



OpenPlant PowerPID

How to Pass Service from Line to Run and on to Inline Component at Creation and Post Creation

Version 1.4

Initial Release December 14, 2012

How to Pass Service from Line to Run and on to Inline Component at Creation and Post Creation

Trademarks

Bentley, the “B” Bentley logo, MicroStation, ProjectWise and AutoPLANT are registered trademarks of Bentley Systems, Inc or Bentley Software, Inc.

Microsoft, Windows and SQL Server are registered trademarks of Microsoft Corporation.

AutoCAD is a registered trademark of Autodesk, Inc.

Oracle 11g, 10g and Oracle 9i are registered trademarks of Oracle Corporation.

Other brands and product names are the trademarks of their respective owners.

Copyrights

© 2010 Bentley Systems, Incorporated

No part of this document may be reproduced, translated, or transmitted in any form or by any means, electronic or mechanical, for any purpose without the express written permission of Bentley Systems, Inc, and then only on the condition that this notice is included in any such reproduction. No information as to the contents of this document may be communicated to any third party without the prior written consent of Bentley Systems, Inc.

Information in this document is subject to change without notice and does not represent a commitment on the part of Bentley Systems, Inc. Bentley Systems, Inc. is not liable for errors contained in this document or for incidental or consequential damages in connection with furnishing or use of this material.

Including software, file formats, and audiovisual displays; may only be used pursuant to applicable software license agreement contains confidential and proprietary information of Bentley Systems, Inc. and/or third parties which is protected by copyright and trade secret law and may not be provided otherwise made available without proper authorization.

RESTRICTED RIGHTS LEGENDS

Use, duplication, or disclosure by the government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Right in Technical Data and Computer Software clause in DFARS 252.227-7013 or subparagraphs (c)(1) and (2) of the Commercial Computer Software – Restricted Rights in 48 CFR 52.227-19 as applicable.

Unpublished – rights reserved under the Copyright Laws of the United States and International treaties.

Downloaded files from Bentley-related Internet websites and files included on MySELECT CDs are subject to the legal terms, conditions, policies and usage restrictions posted on the website(s), which may significantly alter the rights granted in the license agreement included with these materials.

How to Pass Service from Line to Run and on to Inline Component at Creation and Post Creation

Table of Contents

Document Version History.....	4
Software Versions.....	4
Assumptions prior to starting.....	6
OpenPlant Power PID and Schema Editing.....	6
Objective.....	6
General Overview of Required Administrative Task.....	6
OpenPlant PowerPID Schemas.....	6
Verifying the default BMF schema.....	7
Configuring Pipeline to Pass Values.....	8
Configuring Pipe Run to Receive and Pass Values.....	9
Configuring Valve and Instrument to Receive and Use Values.....	12
Setting SERVICE Property on the Component to Read Only.....	12
Testing the Results.....	13
Notes.....	15

How to Pass Service from Line to Run and on to Inline Component at Creation and Post Creation

Document Version History

Version	Date	Author	Comments
1.0	12/14/2012	Tony DeRosa	Initial Release
1.1	12/17/2012	Tony DeRosa	Removed Value Property Template attribute – not needed for passing property only.
1.3	06/10/2013	Tony DeRosa	Made Notify for piping components class specific; Made SERVICE read only on specific components; Miscellaneous document format changes.
1.4	06/26/2013	Tony DeRosa	Added special provision for inline Instruments (Flow Elements)

Reviewed By	Date	Approved By	Date

Software Versions

Application Name	Version
OpenPlant PowerPID	08.11.10.38 (SS5 and higher)
Bentley Class Editor	From PowerPID install

How to Pass Service from Line to Run and on to Inline Component at Creation and Post Creation

How to Pass Service from Line to Run and on to Inline Component at Creation and Post Creation

Assumptions prior to starting

OpenPlant Power PID and Schema Editing

- A general understanding and use of the Class Editor.

Note: A good understanding of OpenPlant PowerPID schemas is also assumed.

Objective

This session will cover the process of using an existing **Associated Item** (In this case SERVICE) and passing this value to an inline component. Specifically you will pass this value to a valve but the same procedure could be used to pass the value to all inline components. This specific method will work with an **ASSOCIATED PROPERTY** and not a standard common property such as Description. Common properties like Description are passed from one class to another using a different method.

