

# L06 - FactoryTalk View Site Edition - Building Applications Lab



*For Classroom Use Only!*



# Important User Information

This documentation, whether, illustrative, printed, 'online' or electronic (hereinafter 'Documentation') is intended for use only as a learning aid when using Rockwell Automation approved demonstration hardware, software and firmware. The Documentation should only be used as a learning tool by qualified professionals.

The variety of uses for the hardware, software and firmware (hereinafter 'Products') described in this Documentation, mandates that those responsible for the application and use of those Products must satisfy themselves that all necessary steps have been taken to ensure that each application and actual use meets all performance and safety requirements, including any applicable laws, regulations, codes and standards in addition to any applicable technical documents.

In no event will Rockwell Automation, Inc., or any of its affiliate or subsidiary companies (hereinafter 'Rockwell Automation') be responsible or liable for any indirect or consequential damages resulting from the use or application of the Products described in this Documentation. Rockwell Automation does not assume responsibility or liability for damages of any kind based on the alleged use of, or reliance on, this Documentation.

No patent liability is assumed by Rockwell Automation with respect to use of information, circuits, equipment, or software described in the Documentation.

Except as specifically agreed in writing as part of a maintenance or support contract, equipment users are responsible for:

- properly using, calibrating, operating, monitoring and maintaining all Products consistent with all Rockwell Automation or third-party provided instructions, warnings, recommendations and documentation;
- ensuring that only properly trained personnel use, operate and maintain the Products at all times;
- staying informed of all Product updates and alerts and implementing all updates and fixes; and • all other factors affecting the Products that are outside of the direct control of Rockwell Automation.

Reproduction of the contents of the Documentation, in whole or in part, without written permission of Rockwell Automation is prohibited.

Throughout this manual we use the following notes to make you aware of safety considerations:

**WARNING**



Identifies information about practices or circumstances that can cause an explosion in a hazardous environment, which may lead to personal injury or death, property damage, or economic loss.

**IMPORTANT**

Identifies information that is critical for successful application and understanding of the product.

**ATTENTION**



Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss. Attentions help you:

- identify a hazard
- avoid a hazard
- recognize the consequence

**SHOCK HAZARD**



Labels may be located on or inside the device that dangerous voltage may be present.

**BURN HAZARD**



Labels may be located on or inside the device that surfaces may be dangerous temperatures.

---

# FactoryTalk View Site Edition - Building Applications Lab

---

## Contents

Before you begin.....	8
About this lab .....	8
What you will accomplish in this lab .....	8
What's New in FactoryTalk View SE v8.0 .....	9
Who should complete this lab .....	9

Tools & prerequisites .....	9
Overview .....	11
About this section.....	11
FactoryTalk View SE components.....	11
FactoryTalk View SE - Network and Local .....	15
Section 1: Overview of InstantFizz – A Network Distributed Application.....	21
About this section.....	21
Specifying the FactoryTalk Directory location .....	22
Explore the InstantFizz application .....	22
Explore the InstantFizz_HMI HMI Server .....	25
Explore the RSLinx Enterprise data server .....	27
Explore the Communications Setup .....	29
Create a display and add a numeric input object .....	31
Section 2: Tags .....	35
About this section.....	35
Import/Export Tags Wizard .....	36
Add an object referencing an HMI tag .....	39
Section 3: Graphic Displays, Objects and Animations .....	45
About this section.....	45
Graphic Displays.....	46
Graphic Libraries.....	47
Modifying a Graphic Display by editing the Associated XML file.....	48
Tooltips .....	55
Docked displays.....	61
Explore docked displays at run-time.....	64
Commands.....	67
Animations and expressions.....	72
The Object Explorer .....	73
Local Messages .....	78
Using Symbol Factory Graphics .....	89
Color Animation Enhancements .....	102
Find and Replace.....	109
Cross Reference.....	118
PNG Support.....	120
Parameter File Supports Tag Browsing.....	124
Digit Grouping .....	125
Section 4: Data logging and trending .....	136
About this section.....	136



Runtime exploration .....	136
Configuration exploration .....	137
Configuring trends.....	143
Viewing the trend at runtime .....	154
Section 5: FactoryTalk Alarms and Events.....	163
About this section.....	163
Introduction to alarming .....	164
FactoryTalk Alarms and Events components .....	165
Working with Device-based Alarms and Events - ALMD.....	176
Working with Device-based Alarms and Events - ALMA.....	186
Working with Tag-based Alarms and Events.....	195
Setting up historical alarm and event logging .....	213
Add a database definition to the FactoryTalk system.....	214
Associate database definition with alarm servers.....	216
Using FactoryTalk Alarm and Event Objects.....	218
Monitor and Interact with Real time Alarms .....	219
Viewing Historical Alarm data with the Alarm and Event Log Viewer.....	233
Out of Scope Alarms.....	242
Section 6: Testing displays .....	245
About this section.....	245
Test display.....	245
Configure a client file .....	246
Running the FactoryTalk View SE client .....	253
Verify runtime edits .....	253
Section 7: Events and Using the Event Detector Editor .....	254
About this section.....	255
Create an Event Detector file.....	255
Testing Event .....	259
Section 8: FactoryTalk View Application Documenter .....	263
About this section.....	263
Starting FactoryTalk View Application Documenter and Opening a Project .....	263
Section 9: Extending FactoryTalk View SE functionality using VBA .....	266
About this section.....	266
Introduction to VBA.....	266
View the FactoryTalk View SE object model .....	266
Understanding VBA Code and Making Edits to the VBA code.....	267
Launching the VBA Editor.....	270
Code Walk-through to understand different components of the VBA applications .....	273

Testing the display with VBA Code.....	277
Section 10: FactoryTalk ViewPoint.....	278
Before you begin.....	278
About this section.....	279
Connecting to the FactoryTalk ViewPoint Administration Site .....	280
Configuring FactoryTalk ViewPoint Administration Security.....	280
FactoryTalk ViewPoint Server Settings .....	288
Activating FactoryTalk ViewPoint .....	288
Publishing a FactoryTalk View SE application .....	289
Running an application in a browser.....	293
Configuring FactoryTalk ViewPoint Application Security.....	300
Tag Write Functionality .....	307



---

## Before you begin

This hands-on lab focuses on the **basic** building blocks of FactoryTalk Site Edition.

This lab is intended to be a compilation of several smaller exercises designed to instruct the user on the features of FactoryTalk View Site Edition. The lab can be done linearly, or the user may select which exercises interests them most without having to be concerned with numerical order.

The following steps must be completed before starting the lab exercise:

1. If the **Log On To Windows** dialog is active, type '**Labuser**' for Username and '**rockwell**' for password.
2. Use the same Login information if prompted to Log On to the FactoryTalk Directory or when creating a FactoryTalk Alarms and Events History Database.

## About this lab

Welcome to the FactoryTalk View SE Hands-On Lab series. This session provides you with an opportunity to explore the features of FactoryTalk View Site Edition (FactoryTalk View SE). The following sections explain what you'll be doing in this lab session and what you will need to do to complete the hands-on exercises.

Factory Talk View SE is an integrated package for developing and running multi-user, networked human-machine interface (HMI) applications. FactoryTalk View SE is designed for automated process or machine monitoring, and supervisory control.

In this lab, you will be working with a network application containing an HMI Server, a data server, and a single HMI Client. For this lab, these servers and clients will all be located on the same computer. FactoryTalk View Studio, the development environment, will also be on this computer. In the deployed system however, these components could actually be on separate computers, and additional HMI clients could be used, since View SE scales easily from small to large systems.

## What you will accomplish in this lab

As you complete the exercises in this hands-on session, you will gain an understanding of the functionality and capability of FactoryTalk View Site Edition by

- exploring the InstantFizz network application components
- utilizing graphics and animation
- test running displays
- configuring and running an HMI client file
- configuring and monitoring alarms (FactoryTalk device and tag based)
- working with data log models and trends
- creating a simple VBA program to extend some of the FactoryTalk View SE features
- exploring how to create reports of the InstantFizz network application
- working with FactoryTalk ViewPoint



## What's New in FactoryTalk View SE v8.0

Look for the “NEW” icon throughout the manual to learn more about new features in FactoryTalk View SE v8.0. The following table lists the new features and the page numbers for each item.

This new feature...	Is on page...
Capturing Value Before Change	63
Button Enable/Disable	86
CurrentUserHasGroup() security function	88
Digit Grouping	129
Confirmation Pop Up	153
Irregular Touch Style	155
Alarm Filtering	237
Out of Scope Alarms	244
Application Documenter File	268

For a complete description of the new features in FactoryTalk View SE v8.0 and FactoryTalk ViewPoint v8.0, see the following AID's:  
[AID 617486 - What's new in FactoryTalk View Site Edition 8.0 CPR9 SR7](#)  
[AID 617834 - What's new in FactoryTalk ViewPoint 8.0 CPR9 SR7](#)

### Who should complete this lab

This hands-on lab is intended for individuals who:

- Have a basic knowledge of HMI software and are involved in the design and implementation of supervisory-level HMI projects.

### Tools & prerequisites

#### Hardware

This hands-on lab does not require any hardware. A Logix5000 controller could be used in place of SoftLogix 5800.

#### Software

This hands-on lab uses the following software:

- FactoryTalk Services Platform v2.70 (CPR 9 SR 7)
- FactoryTalk View SE v8.00
- RSLinx Enterprise v5.70
- FactoryTalk Alarms and Events v2.70
- FactoryTalk Diagnostics v2.70

- FactoryTalk Activation Manager v3.62.01
- RSLinx Classic v3.70
- RSLogix 5000 v20.01
- SoftLogix5800 v20.01
- FactoryTalk ViewPoint v8.00
- Microsoft SQL Server 2008 R2 Express
- Microsoft SQL Server 2008 R2 Standard
- Microsoft Office 2010
- Microsoft Silverlight 5.1
- Internet Explorer 9
- doPDF 7.3
- Adobe Reader 11

### **Lab Files**

This hands-on lab uses the following files located in the following folders:

In '**C:\Lab Files\FTView SE**' folder:

- **IF\_Client\_1280x1024.cli** - a preconfigured FactoryTalk View SE Client file
- **ODBC\_InstantFizz.mdb** – a Microsoft Access database
- **Lock.png** – an image file

In '**C:\Lab Files\RSLogix 5000**' folder:

- **IF2\_DEMO.ACD** – control program to be used in this lab

---

## Overview

### About this section

This section will outline the major components and fundamental ideas of FactoryTalk View Site Edition. It will specifically:

- Discuss the components of FactoryTalk View SE
- Discuss the differences between FactoryTalk View SE Local and Network Applications
- Discuss HMI Servers, data servers, and Tag Alarm and Event Servers
- Discuss FactoryTalk Alarms and Events Services for Device Based and Tag Based Alarms

### FactoryTalk View SE components

#### FactoryTalk View Site Edition

FactoryTalk View Site Edition is an integrated software package for developing and running human-machine interface (HMI) applications that involve multiple users and servers, distributed over a network.

A member of the FactoryTalk family of products, FactoryTalk View Site Edition (also called FactoryTalk View SE) provides all the tools you need to create powerful, dependable process monitoring and supervisory control applications.

FactoryTalk View SE software is designed for use with Microsoft Windows Server 2003, 2008/2008 R2 and 2012/2012 R2,

Windows 7, Windows 8/8.1, Windows XP, and Windows Vista Business. It can run on 32-bit or 64-bit versions of Windows.

FactoryTalk View Site Edition consists of several pieces of software you can use to build automation applications. Depending on the particular software packages installed, you will have one or more of the following pieces of software: FactoryTalk View Studio Enterprise, FactoryTalk View SE Client, FactoryTalk View SE Server, FactoryTalk Alarms and Events, FactoryTalk Services Platform, FactoryTalk View Administration Console, FactoryTalk Directory, FactoryTalk Activation, RSLinx Enterprise and RSLinx Classic.



#### FactoryTalk View Studio FactoryTalk View Studio

**Start > Programs > Rockwell Software > FactoryTalk View >**

FactoryTalk View Studio is configuration software for developing and testing FactoryTalk View SE applications. FactoryTalk View Studio contains editors for creating complete applications, and includes client and server software for testing the applications you create. Use the editors to create applications that are as simple or as complex as you need. You can use FactoryTalk View Studio to develop FactoryTalk View Site Edition (SE) and FactoryTalk View Machine Edition (ME) applications.

FactoryTalk View comes with process faceplates, graphic libraries and Symbol Factory graphics objects that can be used in your applications. Process faceplates are preconfigured to work with various Logix5000 instructions (for example, PIDE, D2SD, and the ALMD and ALMA instructions). Many of the graphic library objects are preconfigured with animation. Use the objects as they are, or change them to suit your needs.

When you have finished developing an application, use FactoryTalk View SE Client to view and interact with the application.



**FactoryTalk View SE Client**  
**FactoryTalk View Site Edition Client**

**Start > Programs > Rockwell Software > FactoryTalk View >**

FactoryTalk View SE Client is a complete runtime operating environment for viewing and interacting with FactoryTalk View SE local and network applications. To set up a FactoryTalk View SE Client, you need to create a configuration file using the FactoryTalk View SE Client wizard. The HMI Server does not have to be running when you configure a FactoryTalk View SE Client. With the FactoryTalk View SE Client you can:

- Load, view, and interact with multiple graphic displays at a time from multiple servers
- Perform alarm management
- View real-time and historical trends
- Adjust set points
- Start and stop components on any server
- Provide a secure operator environment



**FactoryTalk View Administration Console**  
**FactoryTalk View > Tools > SE Administration Console**

**Start > Programs > Rockwell Software >**

FactoryTalk View Administration Console is for administering FactoryTalk View applications after they have been deployed. FactoryTalk View Administration Console contains a sub-set of the FactoryTalk View Studio editors, so you can make minor changes to an application without the need for installing FactoryTalk View Studio. The FactoryTalk View Administration Console has a two hour run-time limit. A warning message is displayed five minutes before the time is up. To continue using it you simply shut it down and restart it.

FactoryTalk View Administration Console allows you to:

- Change the properties of an HMI server.
- Change the properties of a data server.
- Add FactoryTalk users to an application, using the Runtime Security editor.
- Set up security for commands and macros, using the Runtime Secured Commands editor.
- Run FactoryTalk View commands from the Command Line.
- Change how HMI tag alarms are logged and annunciated, using the Alarm Setup editor.
- Change the path of data log models.
- Change which system activities are logged and how frequently, using the Diagnostics Setup editor (on the Tools menu).
- Change the location alarms are logged to, and manage log files, using the Alarm Log Setup editor (on the Tools menu).
- Import and export HMI tags using the Tag Import and Export Wizard (on the Tools menu)



## FactoryTalk View SE Server

The FactoryTalk View SE Server, also called the HMI server, stores HMI project components (for example, graphic displays, global objects, and macros) and serves them to clients. The server also contains a database of tags, performs historical data logging, and HMI alarm monitoring. FactoryTalk Alarms and Events can be used instead of FactoryTalk View SE HMI alarm monitoring. To maintain compatibility with existing applications, FactoryTalk View still supports the traditional HMI alarm monitoring.

The FactoryTalk View SE Server has no user interface. Once installed, it runs as a set of 'headless' Windows services that supply information to clients as they request it.

## FactoryTalk Alarms and Events

Before FactoryTalk Alarms and Events, FactoryTalk View SE only supported HMI tag alarm monitoring. To maintain compatibility with existing applications, this type of alarm monitoring is still supported.

However, FactoryTalk Alarms and Events now allows multiple FactoryTalk products to participate together in a common, consistent view of alarms and events throughout a FactoryTalk system. FactoryTalk Alarms and Events support two types of alarm monitoring:

**Device-based alarm monitoring** - Pre-built alarm instructions, available in RSLogix 5000 v. 16 or later, are programmed in a logic project and then downloaded into a Logix5000 controller. The controller detects alarm conditions and publishes event information, which is routed through the system for display and logging.

**Tag-based alarm monitoring** - If you are not using Logix5000 controllers, or if you do not want to use the pre-built alarm instructions available with RSLogix 5000, tag-based alarm monitoring offers the equivalent of HMI Tag Alarm Monitoring, but with an expanded feature set. Software-based Tag Alarm and Event Servers monitor controllers for alarm conditions through data servers and publish event information for display and logging. Tag-based alarm monitoring is supported for Logix5000 controllers, PLC-5, and SLC 500 devices communicating through Rockwell Automation Device Servers (RSLinx Enterprise), or for third-party controllers communicating through OPC data servers.

## FactoryTalk Services Platform

FactoryTalk Services Platform provides common services (such as diagnostic messages, health monitoring services, and access to real-time data) to products and applications in a FactoryTalk system.

## FactoryTalk Directory

FactoryTalk Directory centralizes access to system resources (for example, FactoryTalk View SE Servers, or OPC servers) and names (for example, data tags, graphic displays, and log models), for all of the FactoryTalk products and components participating in an automated control system.

FactoryTalk Directory software works like a telephone directory, or electronic address book, providing a lookup service for parts of an application to find each other on a single computer, or across a network.

Through the lookup service, application components such as tags and graphic displays can be stored in their original environments, and yet be made available to all clients participating in an application. No duplication is necessary.

FactoryTalk View Site Edition applications use two types of FactoryTalk Directory:

- **FactoryTalk Local Directory** (also called the Local Directory) manages local applications. All local application components, except for OPC data servers, must be located on the same computer.

- **FactoryTalk Network Directory** (also called the Network Directory) manages network distributed and network station applications. Network distributed applications can consist of multiple clients and servers, distributed across several computers connected over a network. Network station applications can consist of multiple clients and a single server, all of which must be located on the same computer. One Network Directory manages all of the FactoryTalk products that participate in a single network application.

Both the Local and the Network Directory are set up on the computer, when you install the FactoryTalk Services Platform.



### **FactoryTalk Administration Console**

**Start > Programs > Rockwell Software > FactoryTalk Administration Console**

Part of the FactoryTalk Services Platform, FactoryTalk Administration Console is an optional, stand-alone tool for developing applications and managing a FactoryTalk system. You can use FactoryTalk Administration Console or FactoryTalk View Studio to develop applications and manage a FactoryTalk system. Only FactoryTalk View Studio can be used to create HMI servers and HMI projects.

FactoryTalk Administration Console allows you to:

- Create and configure application, area, and data server elements in a FactoryTalk Directory.
- Create and configure alarm and event servers, including both tag-based and device-based servers.
- Configure alarm conditions for tag-based alarm detection.
- Organize securable actions into groups.
- Create database definitions for logging historical alarm and event messages.
- Configure options for routing, logging, and viewing diagnostic messages.
- Backup and restore an entire directory, an individual application, or system settings.
- Set up redundancy for OPC data servers and Tag Alarm and Event Servers.
- Configure client computers to recognize the location of a Network Directory Server computer.
- Configure system-wide policy settings.
- Secure a FactoryTalk system with security services.



### **FactoryTalk Activation**

**Start > Programs > Rockwell Software > FactoryTalk Activation > FactoryTalk Activation Manager**

FactoryTalk Activation provides a secure, software-based system for activating Rockwell Software products and managing software activation files. With FactoryTalk Activation, there is no need for a physical 'master disk' or any physical media; instead, activation files are generated and distributed electronically.

FactoryTalk Activation provides these types of activations:

- Local node-locked activations are locked to a single computer.
- Mobile node-locked activations are locked to a hardware dongle.

- Shared concurrent activations are locked to an activation server computer, and shared by client computers on the network.

There are two types of shared concurrent activation: floating and borrowed. Floating concurrent activation requires a continuous network connection, while borrowed concurrent activation does not.

### **RSLinx Enterprise**

RSLinx Enterprise is a FactoryTalk Live Data server that can run on multiple platforms, from PanelView Plus dedicated terminals to desktop computers.

For communications with Allen-Bradley local and remote devices—particularly with Logix5000 controllers—RSLinx Enterprise is the recommended data communications software for FactoryTalk View applications.

### **RSLinx Classic**

RSLinx Enterprise is the preferred choice when using Logix controllers, but there are some cases where RSLinx Classic must be used. For example, install and use RSLinx Classic to serve data through DH+ (Data Highway +) networks, to support complex bridging and routing, and to support unsolicited messaging from a controller to RSLinx.

With RSLinx Classic, you can also create alias topic shortcuts and perform online tasks such as uploading and downloading RSLogix 5000 files.

## **FactoryTalk View SE - Network and Local**

### **Network Distributed Applications**

A network distributed application can contain several servers, running on multiple computers on a network, with multiple client users connecting to the application simultaneously, from anywhere on the network. For example, you may use separate servers for different functional areas or locations within your enterprise, and allow clients to interface to any of the servers. Network distributed applications have one or more areas (see Areas definition below), one HMI server per area, and one or more data servers. An area may contain another area within it.

Once you have created the applications and an HMI server, you can use the FactoryTalk View Studio editors in the HMI server project to create application components such as graphics displays, global objects, and data log models.

#### **Areas**

A key part of the network architecture system is the area. An area is a logical division within your application. You can think of areas as partitions of your hard drive. The partitions are all on the same main disk (or application, in this analogy), but they divide it logically and hold information independently of each other. An area can also be used to organize the application in a way that makes sense for the process it is controlling.

For example, an area might represent a portion of a process, or a region within the process facility. An automotive plant could be divided into areas called Press and Fabrication, Body Shop, Paint Shop, Engine, and Transmission; a bakery could be divided into areas called Ingredients, Mixing, Baking, and Packaging. Alternatively, a plant with identical production lines could be divided into areas called Line 1, Line 2, Line 3, and so on. This would allow you to add new, identical production lines to the application by copying HMI server projects into new areas.

### **Root Area**

All FactoryTalk View applications have one system-defined area called the root area, which has the same name as the application. The application root area can contain one HMI server, and one or more data servers.

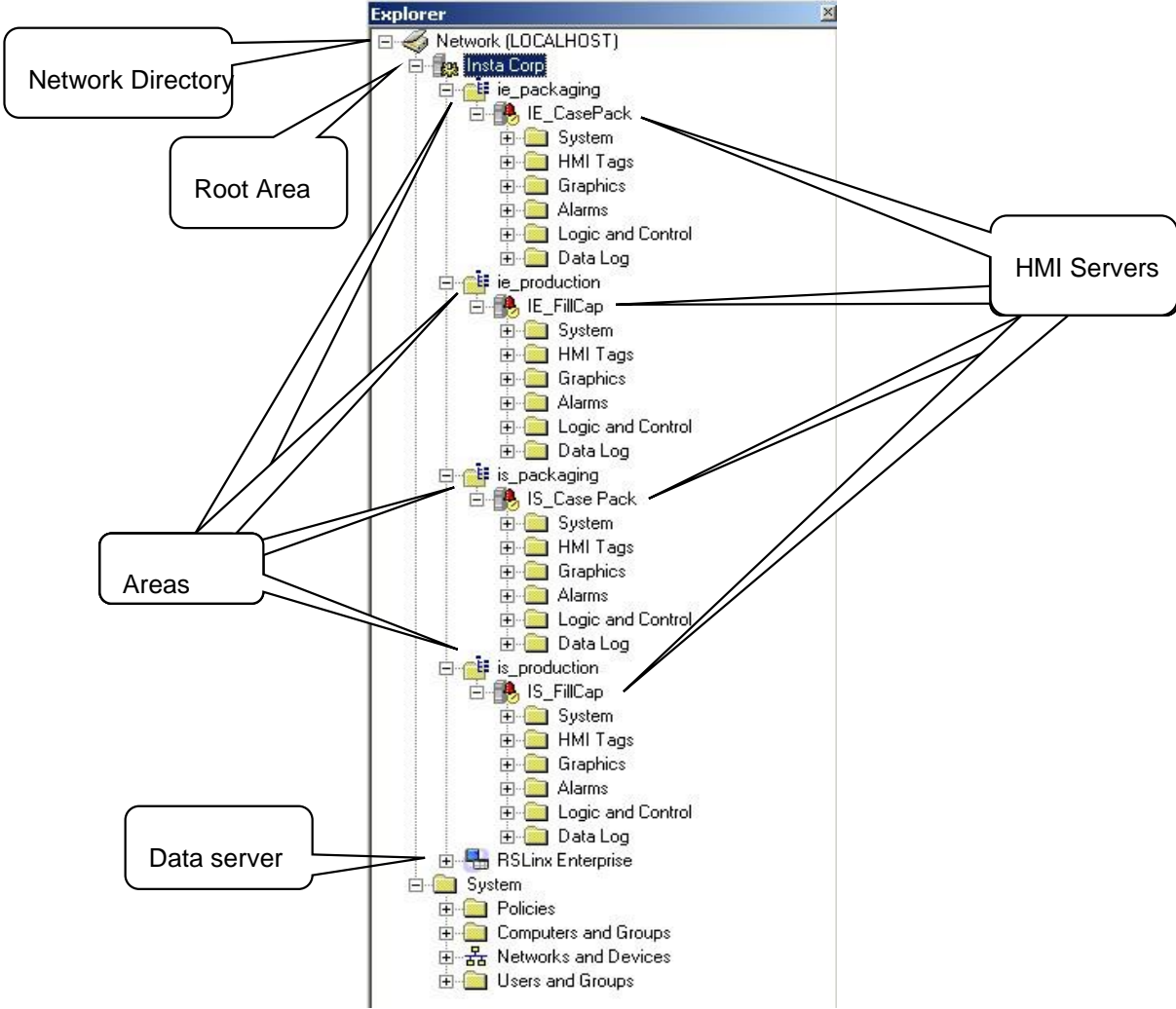
### **Best practice**

Since an area is nothing more than a logical method of organizing the application, and not a physical entity, there is not a limit to the number of areas that can reside within an network distributed application. However, there is a limit of 1 HMI server per area and 10 HMI servers per application\*.

The recommended limit of data servers within an application is 10\*. However, it makes sense to logically organize the data servers that are serving alarms in order for the alarm summary to filter alarms appropriately at runtime. For example, a single area may contain an HMI server for a physical location of a facility, an RSLinx Enterprise data server (configured as a FactoryTalk device based alarm server), and a 3rd party OPC server (configured with the FactoryTalk tag based alarm server). This configuration allows for the alarm summary to filter alarms based on the area name, regardless of which server the alarm comes from.

What you want to avoid is one physical installation of a data server to be referenced multiple times from different areas of the application. This is not necessary because FactoryTalk allows any client to see any data point within the application, regardless of which area it comes from.

This is an example of a FactoryTalk View SE network distributed application. This is NOT the application you will be using in this lab. It is just an example.



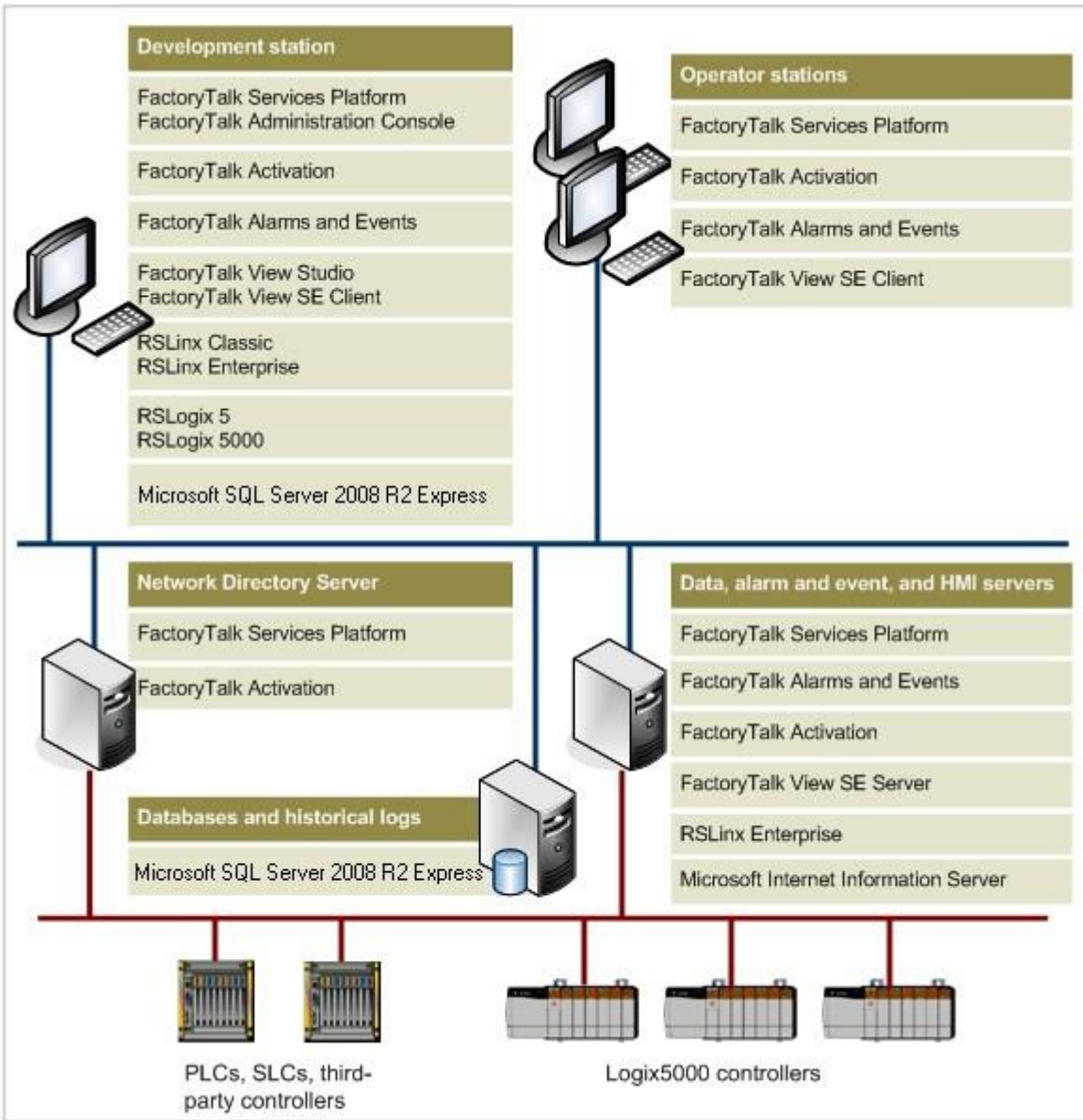
The **Insta Corp** application consists of four different defined areas: **ie\_packaging**, **ie\_production**, **is\_packaging**, and **is\_production**. The areas are marked by the folders that are right off the root, which is the application **Insta Corp**.

