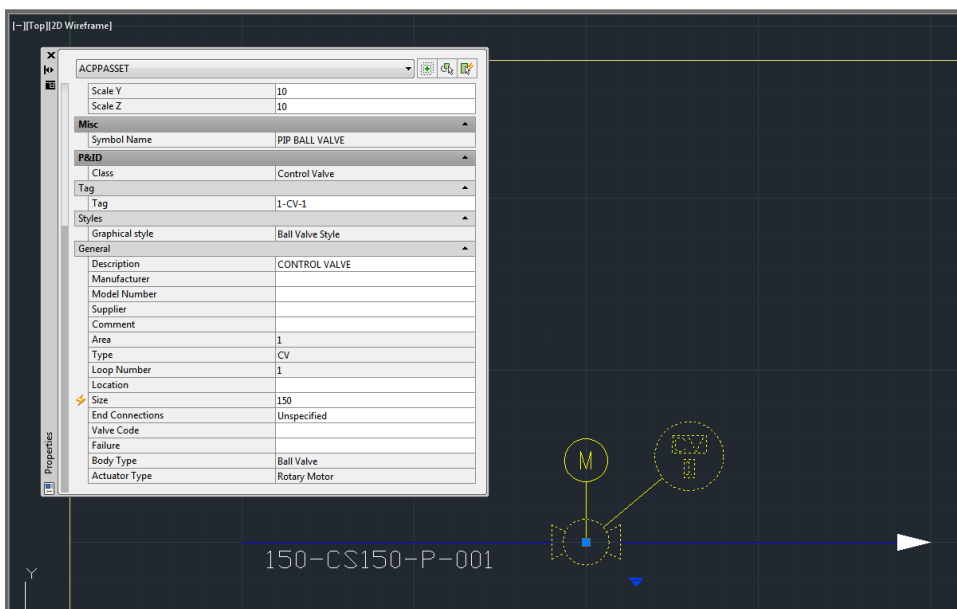


Linking Actuators to Control Valves

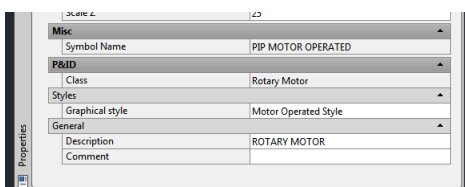
AutoCAD P&ID 2014

When you place a Control Valve in a drawing in AutoCAD P&ID, you can use the Control Valve Browser window to select a valve body and an associated type of actuator. When you do this, a grouped symbol is created that combines the two. In the Data Manager, 2 records are created, one Control Valve record and one record for the type of actuator that you selected; we will assume a Rotary Motor in this discussion.

The default project configuration copies the class name of the chosen valve body into the Valve Body property of the Control Valve record, and the class name of the chosen actuator into the Actuator Type property, otherwise no further link is created.



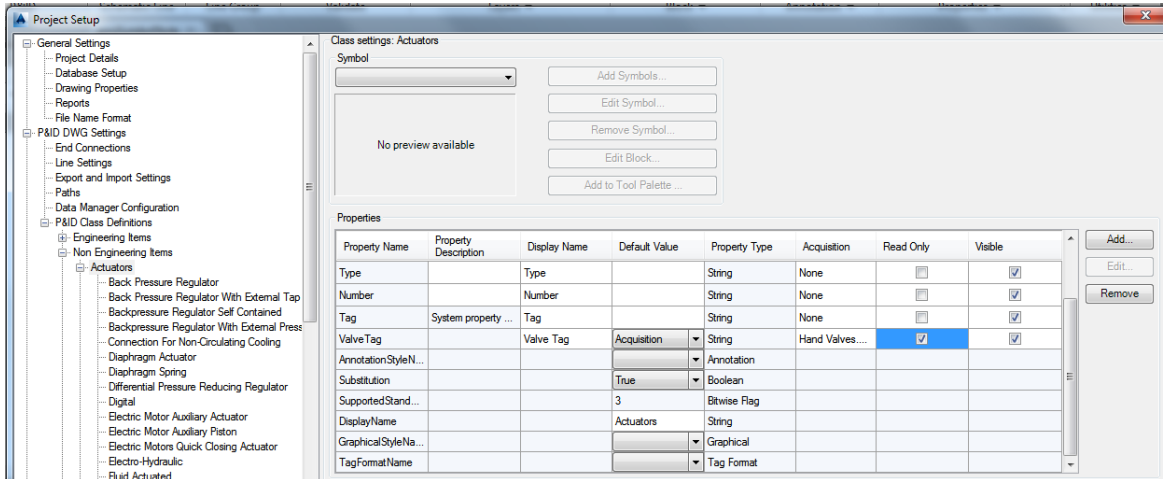
Additionally, it is not possible to add a Tag to the Rotary Motor as the project configuration does not provide a Tag property or Tag Format for the Actuators class.