General Overview of Required Administrative Task

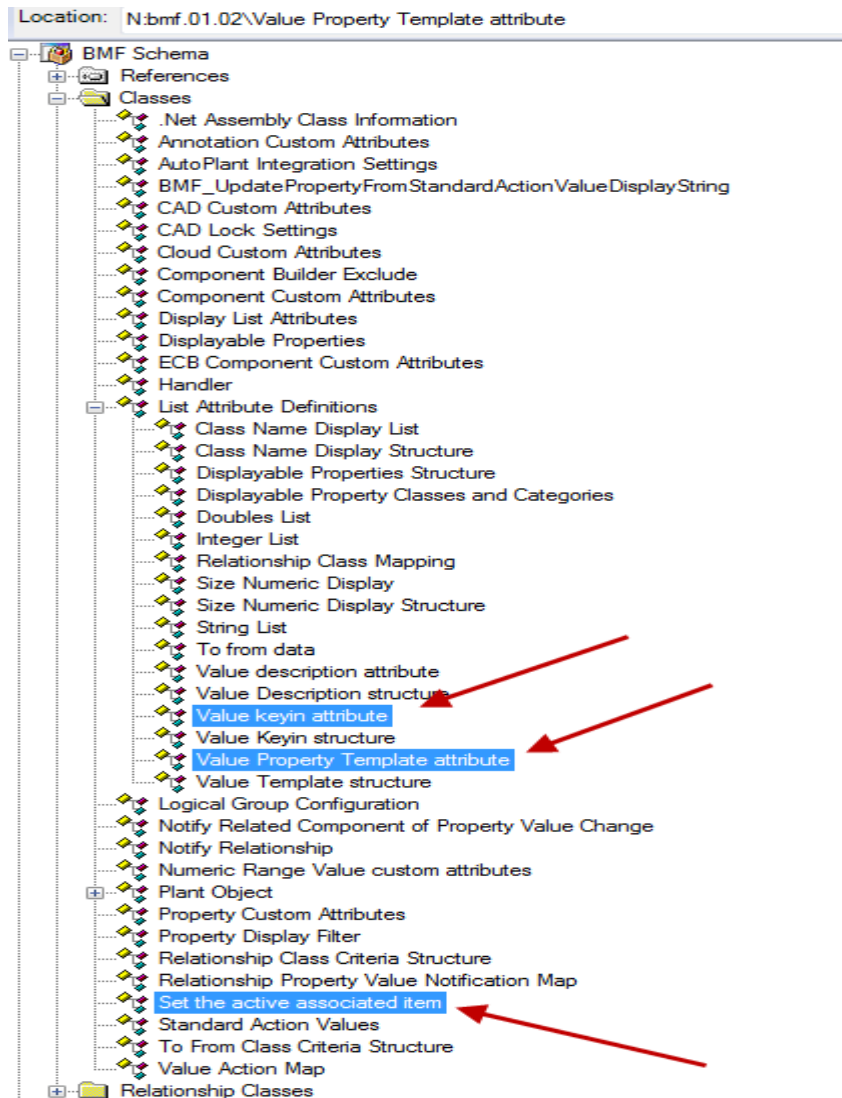
OpenPlant PowerPID Schemas

- Verify the BMF default schema.
- Set the class for pipeline to use the correct properties to set Service.
- Pass the Service property on Pipeline to Design Service on Pipe Run.
- Add Design Service to Pipe Run and pass this to Valve.
- Set Valve to use the inherited Service value and not the default Service value.

How to Pass Service from Line to Run and on to Inline Component at Creation and Post Creation

Verifying the default BMF schema

- Load the **Class Editor**
- From within the Class Editor load the **bmf.XX.XX.ecschema.xml** schema
- Expand the **Classes** node tree in the left hand pane and verify that the class names shown below are present in the schema.



How to Pass Service from Line to Run and on to Inline Component at Creation and Post Creation

Configuring Pipeline to Pass Values

- Load the **Class Editor**
- From within the Class Editor load the **OpenPlant_PID.XX.XX.ecschema.xml** file located in the project schemas folder to be modified.
- Once loaded right click on the schema and from the context menu select **Supplement Schema...** and pick the **OpenPlant_PID_Supplemental_Imperial.XX.XX.ecschema.xml** file.
- Expand the tree in the left hand pane and pick on the **Pipeline** (Class Name: PIPING_NETWORK_SYSTEM) class. Alternatively you can use the search tool to locate the class.
- Select the **Properties** tab.
- Find the **SERVICE** property and pick the **Override** button.
- With the **SERVICE** property selected pick the **Custom Attributes...** button then select **Add/Remove...**
- Add the **Notify Related Component of Property Value Change** (Class Name: BMF_NOTIFY_RELATED_COMPONENT_OF_PROPERTY_VALUE_CHANGE)
- Fill in the custom attribute as shown below.

