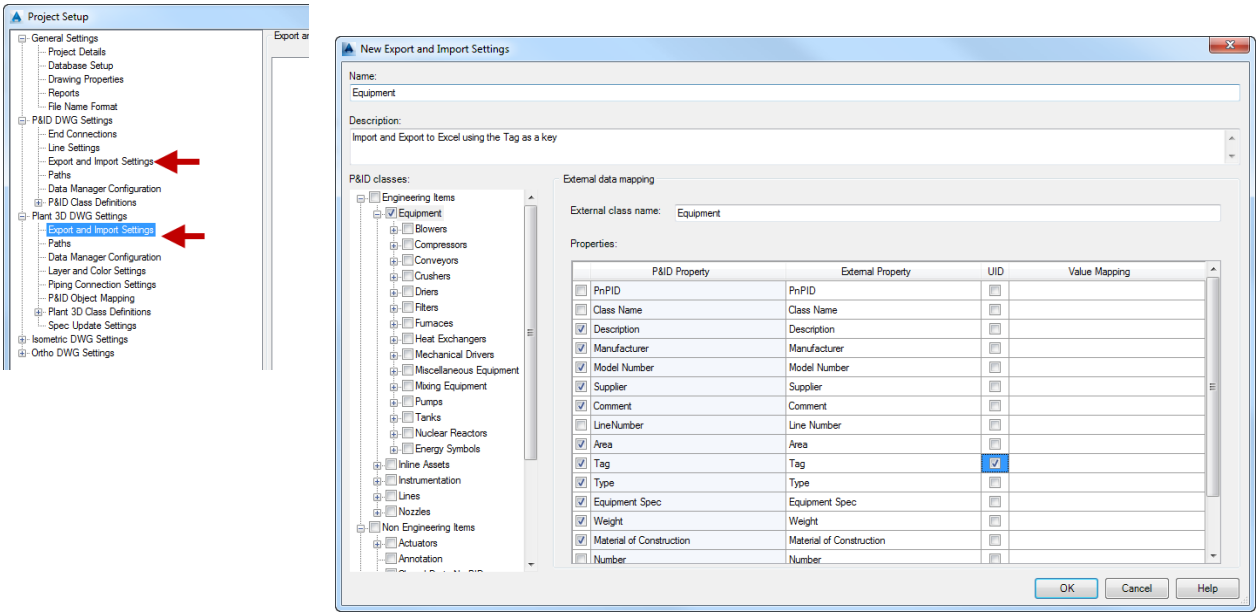


Figure 244: Settings for AutoCAD P&ID and AutoCAD Plant 3D

Setting up the Export Import settings, make a selection available when pick on the class in the Data Manager, and the exported Excel spreadsheet includes only the columns chosen.

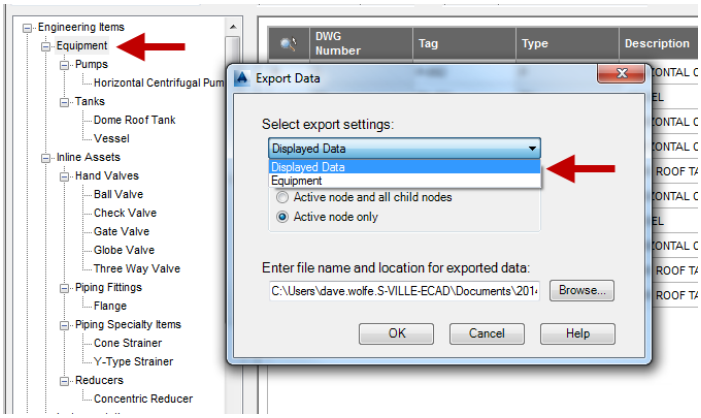


Figure 245: Select Export Settings

A	B	C	D	E	F	G	H	I	J	K
Tag	Type	Description	Manufacturer	Model Number	Supplier	Comment	Equipment Spec	Weight	Material of Construction	Area
P-002	P	HORIZONTAL CENTRIFUGAL PUMP								
TK-001	TK	VESSEL								
P-003	P	HORIZONTAL CENTRIFUGAL PUMP								
P-001	P	HORIZONTAL CENTRIFUGAL PUMP								
TK-002	TK	DOME ROOF TANK								
P-004	P	HORIZONTAL CENTRIFUGAL PUMP								
TK-003	TK	VESSEL								
P-005	P	HORIZONTAL CENTRIFUGAL PUMP								400
TK-004	TK	DOME ROOF TANK								400
TK-005	TK	DOME ROOF TANK								401

Figure 246: Custom Export

Data Manager Reports

The final way to output information from the Data Manager is using Data Manager Reports. Reports are available for AutoCAD P&ID items only. The default reports are:

- Control Valve List
- Document Registry
- Equipment List
- Instrument List
- Line List
- Line Summary List
- Nozzle List
- Specialty Items List
- Valve List

Reports are useful to customize which columns are displayed and exported/imported to Excel. You also have a good degree of freedom to use filters. While the data manager displays every property in the normal view, you can use reports to display specific sets of items and a handful of properties to make editing easier and faster.

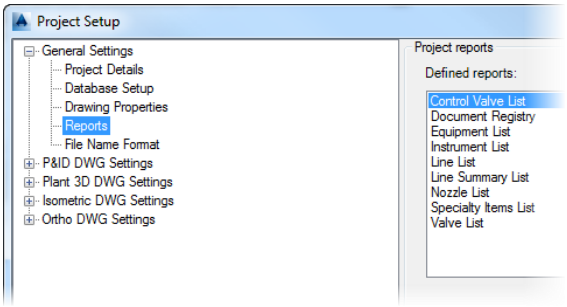


Figure 247: Data Manager Reports

Create a New List

To create a new list, click the New button. You can create a new report for flow meters by entering the report name and then choosing Replace table.

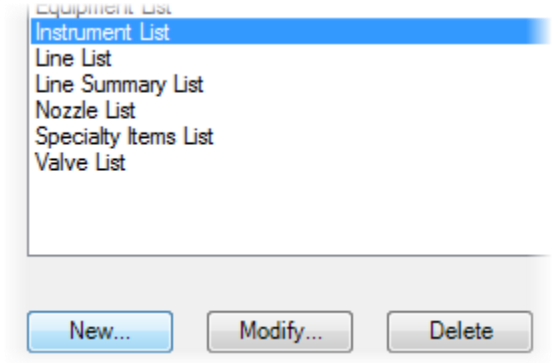


Figure 248: New List

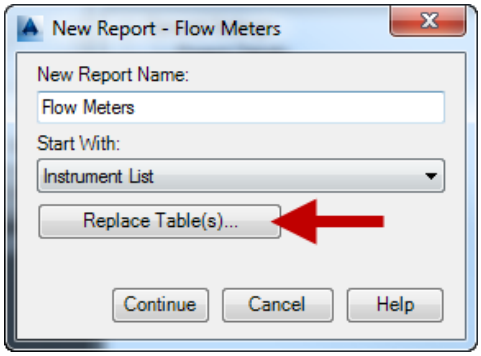


Figure 249: Replace Tables

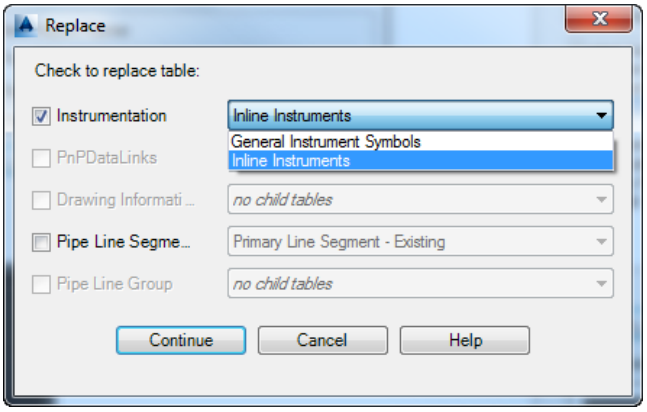


Figure 250: Select Child Table

In the Replace Table dialog, you can choose subclasses from the currently selected parent class. For the flow meters report, the Inline Instruments class was selected.

Choose the minimum number of columns to make editing easier. In this report, the line properties were given their own columns in order to allow users to filter and sort by individual properties more easily.

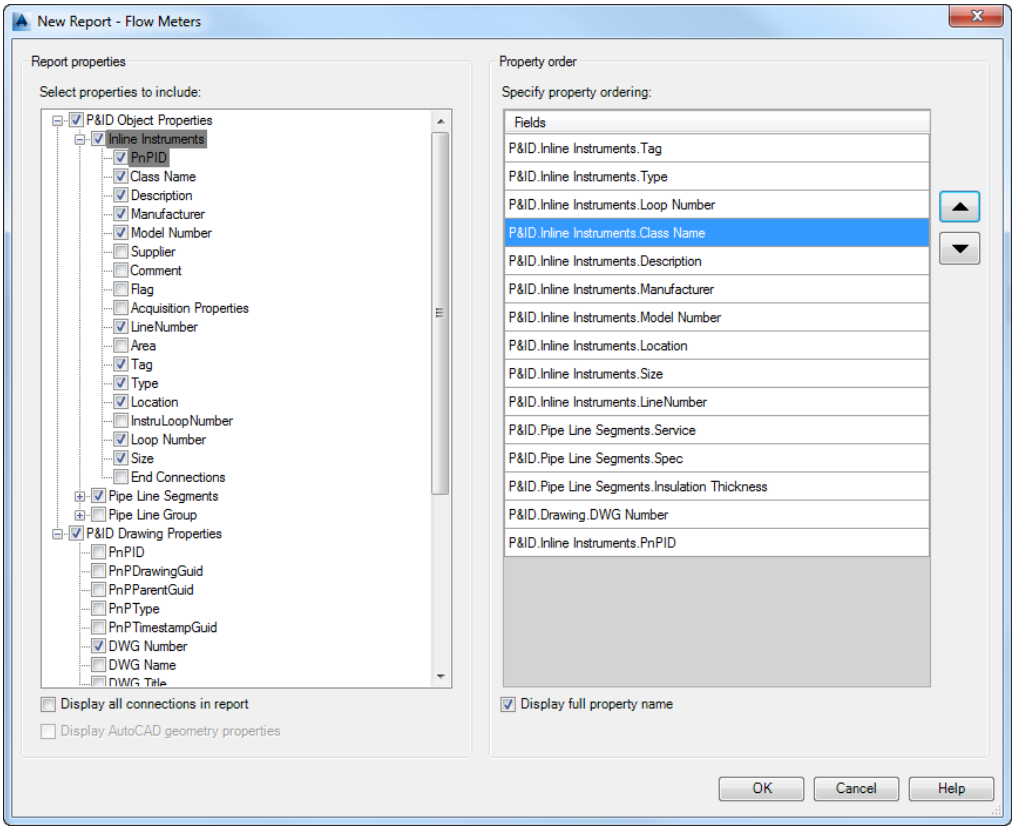


Figure 251: Column Selection and Ordering

In the Report preview, you can use filters to exclude or include items. In this example, the FE type is the only allowed Type. The most important step in this process is to **click save** after you have made the changes. If you do not click save, the report will use the default columns without filters.

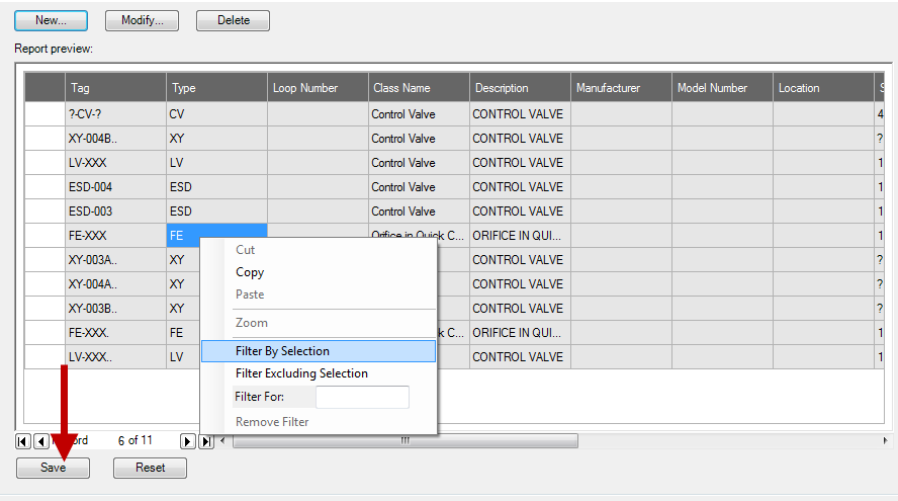


Figure 252: Save

The report will be available in the Data Manager under Project Reports.

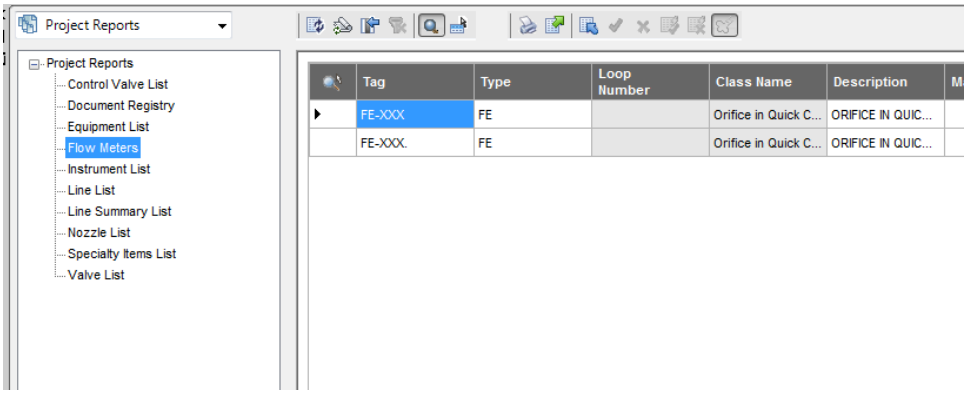


Figure 253: View Report

Modifying reports follows the same steps. Remember to click save after changing the columns and reports before clicking apply or ok.

You can click the reset button to clear any filters applied to the Report preview.

Report Creator

The report creator is a stand-alone app for creating project reports. By using the report creator, you can predefine report layouts, the data supplied to the reports, and the generated report locations.

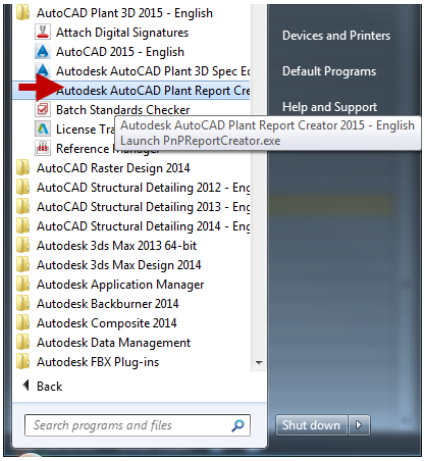


Figure 254: Launch Report Creator

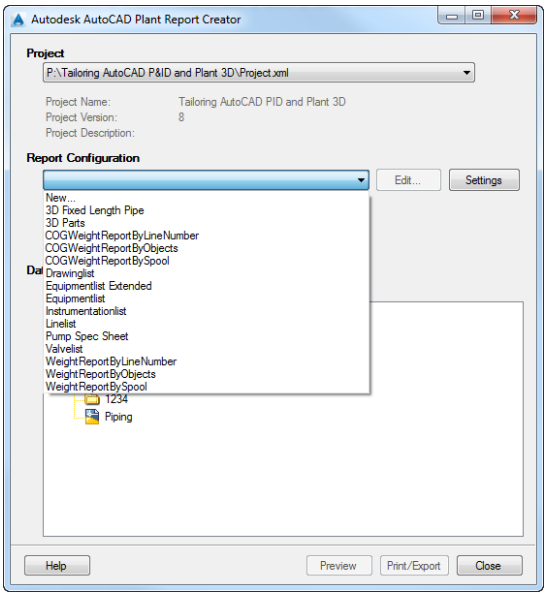


Figure 255: Report Selection

Generating Reports

The report creator has five steps when generating reports:

1. Select the project
2. Select the report location
3. Select the report
4. Choose the data source
5. Preview or print the report

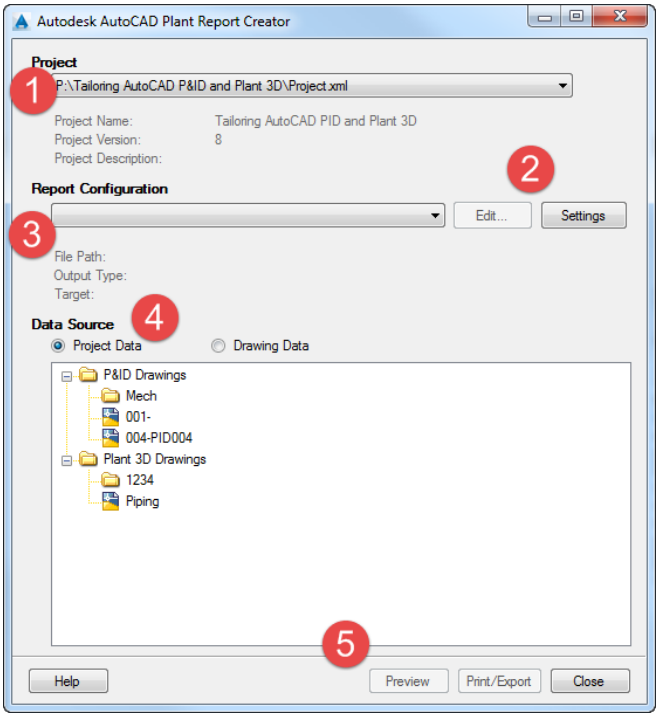


Figure 256: Report Generation Steps

Reports can be loaded from three locations – the default installation location, similar to C:\ProgramData\Autodesk\AutoCAD Plant 3D 2015\R20.0\enu\ReportCreator\ReportFiles, a report folder within the project folder, or a custom location.