Look at one of the areas – **ie\_packaging**, the topmost area. Notice that the HMI server called **IE\_CasePack** is located inside the area.

The folders under the **ie\_packaging** HMI Server titled System, HMI Tags, Graphics, Alarms, Logic and Control, and Data Log are all different components you can configure under each HMI server – they are not areas within the area, but are actually components of an HMI server.

There is a data server called **RSLinx Enterprise** located under the root area (**Insta Corp**).

The diagram below shows an example system architecture using a Network application as part of a distributed FactoryTalk system.



### Network Station Applications

A network station application can consist of one or more areas, one HMI server, and, if necessary, multiple data servers and FactoryTalk Tag Alarm and Event Servers. A network station application is similar to a network distributed application with the following exceptions:

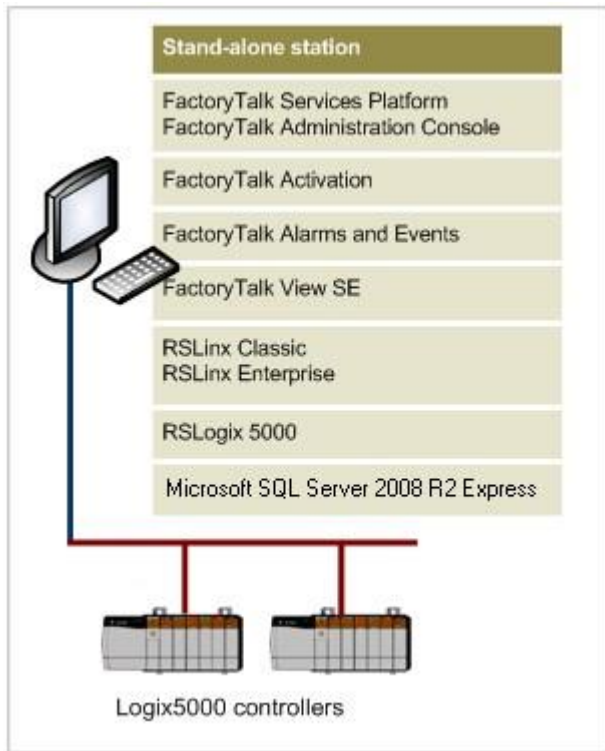
- Only one HMI server can be added to a network station application
- The HMI server will not support redundancy
- HMI clients can only connect to the network station application from the same computer as the HMI server. FactoryTalk View Studio, the FactoryTalk View SE Administration Console, and the FactoryTalk View SE Client are all HMI clients.

As of FactoryTalk View SE v8.0, you can remotely edit the HMI server in a network station application. The benefit of this is that now you don't need to install Studio on the same computer as the runtime SE client.

## Local Applications

A local application is similar to an RSView32 project; all application components and the FactoryTalk View SE client are located on a single computer. There is only one HMI server that is created for you in the root area when the application is created. You may use local applications for parts of the plant or process that are self-contained and are not related to other parts of the process.

The diagram below shows an example system architecture using a Local application as part of a stand-alone FactoryTalk system.



### Creating a new FactoryTalk View SE

**application** Here are the general steps for creating an application:

1. Create a local or network application.
2. If it's a network application, add one or more Areas.
3. If it's a network distributed application, one HMI server can be added per area (local creates one automatically). If it's a network station application, you can only add one HMI server (to any area). Choose to add any of the faceplate displays into the HMI server.
4. Set up data server communications. Add one or more of the following data servers:

- Rockwell Automation Device Server.
  - OPC Data server.
5. Set up Tag Alarm and Event Server.
  6. Create graphic displays, global objects, and other components into your HMI server.
  7. Set up historical FactoryTalk alarm and event logging.
  8. Set up Security.
  9. Set up a run-time FactoryTalk View SE Client.



---

## Section 1: Overview of InstantFizz – A Network Distributed Application

In this lab you will modify and run a **network distributed** FactoryTalk View SE application named **InstantFizz** that we have created for you. This section will introduce you to the **InstantFizz** application.

### About this section

In this section of the lab you will:

- Specify the FactoryTalk Directory location
- Explore the Network Application called **InstantFizz**
- Explore the HMI Server called **InstantFizz\_HMI**
- Explore the data server called **RSLinx Enterprise**
- Configure a Communications path called **Shortcut** that will point to the SoftLogix controller
- Verify communications
- Create a display, add a numeric input object
- Test the display

## Specifying the FactoryTalk Directory location

A network application requires a Network FactoryTalk Directory to be configured. The configuration of the FactoryTalk Directory is generally done during the install, but it can be configured subsequently using the FactoryTalk Directory Configuration Wizard.

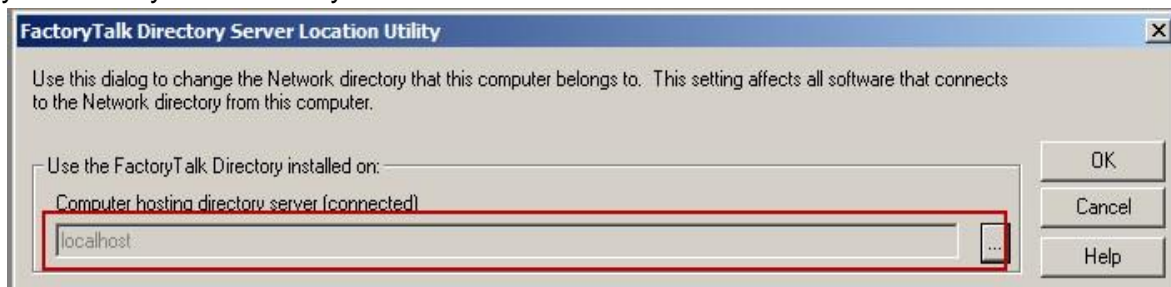
Once the FactoryTalk Directory has been configured you can specify what computer will be hosting the Directory. The default location of the FactoryTalk Directory is the local computer, but it can be any computer on your network. Note: For a FactoryTalk View Local application, the Local FactoryTalk Directory must be used.

For the purposes of this lab, the FactoryTalk Directory location is the local computer. To confirm that, do the following:

1. Go to **Start > All Programs > Rockwell Software > FactoryTalk Tools > Specify FactoryTalk Directory Location**.

In order to specify the FactoryTalk Directory Location you must have administrator rights. During the install of the FactoryTalk Services Platform, the 'Windows Administrator' and 'Administrator' groups are automatically added to the Network FactoryTalk Directory. Hence, you can log in with any user belonging to either of these groups.

2. The **FactoryTalk Directory Server Location Utility** will open. Notice that the FactoryTalk Directory Location is set to the local computer (**localhost**). In a FactoryTalk View SE Network application, the FactoryTalk Directory can be located on any computer on the network that has the FactoryTalk Services Platform installed. You would use this utility to modify the location of your FactoryTalk Directory.



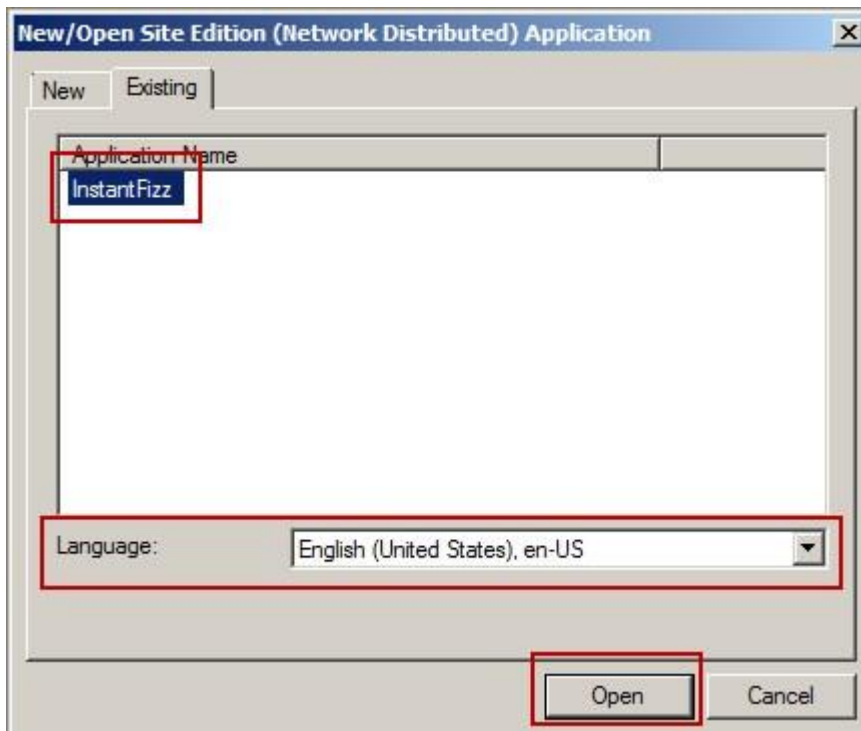
3. Press **Cancel** to close.

## Explore the InstantFizz application

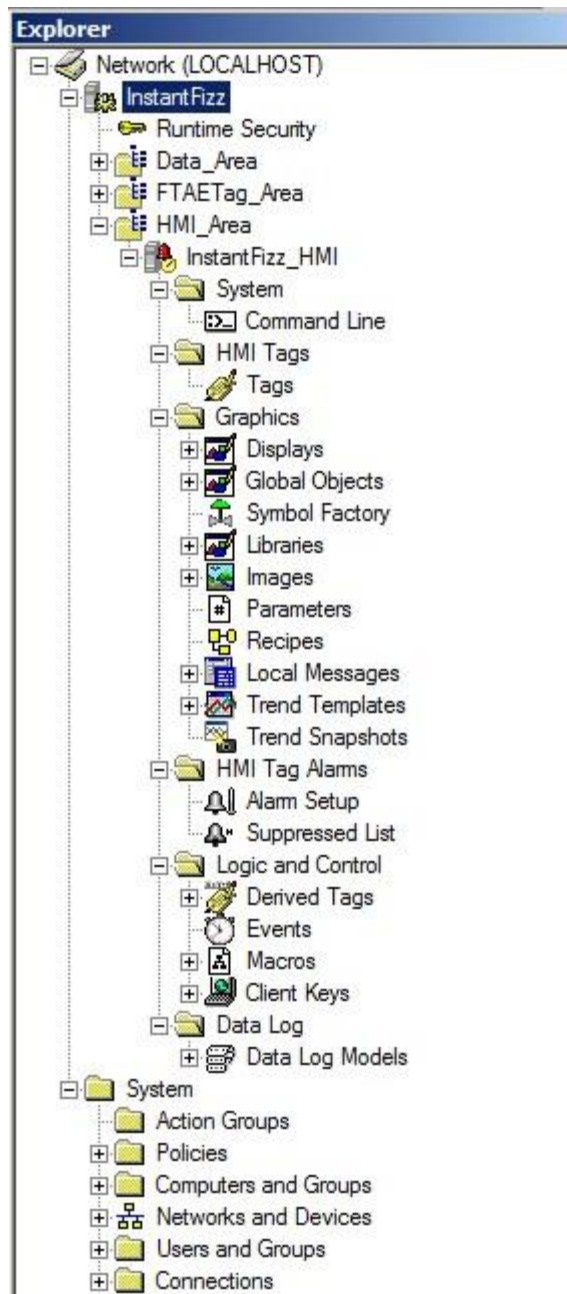
1. Open FactoryTalk View Studio. Select the **Start > All Programs > Rockwell Software > FactoryTalk View > FactoryTalk View Studio** menu item. (For convenience, a shortcut to **FactoryTalk View Studio** has also been added to the Start menu.)



- 2. Open the Network application called **InstantFizz** by selecting **View Site Edition (Network Distributed) > Continue** as shown below:

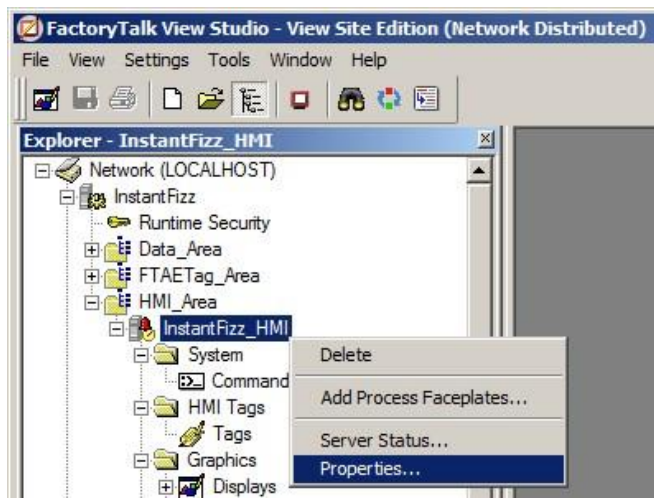


Wait for several seconds to allow FactoryTalk View Studio to open the application. The diagram below shows the InstantFizz application once it is open.

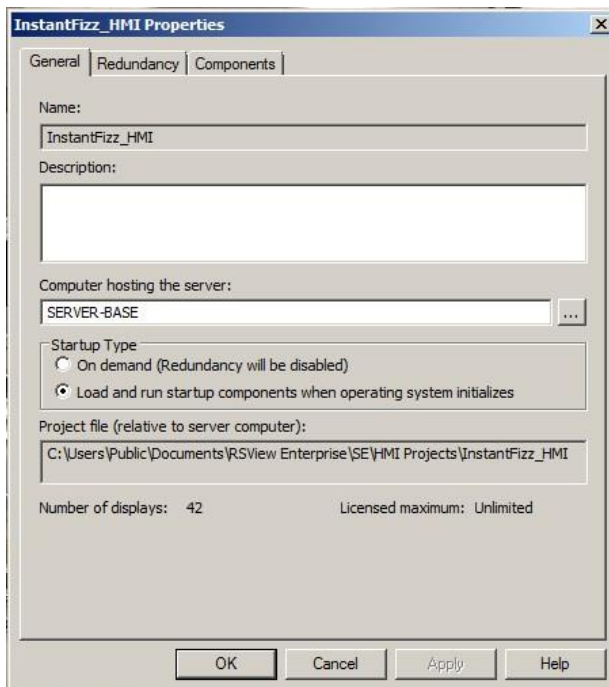


### Explore the InstantFizz\_HMI HMI Server

1. Expand **InstantFizz** to see the **HMI\_Area** icon.
2. Expand **HMI\_Area** to see the **InstantFizz\_HMI** HMI server if it is not already expanded.
3. Right-click on **InstantFizz\_HMI** and select '**Properties...**' to see the server properties.



4. Note the properties of the HMI server.



Notice that the server is set to start up when the computer boots up.

### Startup Type

The Startup type specifies when the HMI Server is to be loaded.

#### On demand

This is the default startup type. The HMI server is loaded as needed, that is when it is needed by FactoryTalk View Studio, the FactoryTalk View Administration Console, or FactoryTalk View SE Clients. When the last client is closed, the HMI server will be unloaded. When you use this startup type, redundancy is disabled.

#### Load and run startup components when operating system initializes

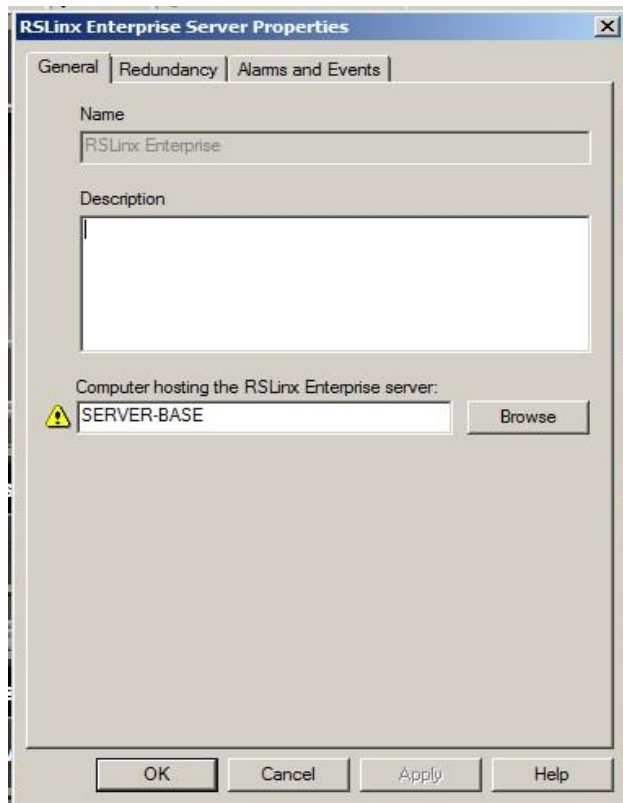
The HMI server and startup components are loaded and run when Windows starts up. The HMI server can be unloaded by changing the startup type to 'On demand' and closing all clients. Use this mode for testing and commissioning applications, and for applications for which you plan to set up redundancy.

5. Click **Cancel**.

### Explore the RSLinx Enterprise data server

To allow our application to monitor and control the tag values within the SoftLogix Controller that is running the control program for this lab, we added an RSLinx Enterprise data server. It provides the best performance when communicating with Logix5000 controllers. Also, the RSLinx Enterprise data server will allow us to subscribe to device-based alarms that originate in the Logix5000 controller program.

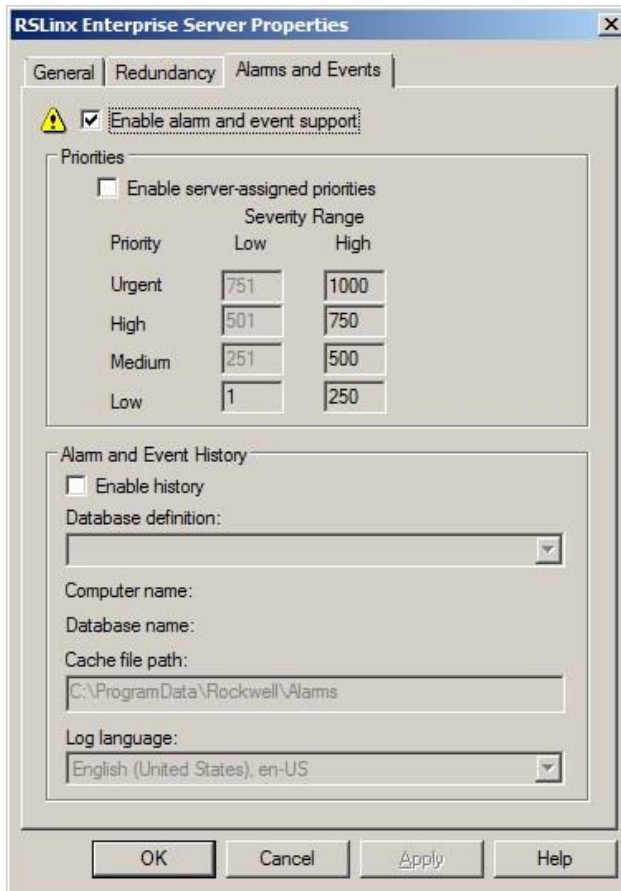
1. To explore the RSLinx Enterprise data server, expand **InstantFizz** to see the **Data\_Area** icon and expand **Data Area**
2. Right-click on **RSLinx Enterprise** and select **Properties...** menu.
3. The RSLinx Enterprise Communication general property page will appear as shown below. Note that the server is running on the current computer (i.e. **SERVER-BASE**).



4. Click on the '**Alarms and Events**' property page.

The Alarms and Events property pages will look like the following. The FactoryTalk Alarms and Events will be discussed in more detail during the Alarming section.





5. Notice that **'Enable alarm and event support'** is enabled.
6. Notice that **'Enable history'** is disabled. It will be enabled later in the lab.
7. Press **'Cancel'** to close the dialog.

## Explore the Communications Setup

We are going to now explore the device shortcut.

**Device Shortcuts:** A Device Shortcut allows you to create a 'pointer' to a device that you can refer to throughout the application while developing displays. This enables the user to change the location of a processor or other such device in one place, which then propagates throughout the rest of the project, without having to change all tag references to that processor. A device shortcut is similar to a Windows shortcut on your computer's desktop that provides easy access to an application.

### Communications Setup Editor Improvements

Offline tag browsing no longer requires you to associate the device shortcut with a controller. In previous releases, you were required to associate a shortcut with both a controller and an offline tag file to be able to browse the tags in the offline tag file. You can now create a shortcut that is associated only with the offline tag file if all you want to do is browse tags in that file.

The Communication Setup editor has been enhanced to prevent the creation of shortcuts that point to devices that do not provide data (such as communication modules and backplanes).

The Communication Setup editor now provides status messages about shortcuts as you create them as well as a summary of all messages via a shortcut verification report.

The Communication Setup editor has an option for enabling alarm and event support at the device level.

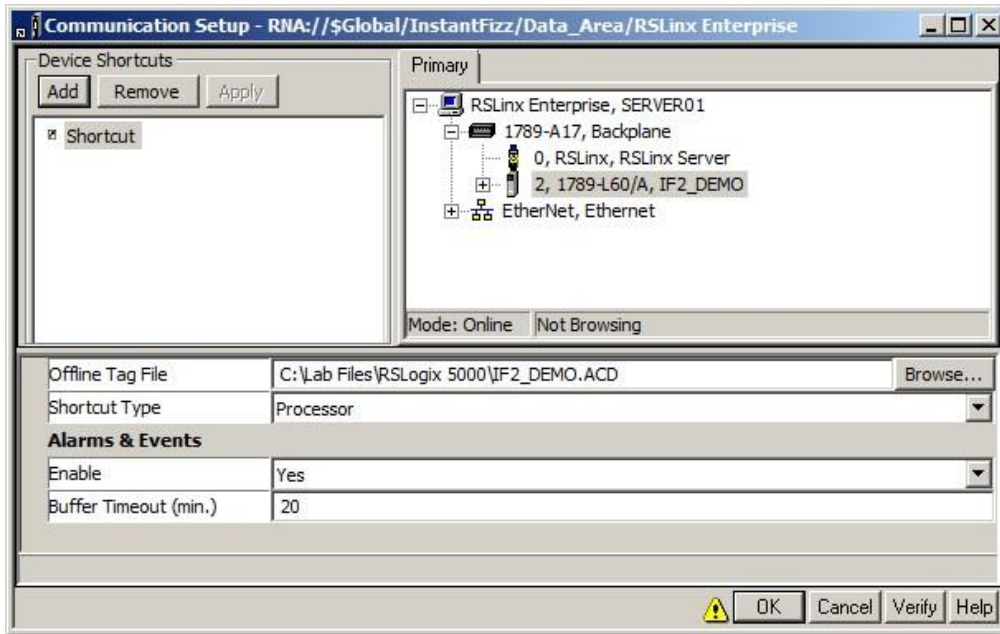
Warnings have been added to FactoryTalk to tell users if making an edit in the development environment will adversely affect the run-time system. If the change is made through a dialog box, this warning icon  appears next to the component where the edit can be made.

1. Open the Communication Setup by expanding the **RSLink Enterprise** device server. Double-click on **Communication Setup**.



The **Communication Setup** dialog will appear to the right of the Explorer tree.

Explore the Device **Shortcut** by following the steps 2-5:



2. Notice the device shortcut named '**Shortcut**' assigned to slot 2 which is the slot number of the SoftLogix module that is running the '**InstantFizz\_Controller**' program.
3. Notice the path to the offline tag file.
4. Notice that '**Alarms and Events**' property is enabled for this shortcut.
5. Press **Cancel** to close.

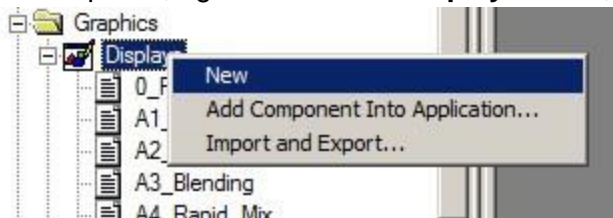
### Offline Tag Browsing

The offline tag file will enable you to browse a ControlLogix controller's tags when that controller is not online. The file must be located on the local PC, not on a networked location.

### Create a display and add a numeric input object

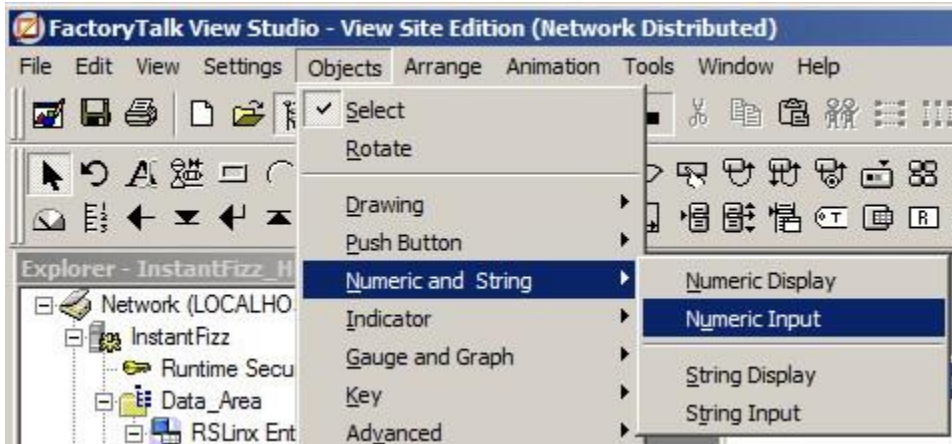
Let's take a minute to verify that communications is working correctly. Verify that you are communicating by adding a Numeric Input Object to a display and then testing that display in FactoryTalk View Studio. Verify that you can access controller tags from the online SoftLogix Controller and the offline tags file by performing tag browses.

1. In the Explorer, right-click on the **Displays** folder, select the **New** context menu item:



An untitled display will be opened.

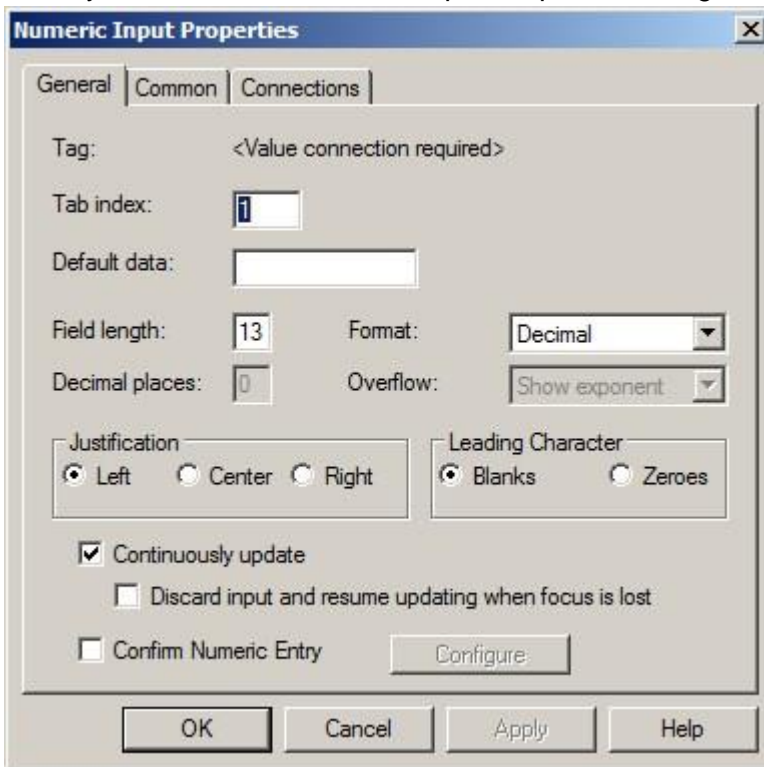
2. Select the **Objects > Numeric and String > Numeric Input** menu.