Notify Related Component of Property Value Change	
<input type="checkbox"/> Notify Relationships	
<input type="checkbox"/> Notify Relationships[0]	
Notify Target of Relationship	True
Relationship Class Name	appid:PIPELINE_HAS_SEGMENT
Related Class Name	appid:PIPING_NETWORK_SEGMENT
Related Class Property Name	DESIGN_SERVICE
Cascade Property Value change	True
Set Default Property Value	True

- **Save** the schemas.

How to Pass Service from Line to Run and on to Inline Component at Creation and Post Creation

Configuring Pipe Run to Receive and Pass Values

- Continuing in the **Class Editor**
- Select the **Pipe Run** (Class Name: PIPING_NETWORK_SEGMENT) class in the left hand pane and in the right hand pane pick on the **Properties** tab.
- On the **Properties** tab pick the **Add...** button. Add the new property, **DESIGN_SERVICE**, as shown below.

DESIGN_SERVICE	
Name	DESIGN_SERVICE
OriginClass	appid:PIPING_NETWORK_SEGMENT
Overrides	False
DisplayLabel	Design Service
TypeName	<input type="checkbox"/> string
Description	
isArray	False
MinOccurs	1
MaxOccurs	1
Read Only	False
Priority	200

- **Save** the schema.
- Continuing in Class Editor and the on the **DESIGN_SERVICE** property add the **Category**, **Notify Related Component of Property Value Change** and **Property Display Filter** custom attributes.
- Fill in the custom attributes as shown below.

How to Pass Service from Line to Run and on to Inline Component at Creation and Post Creation


Property Display Filter	
IsVisible	True
IsReadOnly	True
Notify Related Component of Property Value Change	
<input type="checkbox"/> Notify Relationships	
<input type="checkbox"/> Notify Relationships[0]	
Notify Target of Relationship	True
Relationship Class Name	RUN_HAS_IN_RUN
Related Class Name	oppid:PIPING_COMPONENT
Related Class Property Name	SERVICE
Cascade Property Value change	True
Set Default Property Value	False
<input checked="" type="checkbox"/> Notify Relationships[1]	
Notify Target of Relationship	False
Relationship Class Name	oppid:DATA_CHANGE_CONNECTS_TO_RUN
Related Class Name	
Related Class Property Name	SERVICE
Cascade Property Value change	True
Set Default Property Value	False
<input type="checkbox"/> Notify Relationships[2]	
Notify Target of Relationship	False
Relationship Class Name	oppid:END_RUN_CONNECTS_TO_RUN
Related Class Name	
Related Class Property Name	SERVICE
Cascade Property Value change	True
Set Default Property Value	False
<input type="checkbox"/> Notify Relationships[3]	
Notify Target of Relationship	True
Relationship Class Name	oppid:OBJECT_HAS_INSTRUMENT
Related Class Name	
Related Class Property Name	SERVICE
Cascade Property Value change	True
Set Default Property Value	False
Category	
Standard	0
Name	SERVICE_VALUE
DisplayLabel	Service Value
Description	Service Value
Priority	200
Expand	

- **Save** the schemas.

NOTE: If you need to pass SERVICE to inline Instruments such as Flow Elements you will need to add an additional Notify Relationship to the Notify Related Component of Property Value Change custom attribute. See below for this additional notification.

How to Pass Service from Line to Run and on to Inline Component at Creation and Post Creation

Cascade Property Value change	true
Set Default Property Value	False
<input type="checkbox"/> Notify Relationships[4]	
Notify Target of Relationship	True
Relationship Class Name	RUN_HAS_IN_RUN
Related Class Name	appid:FLOW_ELEMENT
Related Class Property Name	SERVICE
Cascade Property Value change	True
Set Default Property Value	False

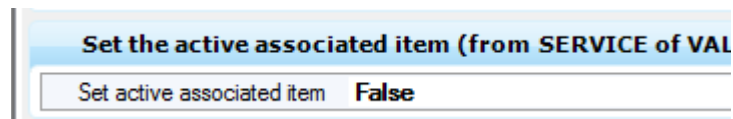


How to Pass Service from Line to Run and on to Inline Component at Creation and Post Creation

Configuring Valve and Instrument to Receive and Use Values

- Continuing in the **Class Editor**
- Select the **VALVE** class in the left hand panel.
- In the right hand panel pick on the **Properties** tab.
- Navigate to the **Service** property, if the property is greyed out use the **Override** button to localize the property.
- Modify it as shown below by adding the **Set the active associated item** custom attribute. Fill the custom attribute in as shown below.