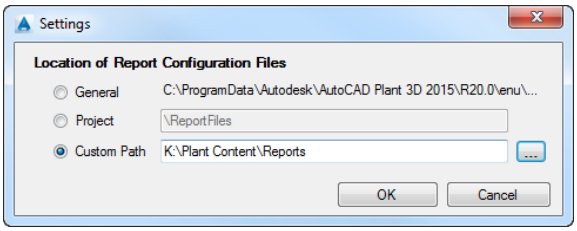


Figure 257: Report Locations

The data source defines how much information you want to display. By selecting the project, everything will be available, but by choosing drawings, you may select a folder or individual drawings to report on.

When previewing, you may export the report to one of many document types.

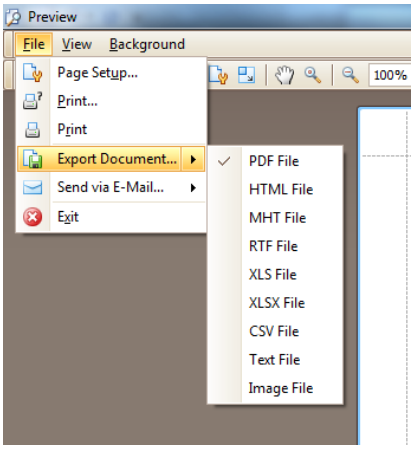


Figure 258: Report Export Formats

Creating Report Templates

The most important aspect of the report creator is its flexibility. With it you can define how your report looks and make sure it uses corporate logos and other required text. For advanced customization and/or scripting help, use the DevExpress documentation:

- http://docs.autodesk.com/PLNT3D/2014/ENU/report_designer.pdf

Use these guidelines to create clean, functional reports:

- Ensure that controls align as much as possible
- Use prefixes to label database properties
- Use Styles to control text font and color
- Use Parameters to create filtered reports

Control Alignment

If you export a default report to Excel, you probably have noticed odd column spacing and placement.

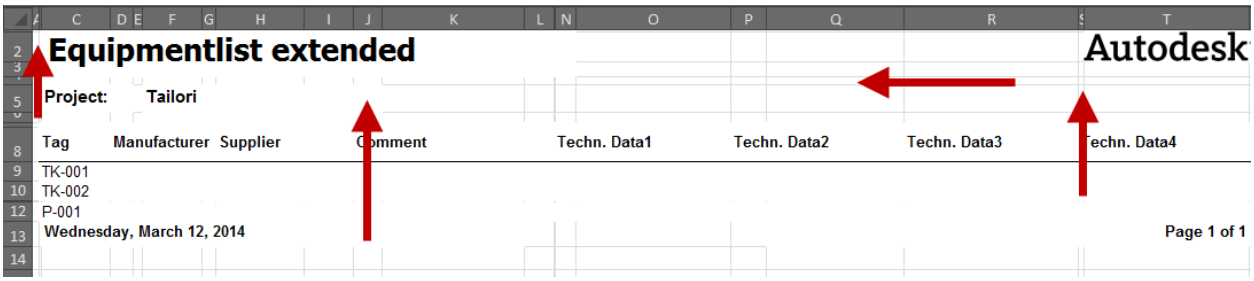


Figure 259: Odd Columns and Rows

This behavior is cause by control alignment. For example, in the Extended Equipment list, you can see the gap on the left side of the controls.

Use the following tips to modify the layout of your report:

- Remove Extra columns by arranging Controls
- Unlock anchoring for tables or controls before dragging sections
- Change the Image sizing to allow stretching
- Add padding or stretch controls to align all columns

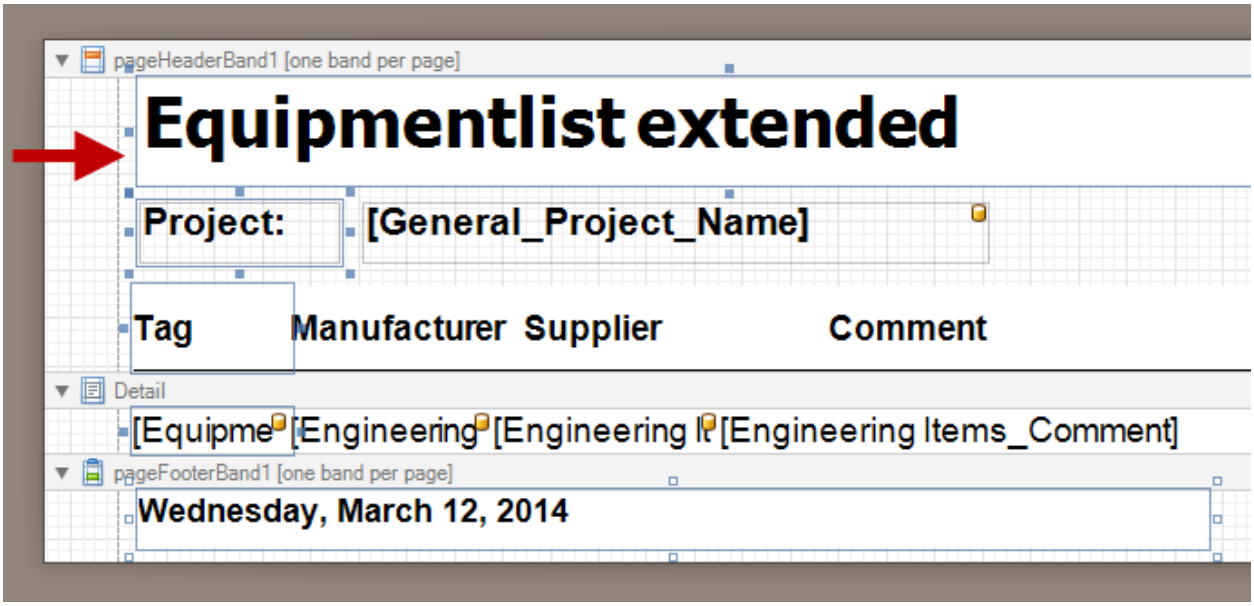


Figure 260: Non-Aligned Controls

To clear up the columns, you can move the controls, or add padding and re-size them.

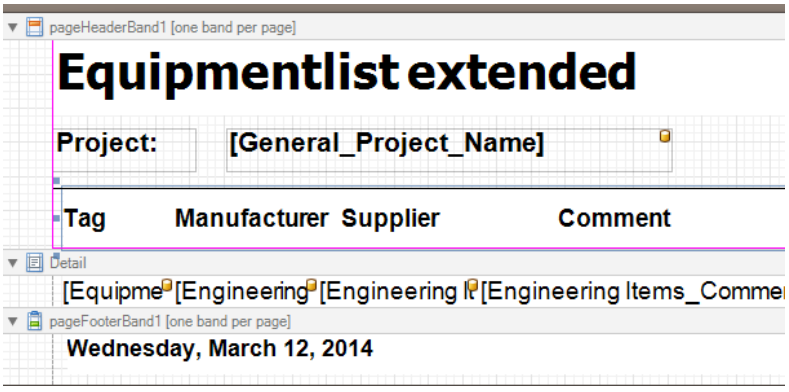


Figure 261: Moving Controls

When moving controls vertically, you should turn off anchoring before moving the section higher.

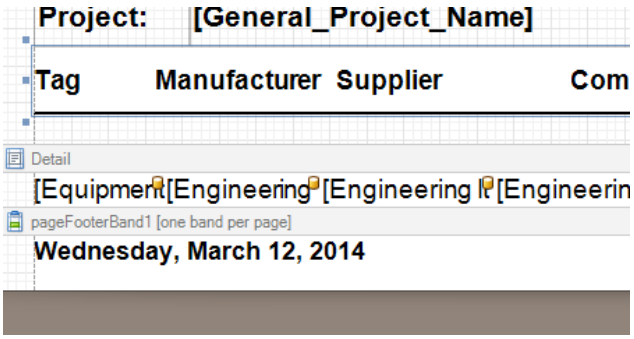


Figure 262: Anchor

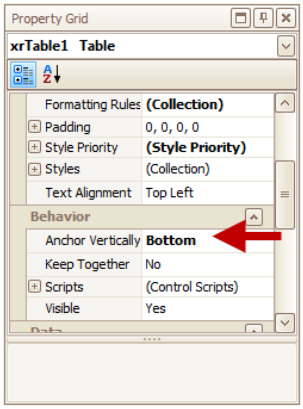


Figure 263: Modify Anchor

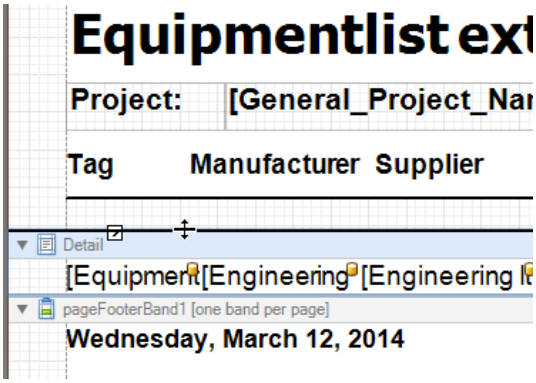


Figure 264: Moving Vertical Slider

Change image sizing to Stretch to allow making them larger.

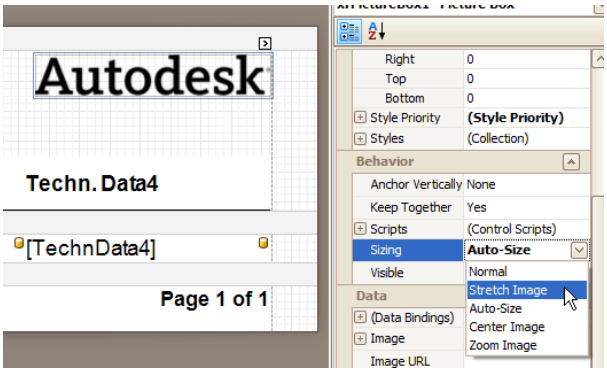


Figure 265: Stretch Image

Align all controls to use fewer columns

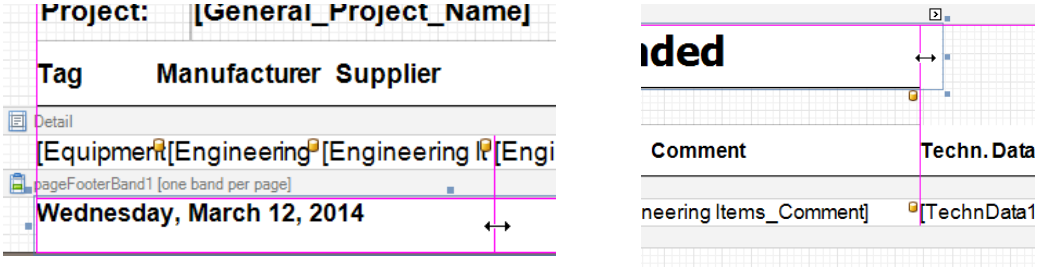


Figure 266: Align Control Edges

The final report with all controls aligned looks like this:

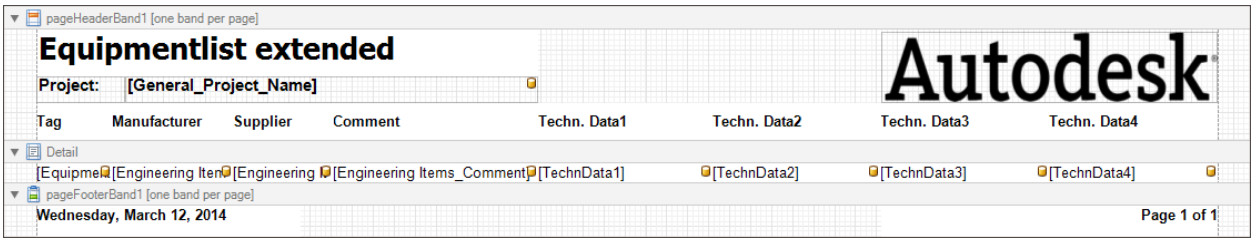


Figure 267: Aligned Controls

The report produces this Excel export:

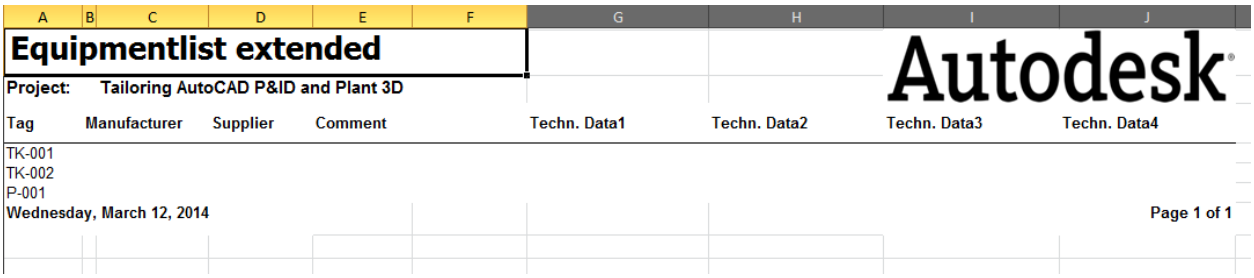


Figure 268: Produced Report with Aligned Controls

Use Prefixes on Database Properties

While the report is mostly cleaned up, you can see column B is unnecessary. Column F is unnecessary as well. You can clear out column B by selecting the General_Project_Name label, and the clicking the > button at the top right. Then click the Format String box expand button.

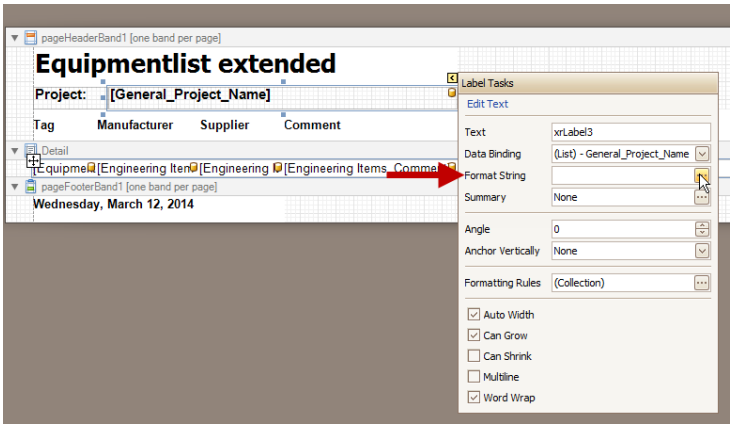


Figure 269: Using Prefixes

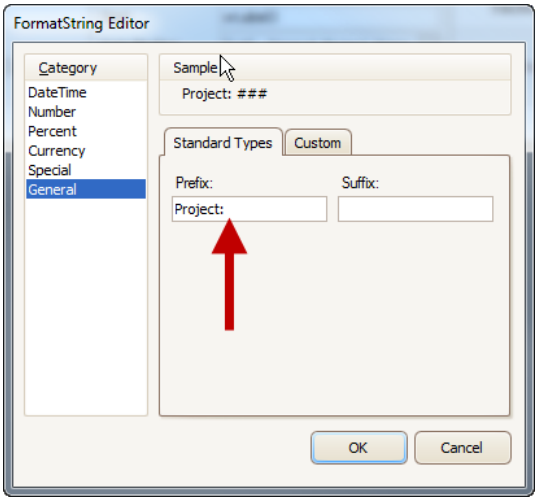


Figure 270: Set Prefix Value

In the general box, you can add a prefix to the value displayed. In this case, enter “Project: “.

With the prefix set, you can remove the Project: label completely.

The final culprit to clear out is column F. This column is created because the Project Name label allows Auto-Sizing which means that the label will always only match the length of the text. Since that is unpredictable, you want to leave a lot of space, but turn off Auto-Sizing.

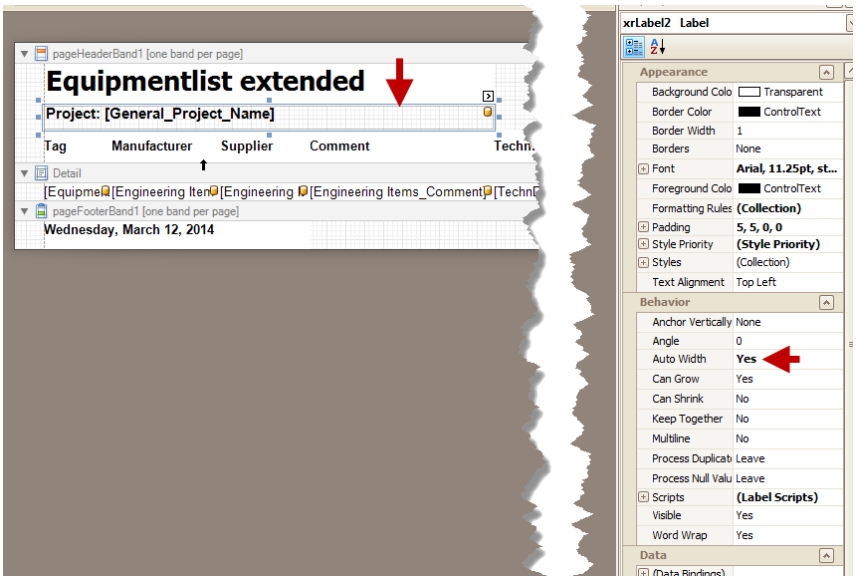


Figure 271: Set Auto Width to No

The final report layout looks like this:

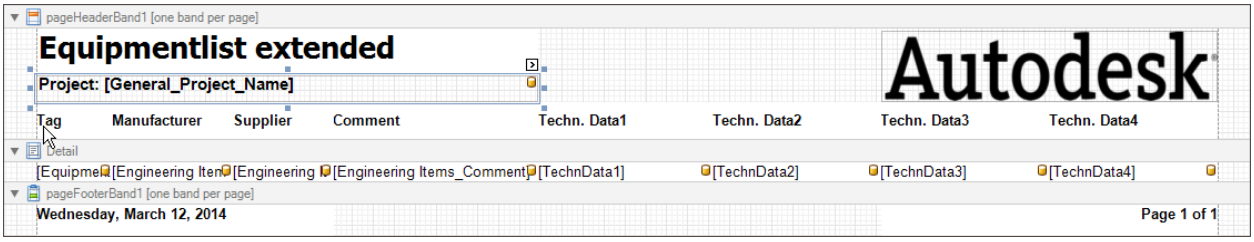


Figure 272: Report with Prefix and No Autowidth

The final report exports to Excel looking like this:

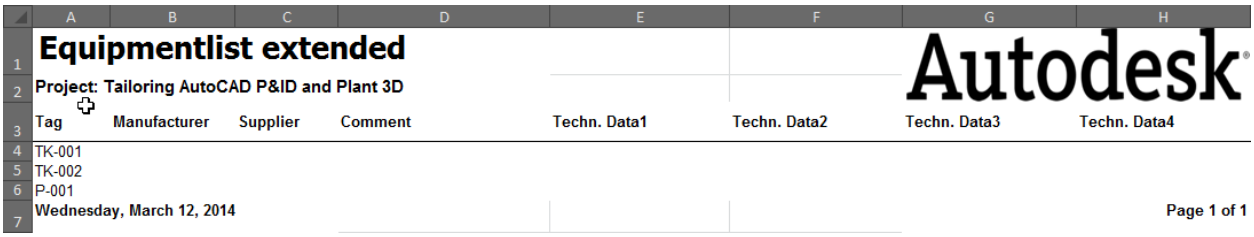


Figure 273: Generated Report

Using Styles to Control Display

While the layout is fixed, you must also control the font color, alignment, and size among other visual settings. The best way to control the visual appearance of text and controls is to use styles.