3. On the empty display, single-click and hold down the mouse button, drag the cursor to draw the numeric input and release the mouse button. As you are dragging the mouse you will see a rectangle to show the size of the object that will be created.



4. When you release, the Numeric Input Properties dialog will appear.



5. Click on the **Connections** tab.

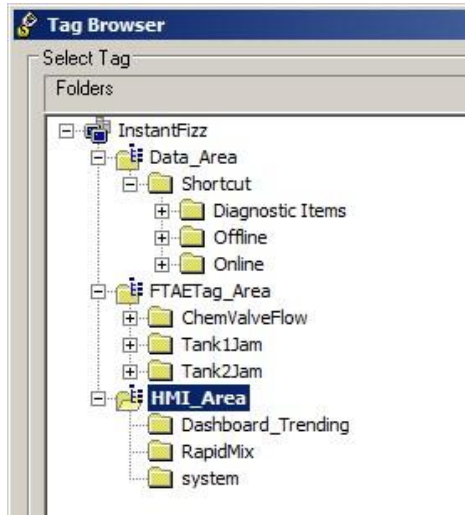
- Click on the **Tag...** button next to the Value field.

General		Common		Connections	
Name		Tag / Expression	Tag	Exprn	
Value	↔		++		
Minimum	←		+++	+++	
Maximum	←		+++	+++	

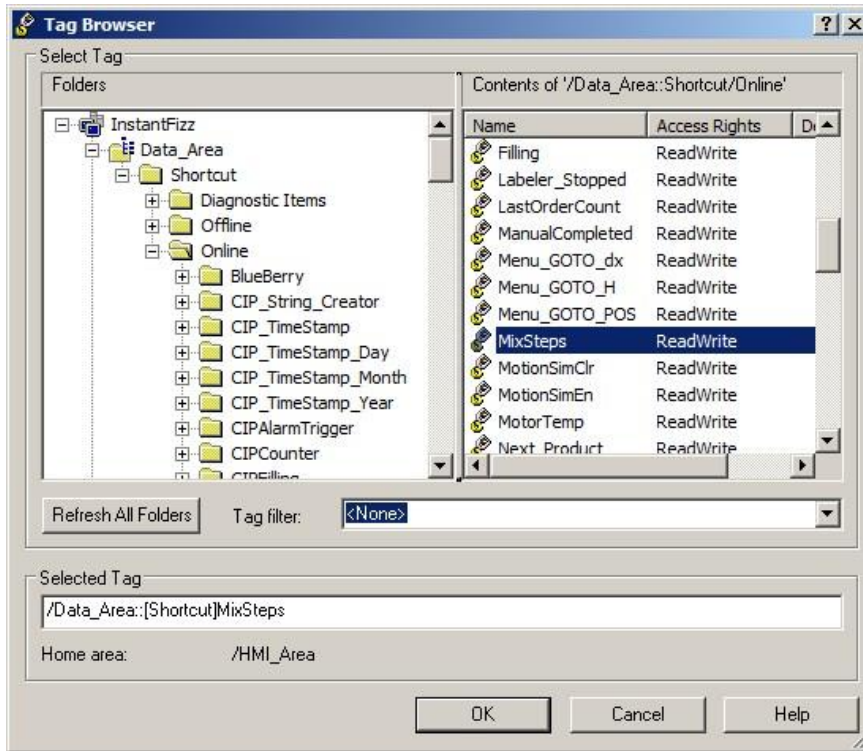
### Numeric Input Minimum and Maximum

Numeric input object allows for user-assigned minimum and maximum values that are validated prior to download.

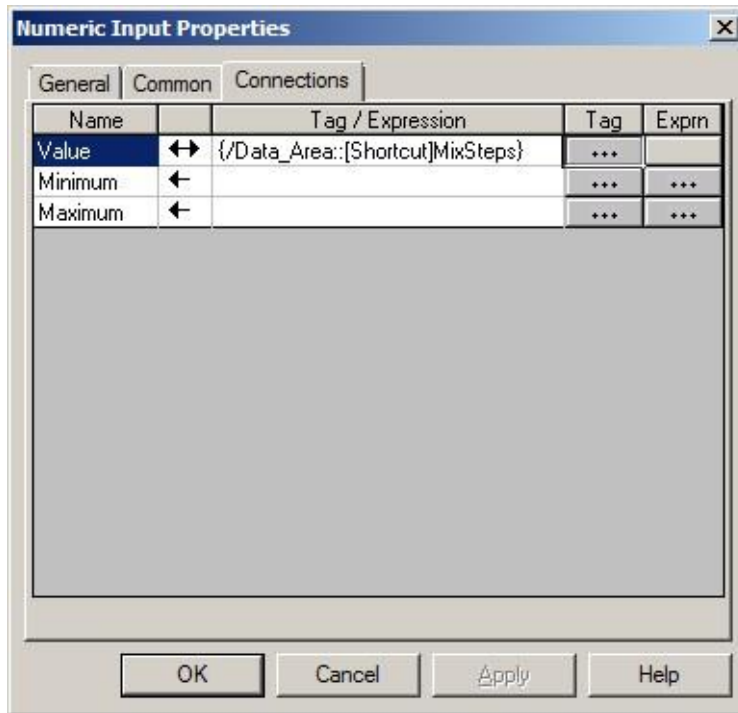
- The Tag Browser will open. The **Shortcut** folder should appear under **Data\_Area**. If you don't see a folder for Shortcut, right-click on **InstantFizz**, and select the **Refresh All Folders** context menu item.




- Expand the **Shortcut** folder, then notice two sub-folders - **Online** and **Offline**. The Online folder references the tags of the online controller that the shortcut is pointing to (such as our SoftLogix controller). The Offline folder references the tags in the ControlLogix controller file (.ACD). Use the Offline tag browsing and Offline Tag File in the RSLinx Enterprise Communications Setup when the online controller is not available, and you only have access to the controller file (.ACD).




- Select the '**Online**' folder.
- Select the **MixSteps** tag from the list of tags on the right, then click **OK**.
- The tag will appear in the Tag/Expression field next to the Value. Your selected tag should look like this:



12. Click the **OK** button on the Numeric Input Properties dialog.
13. After adding an object to our display, we can verify the communications to our controller and test the display by selecting **Test Display** button  on the Graphics toolbar.
14. A value (ex. 3) should appear on the display. This actually verifies that you got the tag from the online tag file and you are communicating with the controller. If you were not communicating with the controller, it would appear as what is called a wireframe, because the data is not available at this time. It would look something like this instead:



15. Click the **Edit Display** button  to get back to edit mode.
16. Close the display, and when prompted to save your changes select **No**.

Section 1 is complete. You have explored a simple network application. You have the building blocks in place and are ready to start creating your graphic displays.

## Section 2: Tags

### About this section

In this section of the lab you will:



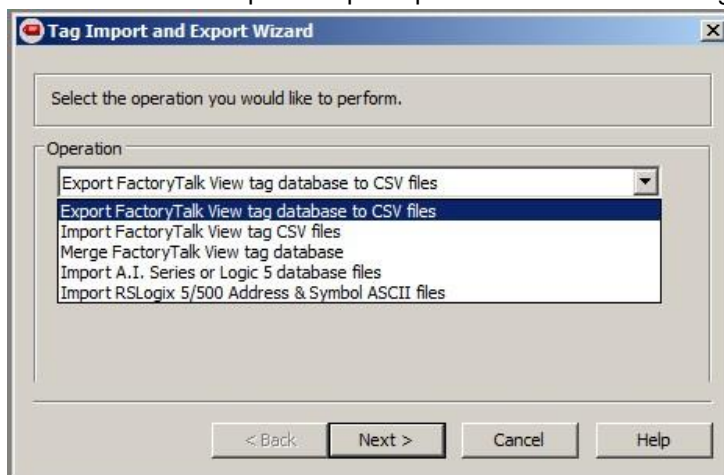
- 
- Export tags
  - Add an object and reference an HMI tag
  - Learn about Tag Labels

## Import/Export Tags Wizard

Let's export the HMI tags using the **Tag Import and Export Wizard**. Use the Tag Import and Export Wizard to convert databases to and from a CSV (Comma Separated Variable) format compatible with Excel. You can also merge one FactoryTalk View tag database with another tag database and import an A.I. Series or Logic 5 text database.

### Import & Export Choices:

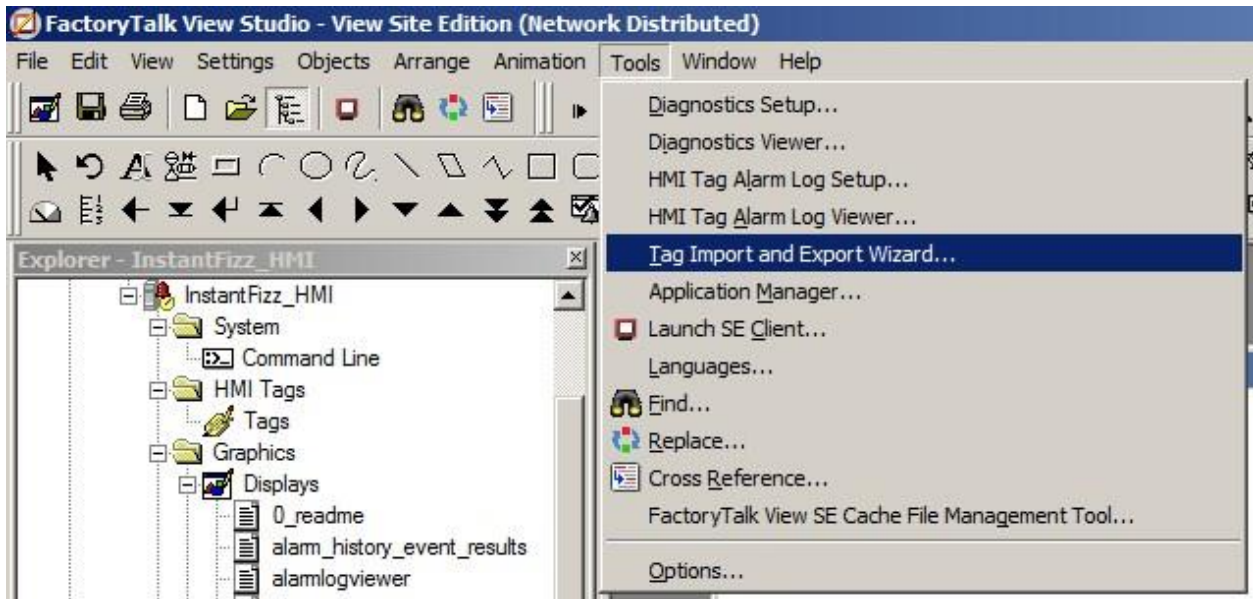
There are several Import / Export options as shown in the Tag Import/Export Wizard.



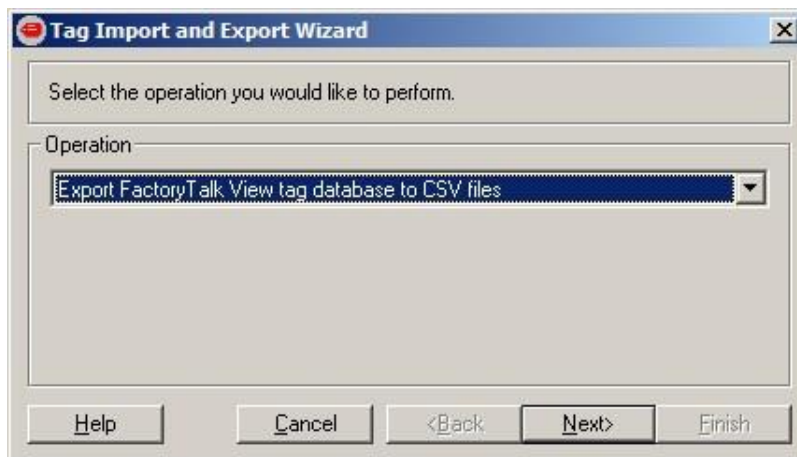
- Export a FactoryTalk View tag database to CSV files - Exports a FactoryTalk View tag database to CSV files.
- Import FactoryTalk View tag CSV files - Imports tag information from CSV files into a FactoryTalk View database. You can create your tags in Excel then import them into the project using this setting in the Tag Import/Export Wizard.
- Merge FactoryTalk View tag database - Merges tag information from one FactoryTalk View project with another FactoryTalk View database.
- Import A.I. Series or Logic 5 database files - Imports symbols from the Rockwell Software A.I. Series or Logic 5 CSV format into a FactoryTalk View database.
- Import RSLogix5/500 Address & Symbol ASCII files - Imports symbols from the ASCII export file of RSLogix5 or RSLogix 500 into a FactoryTalk View database.

1. Return to FactoryTalk View Studio and open the **Tag Import and Export Wizard** from the **Tools** menu.

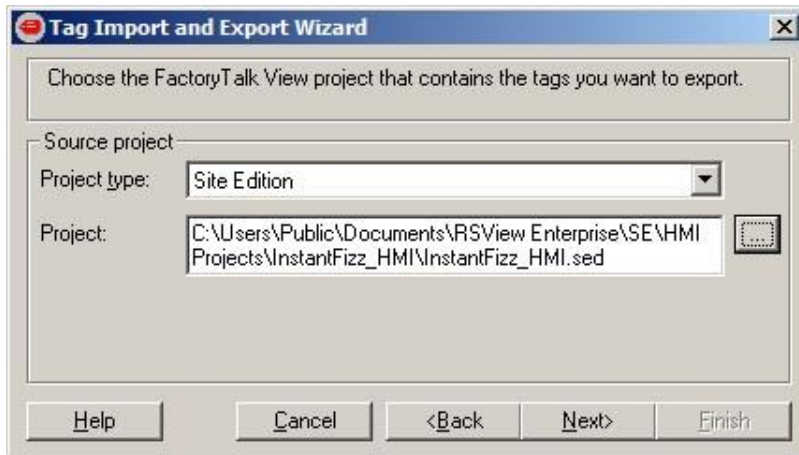




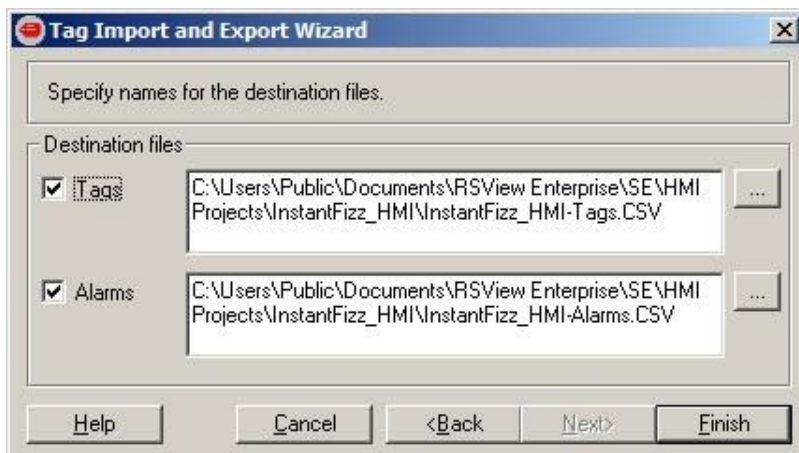
2. Select the 'Export FactoryTalk View tag database to CSV files' option from the drop-down and then click **Next**.



3. Select **Site Edition** as Project Type.
4. Browse to our HMI project under C:\Users\Public\Documents\RSView Enterprise\SE\HMI Projects\InstantFizz\_HMI. Select the file **InstantFizz\_HMI.sed**.



5. Select **Next** to continue.
6. Select **Finish** to export.



7. Let's open up the exported tag (.CSV) file in Excel. Double-click the following file to see it in Excel:

C:\Users\Public\Public Documents\RSView Enterprise\SE\HMI  
Projects\InstantFizz\_HMI\InstantFizz\_HMITags.csv

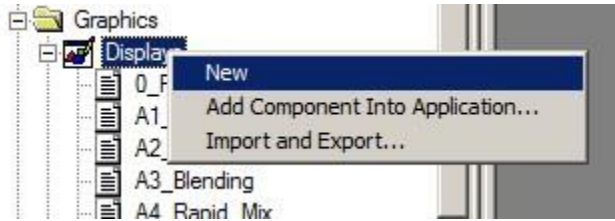
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R
	Tag Type	Tag Name	Tag Descr	Read On	Data Sour	Security C	Alarmed	Native Ty	Value Ty	Min Anal	Max Anal	Initial An	Scale	Offset	DeadBanc	Units	Off Label	On L
1																		
2		;###002 - THIS LINE CONTAINS VERSION INFORMATION. DO NOT REMOVE!!!																
3																		
4		;Folders Section (Must define folders before tags)																
5	F	Dashboard_Trending	F															
6	F	RapidMix	F															
7																		
8		;Tag Section																
9	A	BlueBerry_Amt	F	M	*	F	D	L	0	100	100	1	0	0				
10	A	BlueBerry_Increase	F	M	*	F	D	L	0	100	0	1	0	0				
11	A	BlueBerry_Target	F	M	*	F	D	L	0	100	0	1	0	0				
12	A	CIP_Valve0	F	M	*	F	D	L	0	100	0	1	0	0				
13	A	CIP_Valve1	F	M	*	F	D	L	0	100	0	1	0	0				
14	A	CIP_Valve2	F	M	*	F	D	L	0	100	0	1	0	0				
15	A	CIP_Valve3	F	M	*	F	D	L	0	100	0	1	0	0				
16	A	Dashboard_Trending	F	D	*	F	F	F	0	1	0	100	0	0				
17	A	Dashboard_Trending	F	D	*	F	F	F	0	1	0	100	0	0				
18	A	Dashboard_Trending	F	D	*	F	F	F	0	1	0	100	0	0				
19	A	DigitGrouping	F	M	*	F	L	L	0	1E+09	0	1	0	0				
20	A	GradientPercentage	F	M	*	F	D	L	0	100	0	1	0	0				
21	A	Kiwi_Amt	F	M	*	F	D	L	0	100	200	1	0	0				
22	A	Kiwi_Increase	F	M	*	F	D	L	0	100	0	1	0	0				
23	A	Kiwi_Target	F	M	*	F	D	L	0	100	0	1	0	0				
24	D	LearningMode	F	M	*	F											Off	On
25	A	Mango_Amt	F	M	*	F	D	L	0	100	300	1	0	0				
26	A	Mango_Increase	F	M	*	F	D	L	0	100	0	1	0	0				
27	A	Mango_Target	F	M	*	F	D	L	0	100	0	1	0	0				
28	D	MoreMenuTag	F	M	*	F											Off	On
29	A	Percentage_Val	F	M	*	F	D	L	0	100	0	1	0	0				
30	A	RapidMix\WaterAm	F	M	*	F	D	L	0	50000	0	1	0	0	0 Litres			
31	A	SF_Test	F	M	*	F	D	L	0	15	0	1	0	0				
32																		
33																		

8. Notice that you have a mixture of Device and Memory tags. Scroll over to Column W (Address) to see the Device tag addresses in the controller.
9. We could make changes to this file and add, delete or modify tags and then re-import them back into the HMI project using the Import option of the wizard.
10. Close the Excel file.

**Add an object referencing an**

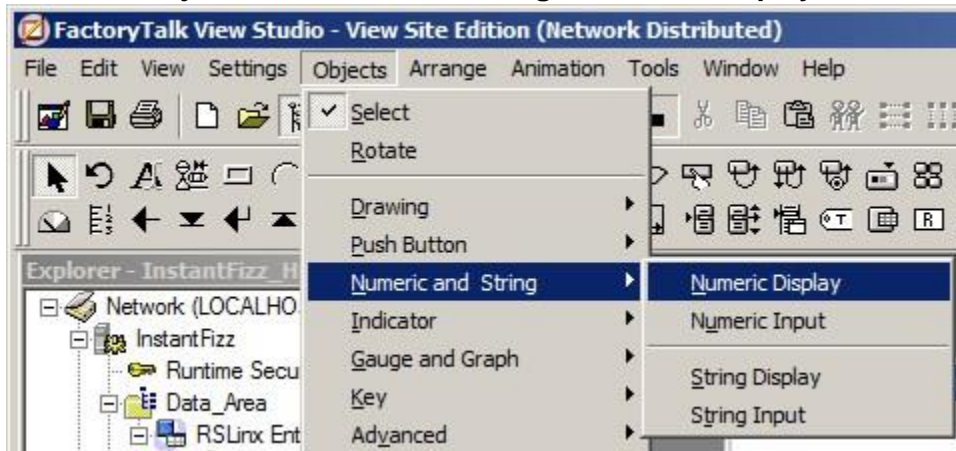
**HMI tag**

1. Create a new display.



An untitled display will be opened.

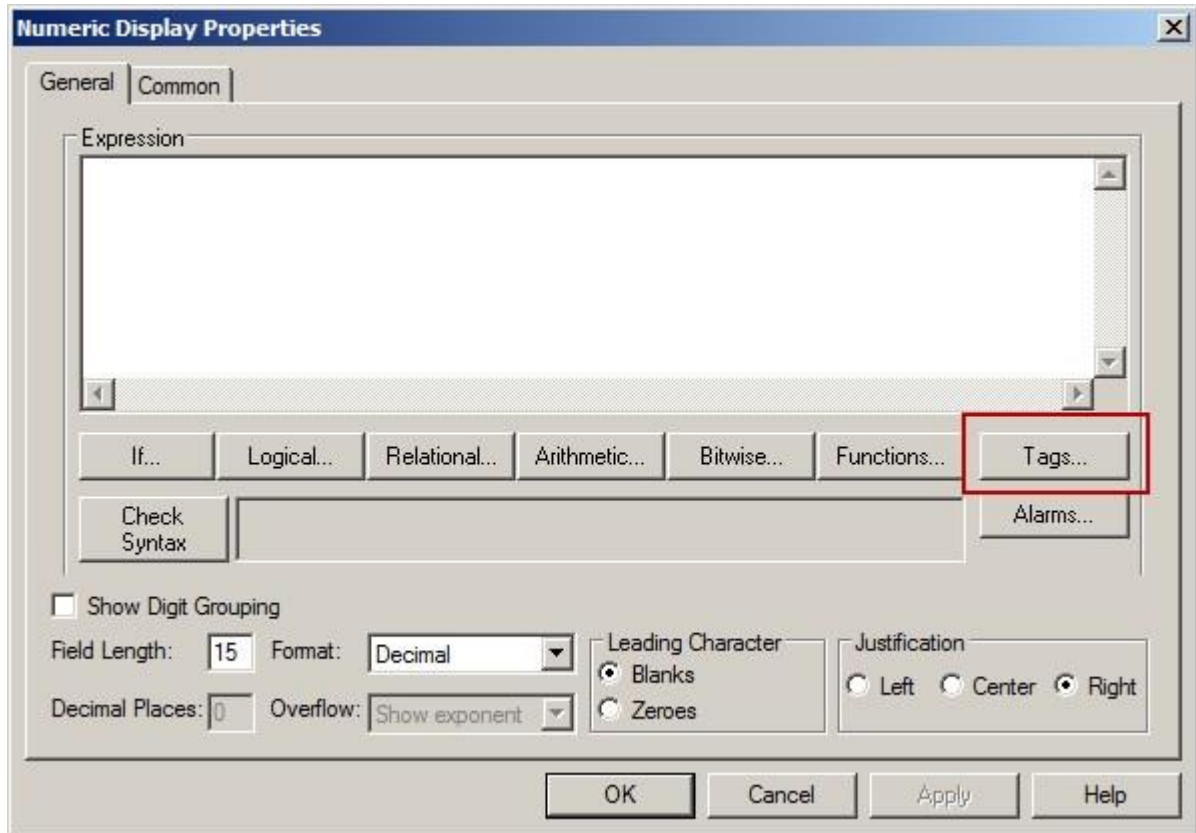
2. Select the **Objects > Numeric and String > Numeric Display** menu.



3. On the empty display, single-click and hold down the mouse button, drag the cursor to draw the numeric display and release the mouse button. As you are dragging the mouse you will see a rectangle to show the size of the object that you are creating:



4. The **Numeric Display Properties** window will open. Select **Tags** to open the Tag Browser and find a tag to reference.



5. The same Tag Browser window is used to browse for **HMI Tags**, **Direct Reference Tags** associated with an online and offline controller, **Diagnostic Items** and pre-defined **System tags**.

Let us review the Tag Browser.

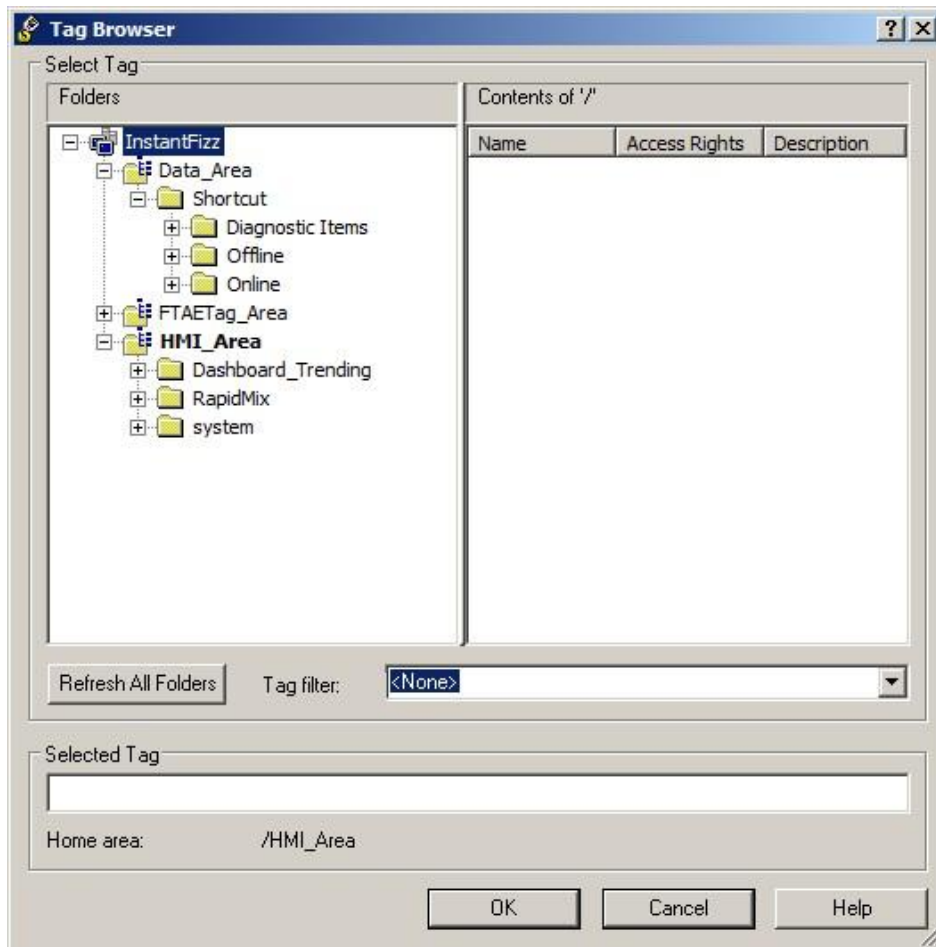
6. '**Shortcut**' is the device shortcut associated with the SoftLogix controller. It contains all the direct reference tags to the controller.

**Diagnostic Items:** Pre-defined diagnostics tags to Logix controllers, such as @Mode, which indicates the current mode of the controller (Run, Program, Remote)

**Offline:** Direct reference tags to the Offline Tag File that was associated with the

**'Shortcut' Online:** Direct reference tags to the controller.

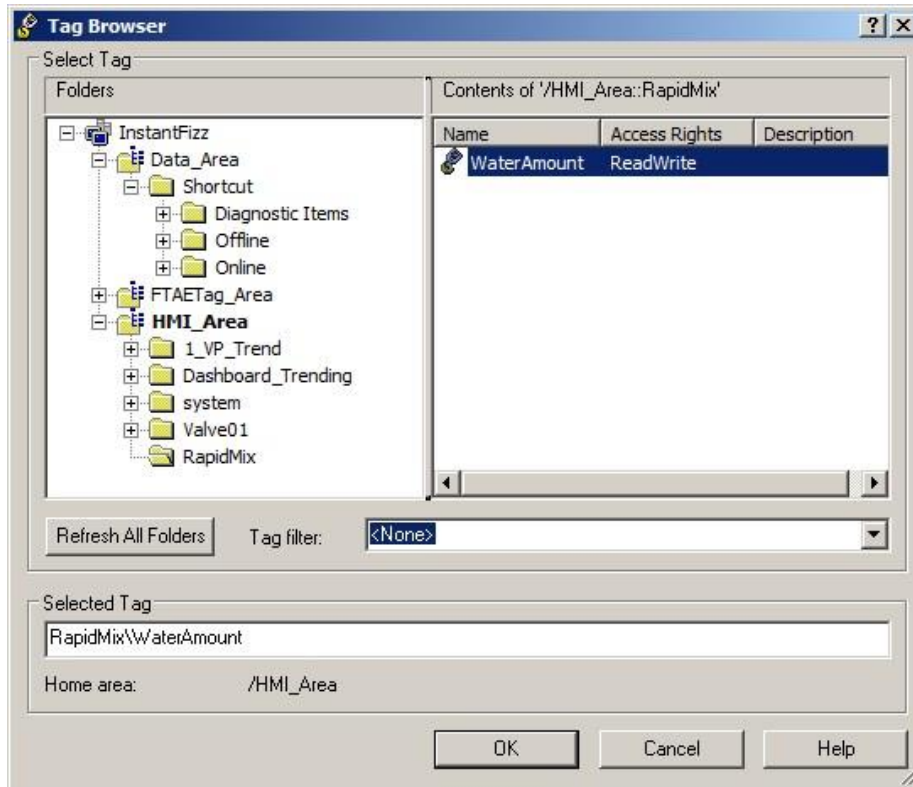
7. **Dashboard\_Trending**, **RapidMix** and **system** are folders created in the HMI Tag Database.
8. **System** contains the pre-defined memory tags and are stored in the HMI Server. For example, **System\Second**.