This White Paper describes how to add enable the tagging of Actuators, how to link the Tag of a Control Valve to its Actuator and how to link the Tag of an Actuator back to the Control Valve that it operates.

Customise the Actuators Class and Subclasses

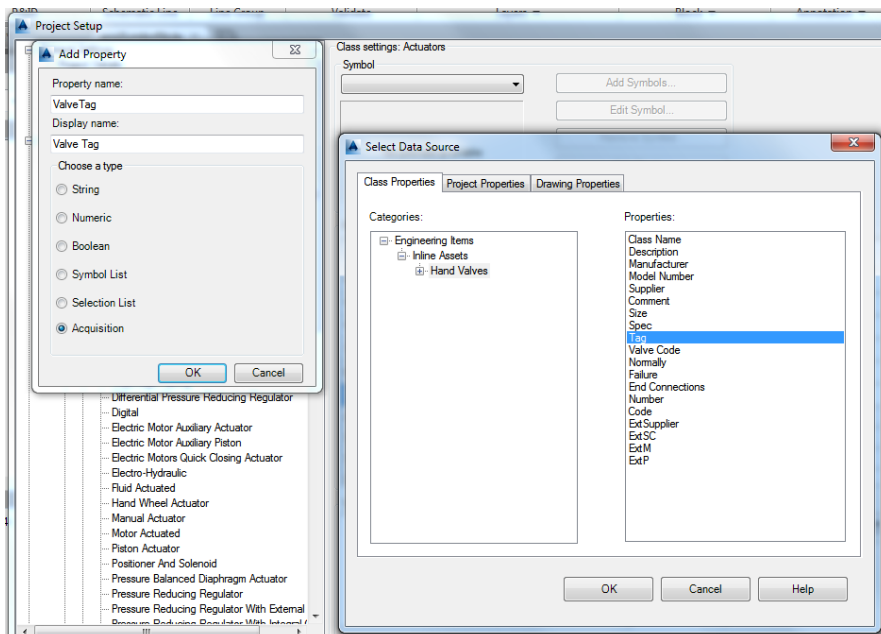
The starting point for our customisation is to add some new properties to the Actuators class in Project Setup.



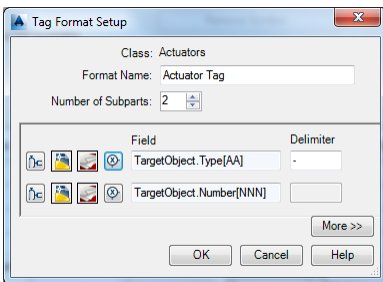
We will add three String type properties: Type, Number and Tag – these will enable us to tag Actuators. Adding a property called Tag automatically adds the tagging functionality to a P&ID class.

We also need to add an Acquisition property that will link the Tag of an associated Control Valve to the Actuator, we will call this ValveTag. Unfortunately, when we try to do this, the user interface only provides us with the ability to link to the Hand Valves class rather than the Control Valve class that we require.

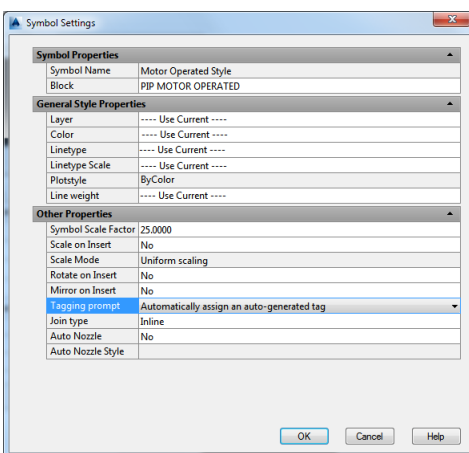
For now, we will go ahead and link the HandValves.Tag property to our new ValveTag property. We will revisit this later.