To access the styles, select an item and click the Styles expand button. Modify each of the styles to enforce your company standards.

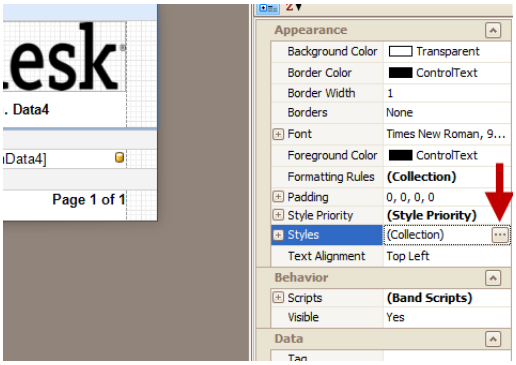


Figure 274: Access Styles

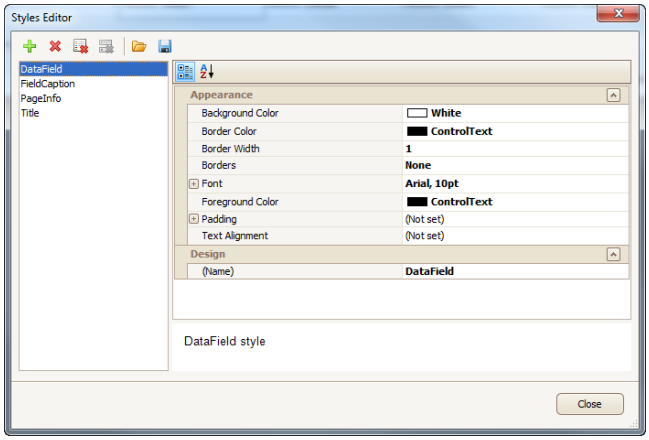


Figure 275: Modify Style Properties

After the styles are configured, you must make sure they are applied to every item. Select an item, expand the styles node and verify that a style is selected. Also, expand Style Priority and make sure that no overrides have been applied.

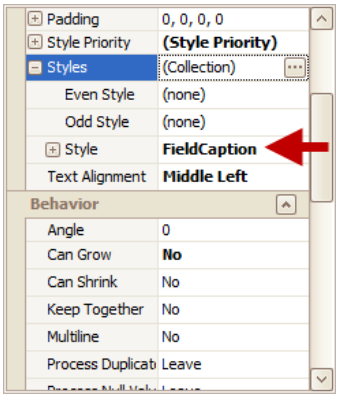


Figure 276: Apply Styles to All Items

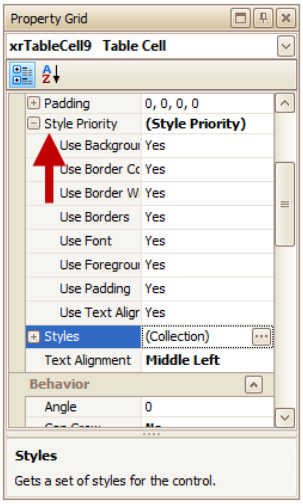


Figure 277: Reset Style Priority

After all the controls have styles applied, any change you make to the style will affect all of its controls.

Using Parameters to Control Content

In the report creator we can choose between project and drawing sources. While that is flexible, the report itself allows another magnitude of flexibility. By using parameters in the report creator, we can prompt the user to enter a value, like Area, and then display the results for only that value or in this case Area.

- Create an Area parameter
- Set the other parameters to not visible
- Set Request Parameters to Yes

Add a parameter by moving to the Field list, right-clicking in the parameters section, and choosing Add Parameter.

Create a parameter called Area. Next, set all other parameters to not visible.

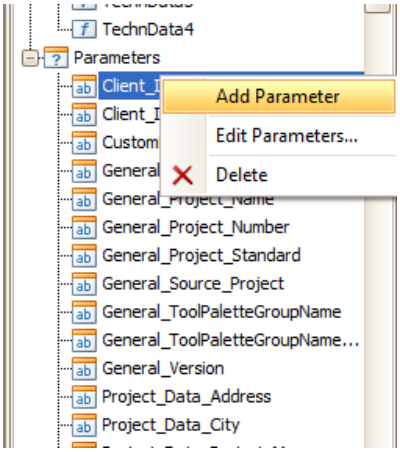


Figure 278: Add Parameter

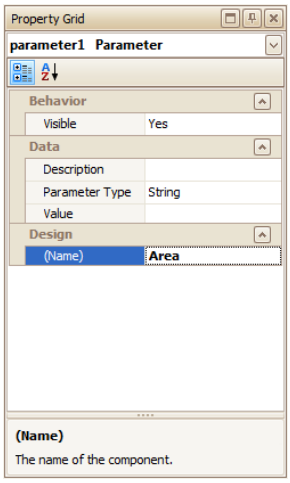


Figure 279: Name Parameter

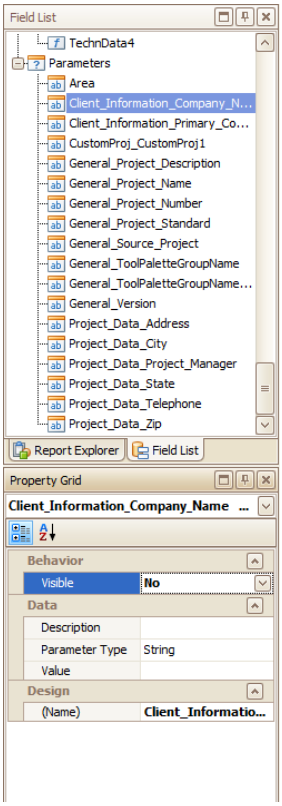


Figure 280: Alter other Parameters to Not Visible

The final step is to select the report template in the Report Explorer, scroll down to Request Parameters and select Yes.

From that point on, users will be prompted to enter an Area for the report. You can use a label to display the parameter that was entered on the report.

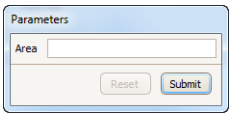


Figure 281: Parameter Prompt

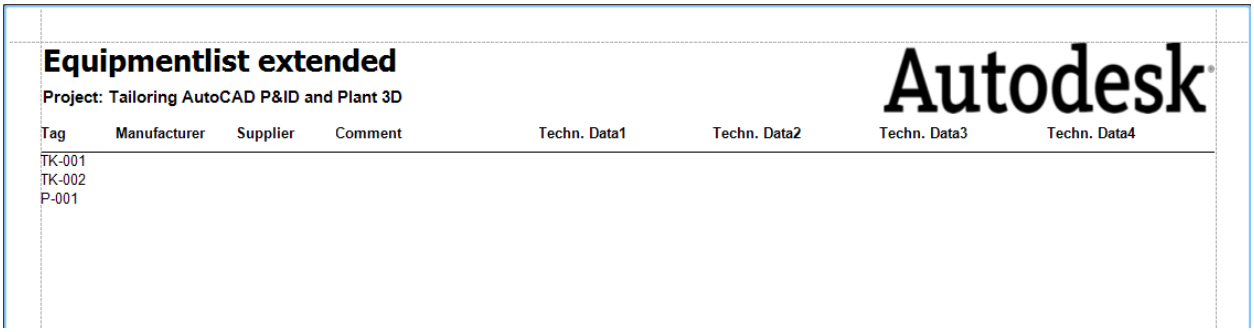


Figure 282: Filter report from Parameter

Interoperability

Because AutoCAD P&ID and AutoCAD Plant 3D target plant design, by nature they must integrate with many programs.

Vault

Document management is used for backing up files, tracking revisions, and sharing across multiple sites. Vault integrates within the project manager to allow managing the project, creating drawings, modifying drawings and removing them.

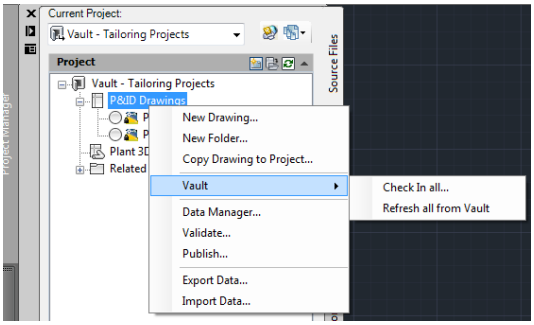


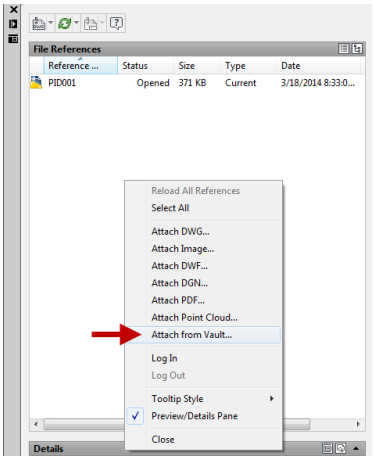
Figure 283: Vault menu

Overview

Autodesk Vault, like most data management programs, consists of two parts, a client and a server. Much like a network file server, the Vault server is responsible for holding the files. In addition, it manages revisions and who owns files. In order to connect to a Vault server, you have to use special programming. In the case of AutoCAD Plant 3D, the programming is written into Plant. For general users, Autodesk also provides a program that can interact with the server independent of CAD software. This program is often referred to as the client.

In order to manage drawing revisions, users must check out a drawing. Checking out allows other users to read and use the file, but they will not be able to alter the master copy by checking in changes.

Client Installation



Plant users do not need the Vault client installed on their machines. However, some find it helpful. With AutoCAD Plant 3D 2014, standard Vault functions like xrefs from Vault do not work.

Figure 284: Vault Version Conflict 2014

However, you can load the Autodesk Vault 2014 client on a computer with AutoCAD Plant 3D 2014. During the installation setup, you need to use a custom installation type:

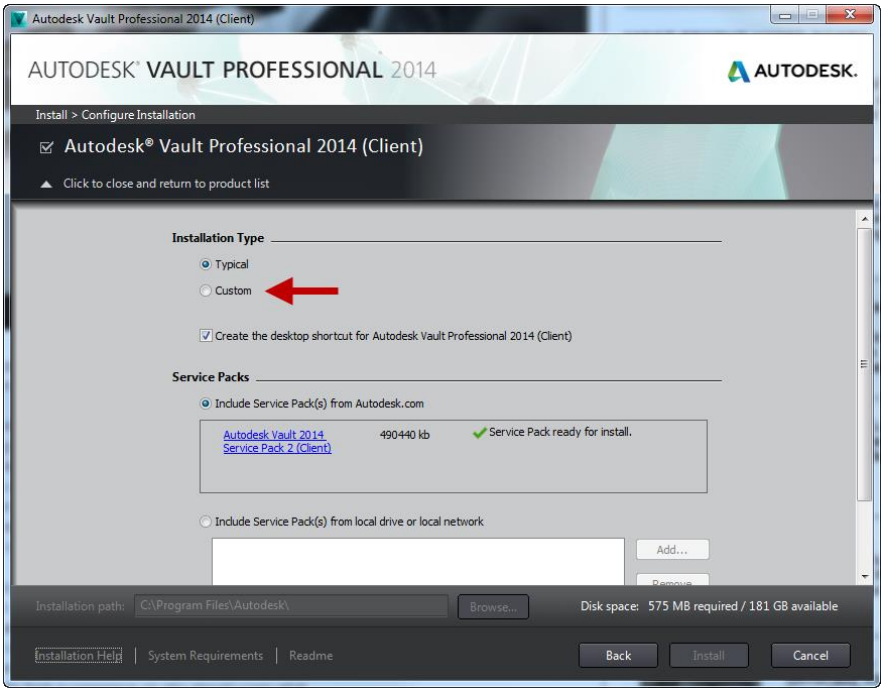


Figure 285: Installing Vault 2014 Client

Then you MUST uncheck the AutoCAD Plant 3D 2014 Addin option.

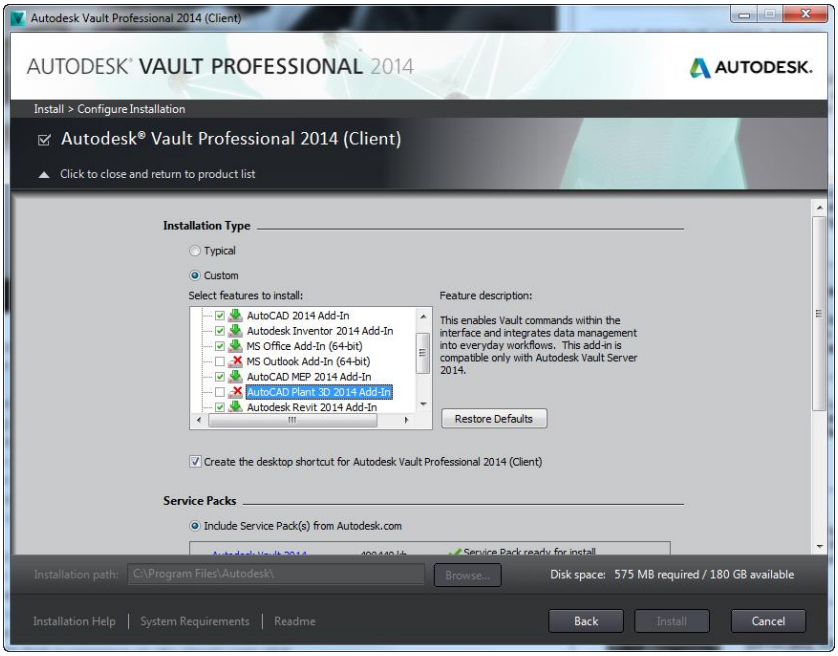


Figure 286: No Plant 3D 2014 Addin

Autodesk Vault 2015 and AutoCAD Plant 3D 2015 are compatible and do not require this extra step.

Using Vault

Get started using Vault here: http://knowledge.autodesk.com/support/vault-products/getting-started

Users who will create, edit and remove projects should be assigned the Administrator role within Vault.

In the Vault Client, go to Tools > Administration > Global Settings to view Users, Groups, and Roles.

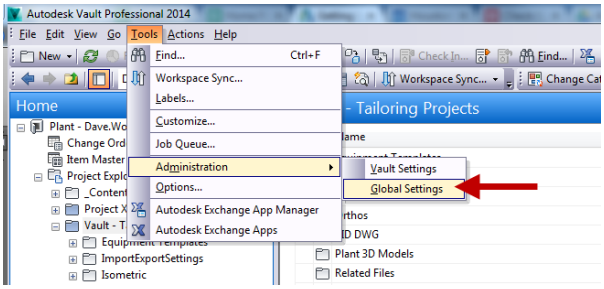


Figure 287: Administrator in Global Settings

Users who do not have Administrator permissions will not be able to check in project settings.

Creating a New Project

You can use the project wizard to create projects in Vault. The project will be created in the Vault working directory. To modify the working folder, you navigate to the Tools menu > Administration > Vault Settings. On the Files tab, click the define button. You have the option of allowing users to define their own folder or setting a specific path. To ensure path consistency, most companies define a custom path.

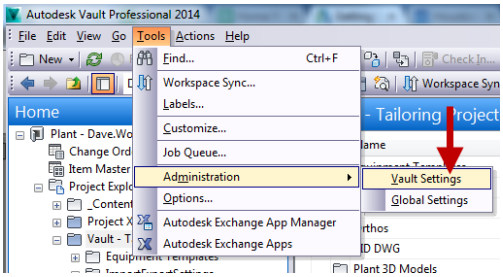


Figure 288: Vault Settings

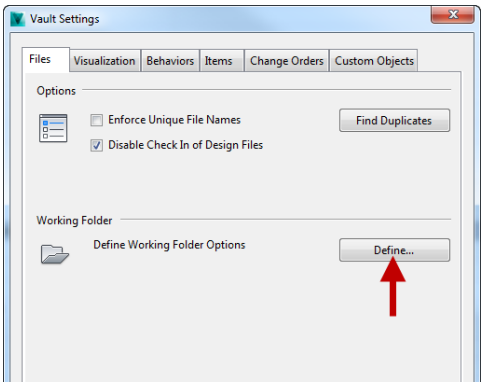


Figure 289: Working Folder

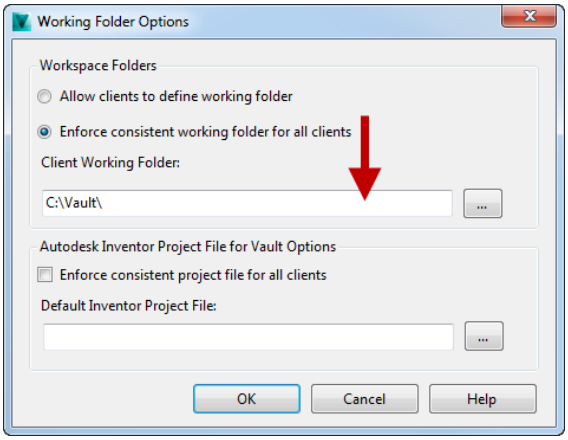


Figure 290: Set Vault Folder

When you start the project wizard, and check the “Create this project in Vault” button, you will be prompted to enter your login credentials.