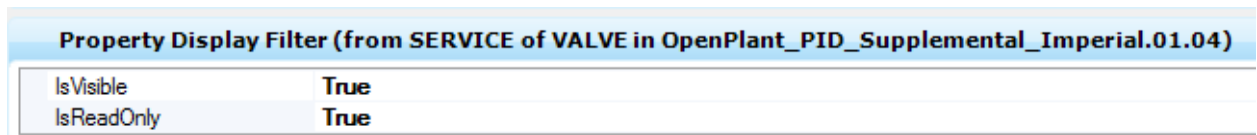
NOTE: This Custom attribute when set to false, instructs the system to not use the default SERVICE but because we have passed values from Pipe Run to use those.



- **Save** the schemas.
- Repeat this process for **INSTRUMENT**.
- **Save** the schema.

Setting SERVICE Property on the Component to Read Only

- Continuing in the **Class Editor**
- Select the **VALVE** class in the left hand panel.
- In the right hand panel pick on the **Properties** tab.
- Locate the **SERVICE** property.
- Right click on the property and from the context menu select **Custom Attribute > Add/Remove...**
- Add the **Property Display Filter** custom attribute and fill it in as shown below.



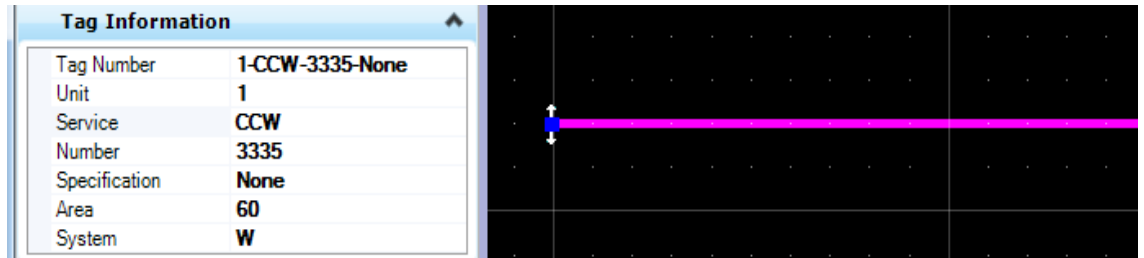
NOTE: Add this to every class listed in the Notify Related Component of Property Value Change custom attribute – VALVE, INSTRUMENT, PIPING_COMPONENT, FLOW_ELEMENT etc.

- **Save** the schema.
- Now after placing the valve the **SERVICE** property will no longer be editable on the valve and can only be edited on the **PIPELINE** and then passed to **PIPE RUN** and onto the **VALVE**. You could repeat this process on instruments and other inline components where you do not want the SERVICE property modified on the specific component.

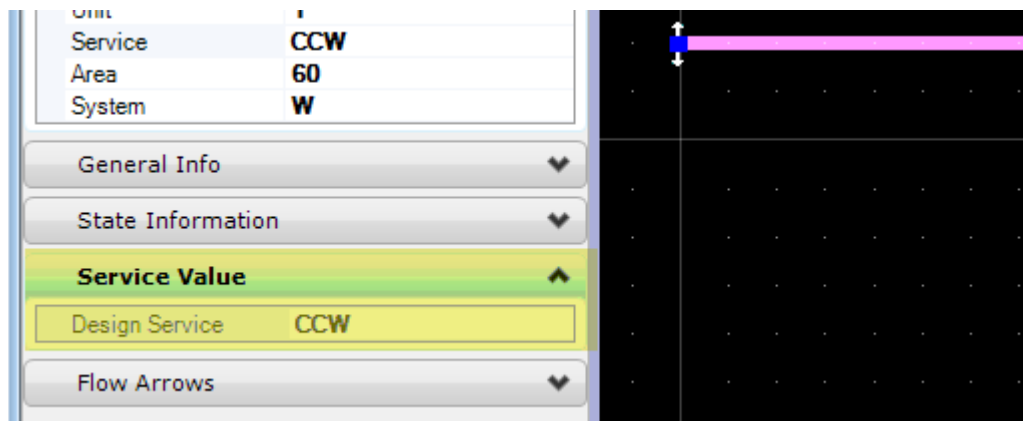
How to Pass Service from Line to Run and on to Inline Component at Creation and Post Creation