Next, we will add a Tag Format called Actuator Tag as shown below and make it the default TagFormatName for Actuators.

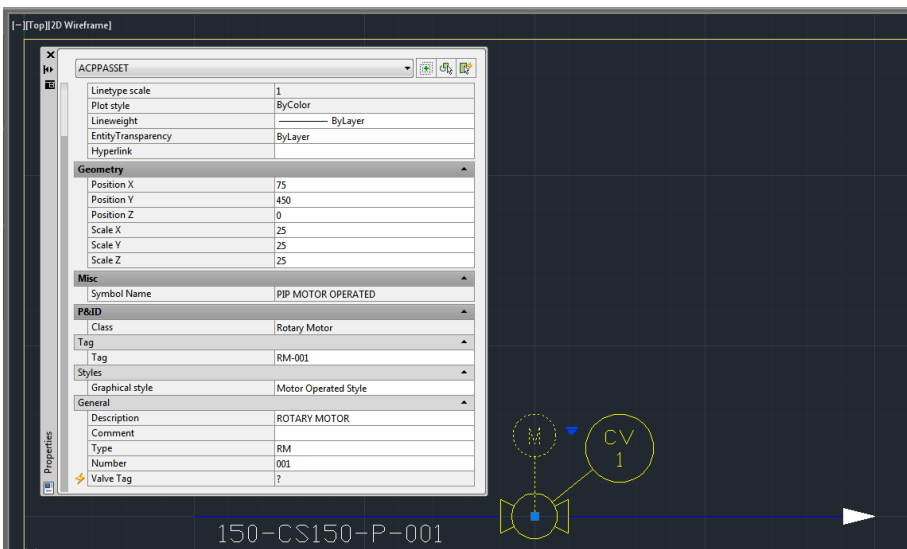


To enable the tagging of an actuator, we must update its Symbol Settings and set the Tagging prompt to a value other than Not a tagged component. Otherwise, we will not be able to assign a tag.



We are now able to assign a tag to our Rotary Motor; the Valve Tag field is currently empty but we will address this shortly.