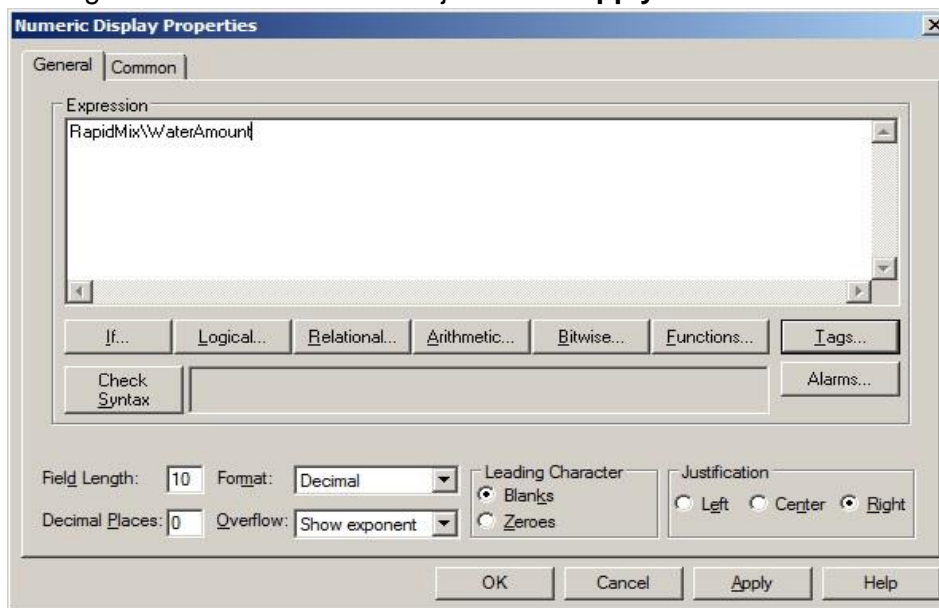
### Structured Tag Support in FactoryTalk Tag Browser

The tag browser lets you select a structure tag in the left-hand pane of the object browser and return a partial tag identifier to the editor that launched the browser. This structured tag can be assigned to a faceplate object to supply values to multiple objects.

9. Browse to the HMI tag **RapidMixWaterAmount** in the **HMI\_Area**.
10. Expand the **RapidMix** folder, then select **WaterAmount** and click **OK**.



11. The tag has been added to the object. Click **Apply** then **OK** to exit.

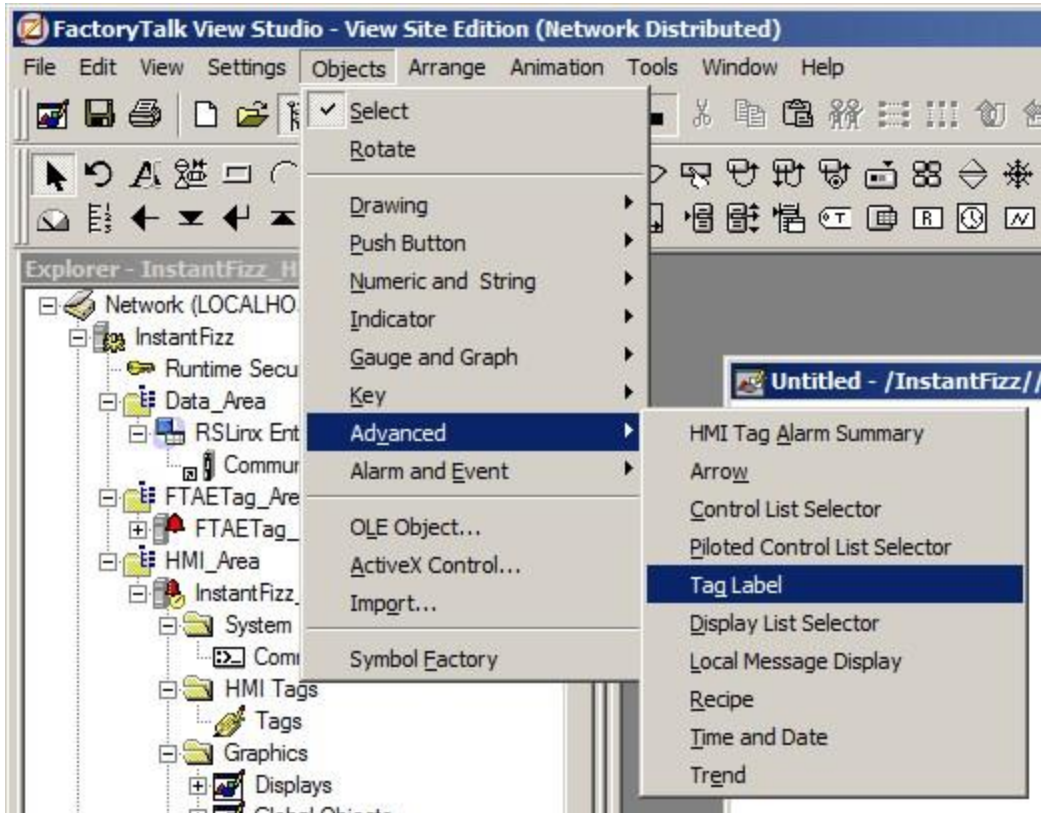


12. Add a Tag Label object on the same display to the right of the Numeric Display object using **Objects> Advanced > Tag Label** menu and drawing out the object on the display.

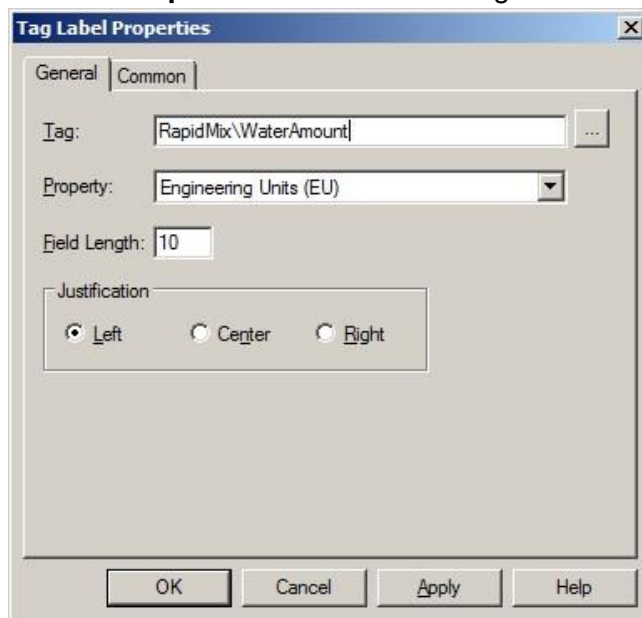
Use Tag Label objects to display information about a tag's properties at run time. You can display the value of one property per tag label. The properties include: Low EU (tags Minimum Value), High EU (tag Maximum Value), Contact Value (tag Status), Engineering



Units (EU) (tag Units), Tag Name, Tag Description, Contact Open Label (tag Off Label), and Contact Close Label (tag On Label).



13. The Tag Label Properties window will open. Set up the properties as shown below. Browse to the same **RapidMix\WaterAmount** tag and set the Property to **Engineering Units (EU)**.



14. Click **OK** to close.



---

15. Test the display by selecting the **Test Display** button  on the Graphics toolbar.

16. Review the display. The **Numeric Display** object will display the **current value** of the **RapidMix\WaterAmount** HMI Tag and the **Tag Label** will display the defined **Engineering Units** of the HMI Tag.



**Note:** The Tag Label object only works with HMI Tags, because we defined the properties (such as Units) of this tag in the HMI Tag Database. It does not work with direct references. To display the engineering units of a direct reference you can add a Text Object and type the applicable units.

17. Click the **Edit Display** button  to get back to edit mode.

18. Close the display **without** saving.

Congratulations! You have learnt how to Export tags, add an object to a display, reference an HMI tag and create Tag Labels for HMI tags.

### Section 3: Graphic Displays, Objects and Animations

#### About this section

In this section of the lab you will:

- Learn about graphic displays
- Modify an XML file
- Configure a tooltip
- Learn about docked displays
- Use Commands
- Configure and use Animations and Expressions
- Use the Object Explorer
- Use Local Messages
- Use Symbol Factory Graphics
- Learn about color animation enhancements
- Learn about Find and Replace

- Learn about Cross Reference
- Learn about PNG file type support
- Adding tag references from within a Parameter file

To help with the navigation within FactoryTalk View Studio for the components used in this section, you can again refer to the Application Explorer image on Page 15.

## Graphic Displays

### Types of Graphic Displays

**Standard Displays** - stored in the Displays folder. These are the displays that the operator sees at run time. They present views of automated plant activity or processes. They can show system or process data and provide operators with a way to write values to a real-time database or network devices such as a controller.

**Global Object Displays** - stored in the Global Objects folder. Global object displays let you link the appearance and behavior of a graphic object on a global object display to multiple copies of that object in standard displays. When you make changes to the original object, the changes are automatically applied to the copies.

**Library Displays** - stored in the Libraries folder. A library display contains ready-made graphic objects that you can use in other displays.

A graphic display represents the operator's view of plant activity. The display can show system or process data, and provide operators with a way to write values to external devices such as programmable controllers. The elements that make up a graphic display are called graphic objects. The Objects toolbar (see below) in the Graphic Displays editor provides simple drawing elements such as line, rectangle and ellipse, as well as ready-made objects such as push buttons, input and display fields, and alarm summaries. Use these elements to create visual representations of processes and activities then animate the display by linking objects to tags so that the appearance of the objects will change as the values of the tags change.

The graphics editor allows you to easily duplicate objects, reshape or resize objects, and arrange them in a variety of ways like stacking them, aligning them with each other, spacing them horizontally or vertically, flipping them horizontally or vertically, rotating them, and grouping them so they behave as a single object. Graphic objects can be:

- Created using the Graphic Display editor.
- Copied and pasted from the Graphics Libraries.
- Copied to the clipboard from another Windows application and then pasted into the graphics display.
- Created by another Windows application and inserted into the graphic display using object linking and embedding.
- Dragged and dropped from another graphic display or library, or another Windows application.

## Objects Toolbar



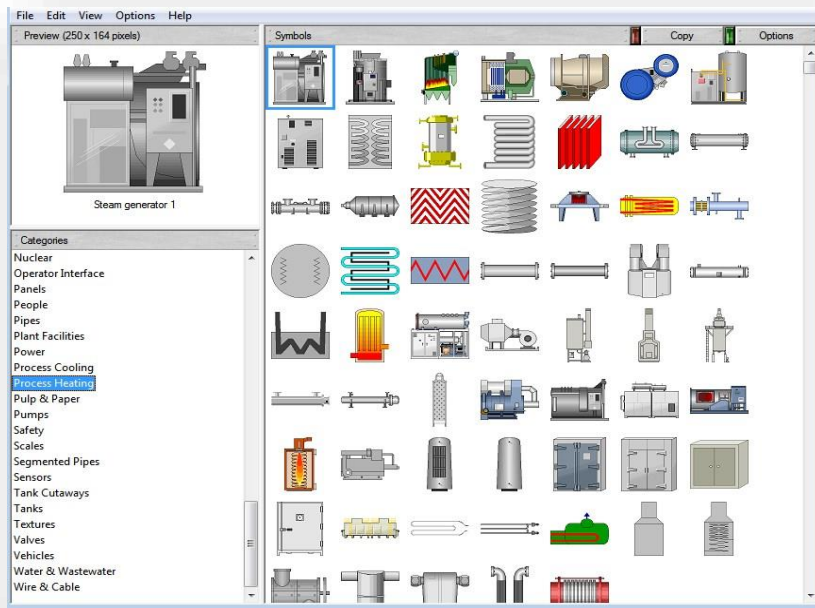
The Object Toolbar contains buttons for commonly-used menu items so you can access the items without opening the menus. When you point to a button on the toolbar, its description is displayed below it. This is called a tooltip. Use the items on the **View > Toolbars** menu to display or hide the various toolbars.

## Graphic Libraries

FactoryTalk View SE offers two kinds of graphics libraries: Symbol Factory and the standard Libraries.

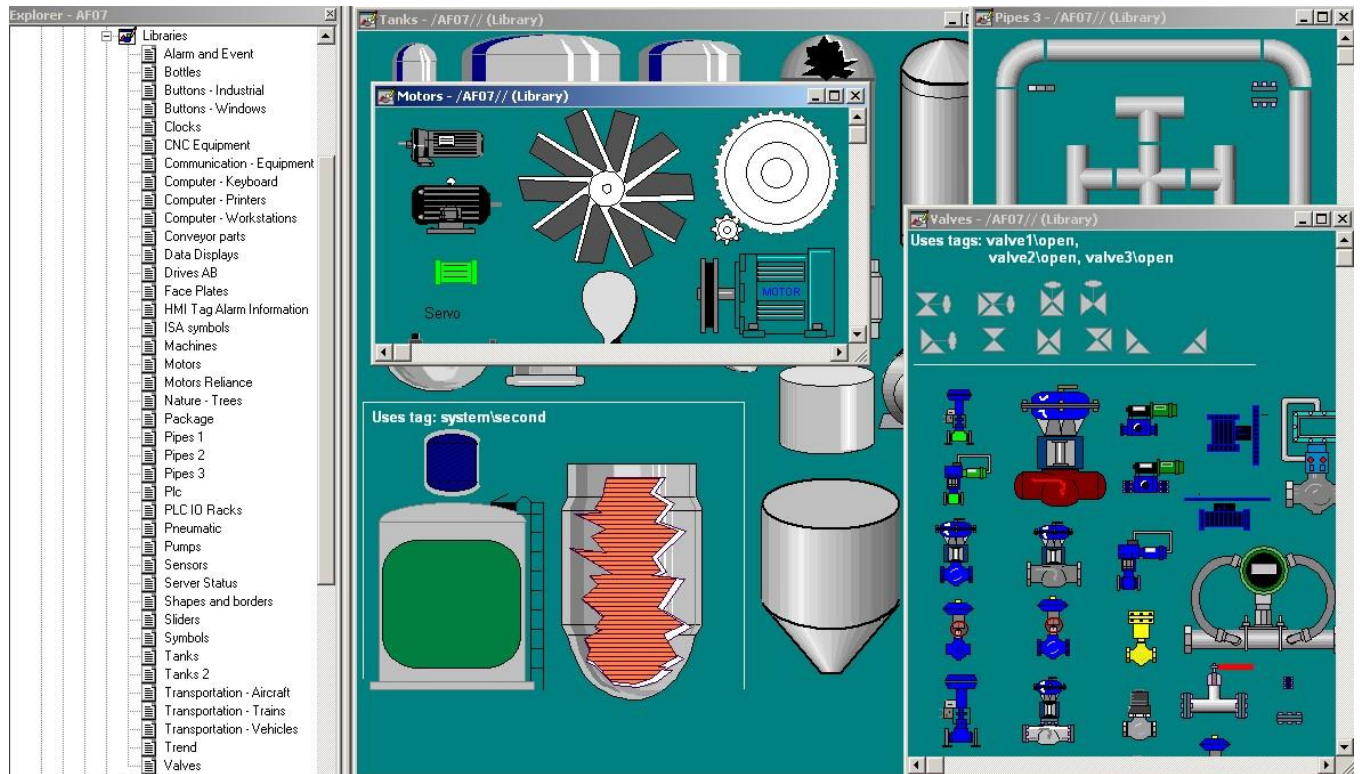
### Symbol Factory

Symbol Factory is a graphics library interface that can be launched from FactoryTalk View Studio. It is a common library between FactoryTalk View Machine Edition and Site Edition. **Symbol Factory** is built on **Software Toolbox's Symbol Factory**, and contains over 5,000 graphical objects. Approximately 4,000 of these are grouped object based graphics with the remainder being static bitmaps. The **Symbol Factory** library supports drag-and-drop and copy/paste onto a native FactoryTalk View display.



### Libraries

The Graphics Library comes with a number of ready-made graphic displays containing objects that you can use in other displays.



There are many different library objects that you can use within your applications.

**Note:** Any animation that has been attached to an object will be included with it when it is copied into a display.

### Modifying a Graphic Display by editing the Associated XML file

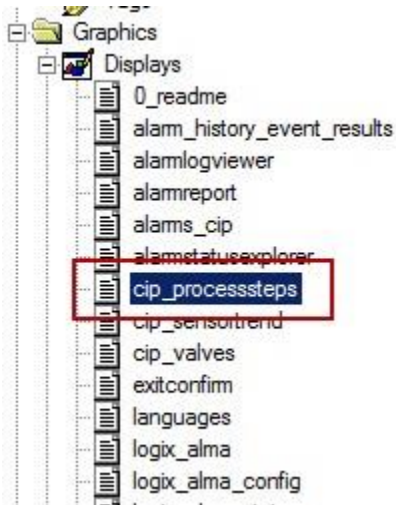
In FactoryTalk View Studio, you can also use the Graphics Import Export Wizard to export graphic display information to an XML file, or to import a graphic display XML file into an application. XML has a standardized format and structure. You can modify the elements and attributes of a graphic display by changing them, or by adding new ones, in the XML file.

You can edit the XML files before importing them back into an application, to modify existing graphic objects, or to add new objects to a display.

You can create or edit graphic displays independent of FactoryTalk View Studio, including display settings, objects, object properties, connections, animations, groupings, key assignments etc.

In this section, we will export a graphic display to an XML file, modify the XML file and then import the XML file back into the graphic display.

1. Double-click the 'cip\_processsteps' display to open it.



We will modify the **Step 1** text of this display in the XML file. Currently, the text displays - 'Step 1: Adding H2O'.



2. Close the display without saving.
3. Let's export the graphic display to an XML file. Right-click on **Displays** and select 'Import and Export....'



4. Select 'Export graphic information from displays', then select **Next**.



5. Press the **Clear All** button and just select the '**cip\_processteps**' graphic display, then select **Next**.

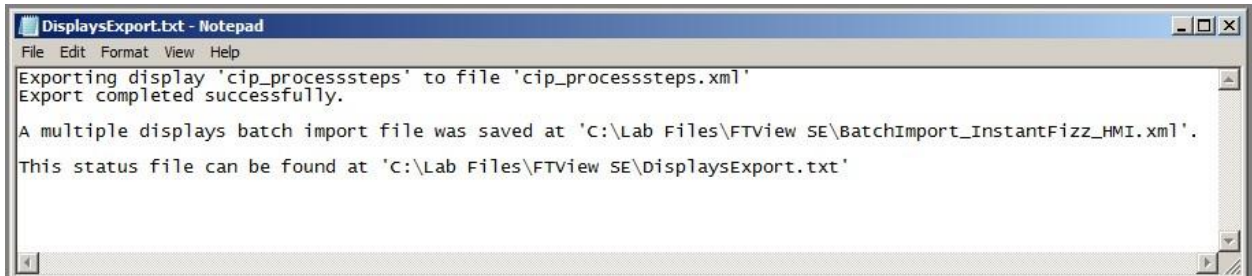


6. If the folder displayed is not **C:\Lab Files\FTView SE** then press the '**...**' button and browse to that folder, then select **Finish**.





7. The export will then begin and a progress bar will appear. When the export has completed, a **status** file will be displayed in **Notepad**. You can ignore any export errors in the **status** file. Press the 'X' in the top right corner of the **Notepad** window to close it.

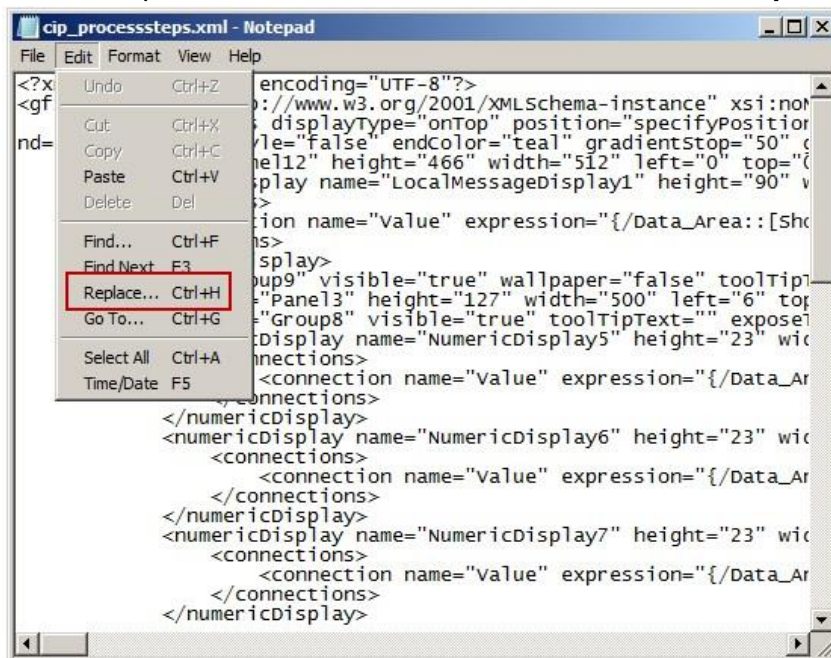


8. Go to the **C:\Lab Files\FTView SE\** folder in Windows Explorer. To modify the file we must open it up in Notepad. Right-click on the file '**cip\_processsteps.xml**' then select **Open With** and choose **Notepad**. The file will now open in Notepad:

```
<?xml version="1.0" encoding="UTF-8"?>
<gfx xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance" xsi:noNamespaceSchemaLocation="Gfx-SE6.xsd">
  <displaySettings displayType="onTop" position="specifyPositionInPixels" positionX="200" positionY="100" se
nd="" useGradientStyle="false" endColor="teal" gradientStop="50" gradientDirection="gradientDirectionHorizontal"
  <panel name="Panel12" height="466" width="512" left="0" top="0" visible="true" wallpaper="false" tooltipText=""
  <localMessageDisplay name="LocalMessageDisplay1" height="90" width="500" left="6" top="4" visible="true" v
  <connections>
    <connection name="Value" expression="{/Data_Area::[Shortcut]Program:Main.CIPSteps}"/>
  </connections>
</localMessageDisplay>
<group name="Group9" visible="true" wallpaper="false" tooltipText="" exposeToVba="notExposed" isReferenc
  <panel name="Panel3" height="127" width="500" left="6" top="334" visible="true" tooltipText="" expose
  <group name="Group8" visible="true" tooltipText="" exposeToVba="notExposed" isReferenceObject="false">
  <numericDisplay name="NumericDisplay5" height="23" width="70" left="60" top="411" visible="true" t
  <connections>
    <connection name="Value" expression="{/Data_Area::[Shortcut]SanConc}"/>
  </connections>
</numericDisplay>
<numericDisplay name="NumericDisplay6" height="23" width="70" left="220" top="411" visible="true"
  <connections>
    <connection name="Value" expression="{/Data_Area::[Shortcut]CIPClarity}"/>
  </connections>
</numericDisplay>
<numericDisplay name="NumericDisplay7" height="23" width="70" left="373" top="411" visible="true"
  <connections>
    <connection name="Value" expression="{/Data_Area::[Shortcut]ChemConc}"/>
  </connections>
</numericDisplay>
<text name="Text6" height="30" width="112" left="39" top="381" visible="true" tooltipText="" expos
<text name="Text11" height="15" width="101" left="205" top="389" visible="true" tooltipText="" exp
<text name="Text12" height="15" width="115" left="351" top="389" visible="true" tooltipText="" exp
<text name="Text5" height="29" width="279" left="120" top="351" visible="true" tooltipText="" exp
</group>
</group>
</gfx>
```

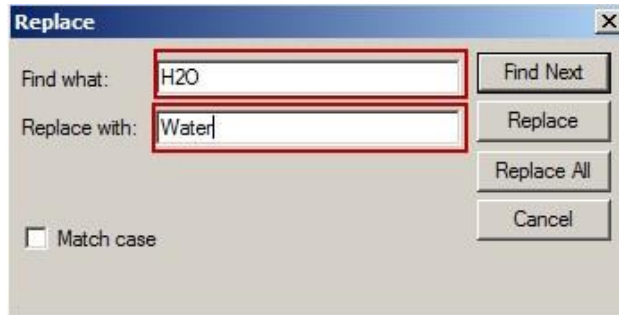
We will change the text 'Step 1: Adding H2O' to 'Step 1: Adding Water'.

9. In the Notepad file, select the **Edit** menu item, then click **Replace**.

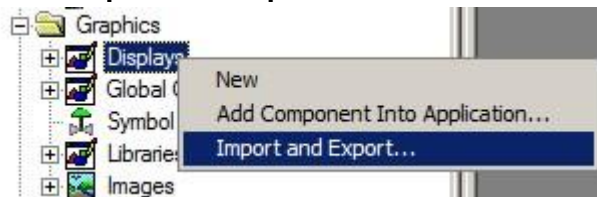


10. Fill the **Replace** window as follows then click the **Replace All** button. The **Find What** text is letter **H**, number **2** and letter **O**.

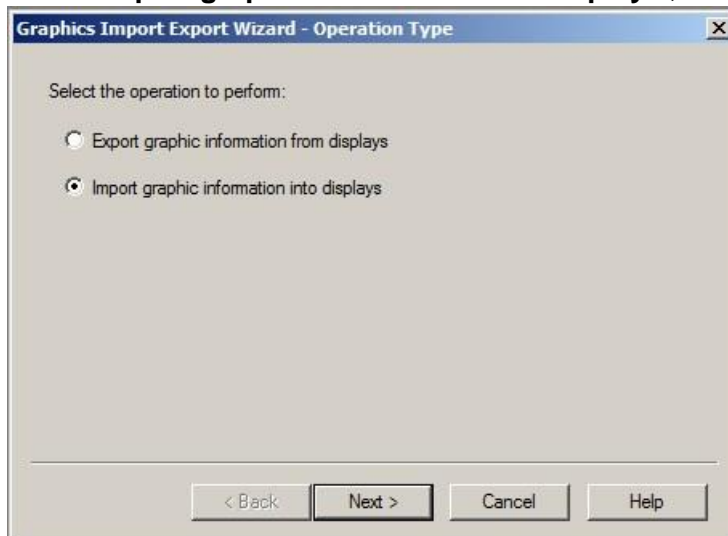




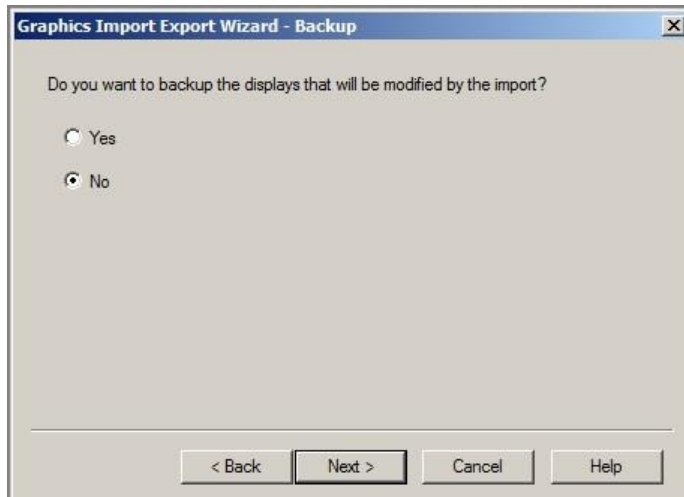
11. Press **Cancel** to exit.
12. Save the file from **File>Save**, then **Exit**.
13. Let's import the modified display. Right-click on **Displays in FactoryTalk View Studio** and select '**Import and Export....**'



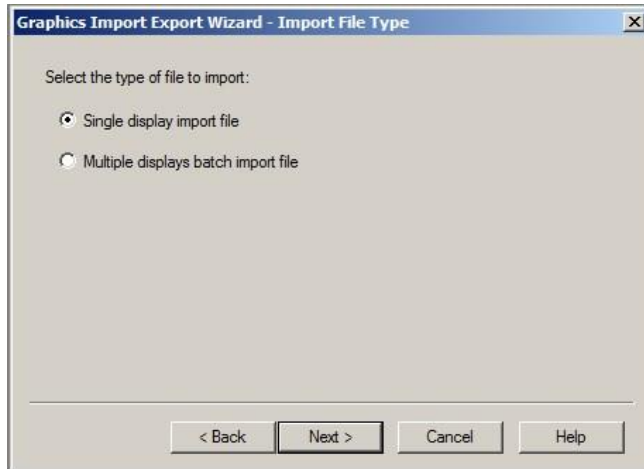
14. Select '**Import graphic information into displays**', then select Next.