Testing the Results

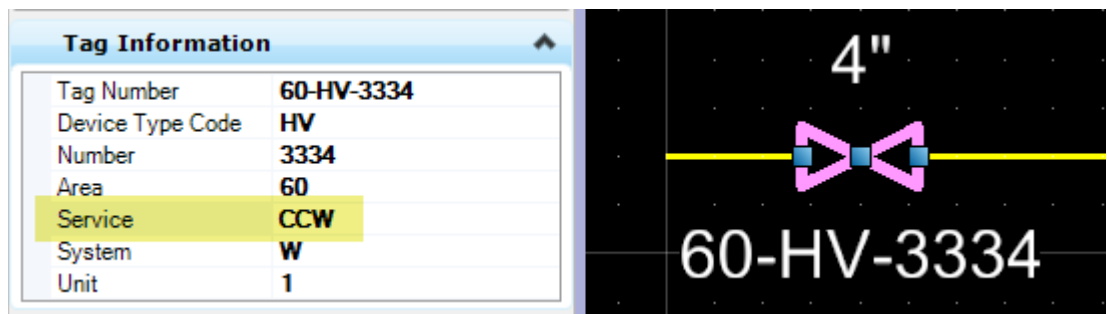
- Launch **OpenPlant PowerPID**.
- Create a new PID file.
- Add several new services to the file. Add "**CW**", "**CCW**", **CWR**". Set the **default** service to **CW**.
- Draw a pipeline and set the service for this new line to **CCW**. Look at the properties for the line and then the run.



- While on the run properties look at the **Service Value** category and verify it matches the **Service** from the Line.



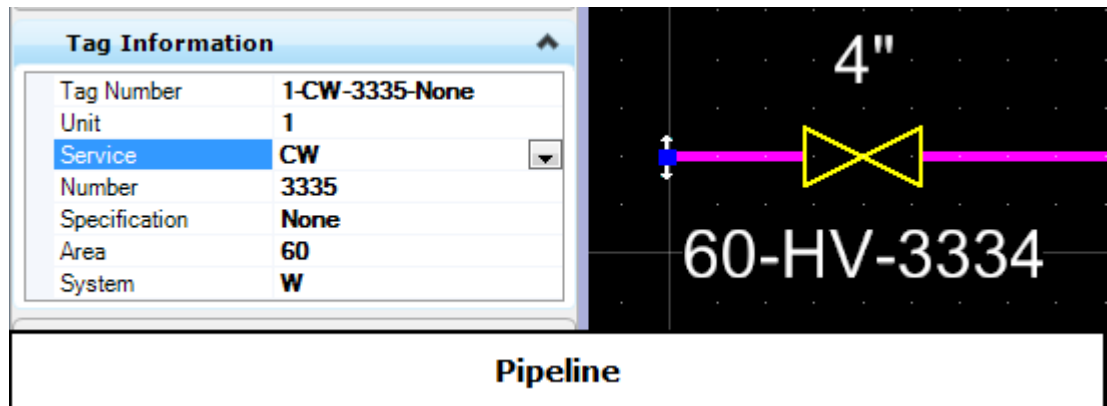
- Insert a valve in the line and note that the valve **SERVICE** is also **CCW** and not the system default of **CW**.



- Edit the **LINE** and change the service to **CW**.