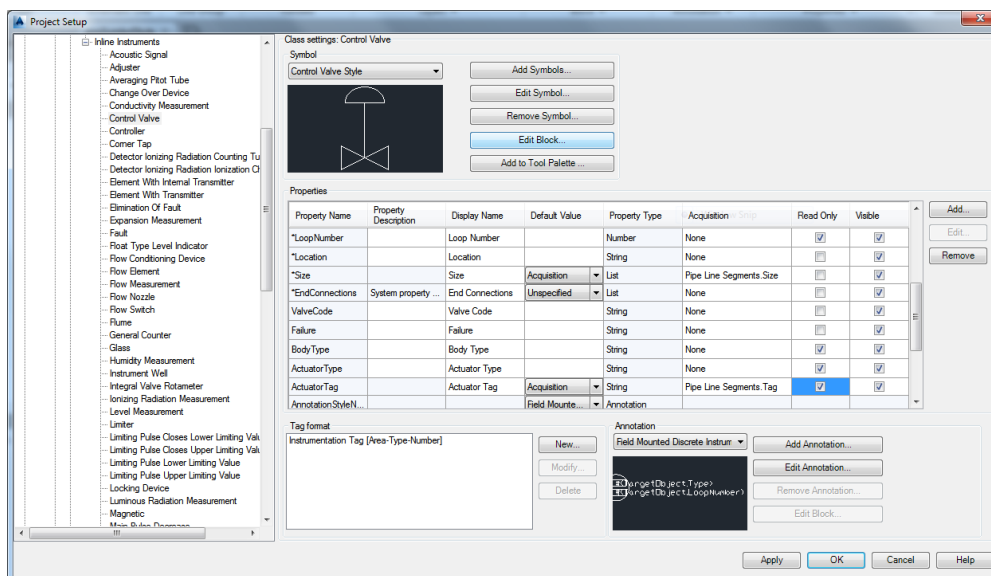
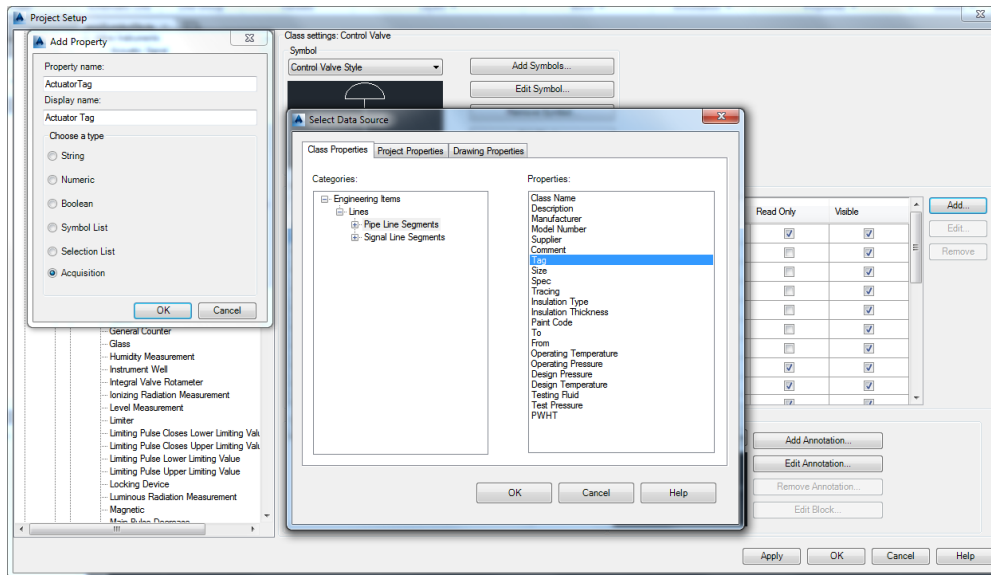
It is important to note that the Assign Tag... option on the Actuator RMB context menu does not allow us to tag our actuator, but we are able to accomplish this through either the Properties window or the Data Manager window.



Customise the Control Valve Class

We now need to turn our attention to the Control Valve class. We need to add an Acquisition property that will link the Tag of an associated Actuator to the Control Valve, we will call this ActuatorTag. Unfortunately, when we try to do this, the user interface only provides us with the ability to link to the Lines class rather than the Actuators class that we require.

We will go ahead and link the PipeLines.Tag property to our new ActuatorTag property. But, we will revisit this later.





Fix-up the ValveTag and ActuatorTag Acquisition Properties



The ValveTag and ActuatorTag properties that we defined through Project Setup must now be modified to link to their correct targets. Before we do this, we must close the project in AutoCAD P&ID.

All user-defined acquisition properties and acquisition rules are stored in an XML format configuration. The name of the configuration file varies depending upon the standard upon which the project is based. Although the initial part of the filename varies, it always ends in 'PnIdPart.xml' and is located in the project folder. The file for a metric project will bear the prefix 'DIN_', 'ISO_' or 'PIP_Metric_'. In our case, we have a metric project based on the PIP standard, so we need to look for the file 'PIP_Metric_PnIdPart.xml'.

We can edit the file using Notepad or any other text editor. When we have opened the file we need to find the following section, searching for the word Actuators should locate this in the file. This is the definition of our 2 acquisition rules. We simply need to change the values within the <TableName> tags from HandValves to ControlValve and from PipeLines to Actuators as highlighted below.