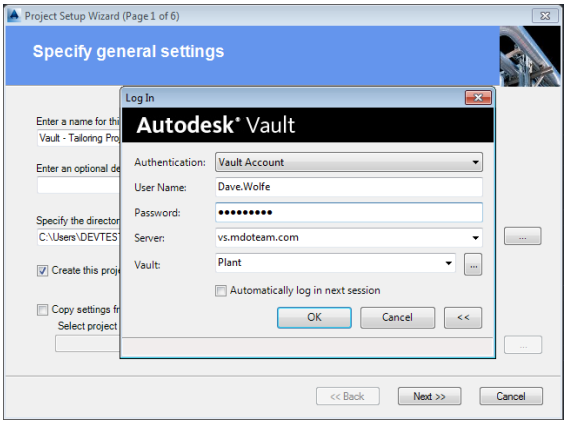


Figure 291: Vault Login

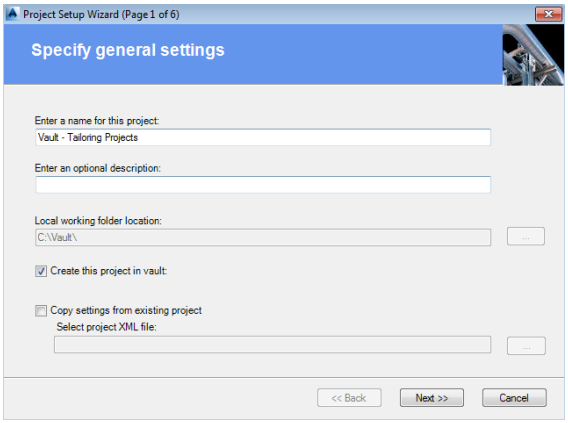


Figure 292: Create Project In Vault

Most of the remaining settings can be picked like a normal project would be, with the exception of paths. When using Vault, the folders selected must be under the project folder.

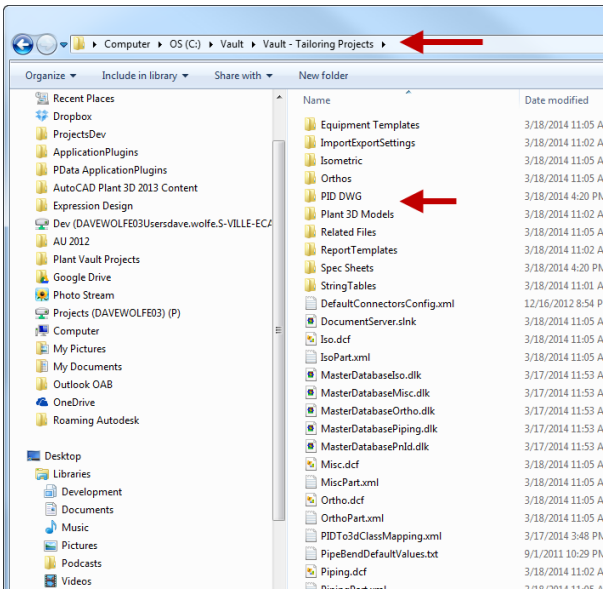


Figure 293: All Drawing Folders in the Project Folder

When using Vault, you also must use SQL Server. Your database settings will look similar to this. Make sure to test the connection before hitting next. After you have completed the wizard, the files will be created and stored in Vault. Notice that the port is separated with a comma in the server name (if applicable).

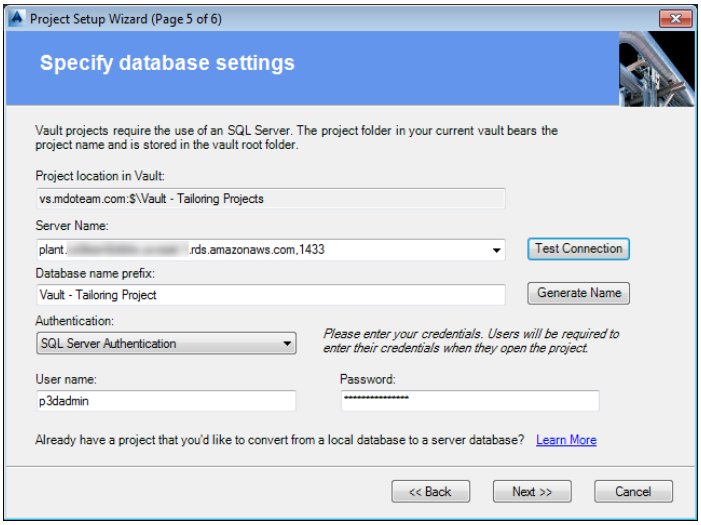


Figure 294: Connect to SQL Server

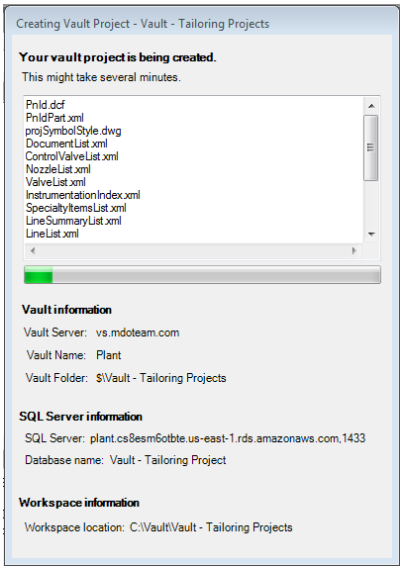


Figure 295: Project Copied Locally

Opening Drawings

You can use the project manager to modify drawing files as is done without Vault. However, to work on a file you need to check it out. Right-click on the file, click Vault, and then Check Out.

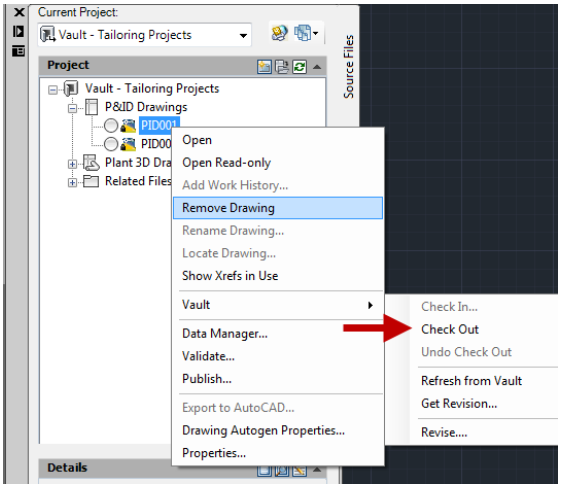


Figure 296: Vault Drawing Menu

You can also double-click on the file in project manager. You will be prompted to check it out or open it read-only. If a file is already checked out, it will have a lock icon beside the file name.

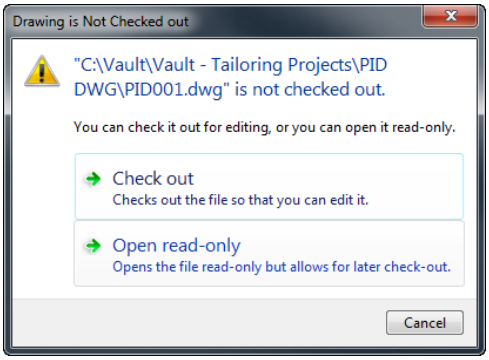


Figure 297: Check Out Drawing

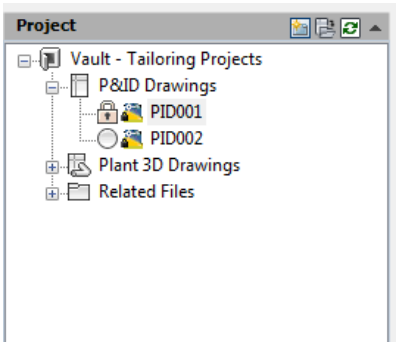


Figure 298: Locked File

Committing Drawings

After modifying a file, you must check it in, to have the Vault server receive the latest changes.

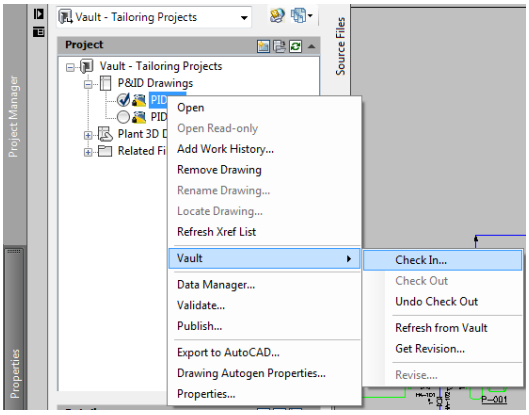


Figure 299: Check In Menu

Organizing Files

After creating the AutoCAD Plant 3D project in Vault, you can move it using Vault within the folder structure to a location that fits your organizational needs. The next time Open Project from Vault is used, the Vault Folder path will show the new location, but the local workspace will use a shortened path. The project will have to be re-initialized to download the files correctly.

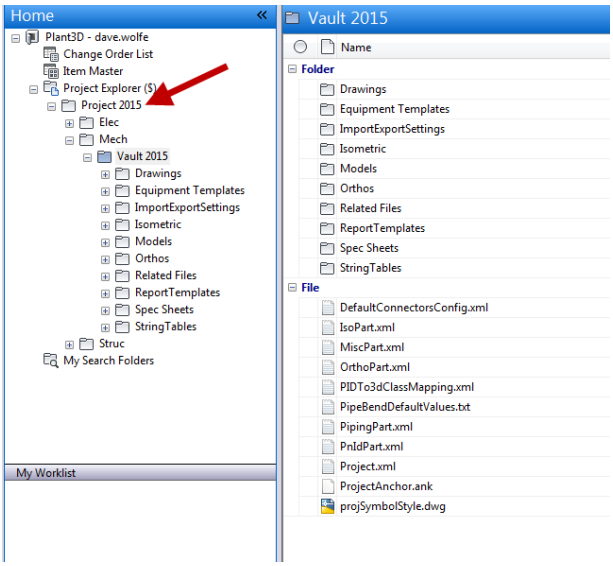


Figure 300: Relocated Project

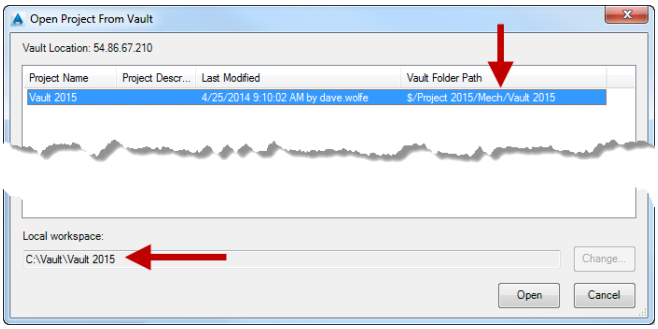


Figure 301: Local Workspace with Relocated project

In 2015, you can use the Xref palette and reference in any Vault file into your Plant project.

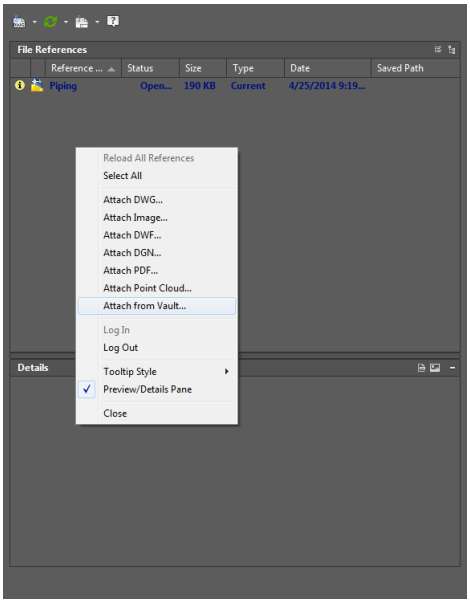


Figure 302: Xref from Vault

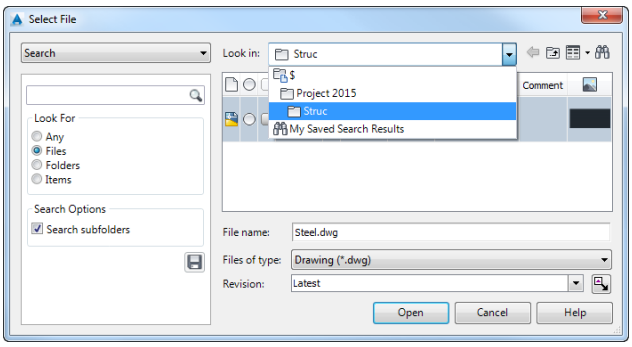


Figure 303: File Selection

Vault Database

AutoCAD Plant 3D takes advantage of both the SQLite local database and a SQL Server database. When checking out or getting a drawing from Vault, the project uses a SQLite database to track the project items. When checking a drawing in, AutoCAD Plant 3D pushes the files to Vault and updates the SQL Server database with any changes. By keeping the files and databases separate, AutoCAD Plant 3D allows anyone to make any changes at any time, while maintaining database integrity.

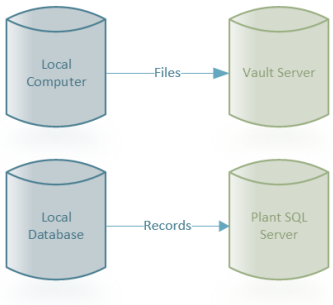


Figure 304: Vault Check In Procedure

Inventor

Autodesk Inventor is used to model equipment or other items that need to come into the piping model. While AutoCAD Plant 3D will not read any properties set in Inventor, the 3D model can be used.

Export to ADSK

The first method is to use the BIM tools to export an Inventor model to an .adsk file format. To start, switch to the Environments tab and click the BIM Exchange button.

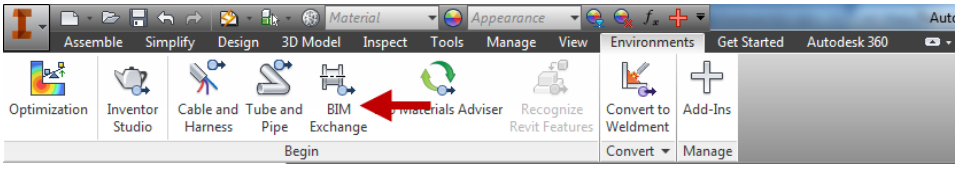


Figure 305: Export to .adsk

Once you get into the assembly editing mode, choose what parts to keep.

Notice that the hole patching range can be customized. The settings from .5 min to 2.5 max were needed to remove the bolt holes in the flanges. Click preview to see what will go away.

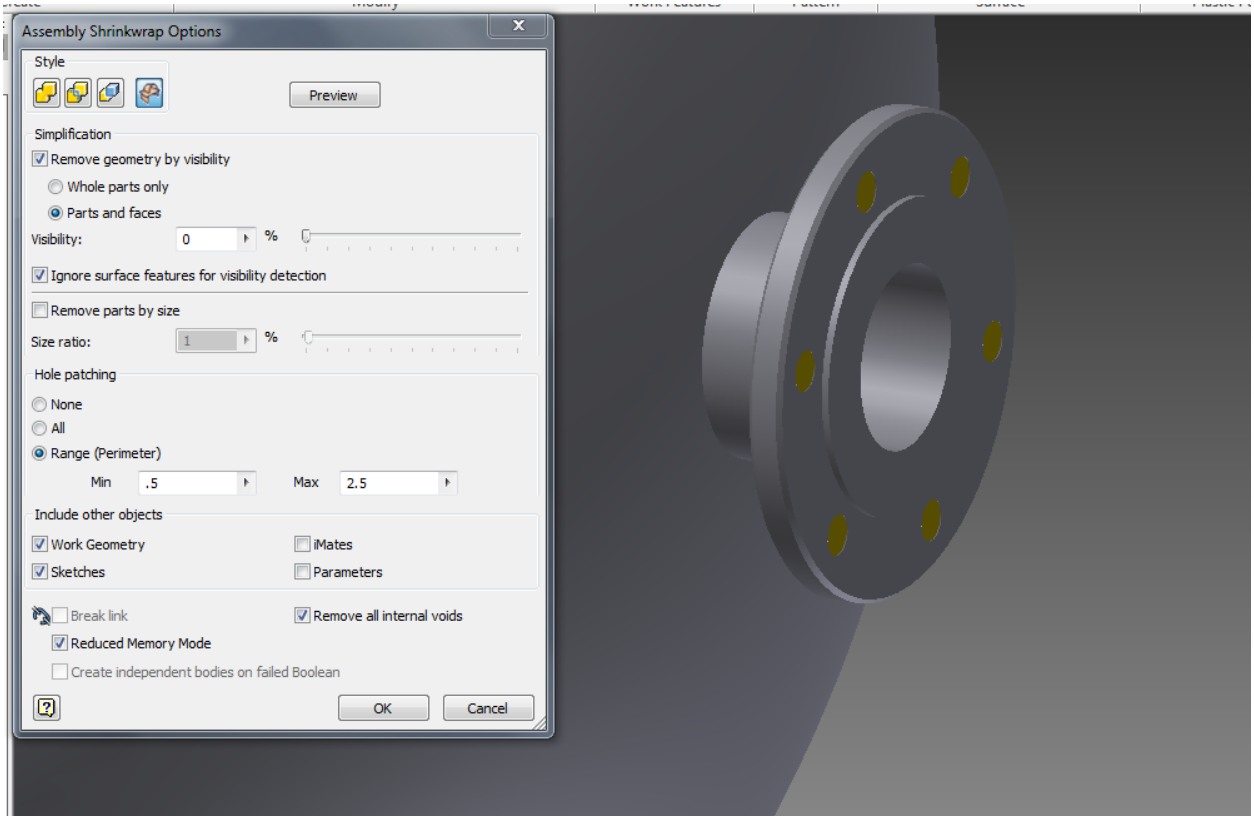


Figure 306: Shrinkwrap

After configuring your settings, click ok. Your model will return to the BIM Exchange mode with changes made. To connect AutoCAD Plant 3D pipes, click the Pipe Connector button.