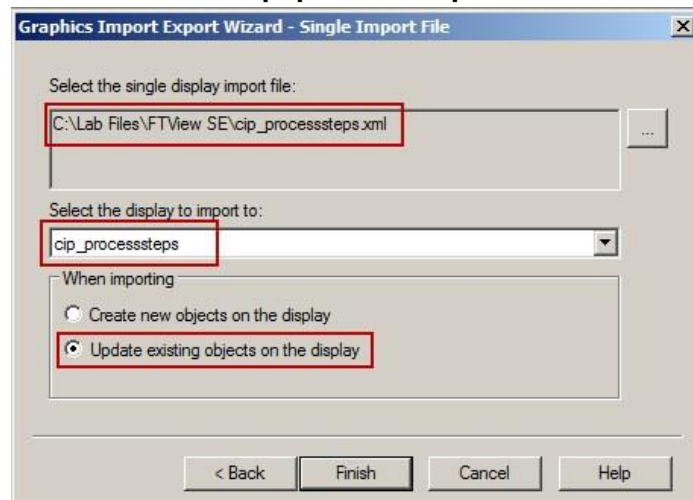
15. Select **No**, then click **Next**.



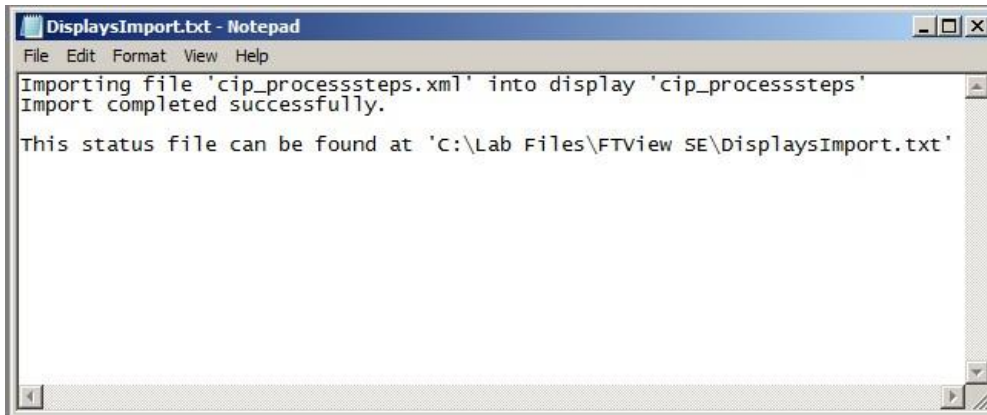
16. Select **Single Display Import File**, then click Next.



17. Configure the window as shown below. The XML file to import can be found in **C:\Lab Files\FTView SE\cip\_processsteps.xml**. This is the file we have just modified.



18. Click **Finish** to complete the import. The Graphics Import Export Wizard provides a **status file** on the success of the import/export. Close the file.



19. Let's review the modified display. Now, open the **cip\_processsteps** display by double-clicking on it from the Displays item in FactoryTalk View Studio.
20. Review the display. The text has been changed from H2O to Water.



**Note:** In this example it would have been easier to simply change the text in the display within FactoryTalk View Studio. There are applications, however, in which you may need to automatically create many objects and/or displays. Creating them in an XML file can save a lot of engineering time.

## Tooltips

To provide information about a graphic object to an operator, you can add a tooltip to objects:

- Tooltips can be added to graphical objects

- Tooltips support embedded variables
- Language switching is supported on tooltip text

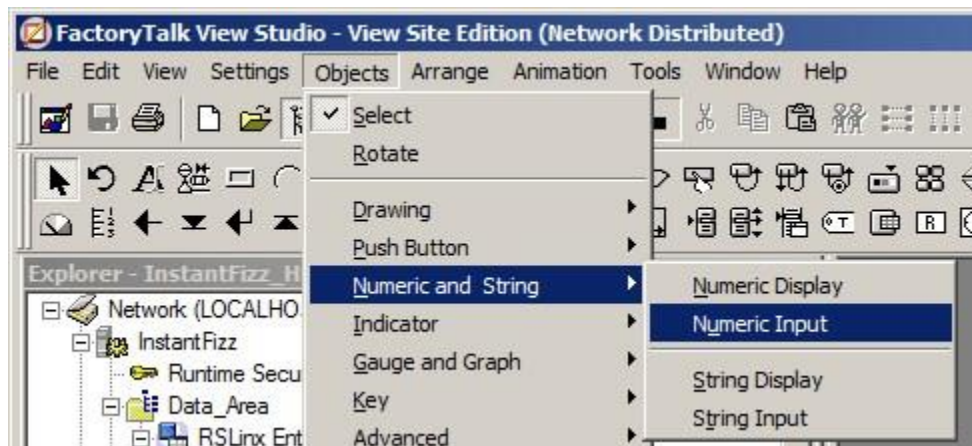
By default, an object has no tooltip. If you add a tooltip, it displays at run time, when the operator positions the pointer over the object for a few seconds.

### Parameter Enhancements

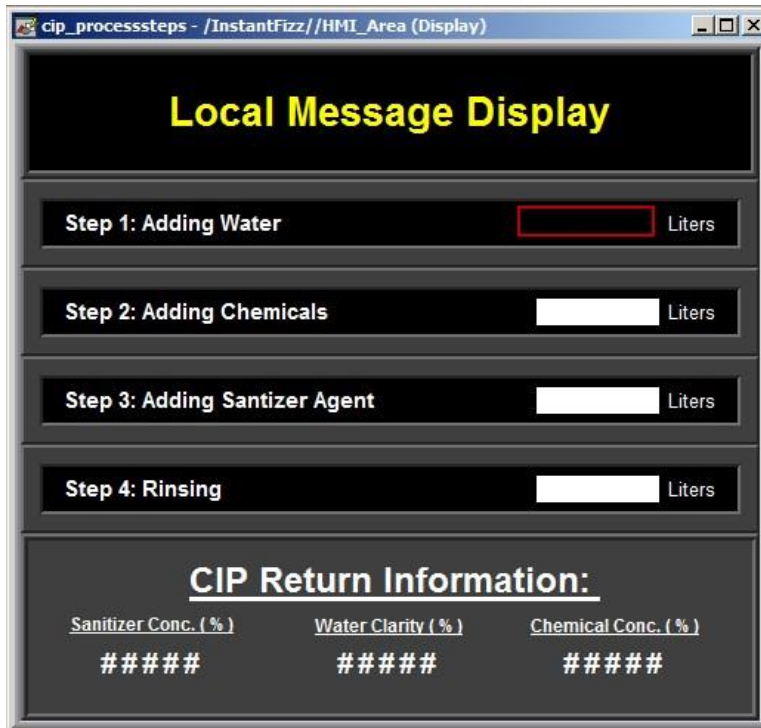
Added parameter enhancements provide support for embedded variables in tooltips and title bars. Literal numbers and strings can now be added into the embedded variable syntax.

In this lab we will add a Numeric Input object to show a tooltip example.

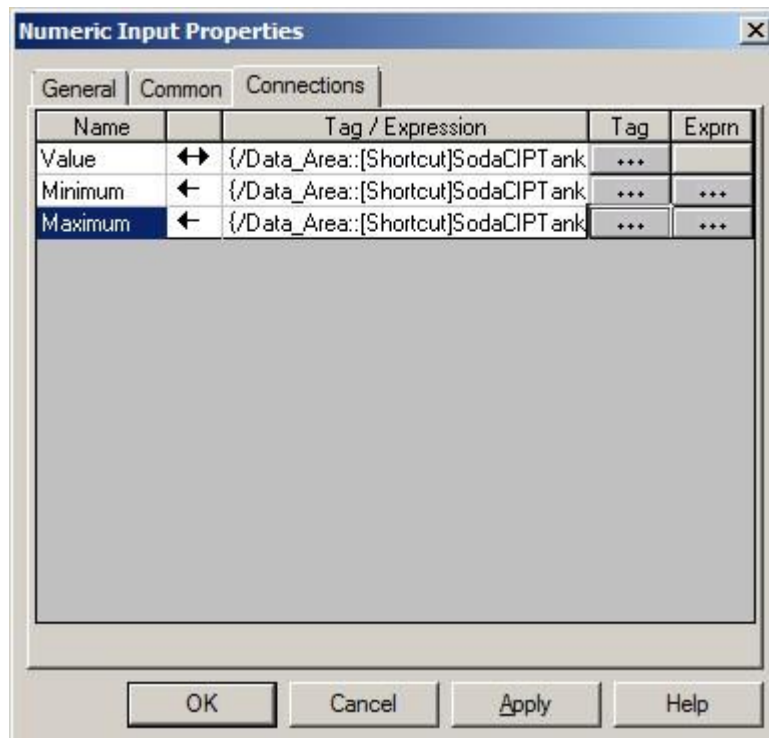
1. In FactoryTalk View Studio, open the **cip\_processsteps** display (this may already be open.)
2. From the **Objects** menu, select **Numeric and String**, then click on **Numeric Input**.



3. Add the **Numeric Input** object in the display location shown below in red.

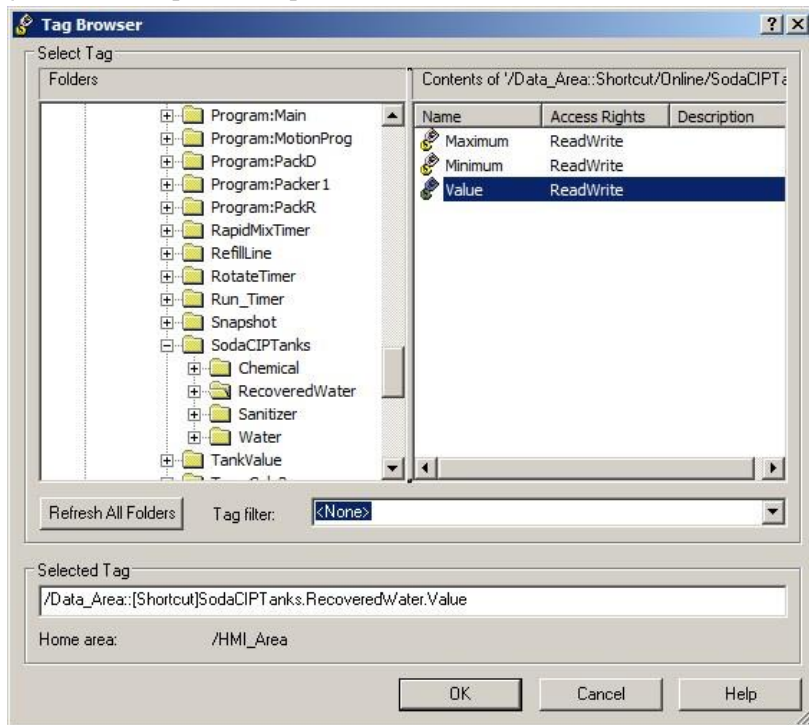


4. Select the **Connections** tab.



5. Click on the ellipses button to browse and add the **Value** tag under **InstantFizz >Data\_Area>Shortcut>Online>SodaCIPTanks>RecoveredWater** folder.

**{/Data\_Area::[Shortcut]SodaCIPTanks.RecoveredWater.Value**



6. Click on the ellipses button to browse and add the **Minimum** tag:  
**{/Data\_Area:: [Shortcut]SodaCIPTanks.RecoveredWater.Minimum**
7. Click on the ellipses button to browse and add the **Maximum** tag:  
**{/Data\_Area:: [Shortcut]SodaCIPTanks.RecoveredWater.Maximum**
8. Click **OK** to close.
9. Save the display.

10. Test the display by selecting **Test Display** button  on the Graphics toolbar.

We have specified the Minimum and Maximum values for the Numeric Input Object. Let's write a value '**22000**' into our object.

This value has to be higher than the Minimum and lower than the Maximum values.

11. Type '**22000**' then press **Enter**.



The object background will turn **red** indicating that the value enter is outside the valid minimum and maximum bounds.

How will the operator know **what** the valid bounds are? Add a tooltip!

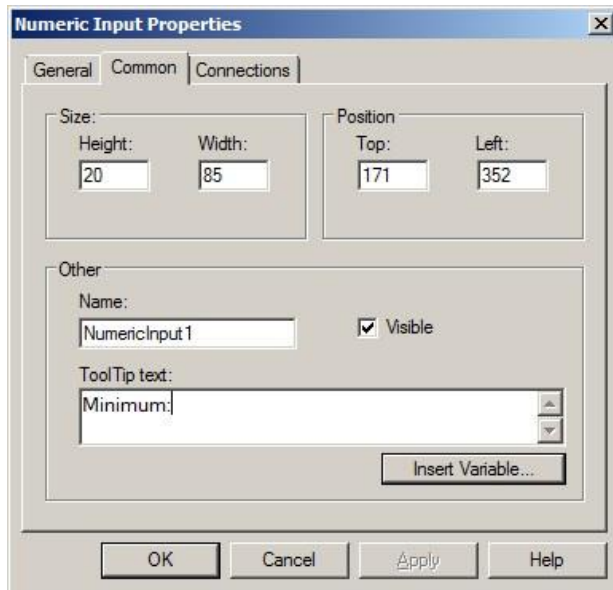
Let's add a tooltip to the Numeric Input object.

12. Click the **Edit Display** button  to get back to edit mode.

13. Double-click on the Numeric Input object to open up its properties. Or right-click and select **'Properties'**.



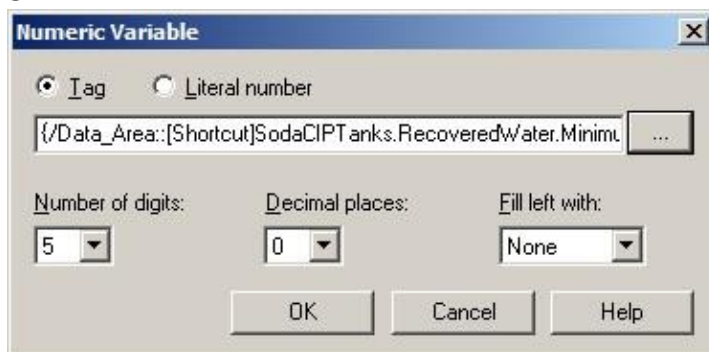
14. Select the **Common** tab.



15. Type **'Minimum:'** in the 'ToolTip text' text box.

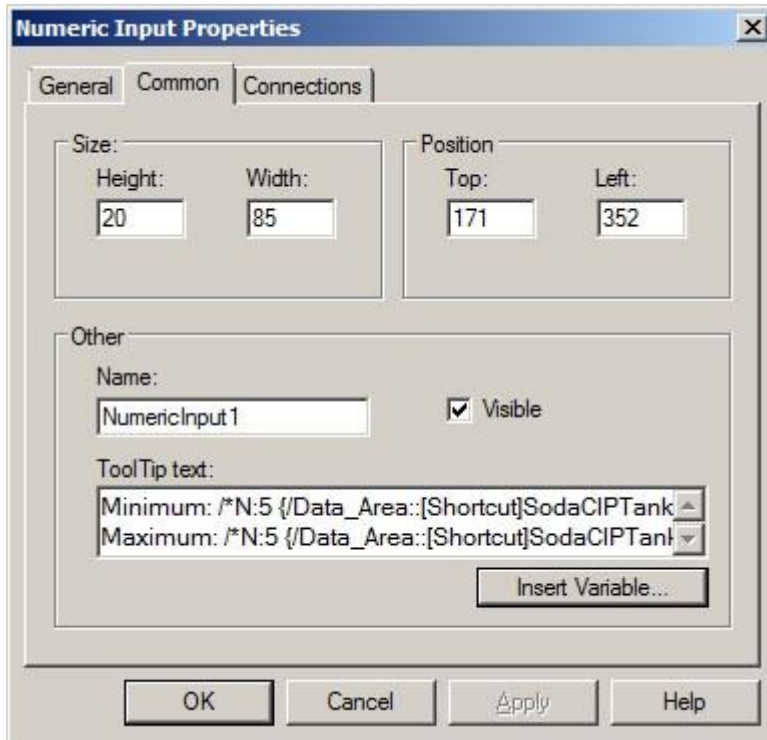
16. Click on **Insert Variable...** and select **Numeric...** to add a variable.

17. Browse to the **{/Data\_Area::[Shortcut]SodaCIPTanks.RecoveredWater.Minimum}** tag. Click **OK**.



18. Repeat steps 15-17 to add the **'Maximum:'** information as well. The Maximum tag to browse to is **{/Data\_Area::[Shortcut]SodaCIPTanks.RecoveredWater.Maximum}**.





19. Click **OK**

20. Save the display.

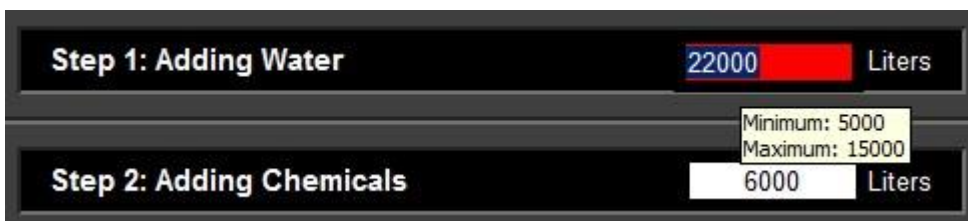
21. Test the display by selecting **Test Display** button  on the Graphics toolbar.

22. Type '**22000**' then press **Enter**.



Again, the object background will turn **red** indicating that the value enter is outside the valid minimum and maximum bounds.

23. Hover the mouse over the object and you should see our tooltip text indicating the proper value bounds.

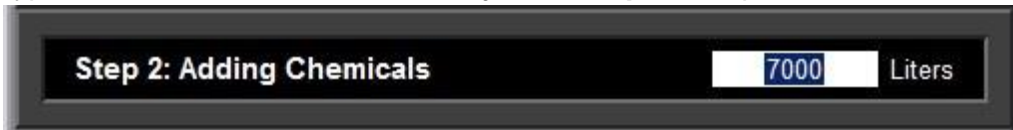


Let's now observe how a change in value in a Numeric Input object appears in FactoryTalk Diagnostics.


1. In the **cip\_processteps** display notice that the value in Step 2 is currently set to 6000.



2. Type '**7000**' into the Numeric Input object on **Step 2** then press **Enter**.



3. Go to the bottom of the screen and notice the diagnostics message that is created showing the new value and the previous value.


 Write '7000' to '/Data\_Area::[Shortcut]SodaCIPTanks.Chemical.Value'. Previous value was '6000'.



New in FactoryTalk View SE v8.0, when you change a Numeric Input or String Input value, the **previous value** and the new value being written are logged to FactoryTalk Diagnostics. In previous versions of SE, only the new value being written was logged.

The benefits of this feature are:

- Provides improved tracking of operator actions.
- Improves support for 21 CFR Part 11 compliance. -
- Streamlines troubleshooting process

4. Click the **Edit Display** button  to get back to edit mode.
5. Close the display '**cip\_processsteps**'.

## Docked displays

### Docked Displays

At run time, graphic displays can be docked to an edge of the FactoryTalk View SE Client window, allowing an operator to gain access to certain displays at all times. Docked displays cannot be accidentally closed by the operator and cannot have other graphics placed on top. They will, therefore, always remain visible to the operator.

For example, you might consider docking:

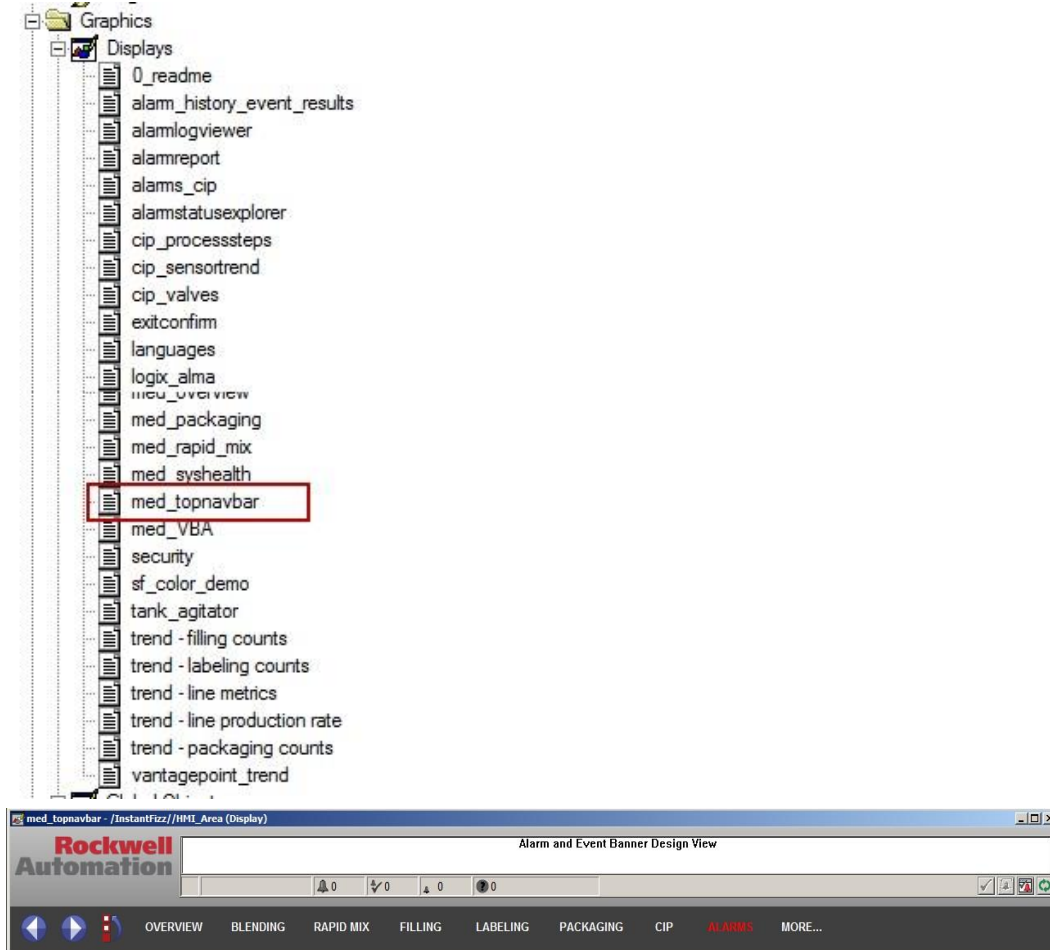
**Navigational menus** - that allow the operator to move among displays in an application.

**Headers or banners** - that provide specific information to the operator, such as the current user's name and area, or information about alarms.

**Control panels** - that contain standard buttons for special purposes, such as changing users, closing open windows, or sending information to a maintenance team.

In this lab we will use a docked display for navigational menus and as an alarm banner. The docked display will be visible to the operator at all times.

1. Open the display 'med\_topnavbar'. This will be our docked display.



2. To dock a display to an edge of the SE Client window, after the client starts up, we will use a start-up macro that utilizes a display command.

A **macro** is a list of commands or command symbols stored in a text file. To run a macro you use its name just as you would a command. The commands in the macro will be executed in the order in which they are listed.

A macro can be specified on startup or shutdown of a client or display. It can be called from a command line in FactoryTalk View Studio, from a button or from the Factory Talk View Administration Console for system administration.

FactoryTalk View has multi-tasking capabilities that you can take advantage of when you create macros.

Some commands (such as Print) finish quickly and the next command can start. Others, such as Set, take longer. In the case of Set, it does not finish until the message has been sent to the controller. In cases like that, you can set up the macro so that the next command can be executed before the previous command is finished. Use the ampersand character (&) to do this.

- Expand the **Macros** menu item and double-click on 'med\_startup\_client' to open.



- Review the first display command in the macro file:

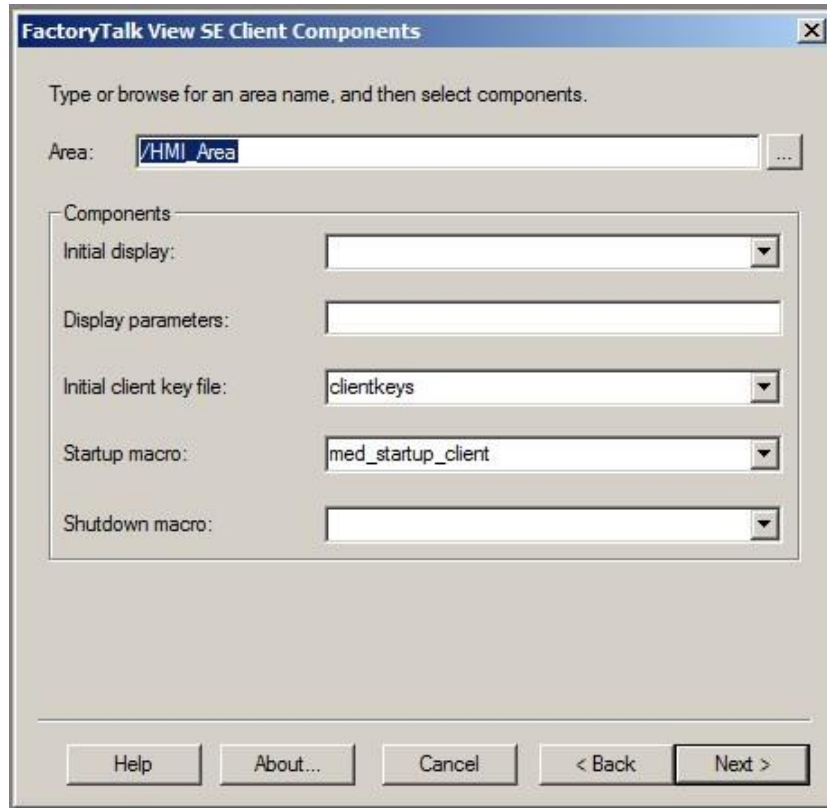
**Display Med\_TopNavBar /DT**



In general, to dock a display use the syntax shown below:

To dock a display in this position	Use this syntax
Top edge of the client window	Display <display name> /DT
Bottom edge of the client window	Display<display name> /DB
Left edge of the client window	Display <display name> /DL
Right edge of client window	Display <display name> /DR

These commands will be executed when the macro is run at the start-up of the FactoryTalk View SE Client. A quick look at how the macro is setup on startup is shown below. We will learn about client set up in further detail in later sections



5. Close the **med\_startup\_client** macro without saving.

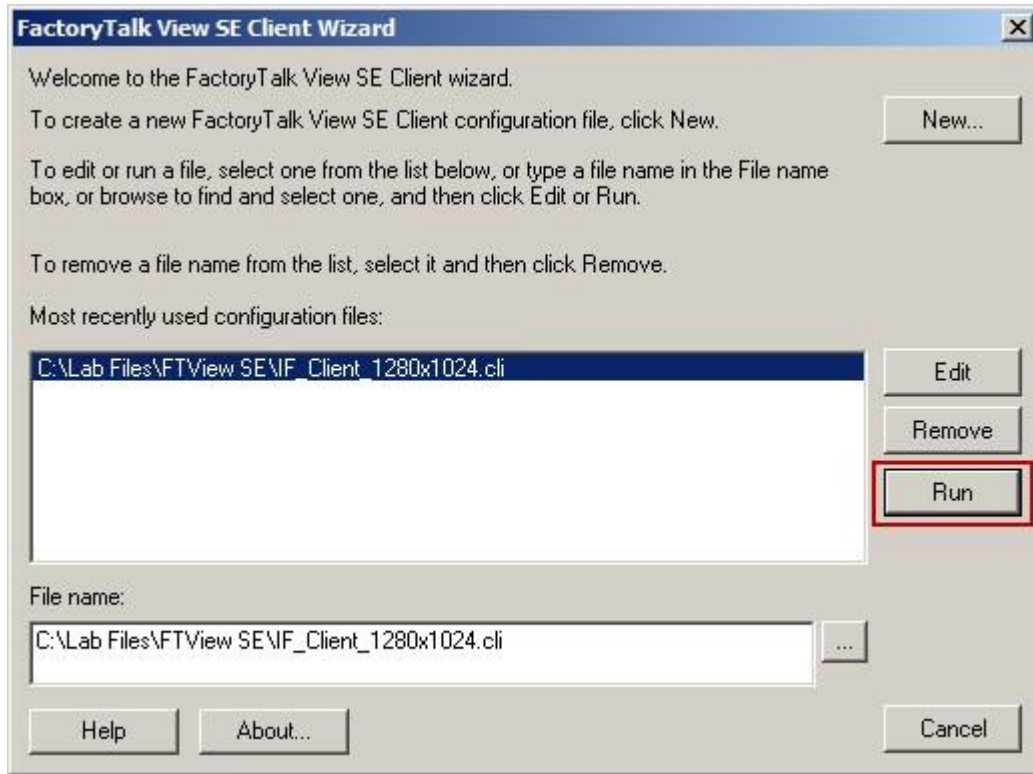
### Explore docked displays at run-time

Let's explore the docked displays functionality at run-time. An SE client file has already been created for you to run with this application. Our StartUp macro will be executed when the SE client file is launched.