| | | |
|--|--|---|
| <pre> </PnPRule> <PnPRule xsi:type="PnPPropertyAcquisitionRule"> <Name>Actuators.ValveTag_PnPPropertyAcquisitionRule</Name> <Classname>Actuators</Classname> <Operator xsi:type="PnPAnyRelatedTableColumn"> <TableName>HandValves</TableName> <ColumnName>Tag</ColumnName> </Operator> <InitializeOnly>false</InitializeOnly> <NotOverridable>false</NotOverridable> <ColumnName>ValveTag</ColumnName> </PnPRule> <PnPRule xsi:type="PnPPropertyAcquisitionRule"> <Name>ControlValve.ActuatorTag_PnPPropertyAcquisitionRule</Name> <Classname>ControlValve</Classname> <Operator xsi:type="PnPAnyRelatedTableColumn"> <TableName>PipeLines</TableName> <ColumnName>Tag</ColumnName> </Operator> <InitializeOnly>false</InitializeOnly> <NotOverridable>false</NotOverridable> <ColumnName>ActuatorTag</ColumnName> </PnPRule> </pre> |   | <pre> <PnPRule xsi:type="PnPPropertyAcquisitionRule"> <Name>Actuators.ValveTag_PnPPropertyAcquisitionRule</Name> <Classname>Actuators</Classname> <Operator xsi:type="PnPAnyRelatedTableColumn"> <TableName>ControlValve</TableName> <ColumnName>Tag</ColumnName> </Operator> <InitializeOnly>false</InitializeOnly> <NotOverridable>false</NotOverridable> <ColumnName>ValveTag</ColumnName> </PnPRule> <PnPRule xsi:type="PnPPropertyAcquisitionRule"> <Name>ControlValve.ActuatorTag_PnPPropertyAcquisitionRule</Name> <Classname>ControlValve</Classname> <Operator xsi:type="PnPAnyRelatedTableColumn"> <TableName>Actuators</TableName> <ColumnName>Tag</ColumnName> </Operator> <InitializeOnly>false</InitializeOnly> <NotOverridable>false</NotOverridable> <ColumnName>ActuatorTag</ColumnName> </PnPRule> </pre> |
|--|--|---|

We also need to fix up 2 other sections within the file that will trigger the values of the acquisition properties to update whenever the tags change. Again, we simply need to change the values within the <TableName> tags from HandValves to ControlValve and from PipeLines to Actuators as highlighted below.

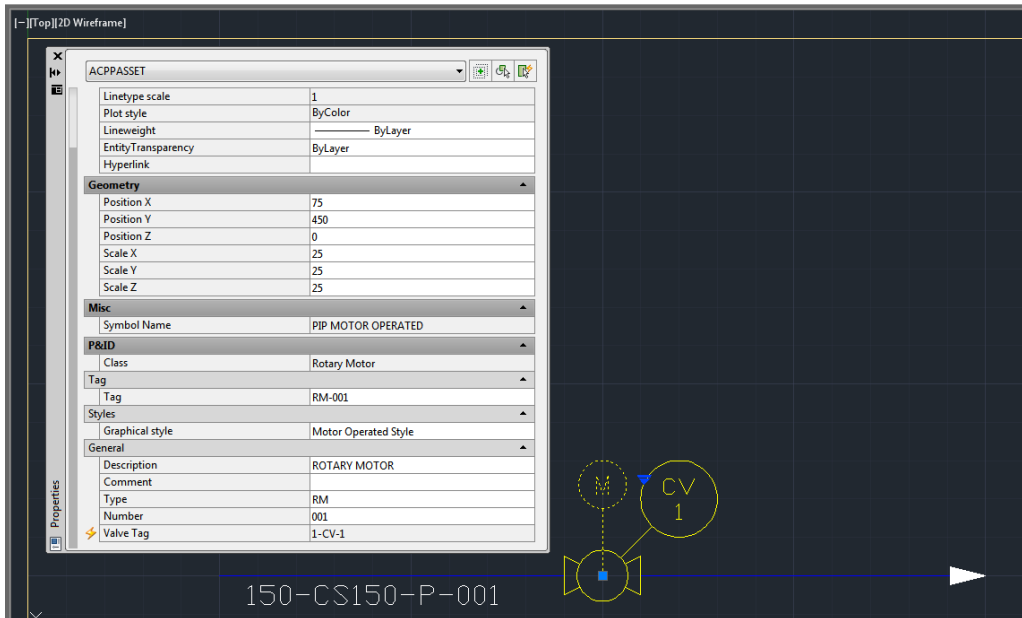
| | | |
|--|---|--|
| <pre> </PnPTrigger> <PnPTrigger> <Name>HandValves_Update_Actuators.ValveTag_PnPPropertyAcquisitionRule</Name> <RuleName>Actuators.ValveTag_PnPPropertyAcquisitionRule</RuleName> <TableName>HandValves</TableName> <Action>Update</Action> <Target>Related</Target> </PnPTrigger> </pre> |  | <pre> <PnPTrigger> <Name>HandValves_Update_Actuators.ValveTag_PnPPropertyAcquisitionRule</Name> <RuleName>Actuators.ValveTag_PnPPropertyAcquisitionRule</RuleName> <TableName>ControlValve</TableName> <Action>Update</Action> <Target>Related</Target> </PnPTrigger> </pre> |
| <pre> </PnPTrigger> <PnPTrigger> <Name>PipeLines_Update_ControlValve.ActuatorTag_PnPPropertyAcquisitionRule</Name> <RuleName>ControlValve.ActuatorTag_PnPPropertyAcquisitionRule</RuleName> <TableName>PipeLines</TableName> <Action>Update</Action> <Target>Related</Target> </PnPTrigger> </pre> |  | <pre> <PnPTrigger> <Name>PipeLines_Update_ControlValve.ActuatorTag_PnPPropertyAcquisitionRule</Name> <RuleName>ControlValve.ActuatorTag_PnPPropertyAcquisitionRule</RuleName> <TableName>Actuators</TableName> <Action>Update</Action> <Target>Related</Target> </PnPTrigger> </pre> |