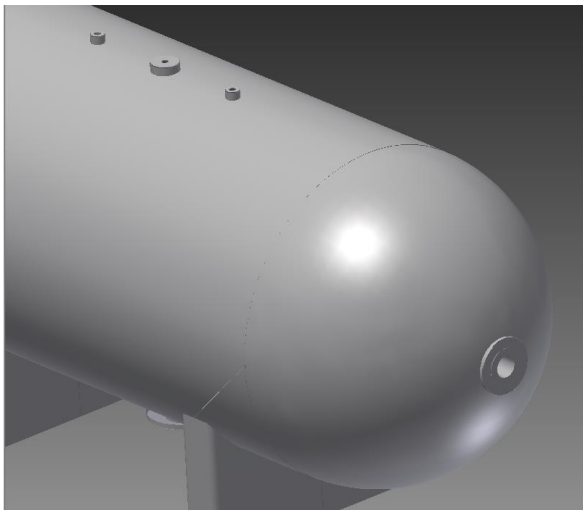


Figure 307: Nozzle

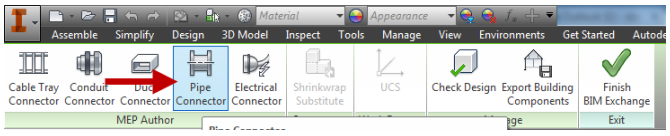


Figure 308: Pipe Connector

For AutoCAD Plant 3D pick the connection size and type, locate the nozzle and pick the direction.

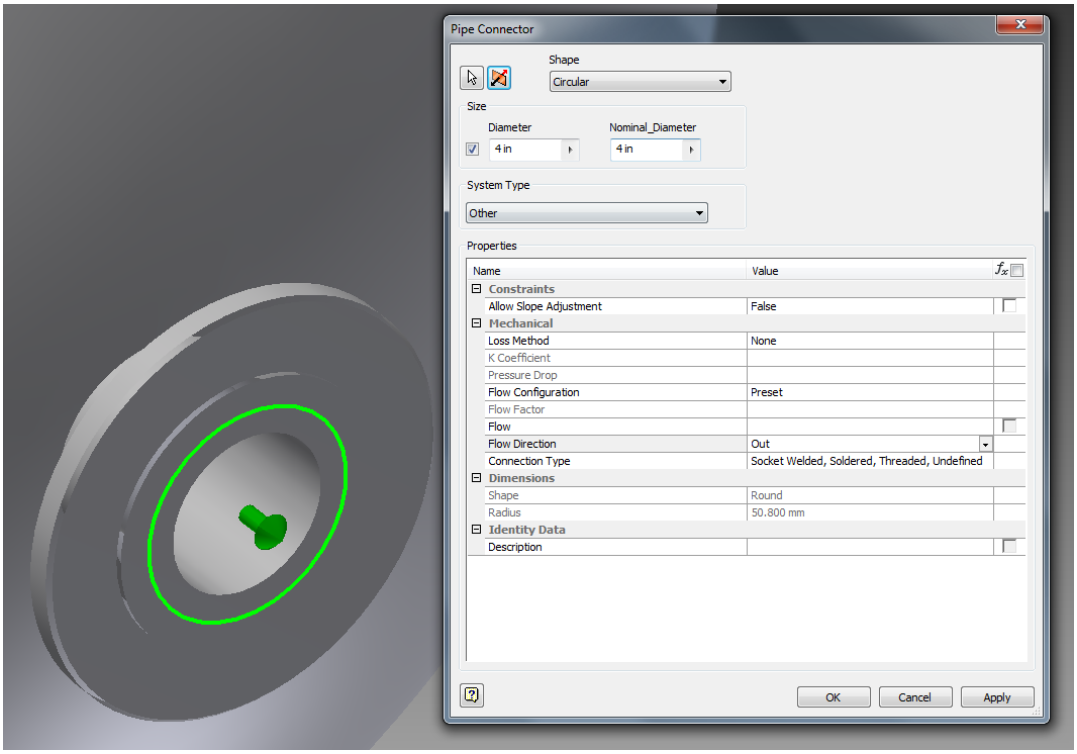


Figure 309: Nozzle Data

After all of the nozzles have been identified, click the Export Building Components button.

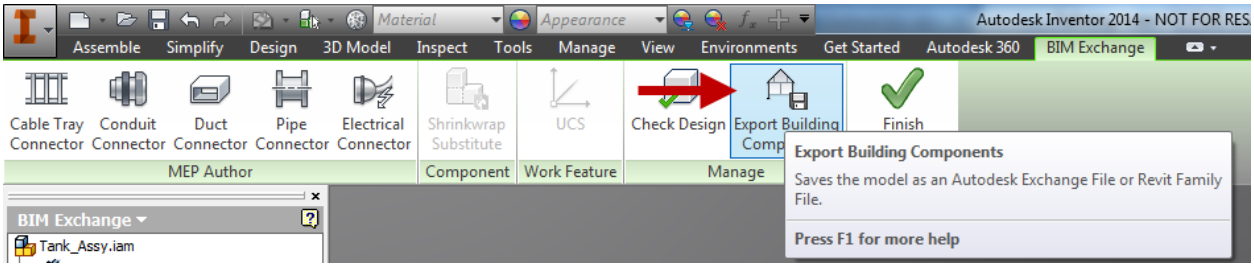


Figure 310: Export to .adsk

Click Ok to export the building components, the routine may take a few minutes to finish.

In AutoCAD Plant 3D, click the Convert Inventor Equipment button and select the exported file.

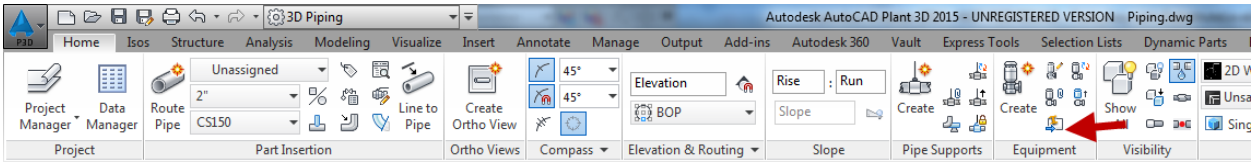


Figure 311: Convert Inventor Equipment

Place the equipment by picking the insertion point, and the rotation.

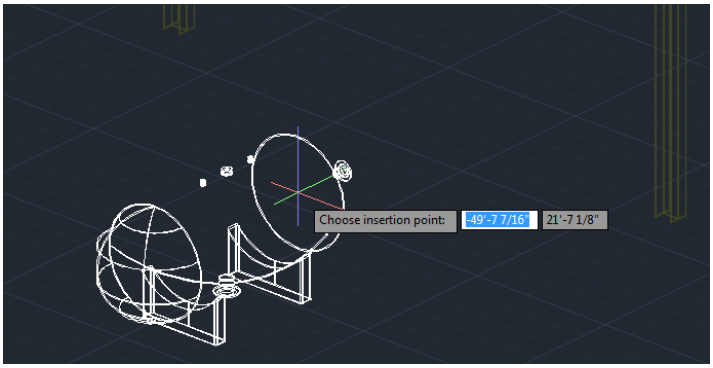


Figure 312: Pick Insertion

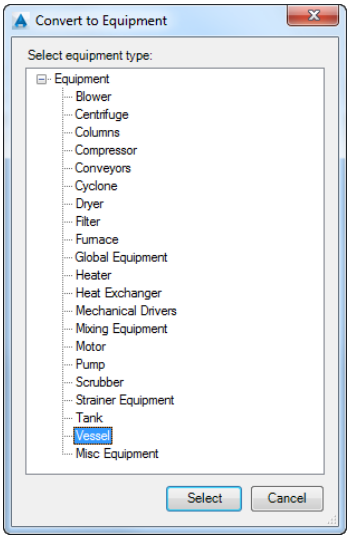


Figure 313: Choose Equipment Class

All of the identified nozzles will be ready for pipe routing. Assign tags and modify properties per standard equipment methods.

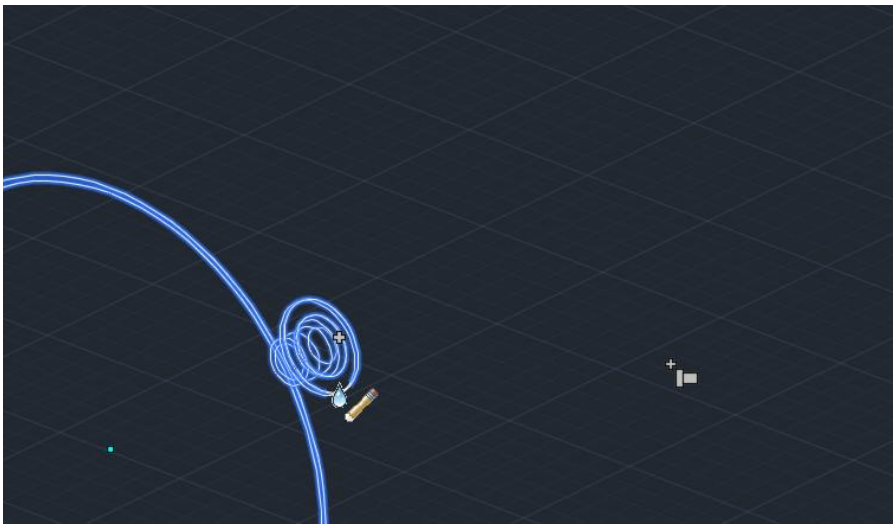


Figure 314: Edit Nozzle

Updating the model requires you to complete all the steps again.

AutoCAD IMPORT

Beginning in AutoCAD 2013, the IMPORT command included the ability to handle several new file formats, one of which is Inventor.

Enter the IMPORT command at the command line, or find the Import button on the Insert panel.

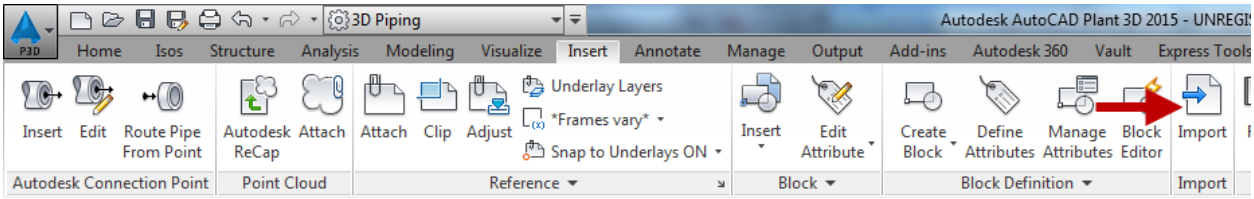


Figure 315: IMPORT

Switch the file type to Inventor, and choose the assembly or part model.

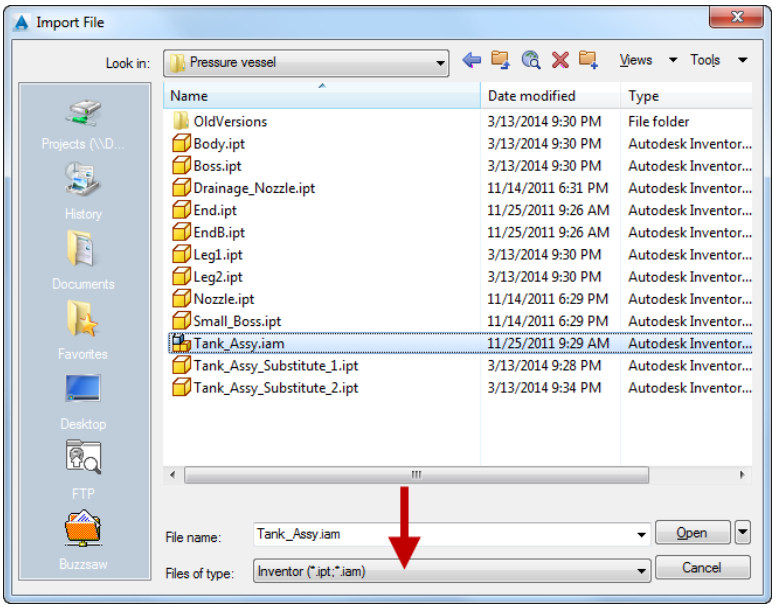


Figure 316: Choose File Type

The import routine will run in the background and you will receive notification when the process is ready.

Click the hyperlink and the model will be inserted at 0,0,0 in the current drawing.

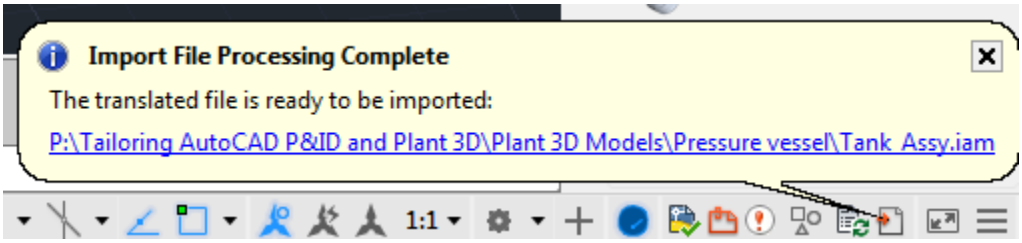


Figure 317: Import Process

Rotate the model as needed (Inventor defaults to +Y be real-world up).

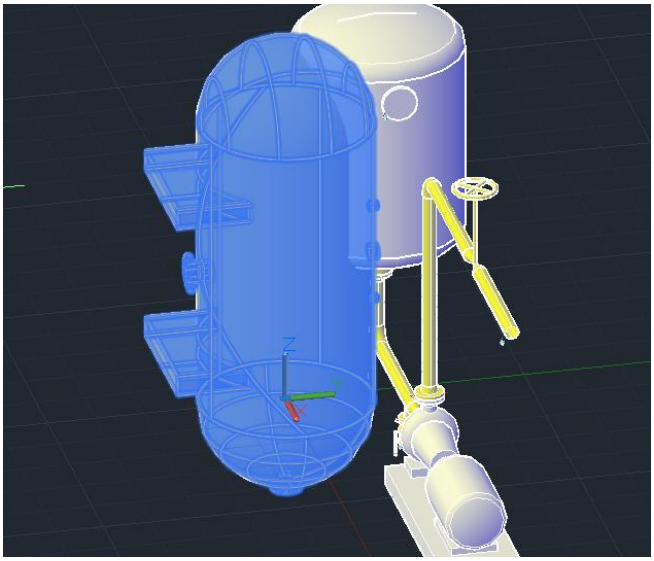


Figure 318: Manipulate Model

Click the Convert Equipment icon, select the block, and choose a class and an origin.

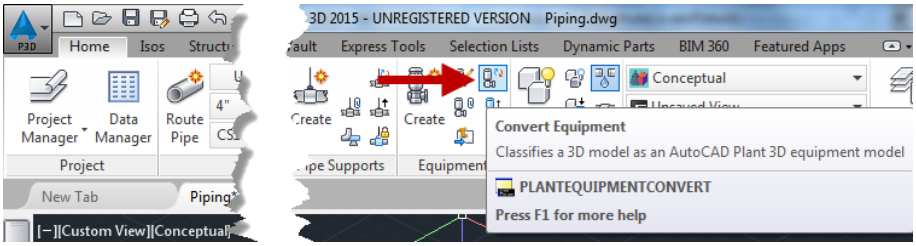


Figure 319: Convert Equipment

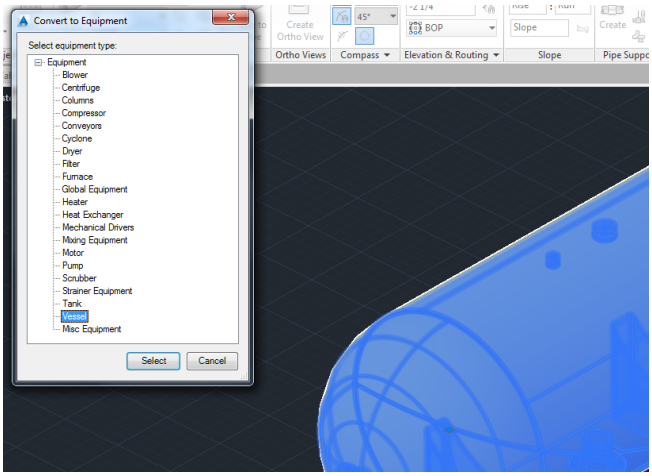


Figure 320: Select Equipment Class

You will be prompted to enter the vessel tag and other information.

Finally, identify the nozzles by click add Nozzle, and picking a nozzle point. Drag away perpendicular to the nozzle to determine the direction.

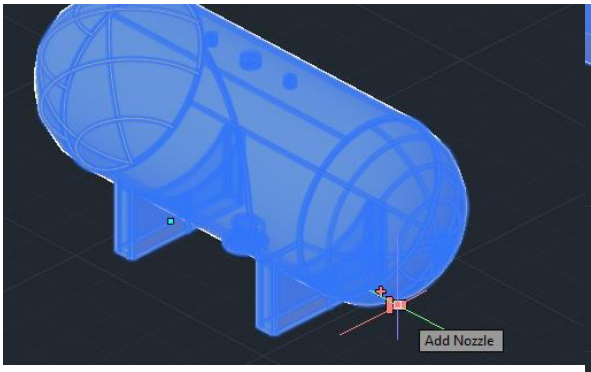


Figure 321: Add Nozzle

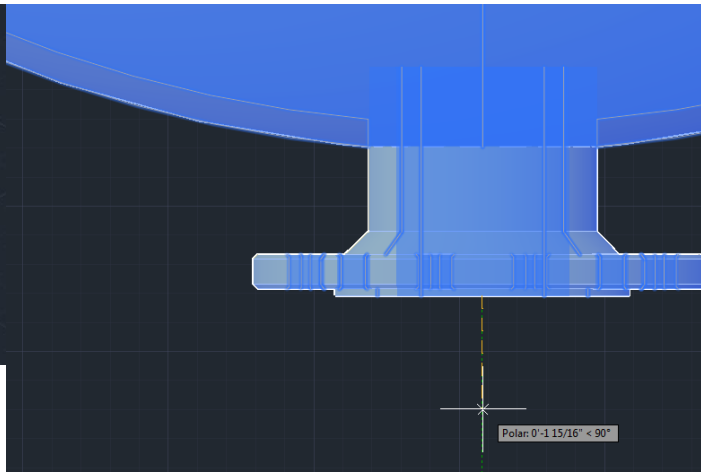


Figure 322: Pick Nozzle orientation

Then you can change the nozzle location again, remove it, or set the nozzle type.