1. Click the **Launch SE Client** icon



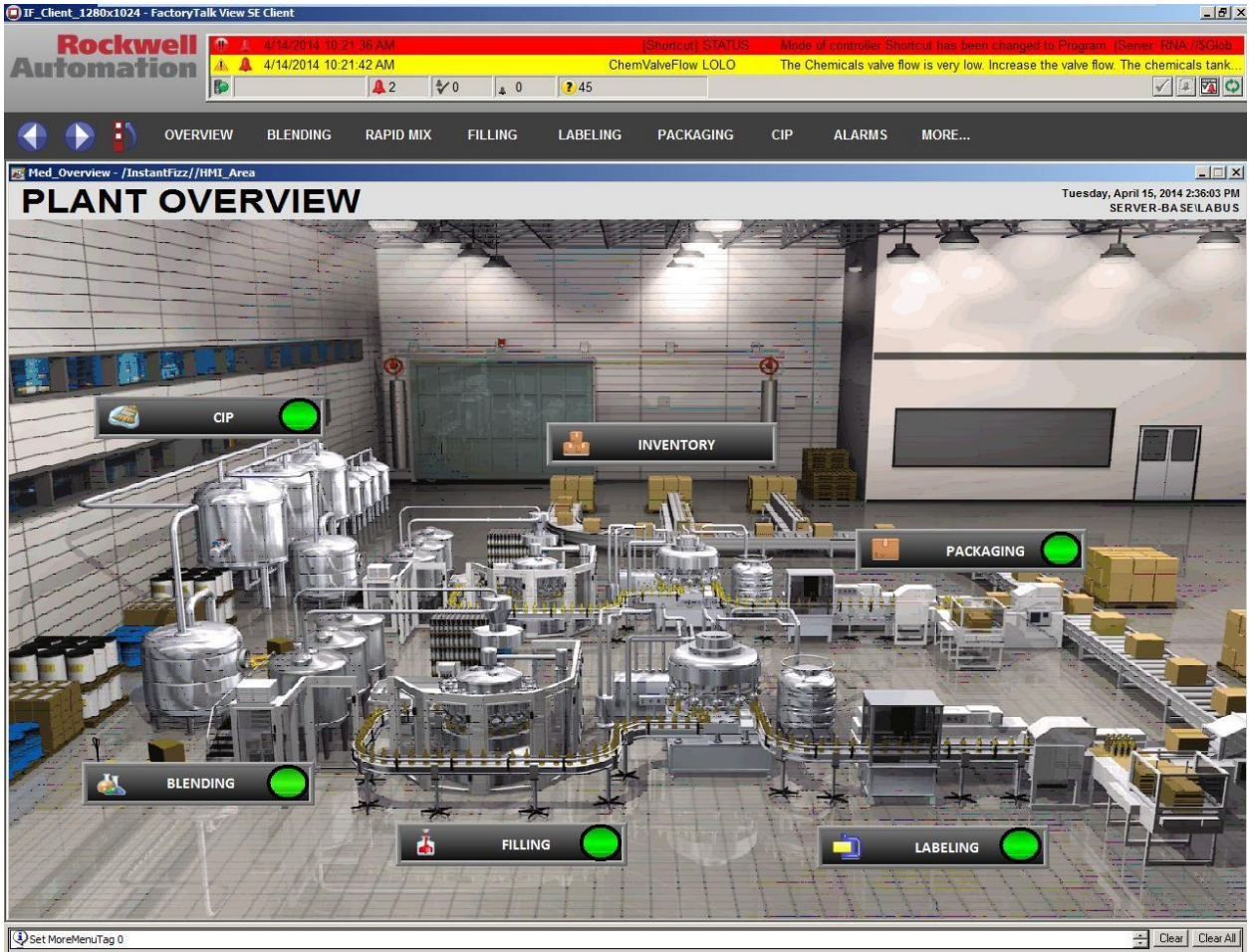
2. Browse to the **C:\Lab Files\FTView SE\IF\_Client\_1280x1024.cli** (if it is not in the list) then click **Run**.



**NOTE: It will take a few minutes for the client to fully load.**

3. Review the layout of the client. The startup macro docked a header display containing navigation buttons and an alarm banner at the top of the client window. Below that is the Plant Overview display.



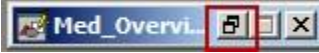


Minimize the **Plant Overview** display.

4.

5. Drag the minimized title bar around to the edges of the display client:

Top edge:  Bottom edge: 

6. Click the Restore button on the **Plant Overview** display. 

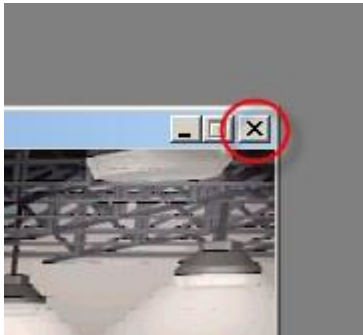
Observe that it will not overlay the docked areas.

7. Drag the **Plant Overview** display to the lower edge or lower right corner of the client.



Observe that it will not overlay the docked areas and scroll bars will appear, so the display will still be visible.

8. Move the **Plant Overview** display so the **Close** button is visible and close the display.



9. Select the **Overview** button on the toolbar display to open the **Overview** display. Notice how it is sized to fit in the main viewable area of the docked displays.
10. Leave the Client running because we will use it in the next section.

## Commands

FactoryTalk View commands allow you interact with and control application components. Most commands accept parameters for added precision and control.

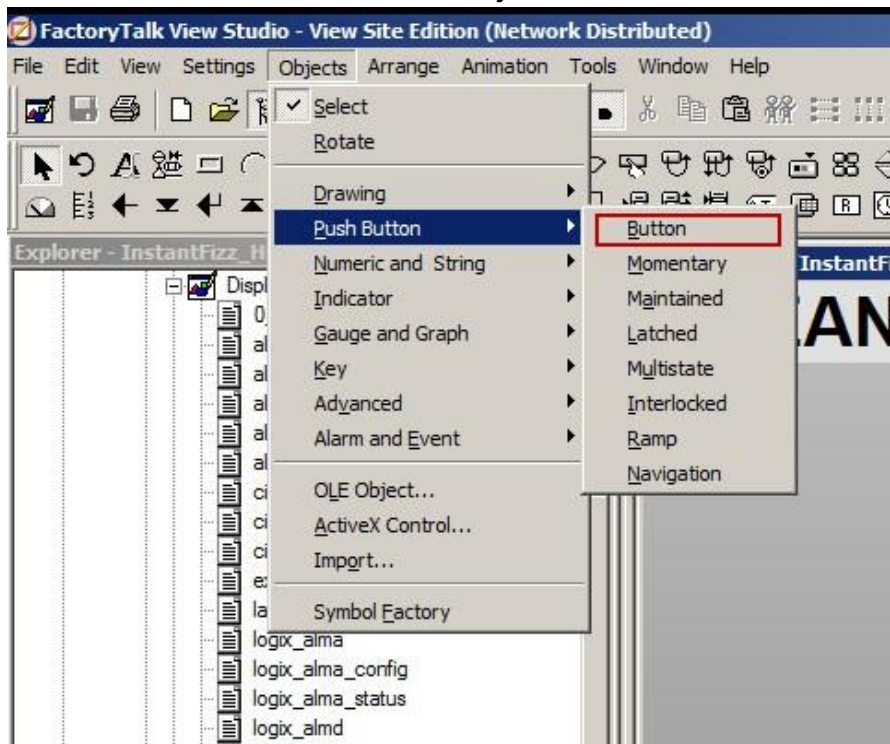
- You can set up keys and graphic objects to run commands at run time. For example, as the press, release, or repeat action when you assign touch animation to an object in a graphic display, or as the action for a button.
- You can run commands from the HMI server's command line.

- You can create a list of commands in a macro, and run the macro in places where the commands are required.
- You can run commands at a particular event using an Event file.

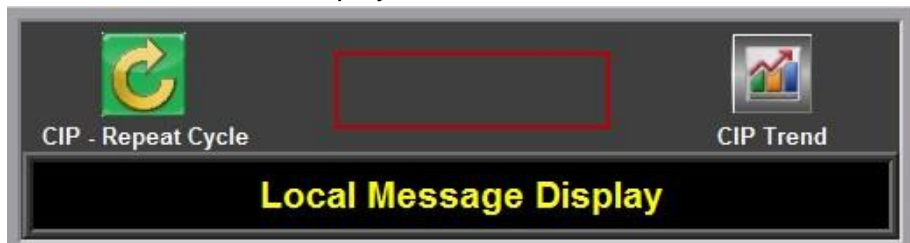
There are approximately 80 different commands. Use the Command Wizard for assistance with selecting and building commands.

We will create a Button object with a display command.

1. In FactoryTalk View Studio, close all the currently open displays without saving.
2. Open the **med\_cip** display.
3. Select the **Button** menu item from **Objects > Push button**.

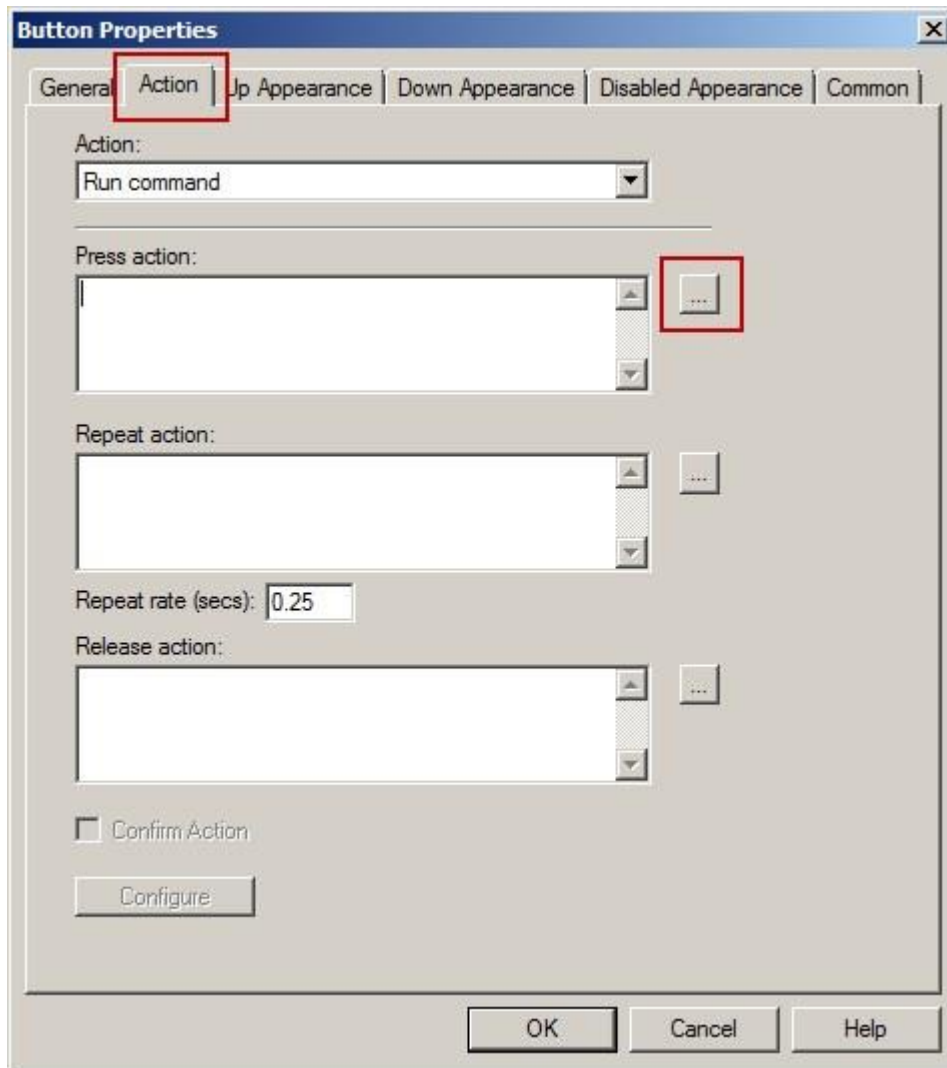


4. Draw the Button in the display as shown below where the red outlined box is shown:

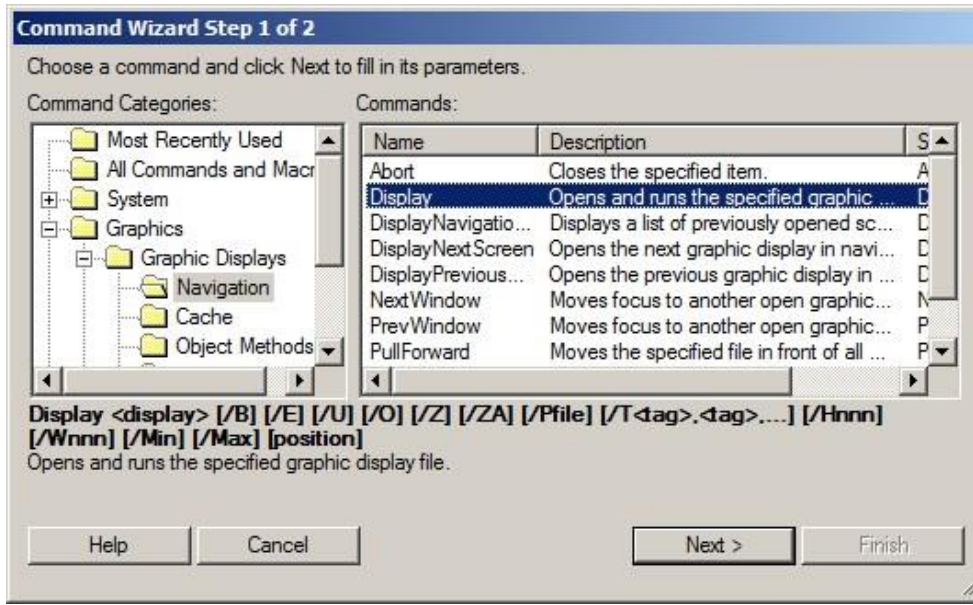


5. In the Button Properties window that pops up, select the **Action** tab





6. Click on the ellipses button next to **Press Action** to open the **Commands Wizard**.
7. Select the **Display** command. You can either select **All Commands and Macros** and find the Display command alphabetically, or select **Graphics > Graphic Displays > Navigation > Display**.

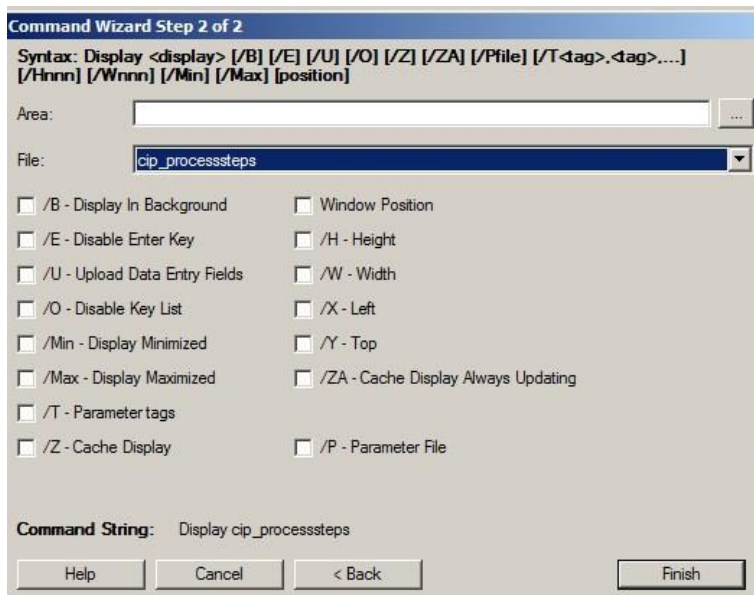


The Command Wizard window can be resized by dragging the bottom right corner. This adds ability to see the list and the description of all command categories and associated commands without the need to use the scroll bars.

8. Click **Next**

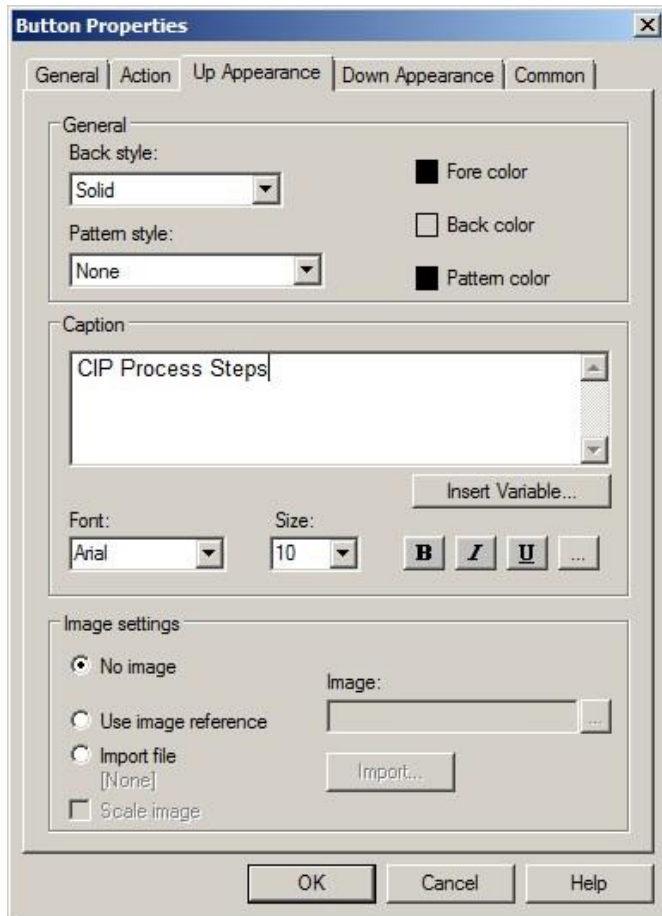
to continue.

- Fill the display command information by selecting **cip\_processsteps** from the File drop down list. If our application had more than one area we could choose to open a display from a different area.



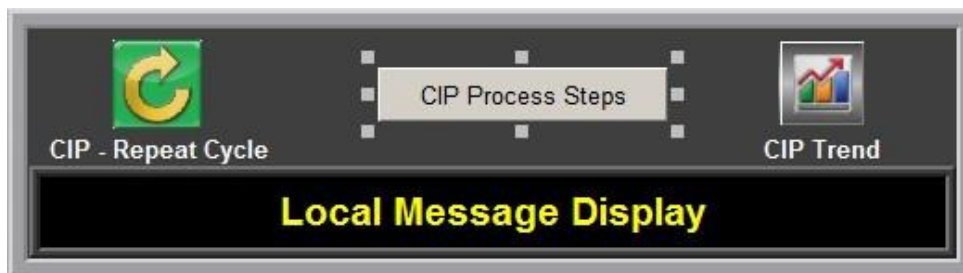
- Leave the optional selections blank and click **Finish** to complete.

- Add a **Button Caption** by selecting the **Up Appearance** tab and typing '**CIP Process Steps**' in the **Caption** field




12. Click **OK** to complete

The new button added should look as shown here. Resize the button if needed.



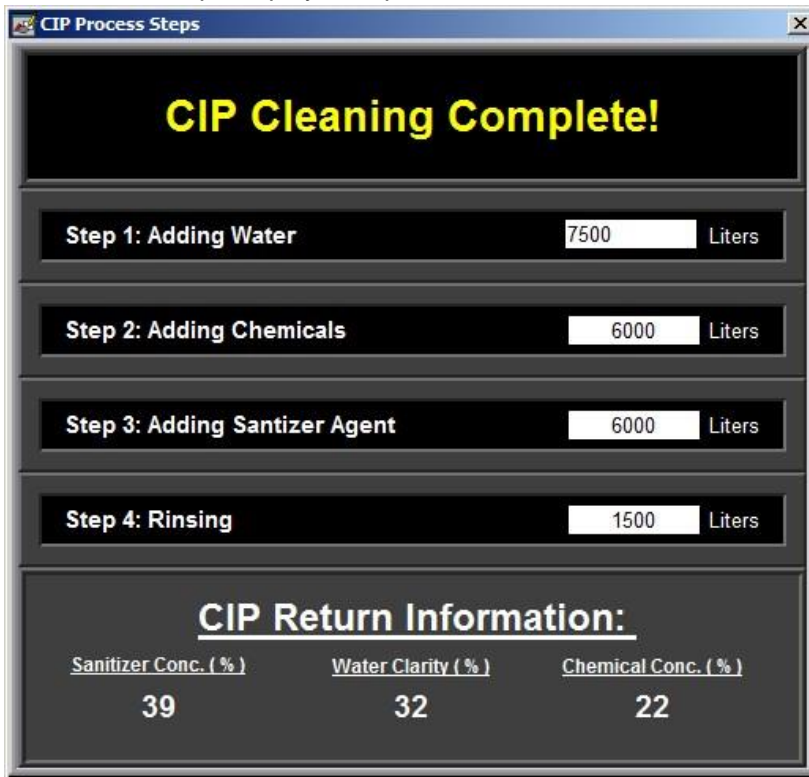
13. Save the display and then close it.

14. Return to the running **client** window and select the **CIP** button  in the navigation display.

15. Open the **CIP Process Steps** display. In the upper right corner of the CIP display, click on the newly added CIP Process Steps button.



CIP Process Steps display will open!



16. Close the CIP Process Steps display and minimize the Client window

Using the Command Wizard we have created a button which when pressed opens another display.

The Commands Wizard contains many more commands related to closing/opening displays, alarming, printing, languages, external applications and many others.

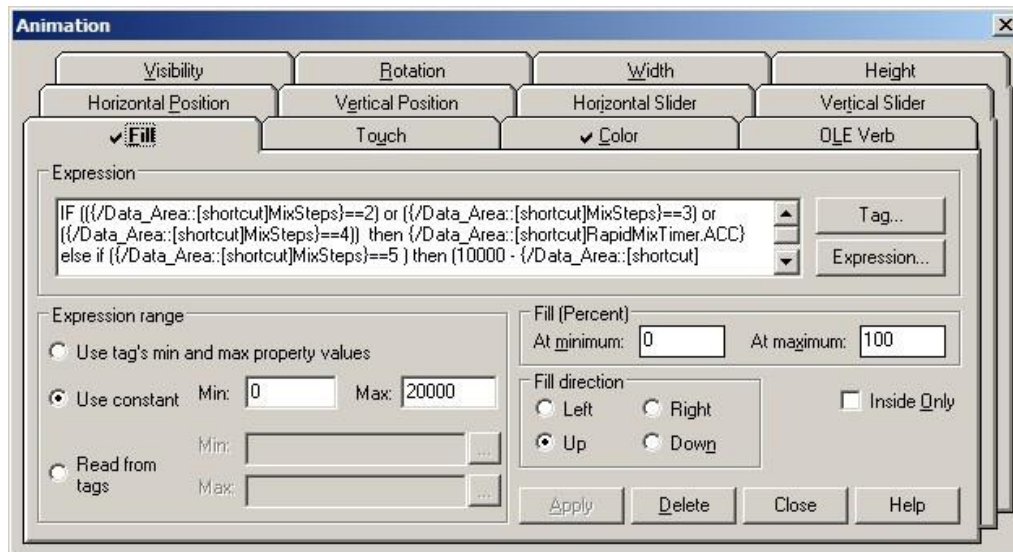
### Animations and expressions

Animation is the ability to add logic to a graphic object so that some characteristic of the object will change when a tag value changes. For example, an object can be made to fill (up, down, left, or right) or change color in relation to a tag value.

**Expression:** An expression is a mathematical or logical equation that returns a value. It can contain tag names, constants and mathematical, relational, logical and/or bitwise operators. A single tag name is often used for simple expressions.

In the picture below, the animation dialog shows that expressions are used to animate objects. There is a tab for each type of animation. If there is a check mark in front of the animation type it means that the

selected object is using that animation. If an animation type is not available for a selected object, the fields on that animation tab will be grayed out. In the example below, the Fill and Color animations are being used on the selected object. Selecting a new object while the Animation dialog is opened will update the Animation dialog for the object that was just selected.



Let us also learn briefly what the Object Explorer is and how we can use it to look at Animations set on an object

## The Object Explorer

The Object Explorer provides a list of all the objects in the current graphic display, including those that are hidden by other objects. A group of objects has a plus sign in front of its name. Click this to expand the list of objects that make up the group.

You can expand or collapse the whole list using the Expand and Collapse buttons.

When you click an object in the display to select it, its corresponding entry in the Object Explorer is highlighted in gray.

When you click an item in the Object Explorer, the object it corresponds to is selected. If an object is hidden by another, or is part of a group, when you select it in the Object Explorer the handles outlining the selected object are visible.

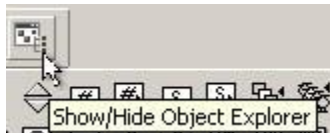
Let's open up one of the displays and take a look at animation.

### To open the Object Explorer

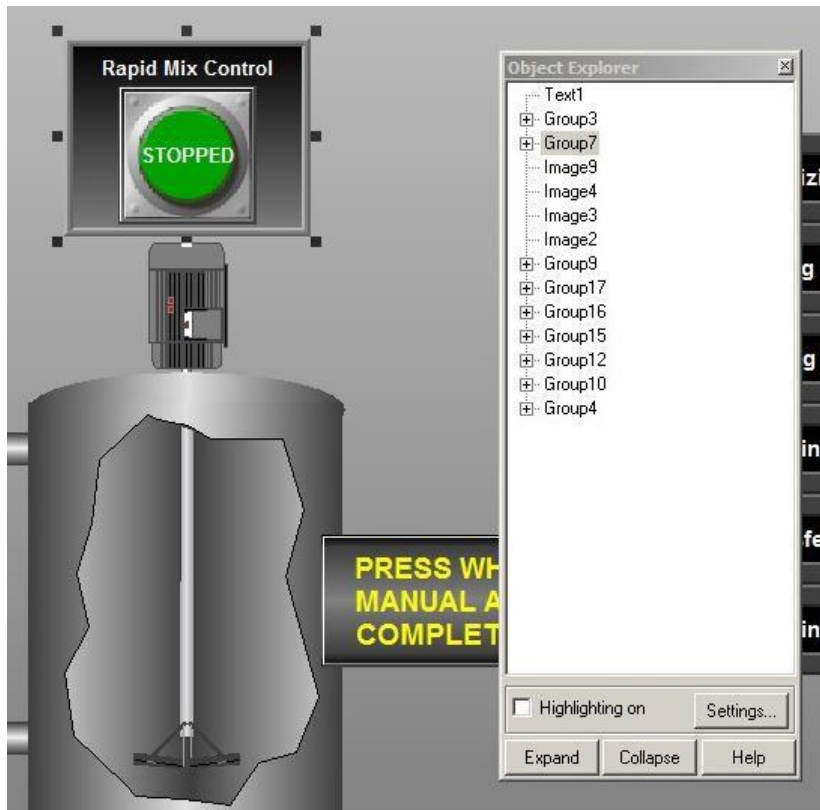
1. Return to FT View Studio and open the **med\_rapid\_mix** display. To open the **Rapid Mix** display, expand the **Displays** folder and double-click on the **med\_rapid\_mix** display. The **Rapid Mix - Overview** display will be opened.



2. Select the **View > Object Explorer** menu item  
-or-  
select the **Show/Hide Object Explorer** button from the menu bar to show or hide it.



Once selected (indicated by a check next to the menu item name), the **Object Explorer** appears. The Object Explorer can be resized and moved. You can click on any of the objects listed, and you will notice that the objects will be highlighted in the display.