We can now save and close the file.

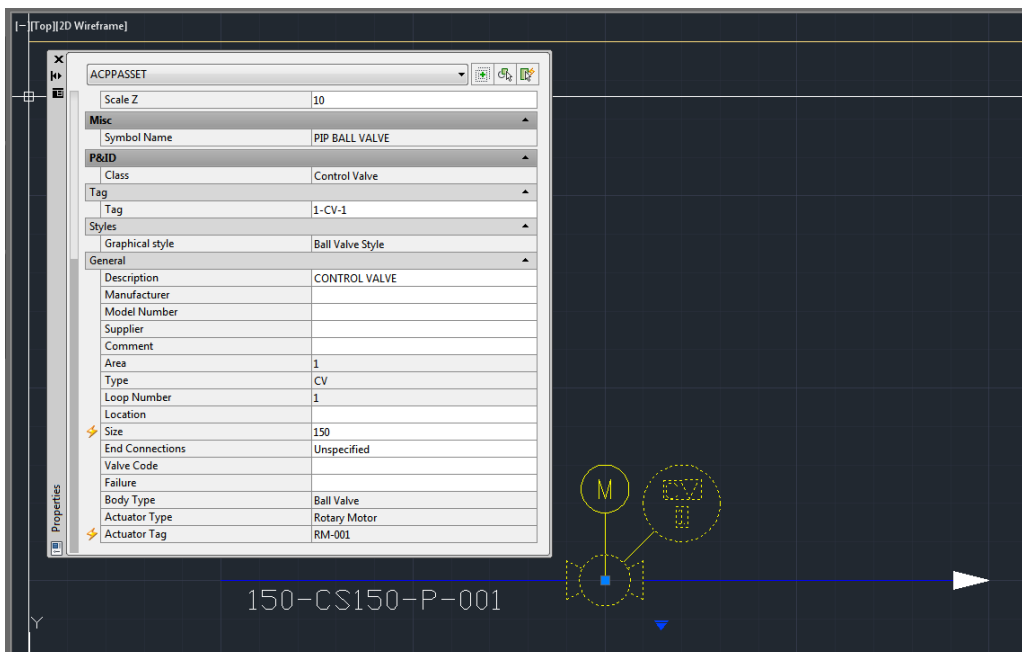
Open the Project

We have now completed our customisation. When we re-open the project our changes will be live.

We can tag our actuator and see the tag of the Control Valve that it operates.



Similarly, we can see the tag of the actuator that we have assigned to our Control Valve.



These new properties are automatically available for use in our schedules and other project reports.