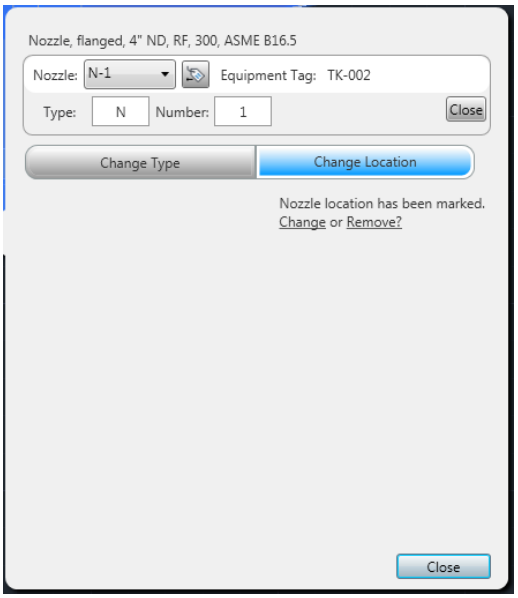


Figure 323: Change Nozzle Location

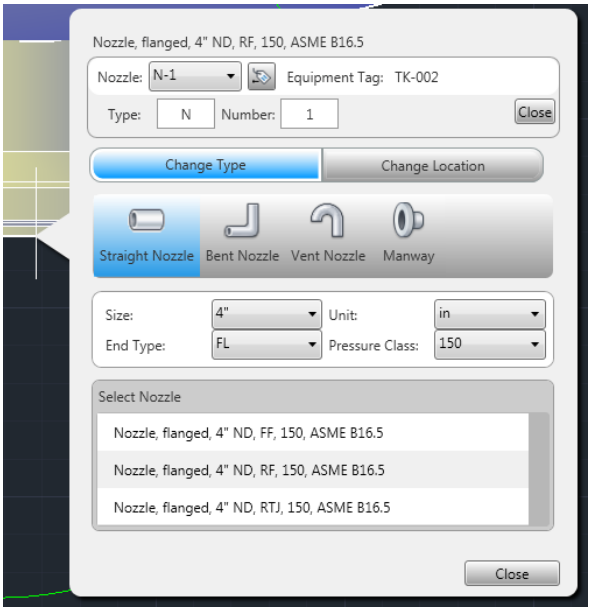


Figure 324: Change nozzle type

Note that in either case, we are only identifying the connection point of the piping. The modeled parts have no direct physical relation to the piping properties.

Revit

Most programs aren't designed to read AutoCAD Plant 3D properties when viewing the model, Revit included. The integration between AutoCAD Plant 3D and Revit lies only within the ability of Revit to load and view AutoCAD drawings, and Revit's ability to export a dwg. Because the data doesn't transfer, you cannot annotate AutoCAD Plant 3D objects from within Revit (and vice versa). However, you can import and export the 3D model to an AutoCAD file.

Exporting from Revit

The Revit export function is available on the application menu > Export > DWG. You can save or choose your export options.

- Export the drawing from Revit to DWG
- Copy the Drawing to Project
- Reference the drawing in a structural model

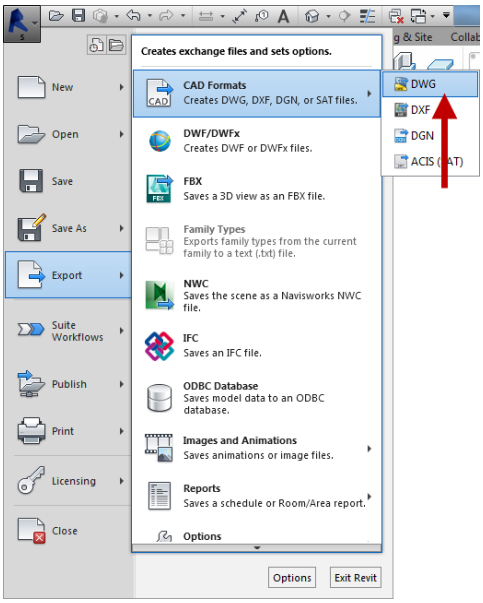


Figure 325: Revit Export to CAD Format

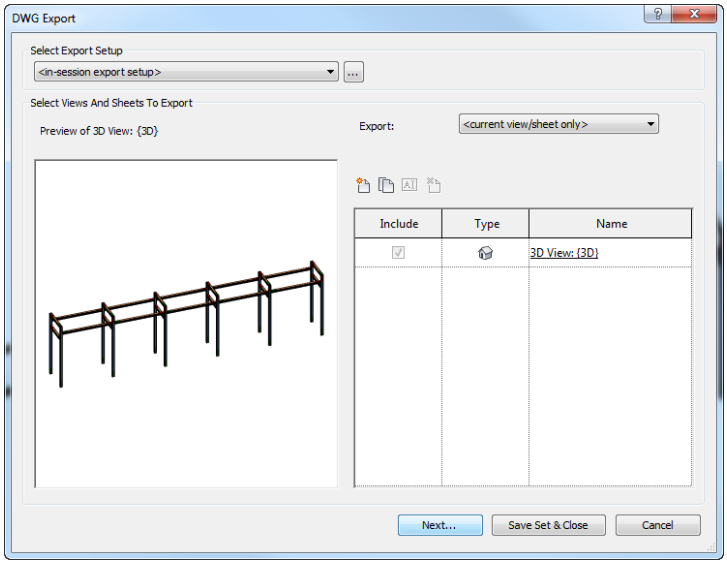


Figure 326: Export Setup

Choose a file location to export to; typically this location would be a structural folder in an AutoCAD Plant 3D project.

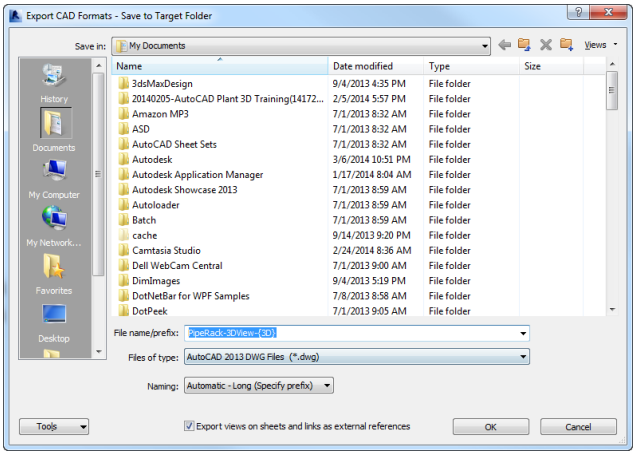


Figure 327: Export Location

Instead of referencing the exported file directly into a piping model, you should create a structural model and use that to reference the exported drawing in as an attachment.

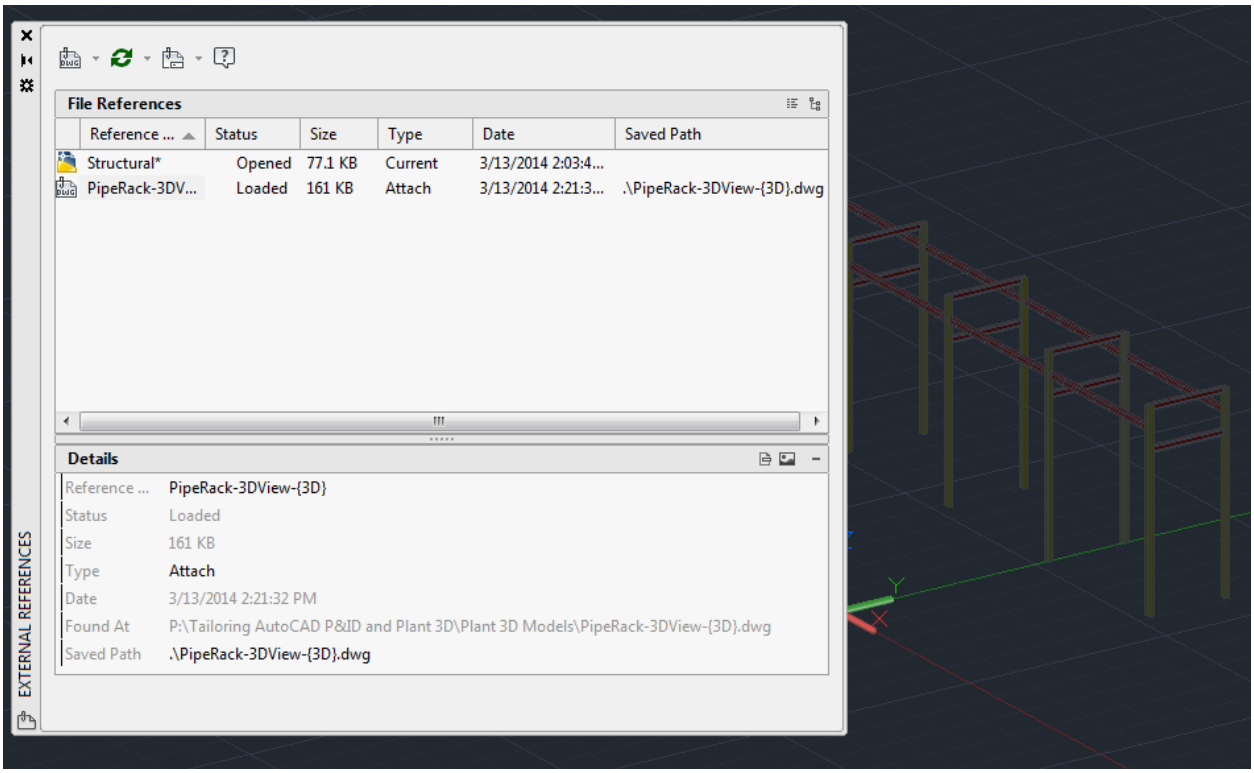


Figure 328: Create Structural Master for References

Using another drawing to host the export allows the exported file to be updated at any time without losing changes, and gives the piper a place to create temporary steel, supports, or pipe racks which have not been designed by the structural engineer.

Importing into Revit

The other direction that models need to be transferred is when importing piping into Revit. Because Revit does not recognize object enablers, linking the piping model directly results in something like this:

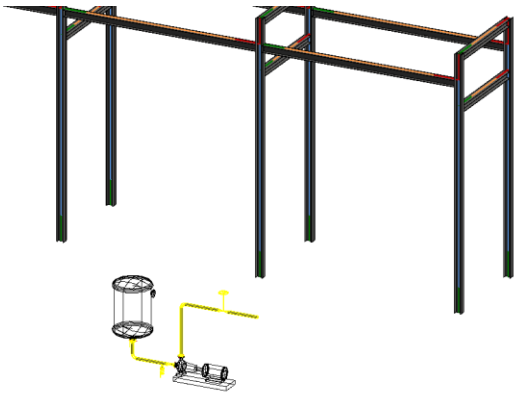


Figure 329: Import Piping Into Revit

In order to import into Revit, you have to export AutoCAD Plant 3D models to AutoCAD shapes first. You can do so by right-clicking on a file that is open, and choosing Export to AutoCAD. You should export files to a location or with a naming system that indicates they are an export. For example, you might export them to a folder called Exports, or simply to the same project folder, but with a suffix like Piping_acad.dwg.

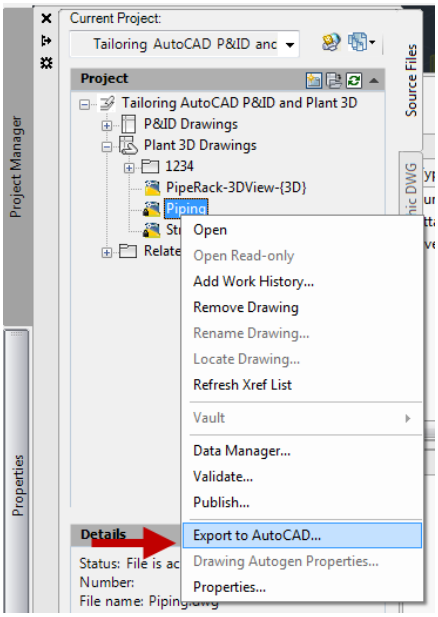


Figure 330: Export to AutoCAD

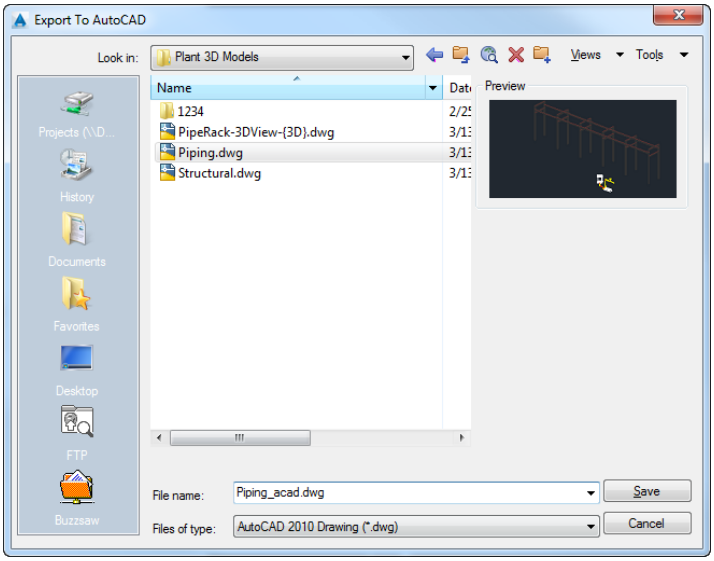


Figure 331: Export Location

The exported file may be brought into Revit by going to the Insert panel, and choosing Link CAD. You should verify the positioning of the model (typically Auto – Origin to Origin) and the level on which the file belongs.

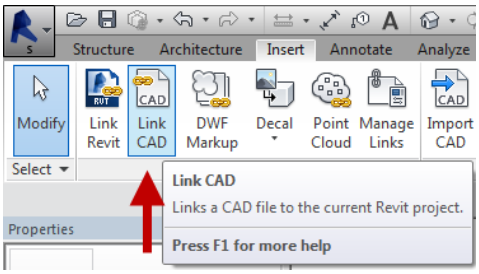


Figure 332: Link CAD File from Revit

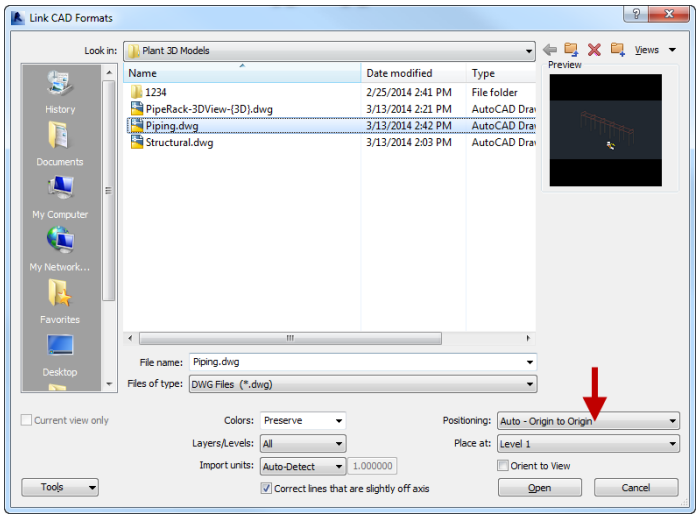


Figure 333: Link CAD file Location

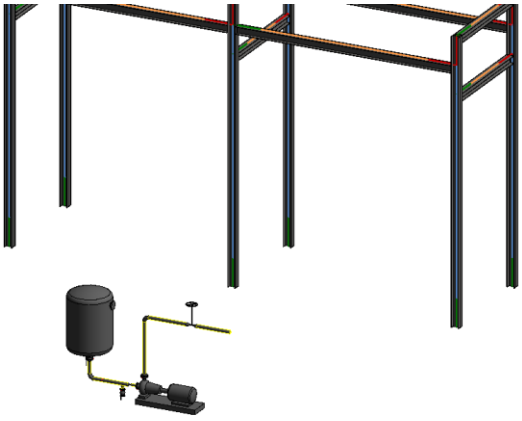


Figure 334: Completed Piping Import

Once the import or export is setup, the destination programs do not have to go through the linking process again. When Revit re-exports the model, AutoCAD will notice the xref has changed and will prompt for updating. Likewise, when AutoCAD Plant 3D updates the model, Revit will see the change and notify that an update is required.

Navisworks

Autodesk Navisworks is a standalone application that allows viewing and combining large models into a single file.

In order to use Navisworks object enablers must be installed. If you have opened a dwg without the object enabler, make sure to remove cache files (.nwc) to allow Navisworks to re-read the file.

Navisworks has several features that every company should be aware of:

- Navisworks Navigator
- Making Insulation Transparent
- Property Display
- Scheduled Publishing
- Viewer Integrations

Navisworks Navigator

When Navisworks is installed, it links a plugin that can be loaded inside of AutoCAD Plant 3D for model viewing.