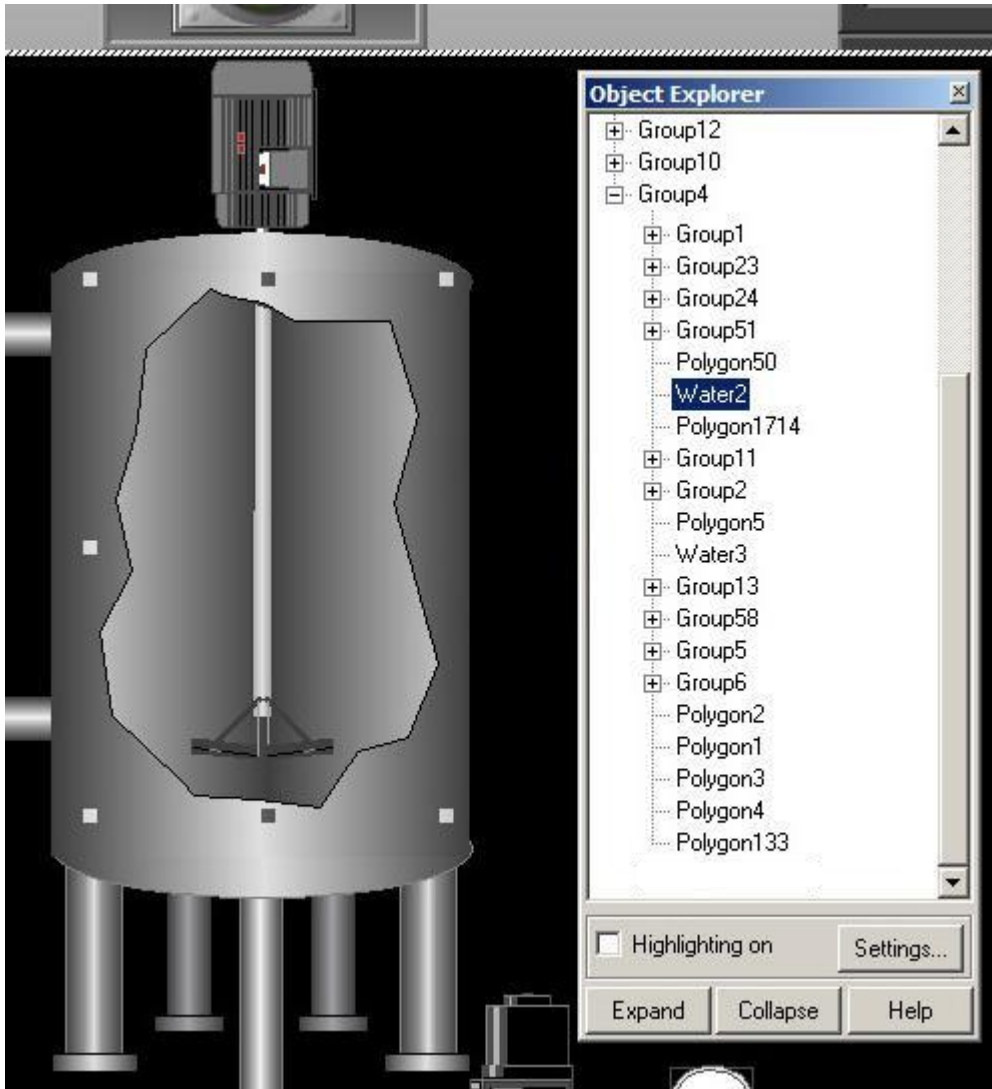
The Object Explorer is truly useful when you group items together and want to reference individual elements within that group.

**Grouping** is useful when you have common objects that you want to move around or apply behaviors toward, for example, animation behavior.

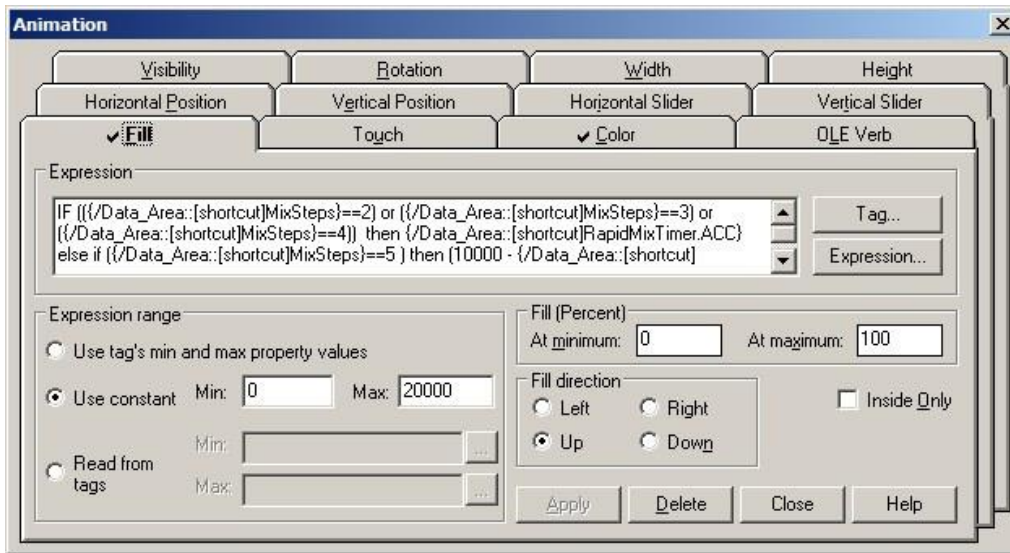
### Look at groups and animation by using the object explorer

1. Select the **Group4** group in the Object Explorer.
2. Click on the **+** next to the **Group4** to expand the object group
3. Right-click on the **Water2** element within the **Group4** group and select the **Animation > Fill** context menu item.






4. The **Animation** dialog will appear and open on the **Fill** tab.



The Fill animation expression has already been pre-entered for you. You can review and familiarize yourself with the Fill and other animation properties, such as Color.

5. Close the Animation window.

6. Return to the running **client** window and select the **Rapid Mix** button  in the navigation display.

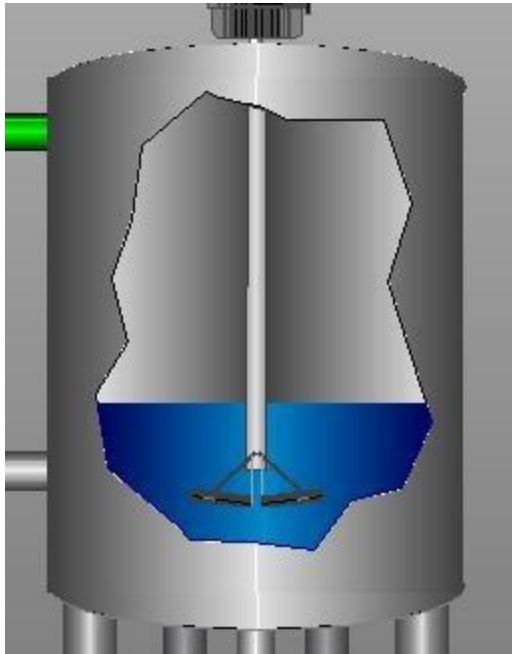
7. If the button at the top of the display says **STOPPED** then press it to start another cycle and watch the animation.



If the button says **RUNNING** then the process is still working on the last processing cycle and is waiting for you

to press the **PRESS WHEN MANUAL ADD COMPLETED** button  to continue.


8. Watch the tank fill!

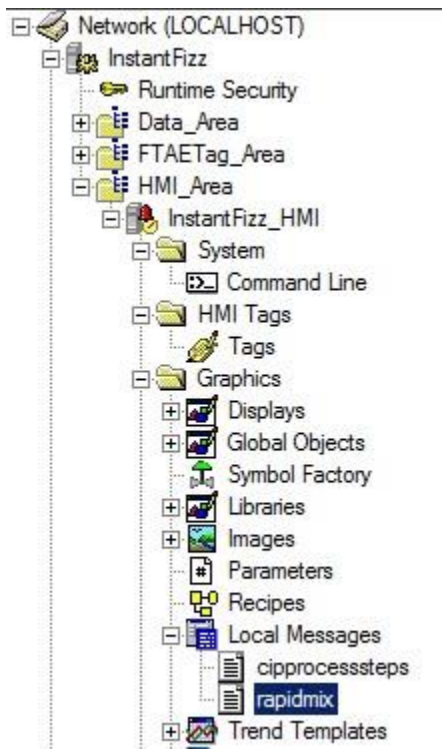


The water filling up in the tank happened as a result of the animation that you observed from the object explorer.

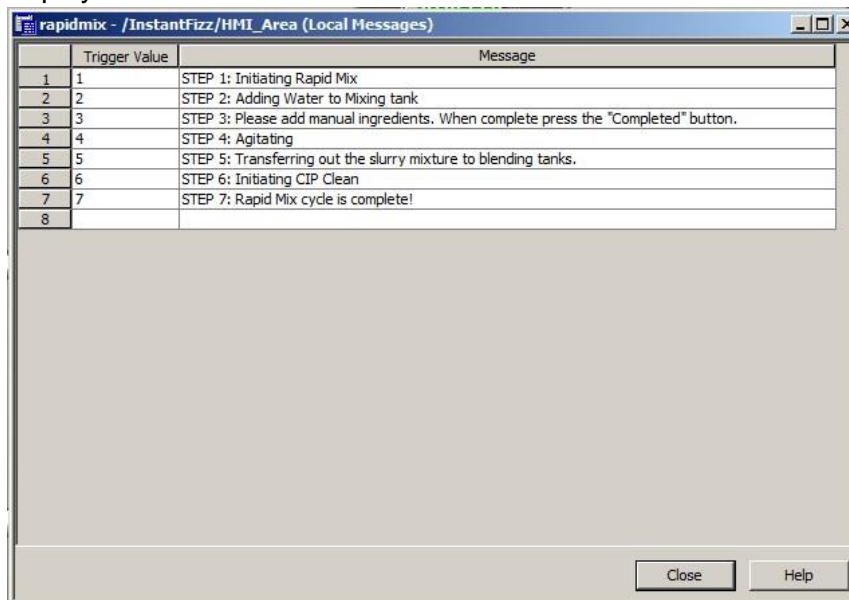
### Local Messages

Use local message displays to provide an operator with information about a process, or about what to do next, at run time. For example, the `med_rapid_mix` display contains a Local Message object that gives the operators instructions on what the current step is in the mixing tank.

1. In the running client window, select the **Overview** button  in the navigation display in order to close the **Rapid Mix display**.
2. Return to FactoryTalk View Studio. Let's review the **rapidmix** Local Message file that has been pre-created for you. Open this file:

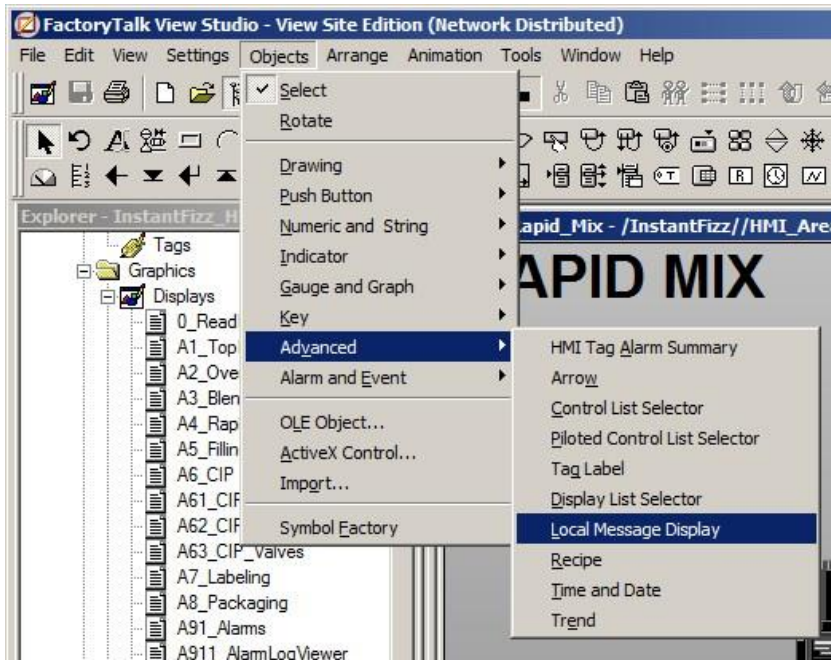


- Local Message editors consist of **Trigger Value** and **Message** columns. When the value of the Local Message tag equals the Trigger Value, the corresponding Message will be displayed. For example, if the tag value is equal to 6, the message '**STEP 6: Initiating CIP Clean**' will be displayed.



- Click **Close**.
- In order to view the messages on a display at run-time, a Local Message object needs to be added. Open the **med\_rapid\_mix** display.

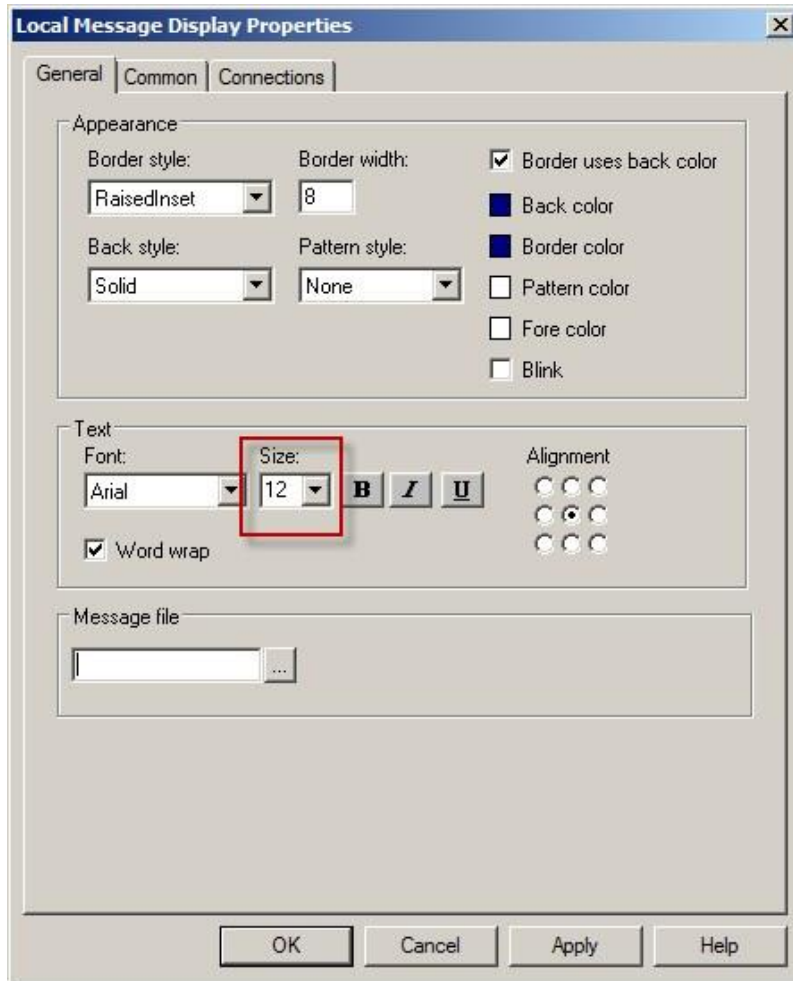
6. Add the Local Message Display to **med\_rapid\_mix** by selecting the **Local Message Display** object from **Object > Advanced** menu.



7. Draw the **Local Message Display** object as shown.

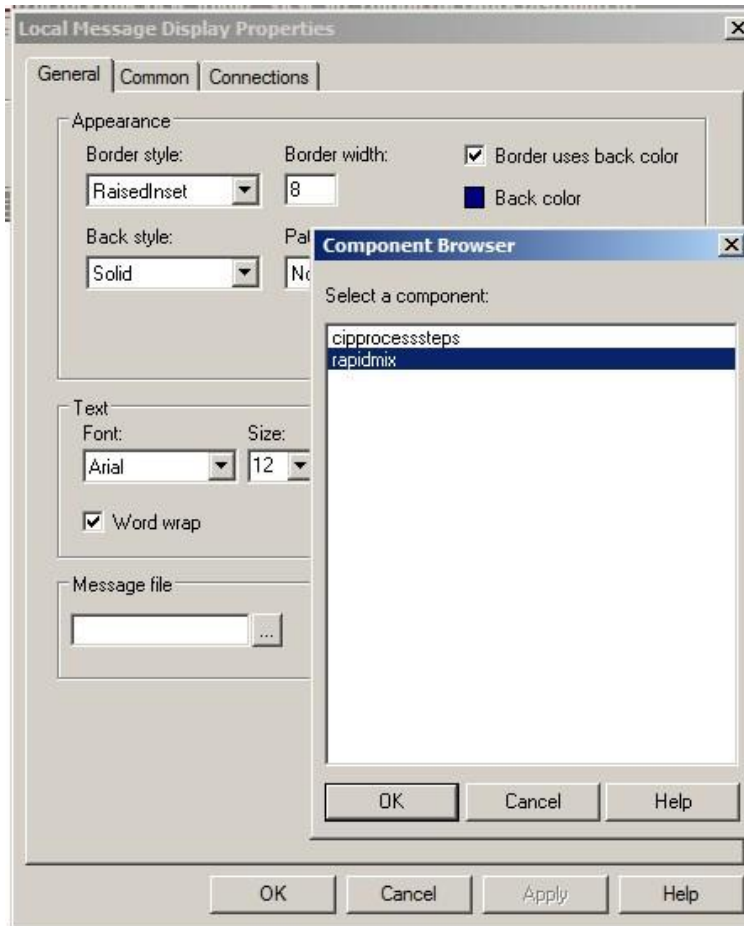


8. We will modify some properties of the message text, such as font and color. On the **Properties** dialog that opens up, change the Font Size to **12**.

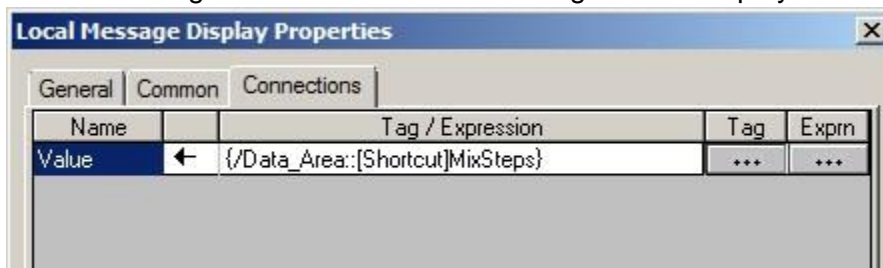


We would like this Local Message Display to display messages from the **RapidMix** Local Message File. The messages will be triggered by the **/Data\_Area::[Shortcut]MixSteps** tag

9. Click on the ellipses button below **Message File** to browse the Local Message File. Select **rapidmix** then click **OK** to close.



10. Select the **Connections** tab and browse to the **/Data\_Area:[Shortcut]MixSteps** tag. In the tag browser expand Shortcut, select Online then on the right side select the MixSteps tag. The value of this tag will determine which message will be displayed.



11. Click **OK** to close.



12. Save and close the display.

Let us view the Local Messages in action

13. Return to the running **client** window and select the **Rapid Mix** button



## RAPID MIX

in the navigation display.

14. If the button at the top of the display says **STOPPED** then press it to start another cycle and watch the Local Message Object display different messages depending on what process step is being executed. If the button says **RUNNING** then the process is still working on the last processing cycle and is waiting for you to press the

**PRESS WHEN MANUAL ADD COMPLETED** button



to continue.




15. Follow the Local Message messages as they change.

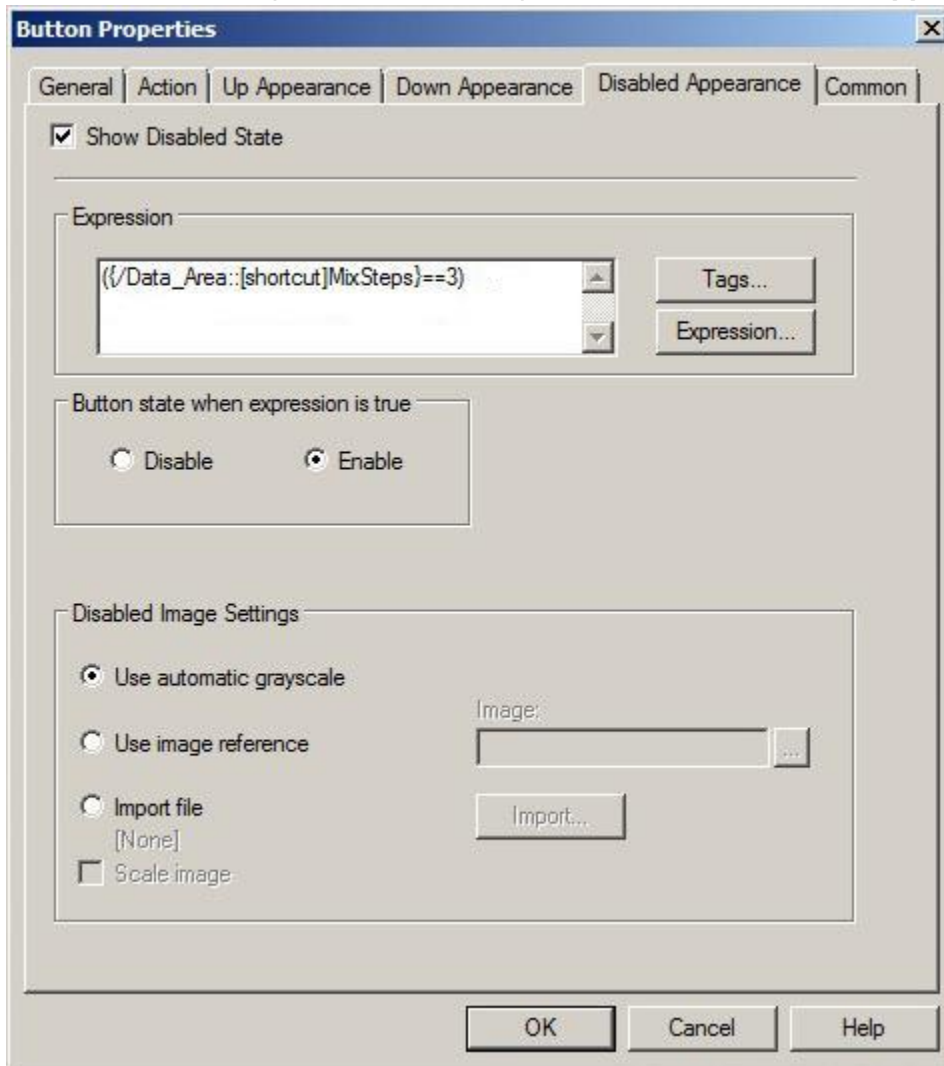


16. Minimize the running client window and return to FactoryTalk View Studio.

Let's now explore the configuration behind the Manual Add button that was used above in the Rapid Mix process.

1. Return to FT View Studio and double click on the Manual Add  button within the **Rapid Mix** display.

2. Once the Button Properties window is open, click on the **Disabled Appearance** tab.



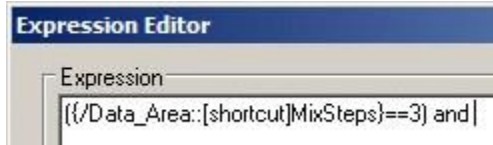
The expression defined indicates that the button will only become enabled once Step 3 of the process has been reached as indicated by the MixSteps tag.



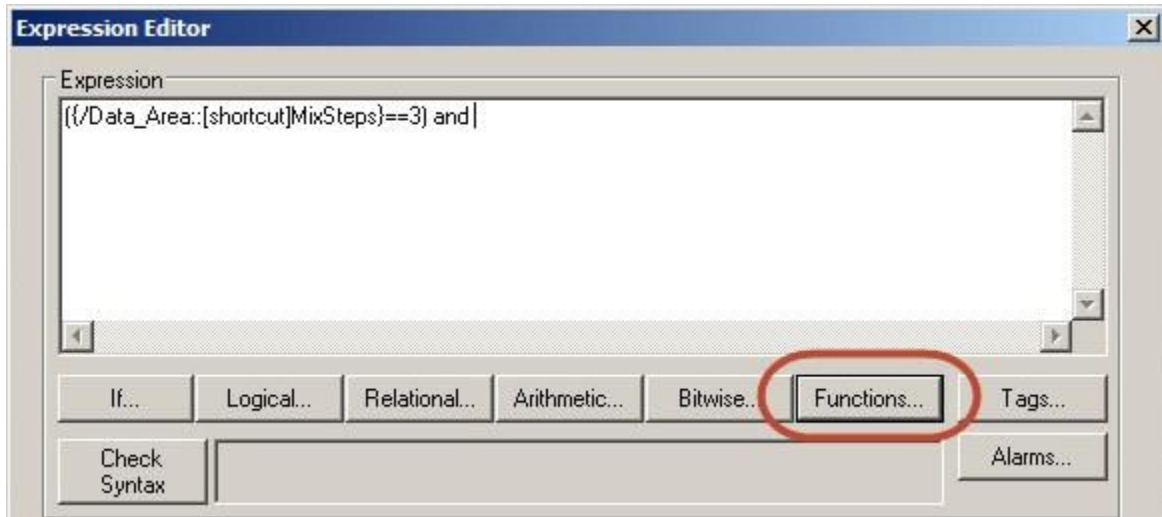
Button objects now have the ability to use a tag name or an expression to determine whether the state of the button should be enabled or disabled. The appearance of the disabled state is configurable with an automatic grayscale option or the use of image references.

We also want to verify that a user is a part of a specific security user group before they can gain access to this button. Let's add to the button state expression to accomplish this.

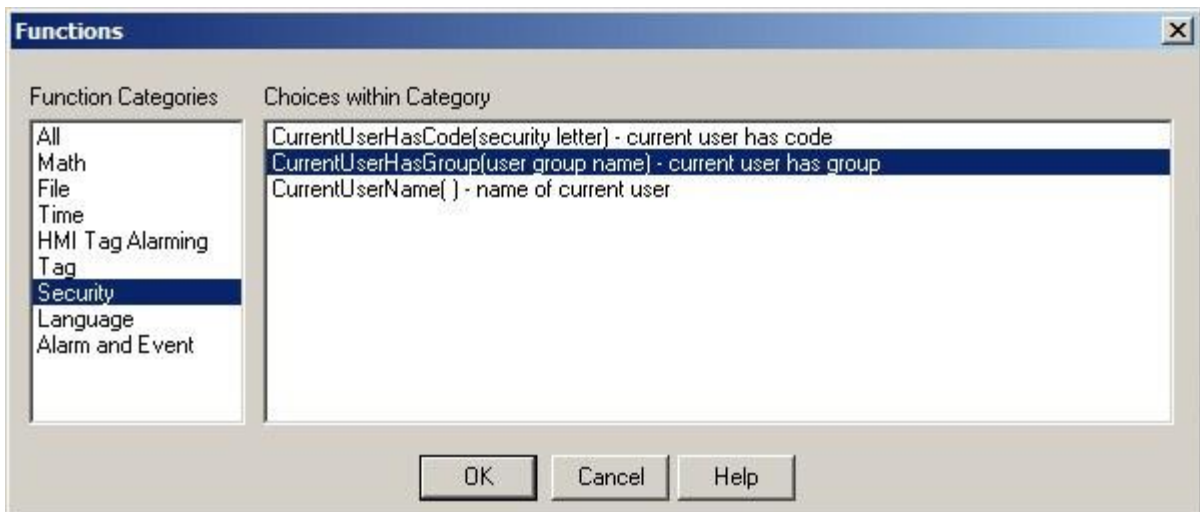
3. Click on the **Expression** button to edit the button state expression.
4. Click at the end of the expression to add an "and" so that your expression now looks like the following:



5. Click the **Functions...** button to bring up the list of function categories.



6. Select **Security** then **CurrentUserHasGroup(user group name)** to add this function to the expression and click **OK**.



The `CurrentUserHasGroup()` security function gives the ability to check whether the current user has been assigned to a particular FactoryTalk user group.

FactoryTalk View SE v8.0 also introduced the `CurrentUserGroups()` security function that return the

FactoryTalk user group list of which the current user is a member. This function is available in VBA only.

- The `CurrentUserHasGroup( )` function has now been added to the expression.



- An "Operators" group has already been created within the FactoryTalk Directory. This is the group we want a user to be a part of before they will have access to the Manual Add button. To specify this group within the function, type 'Operators' within double quotes. The expression should now look like the following:



- Click **OK** twice to save the changes to the Expression Editor and the Button Properties.

- Save** the changes to the Rapid Mix display.

- Return to the running **client** window and navigate away from the Rapid Mix display.

- Once again select the **Rapid Mix** button  in the navigation bar so that we can load the display with our changes.

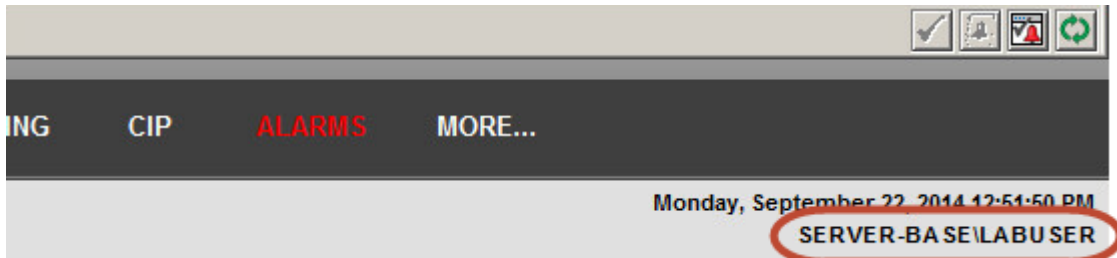
- If the button at the top of the display says **STOPPED** then press it to start another cycle and watch the Local Message Object display different messages depending on what process step is being executed.