How to Pass Service from Line to Run and on to Inline Component at Creation and Post Creation

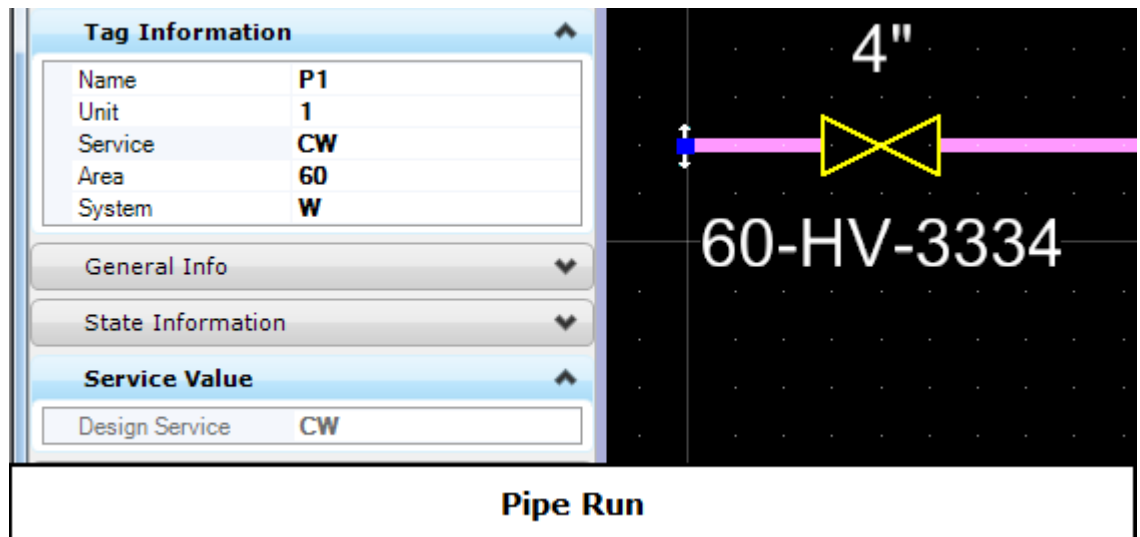
- Note the change passes from line to run to valve.



The screenshot shows the 'Tag Information' window for a pipeline. The 'Service' field is set to 'CW'. To the right, a 4-inch pipeline is shown with a yellow valve symbol. The tag number '60-HV-3334' is displayed below the pipeline.

Tag Information	
Tag Number	1-CW-3335-None
Unit	1
Service	CW
Number	3335
Specification	None
Area	60
System	W

Pipeline



The screenshot shows the 'Tag Information' window for a pipe run. The 'Service' field is set to 'CW'. Below the 'Tag Information' section, the 'Service Value' section is expanded, showing 'Design Service' as 'CW'. To the right, a 4-inch pipe run is shown with a yellow valve symbol. The tag number '60-HV-3334' is displayed below the pipe run.

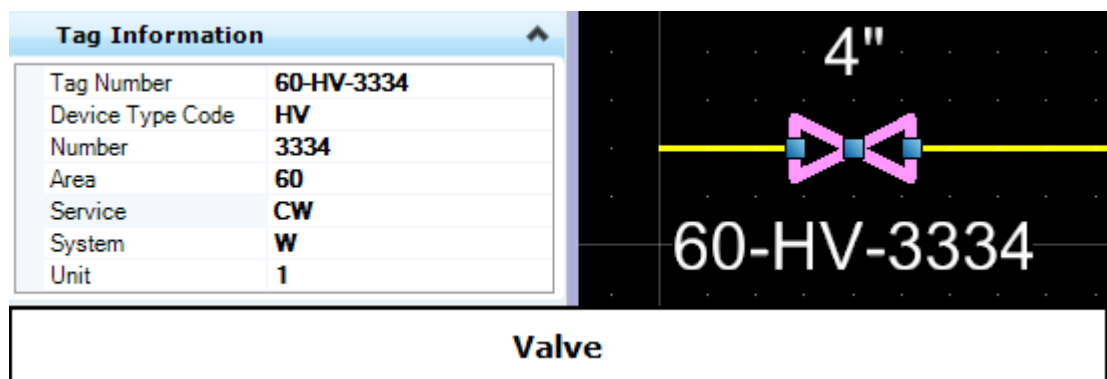
Tag Information	
Name	P1
Unit	1
Service	CW
Area	60
System	W

General Info
State Information

Service Value

Design Service	CW
----------------	----

Pipe Run



The screenshot shows the 'Tag Information' window for a valve. The 'Service' field is set to 'CW'. To the right, a 4-inch valve is shown with a pink valve symbol. The tag number '60-HV-3334' is displayed below the valve.

Tag Information	
Tag Number	60-HV-3334
Device Type Code	HV
Number	3334
Area	60
Service	CW
System	W
Unit	1

Valve

- Repeat this test process by adding an instrument to the line. Note that the SERVICE value is populated based on the pipe run.