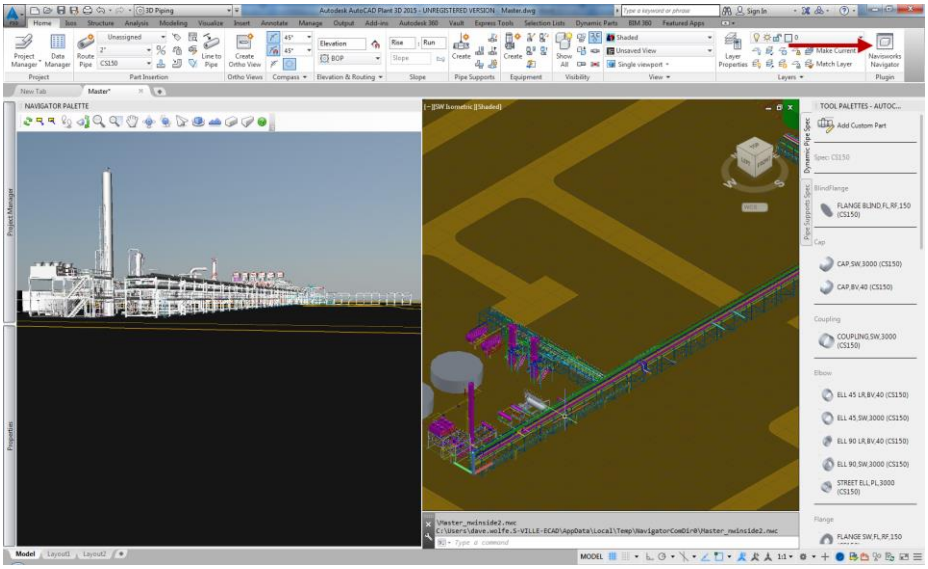


Figure 335: Navisworks Navigator

If Navisworks is installed, you will be able to click the Navisworks Navigator to open a palette that can load the current drawing in Navisworks. While the Navisworks palette won't allow you to display properties of lines, it provides a great way to review the model while working in a 2d wireframe mode for best performance.

Use the Send View to ACAD button to have AutoCAD switch to the same view as the palette, or use the Get View from ACAD button to load the current AutoCAD view in the palette.

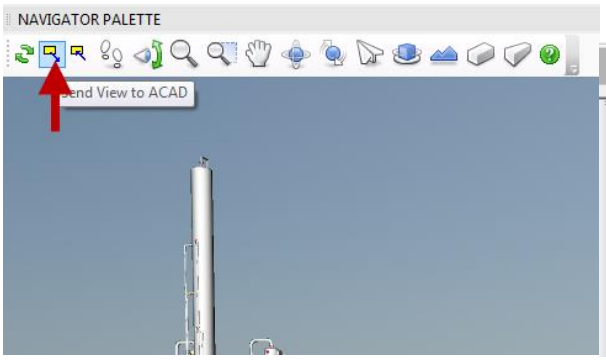


Figure 336: Send View

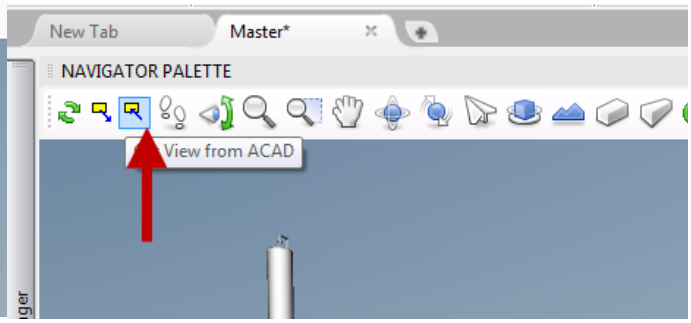


Figure 337: Get View

Using Send View to ACAD often puts AutoCAD in perspective mode, so use the OrthoCube flyout to switch back to parallel when necessary.

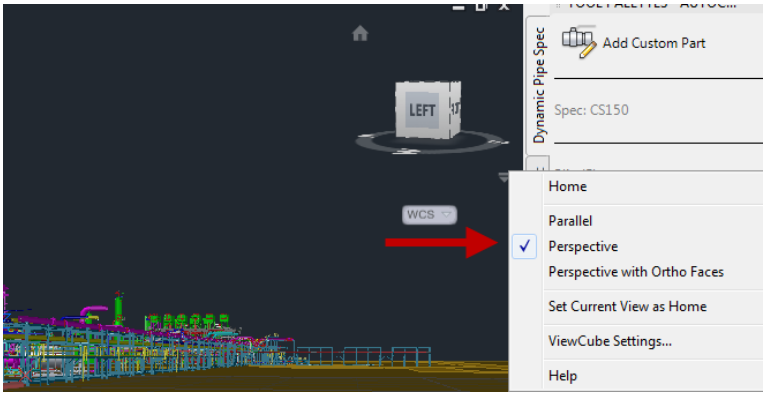


Figure 338: Change Perspective to Parallel

Insulation Display

Many times users want to display insulation in the model to aid in class detection. Having the insulation be partially transparent helps identify that there is insulation and to see the underlying pipe.

In order for insulation to show in Navisworks, PLANTINSULATIONDISPLAY must be set to 1 in AutoCAD Plant 3D.

By default, PLANTINSULATIONMODE is set to simplified partial, which shows the insulation around pipe and fittings, but breaks it for flanges and valves.

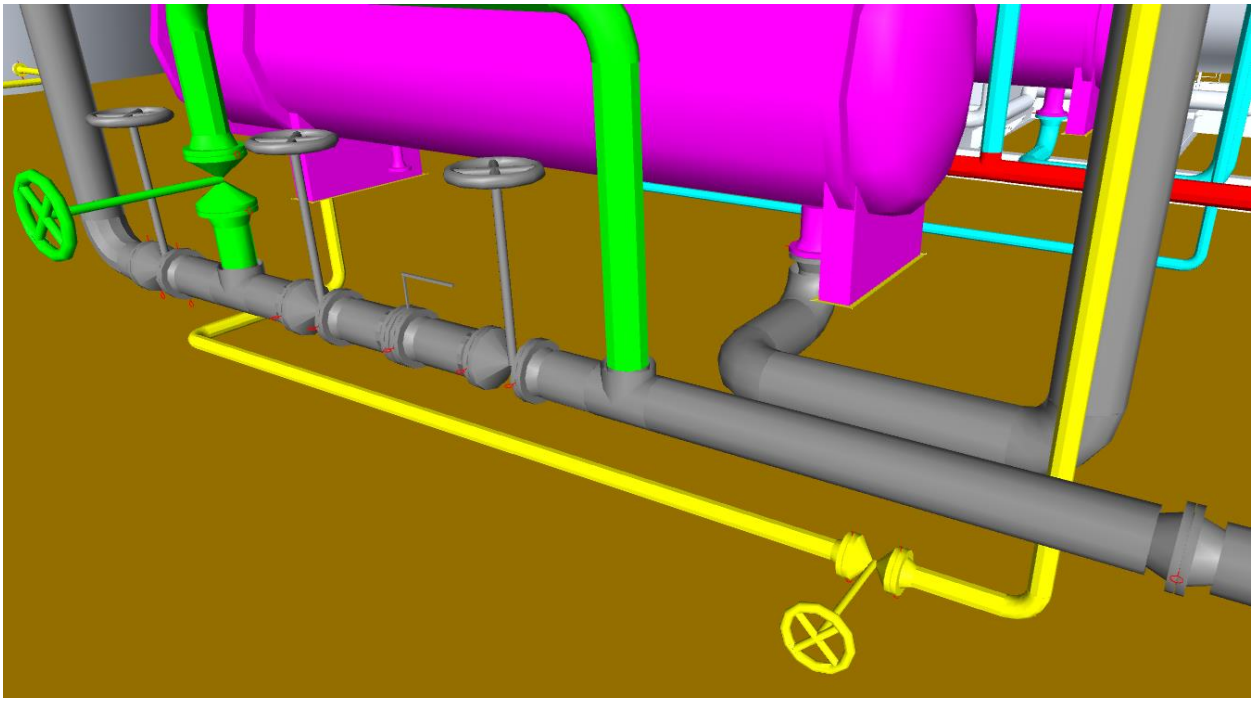


Figure 339: Insulation Non-Transparent

By using the Appearance Profiler, you can turn all the insulation in any model transparent.

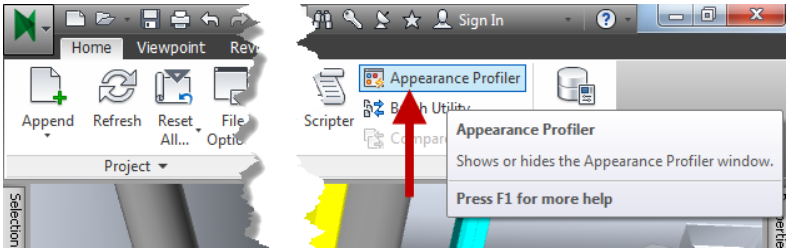


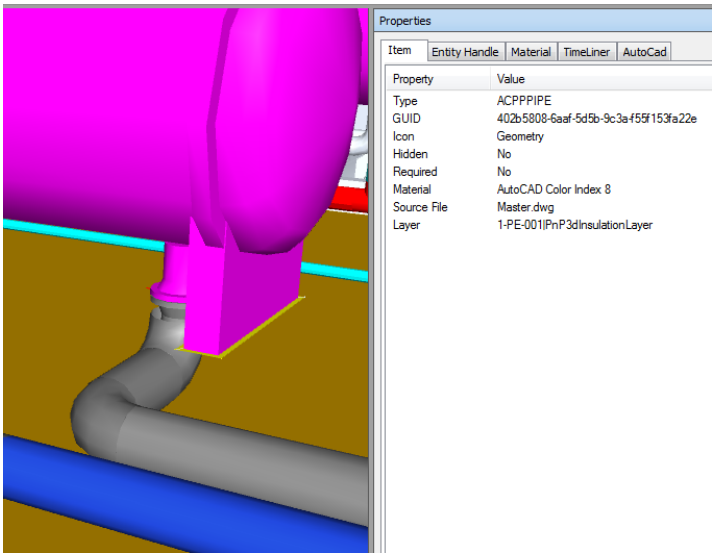
Figure 340: Appearance Profiler

You will have to perform a few setup steps:

- Create a search set for the insulation layer
- Create the profiler setting
- Run the profiler

Creating a Search Set

While many people use the selection try to navigate to items, a search set allows you to locate all instances of items with a specific property. In this case, you have to find all the objects on the P3dInsulationLayer. Typically, when using Navisworks you have xrefs loading up pipe, so you have to search within the layer name.



In this situation Naviworks loaded a model that has insulation on 1-PE-001|PnP3dInsulationLayer, so our search set will have to be able to account for nested layer names.

Figure 341: Identify Property for Search

Open the Find Items window by holding Shift+F3 or using the ribbon.

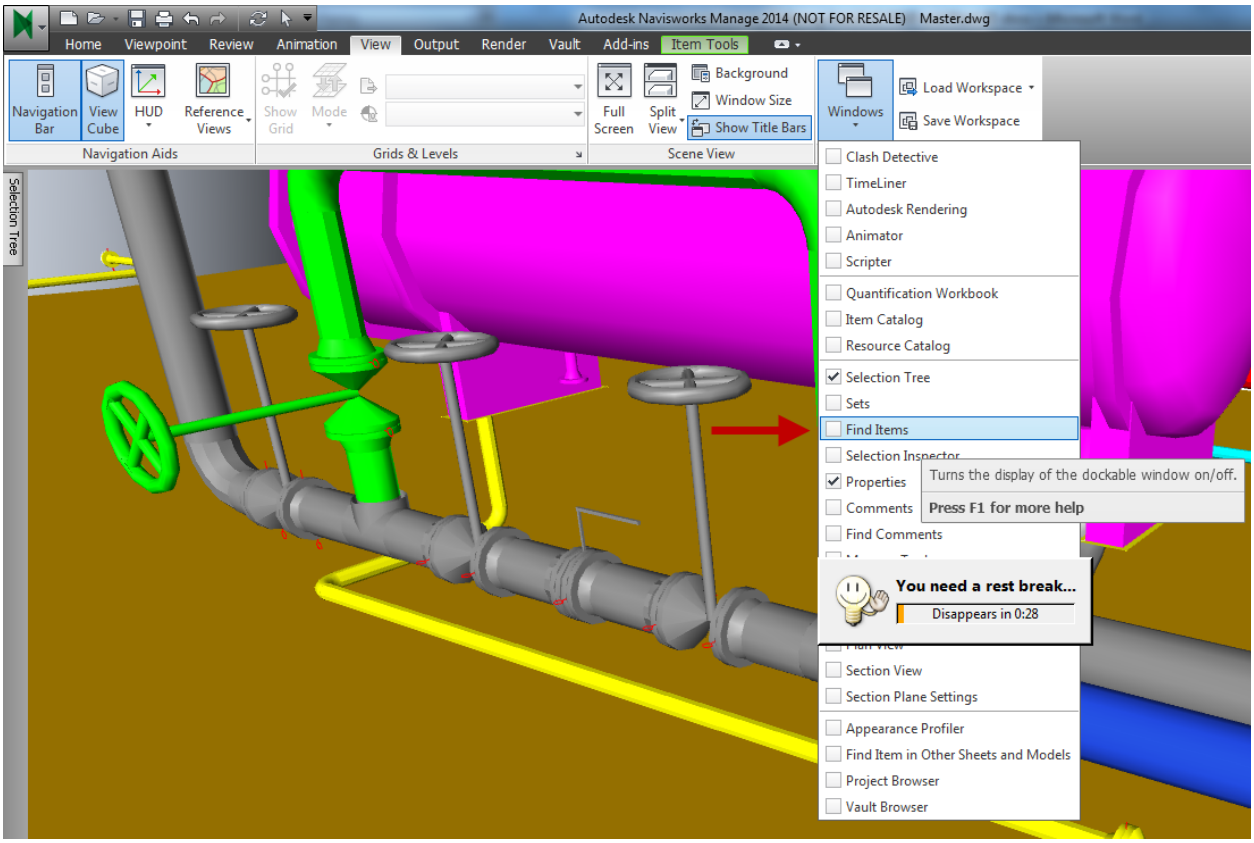


Figure 342: Open Find Items

Set up a find filter using Category=Item, Property=Layer, Condition=Contains, and Value = PnP3dInsulation. You can click Find All to test the search.

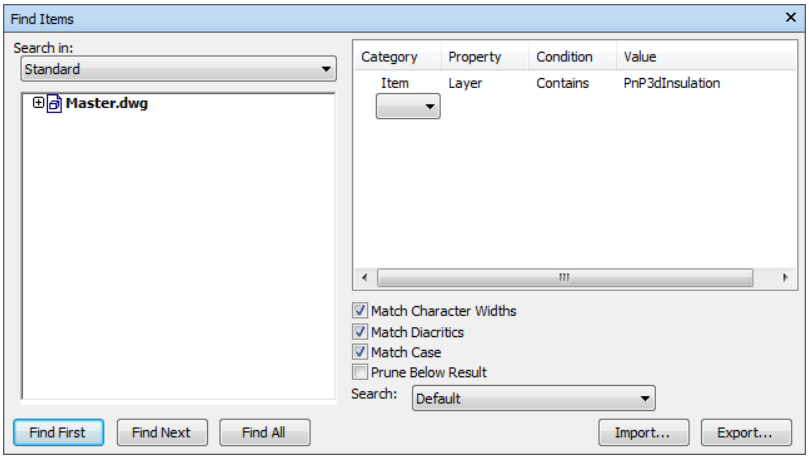


Figure 343: Find All

Open the Sets palette.

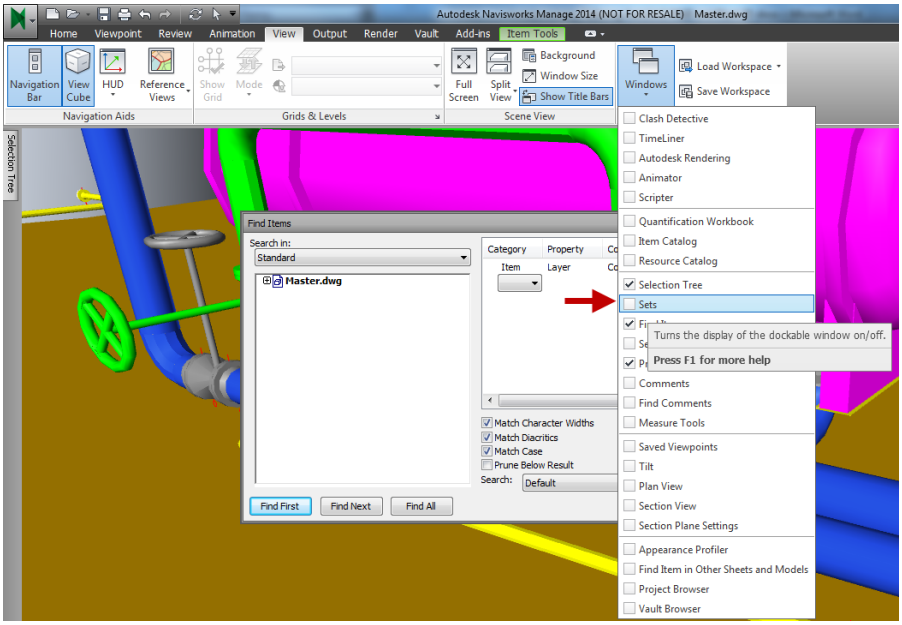


Figure 344: Open Sets palette

Click the Save Search button. You can use F2 to rename the set to Insulation. Exporting the search set will make it available on other models.

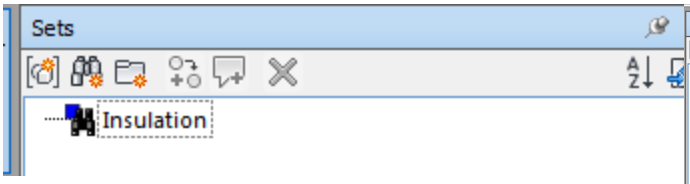


Figure 345: Save the Search

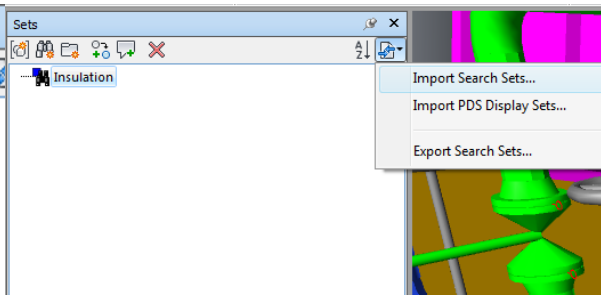


Figure 346: Export Search

Create Profiler Settings

Open the Appearance Profiler, and switch to the By Set tab. You may have to click Refresh to see the search set. Select the search set, change the transparency to 55% and click Add. Click Run to apply the profile. Save the settings to apply the appearance profile settings to another model.

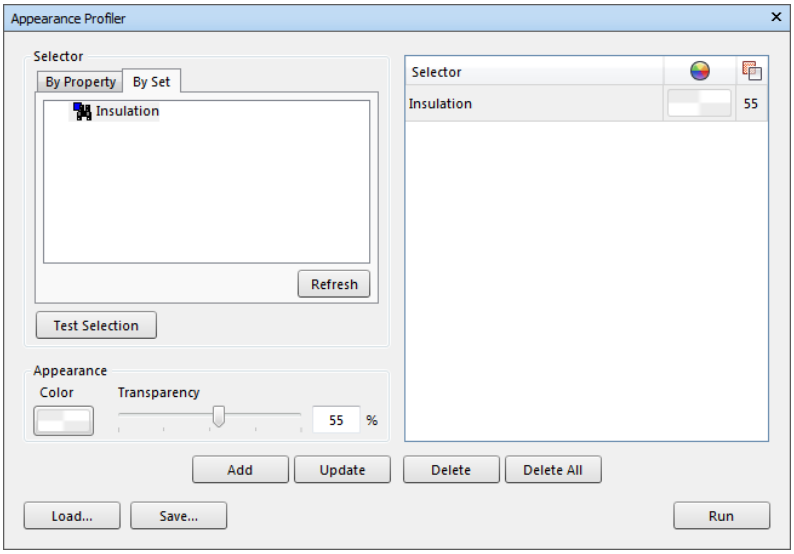


Figure 347: Create Profiler Settings

Displaying Properties

Selecting an object allows you to view its properties in the properties palette. You can also use the selection tree in conjunction with properties. Because the AutoCAD Plant 3D properties are integrated into Navisworks, you can create searches on them, as shown for the Layer. The properties on the AutoCAD tab are pre-defined and include basic part information.

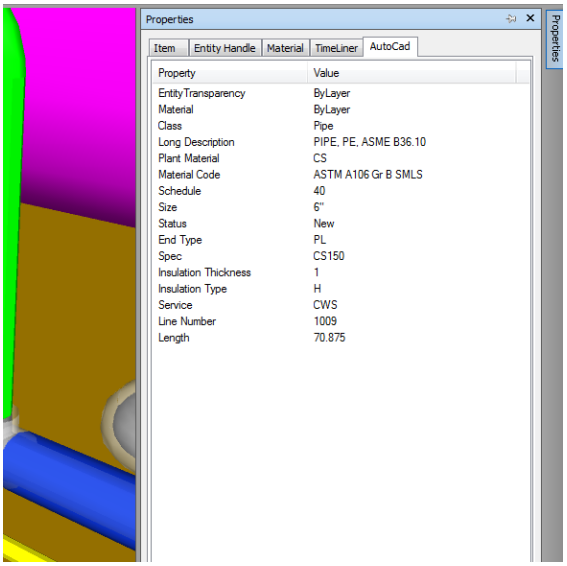


Figure 348: View Properties

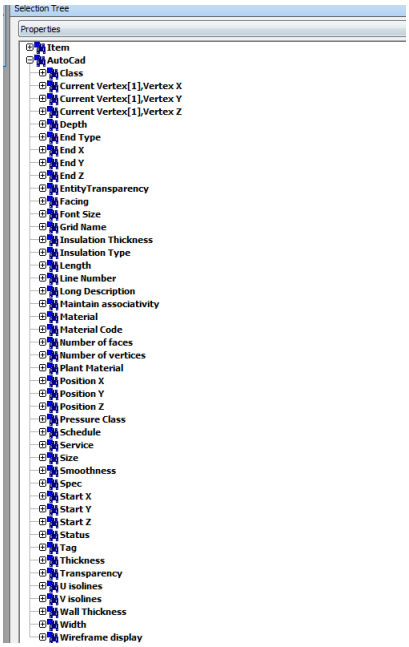


Figure 349: Selection Tree Properties

To show additional properties, you can use the DataLink plugin, which embeds all the project properties into the Navisworks model.

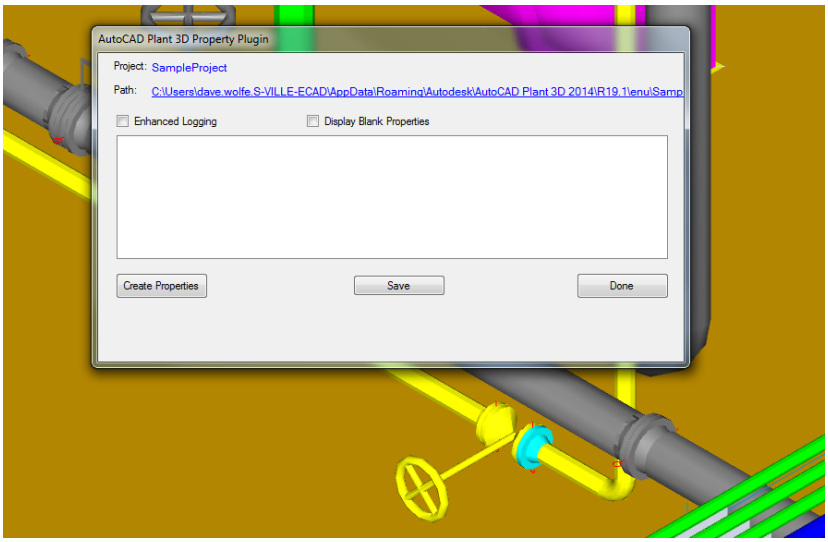


Figure 350: Embed Properties

All non-blank AutoCAD Plant 3D part properties are applied as well as line group information. In addition, url's are converted so they can be clicked inside of Navisworks.

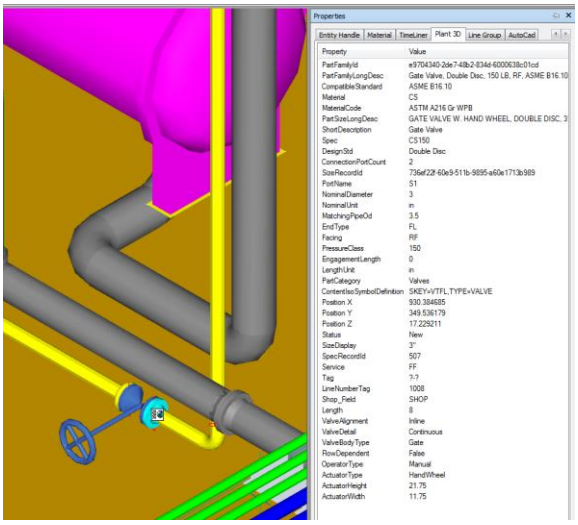


Figure 351: All Properties

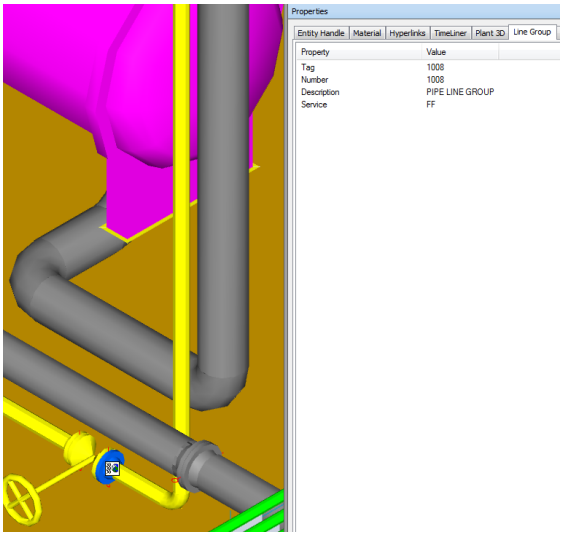


Figure 352: Line Group Properties

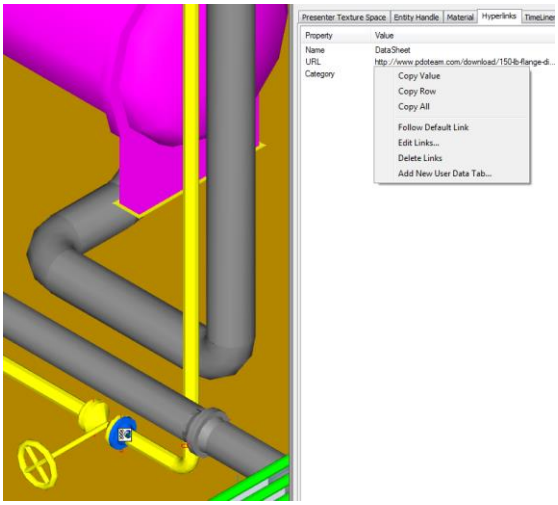


Figure 353: Published Hyperlinks

Publishing

One of the most important aspects of Navisworks is publishing. When publishing Navisworks models, you can take all models from different systems and platforms (Revit, AutoCAD, Inventor, Autoplant, etc.) and include them in one single file that can be distributed and reviewed. All of the properties can be included or excluded. If they are included in the published file, users viewing the model using Navisworks Freedom (free viewer) can identify object in the properties palette as well.

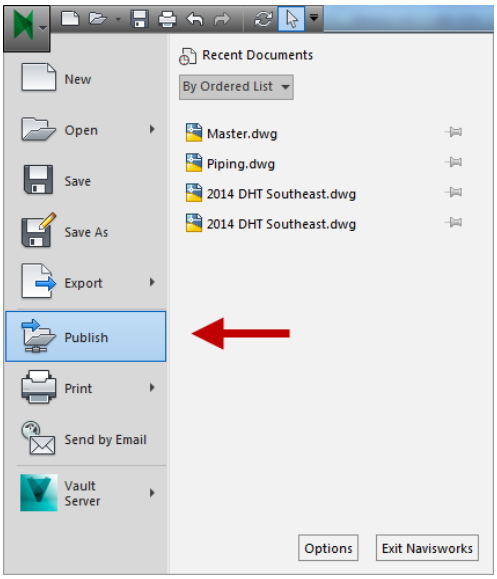


Figure 354: Publish File

In addition to manually publishing files, Navisworks has a scripting api which allows you to build a script, and then publish several files into one model on a scheduled basis. For an example, visit http://www.pdoteam.com/2012/04/automating-model-coordination-for-navisworks/

Reviewing

Autodesk offers many applications that allow you to review your design information.

Autodesk Navisworks Freedom is a free viewer that allows you to view property information.

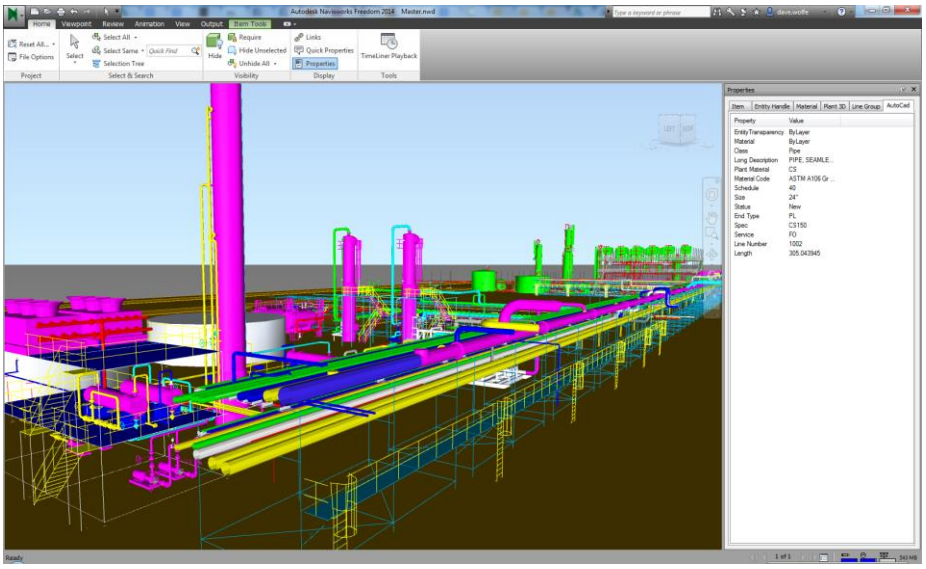


Figure 355: Navisworks Freedom

Autodesk BIM 360 Glue is available for ios devices, and Windows desktop computers.

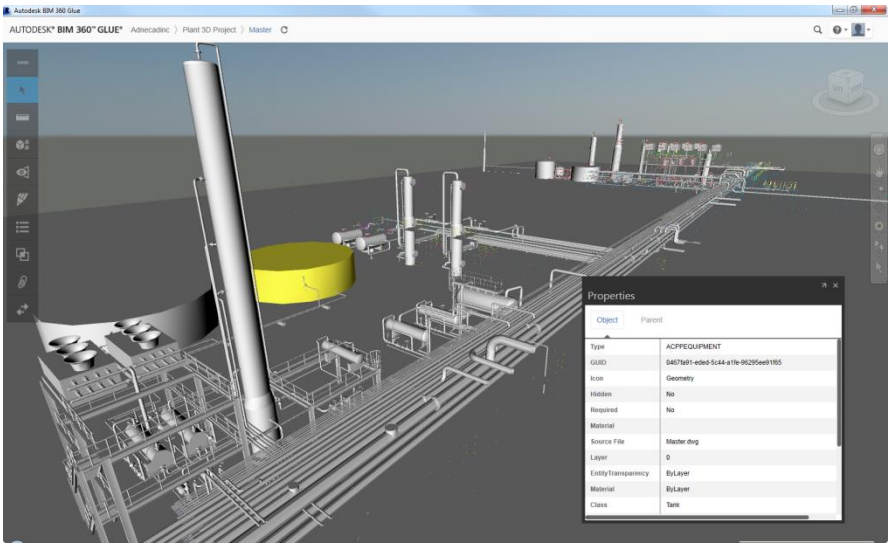


Figure 356: Mobile Devices

Autodesk 360

Autodesk 360 is a program that allows storing and sharing information in various formats. For a Navisworks model of the sample project hosted on Autodesk 360, click this link:

- http://a360.co/1guzm6J

AutoCAD Structural Detailing

AutoCAD Structural Detailing, or ASD, also uses object enablers. However, the best practice for ASD models it to export the files to AutoCAD as well. Due to ASD models having high detail, and the way the dlls load, loading the object enablers into AutoCAD Plant 3D can cause performance issues.

From within AutoCAD Structural Detailing, click the Tools Expand button, and then choose Save model as ACIS solids. Choose a file location, and select object (type in all and hit enter). Choose simplified solids after all the selections have been made. Simplified solids clean up bolts and other detailed objects which helps model performance.

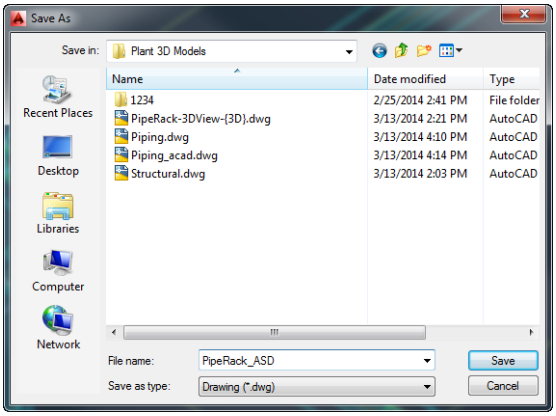


Figure 357: ASD Export

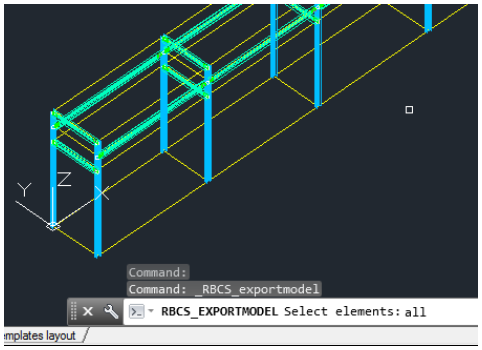


Figure 358: ASD Select Elements

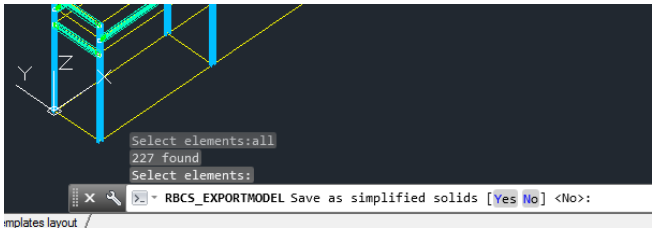


Figure 359: ASD Simplified Model Save

You may use the exported model as an AutoCAD reference. Like the Revit export, you should reference it into a master structural model so that updating the file does not cause you to lose modifications.

AutoCAD MEP

AutoCAD MEP models can be used like any AutoCAD external reference, and referenced when generating orthographics. The drawing file must be added to the project or referenced as an attachment.

Contributions

Successful Projects using AutoCAD P&ID – Greg Hoover, Veolia Water

AutoCAD P&ID in a Multi-User Environment – Barend-Jan van Zijtveld and Gerald te Wierik

De-mystifying AutoCAD Isometrics – David Wolfe

Managing a complete site over multiple AutoCAD® Plant 3D Projects – Jarrod Mudford

How to Manage Large Projects in AutoCAD Plant 3D

Extras in P&ID Customization – Lynn Levy, Mangesh Kalbhor, Atilio Zelaya

Customizing AutoCAD Plant 3D Isometrics – Joel Harris, Ian Matthew