- Follow the Local Message messages as they change. Once Step 3 has been reached we had previously seen that the button would become enabled. However, we have since added the criteria of the Operators user group to which Labuser is not a part of and the button remains disabled.



The currently logged in user is indicated in the upper right hand corner of the display.



15. We will now log in Operator1 who is a member of the Operators group. To log this user in click on the **More...** menu and select **Security**.



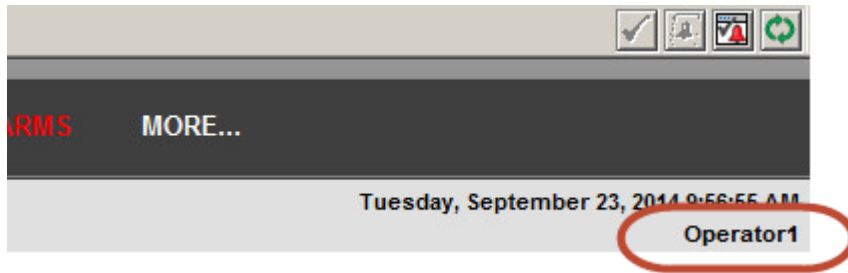
16. Select the **Log In** button and enter the credentials 'Operator1' for both user name and password.

Note that the password is case sensitive.



17. Click **OK** and **close** the Security display.

18. We have now logged in a user that is part of the Operators user group.



Notice that the button has changed from disabled to enabled because the button state expression is now evaluating to true.

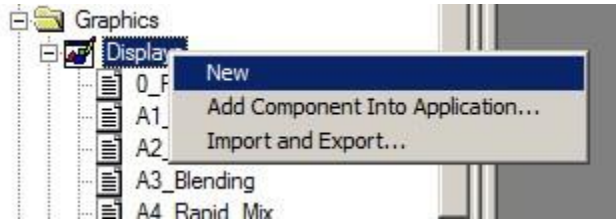


With the `CurrentUserHasGroup()` security function now a part of the expression, click the enabled button and watch the Local Messages as the process completes

## Using Symbol Factory Graphics

In this section you will create a new display and learn the ways to launch the **Symbol Factory** library as well as how to select graphics and place them on the new display. This will allow you to get a look at the types of graphics that can be found in the **Symbol Factory** library, as well as how the library fits into the FactoryTalk View Studio interface.

1. Return to FactoryTalk View Studio and expand the **Graphics** folder in the Explorer pane. Right-click **Displays** and select **New** to create a new Display.

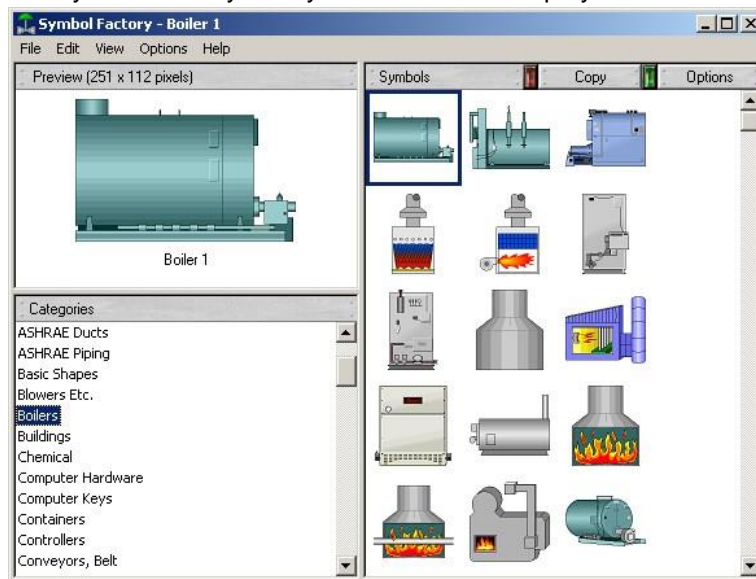


A new display is opened in the workspace area.

2. Double-click **Symbol Factory** in the Project Explorer.



The Symbol Factory library browser will be displayed.



3. Close library browser.  
We will now explore another way to open and view the Symbol Factory.
4. Click anywhere on the **Untitled** display that we just created to return focus to the display.
5. With library browser closed after the last step, select the **Symbol Factory** icon on the **Objects** Toolbar. This is an alternate way to launch the Symbol Factory.



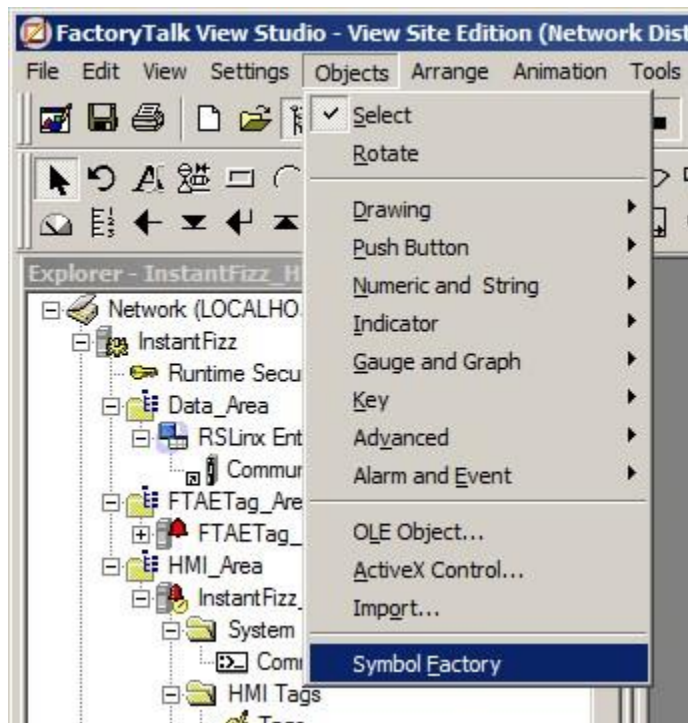
Symbols  
Thumbnail  
window



5. Close library browser.

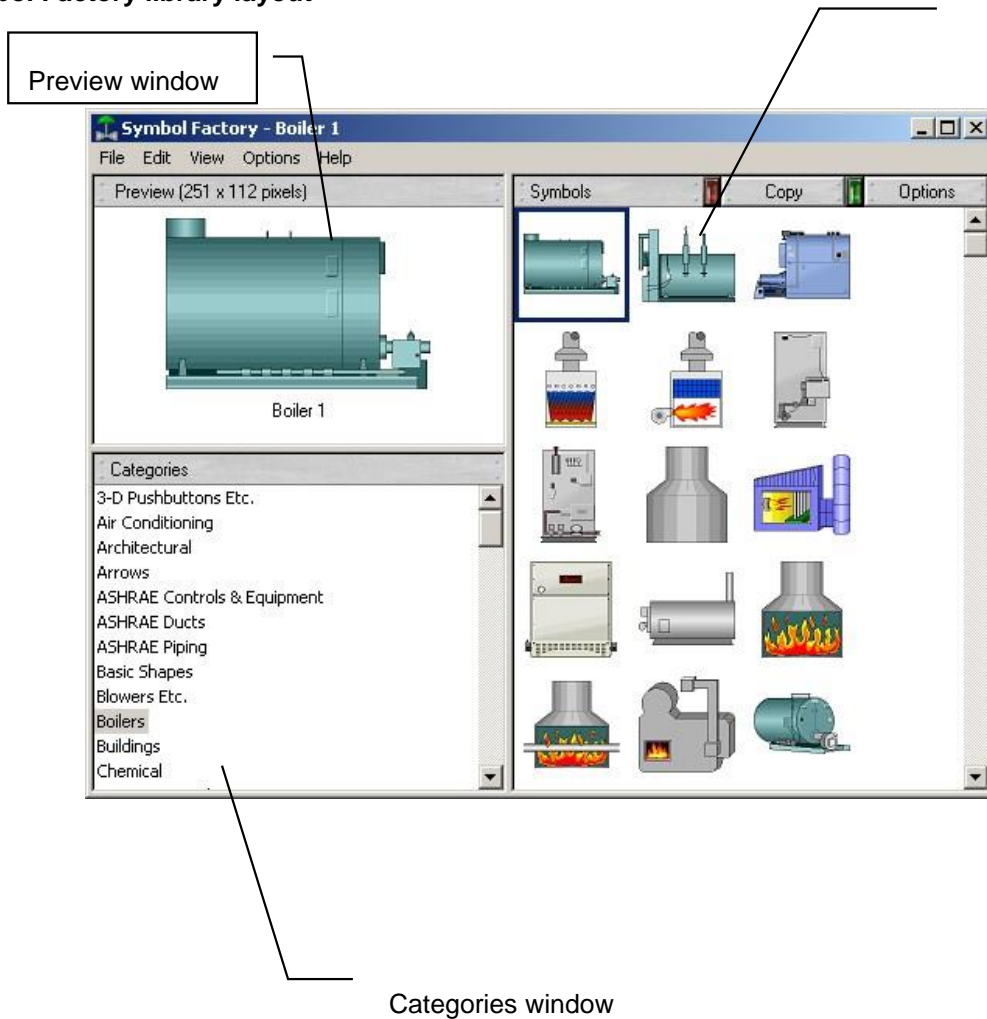
Here is another way to launch the Symbol Factory

6. With library browser closed after the last step, select **Symbol Factory** from the **Objects** pull down menu.



7. Do not close the library browser.

## Symbol Factory library layout

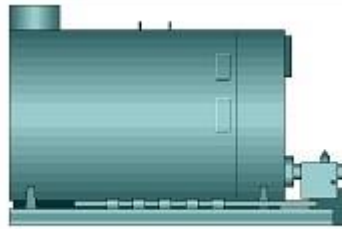


8. Select different categories in Category window. You will notice that the graphical objects in each category are displayed in the Symbols Thumbnail window.
9. When a graphical object is selected in the Symbols Thumbnail window, a larger thumbnail image is displayed in the **Preview** window.


There are a few different ways to include a Symbol Factory graphic object onto the display. Let us explore how an object can be added using the Copy button, drag and drop and by using Copy-Paste commands.

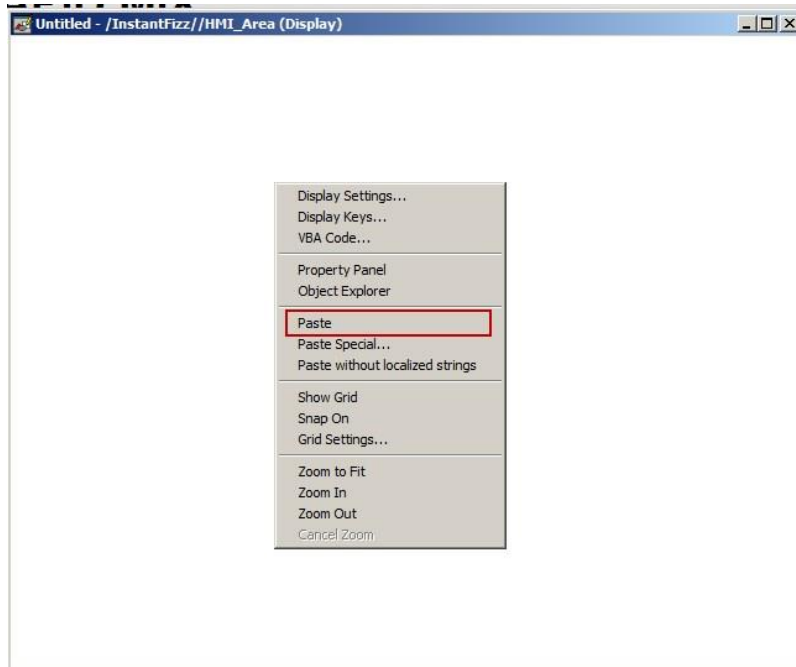
### ***Adding a Symbol Factory Graphic to a display using the Copy Button***

1. Select the **Boilers** category in the Symbol Factory library.

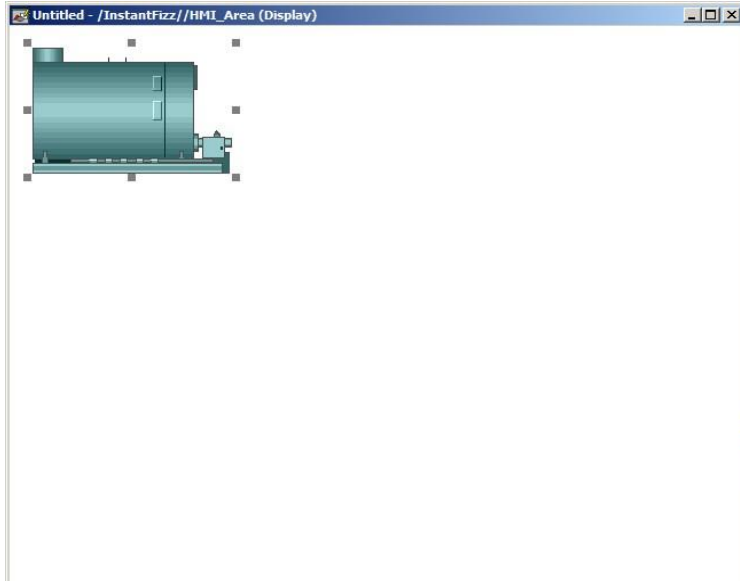


Boiler 1

2. Select the **Boiler 1** graphic
3. Select  from menu above Symbols Thumbnail Windows.  
When **Copy** is selected, the library window will minimize.
4. On the new Graphic Display, right-click to display menu and select **Paste**.



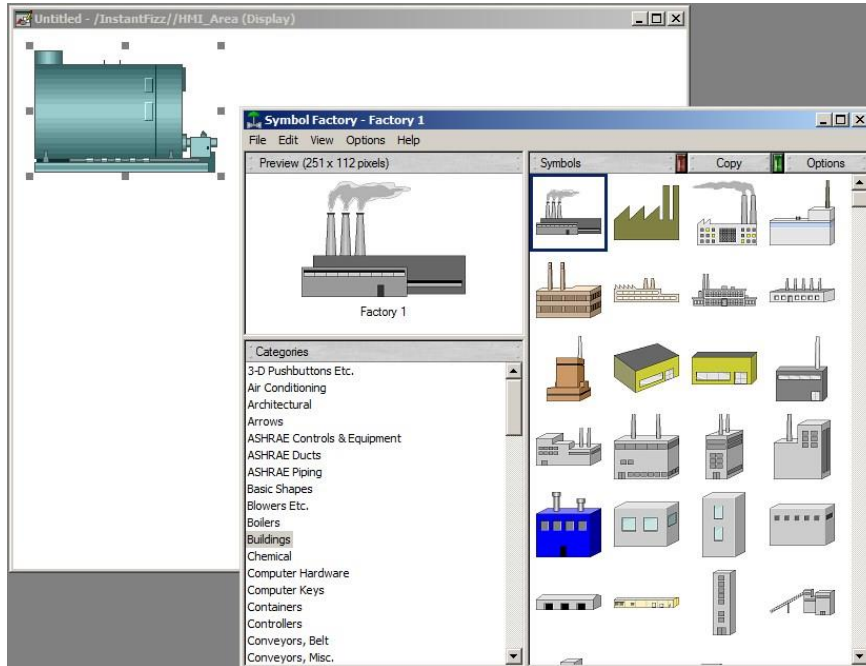
5. **Boiler 1** graphic will be copied to display as shown below.



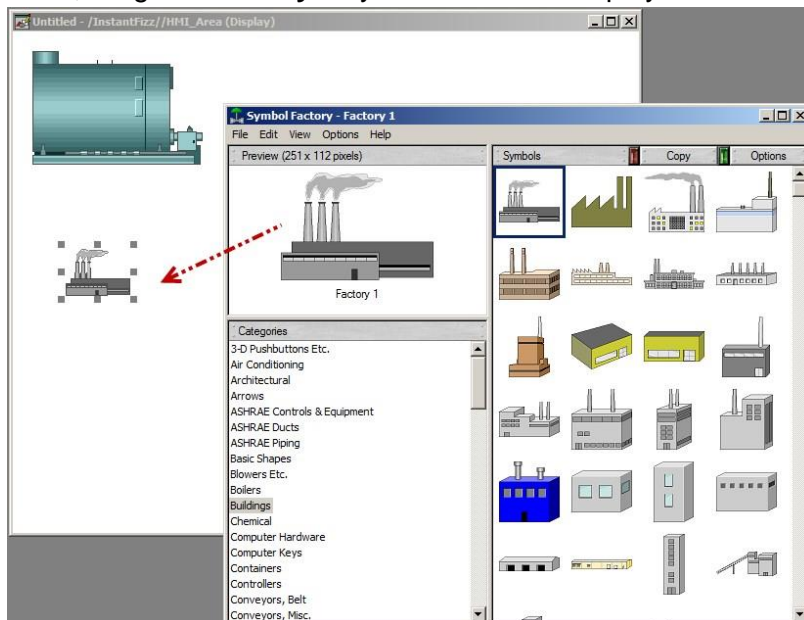
This is one way to copy and paste a Symbol Factory graphic to a display.

## Adding a Symbol Factory Graphic to a display using Drag and Drop

1. The library browser minimizes to the Windows Task Bar when a symbol is copied. You might need to select it from the Task Bar to display it again.
2. With Symbol Factory library displayed, select the **Buildings** category.
3. Select the **Factory 1** symbol from the library.



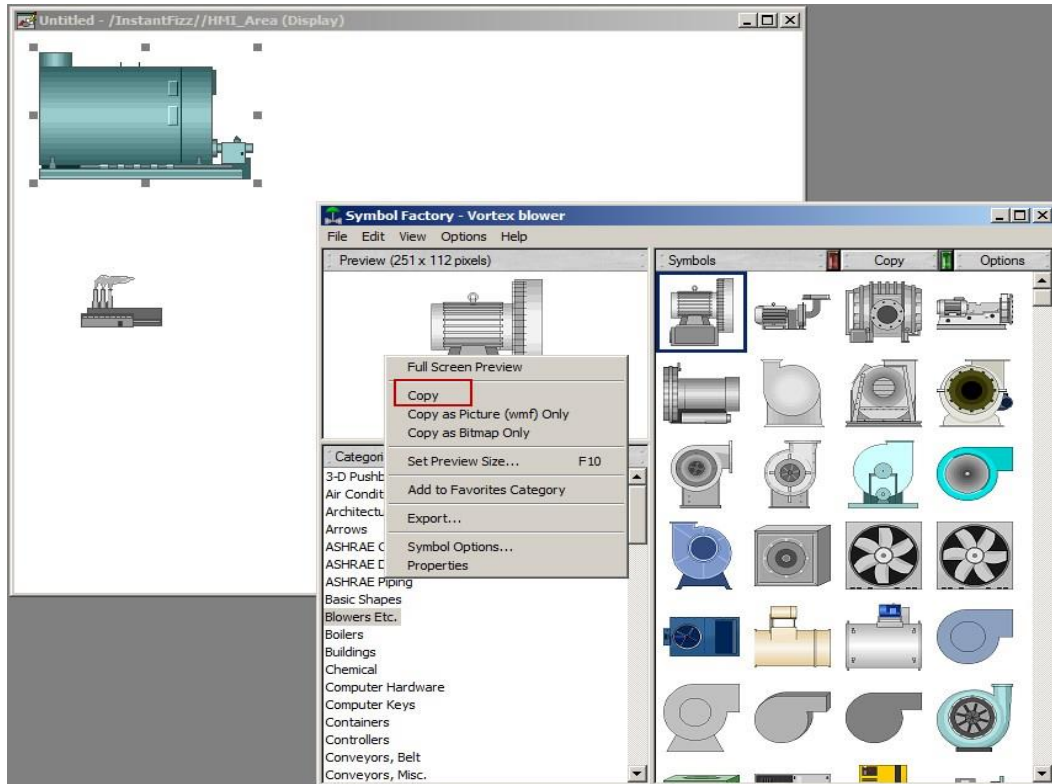
4. Select the **Factory 1** symbol in the **Preview** window and with the left mouse button held down, drag the **Factory 1** symbol on to the display screen under the boiler symbol.



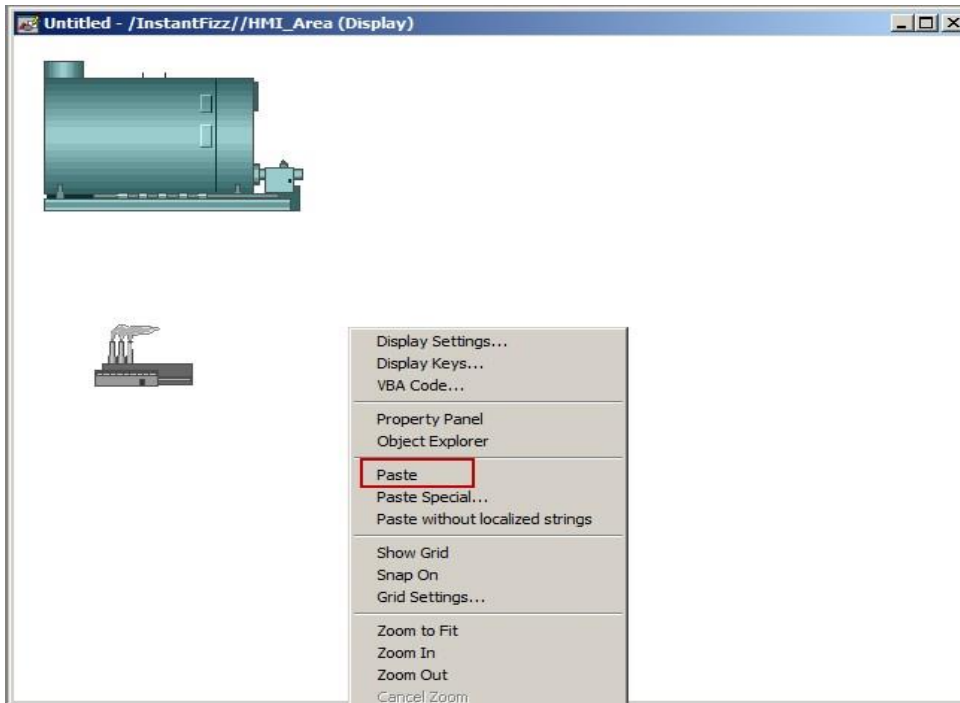
### **Adding a Symbol Factory Graphic to a display with Copy and Paste**

A third method to add a graphic to a display from the library is Copy and Paste or Paste Special.

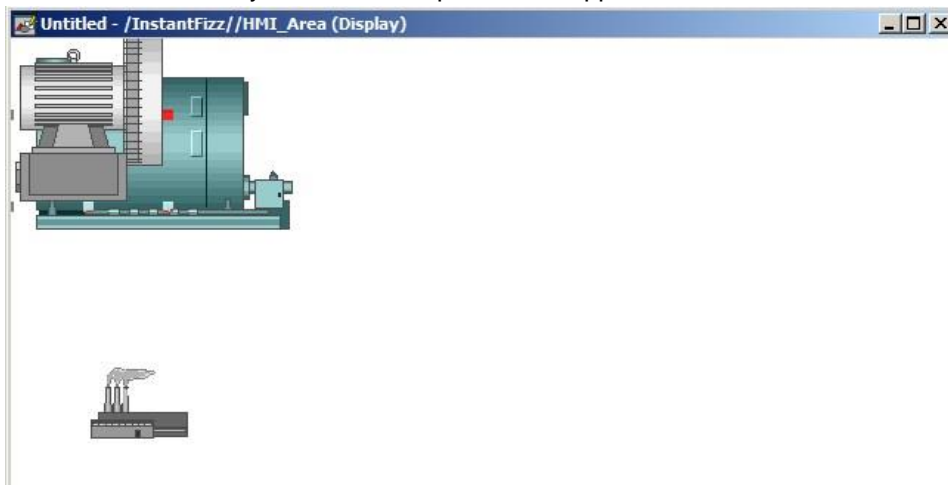
1. With Symbol Factory library displayed, select the **Blowers Etc.** category.
2. Right-click the **Vortex blower** symbol in the Preview window, to display the menu.



3. Select **Copy** from menu.
4. On the display screen, right-click and select **Paste**.

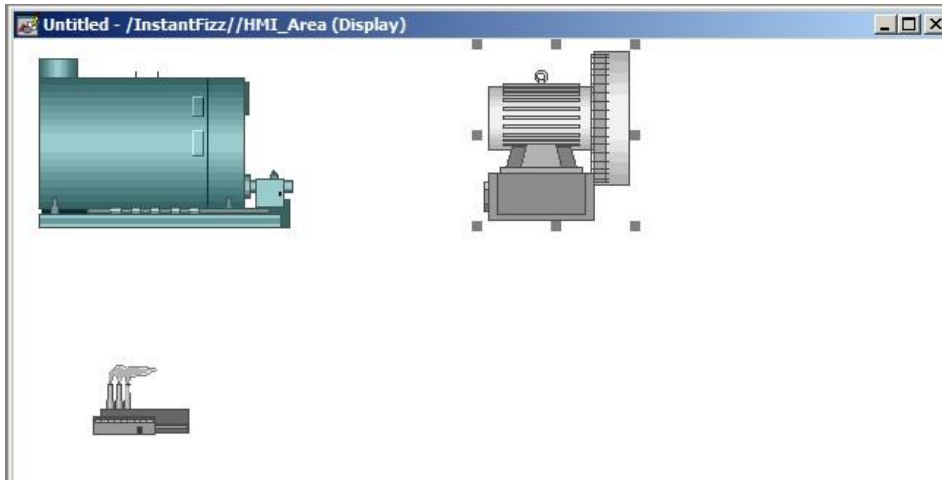


The **Vector blower** symbol will be copied to the upper left-hand corner of the display.



5. Click and drag to move the symbol from that position to another location on the display.





Those are the three main ways to copy a Symbol Factory graphic from the library to a display screen.

6. Close this display without saving.

## Adding a Symbol Factory Graphic to Images for use on a button

1. In the Explorer, right-click on the **Display** folder and select the **New** context menu item:

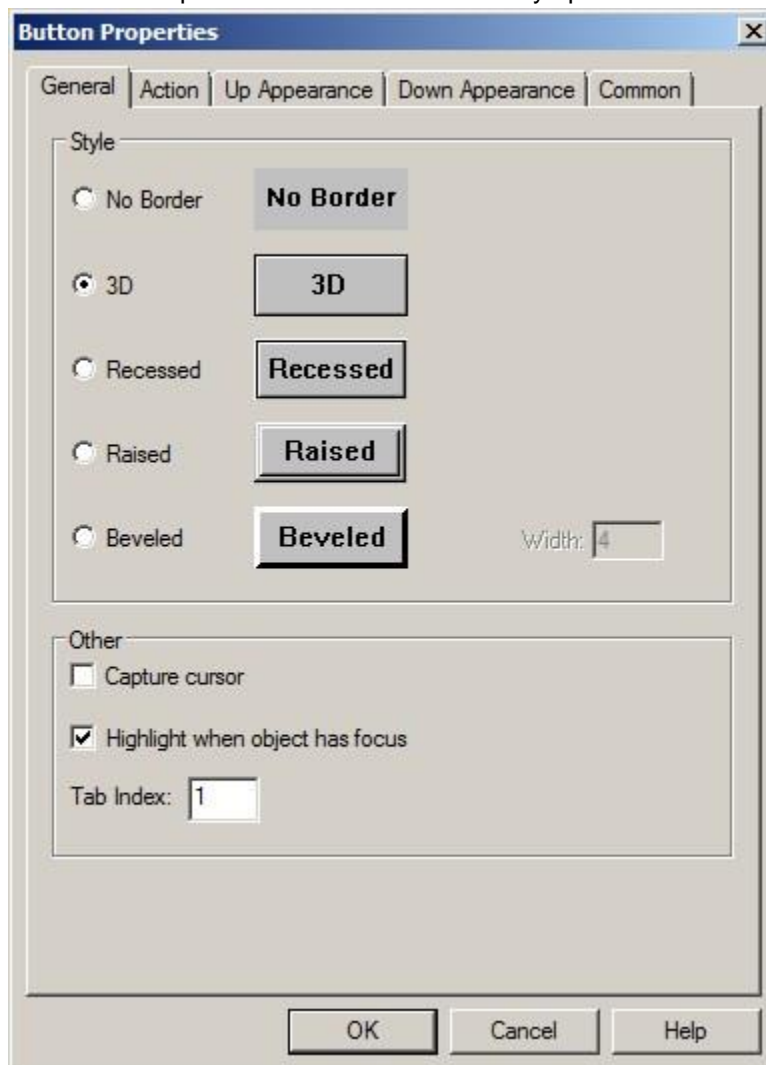


An untitled display will be opened.

2. Select the button icon from the objects toolbar and add a **Command Button** to the new display.




The Button Properties window is automatically opened:

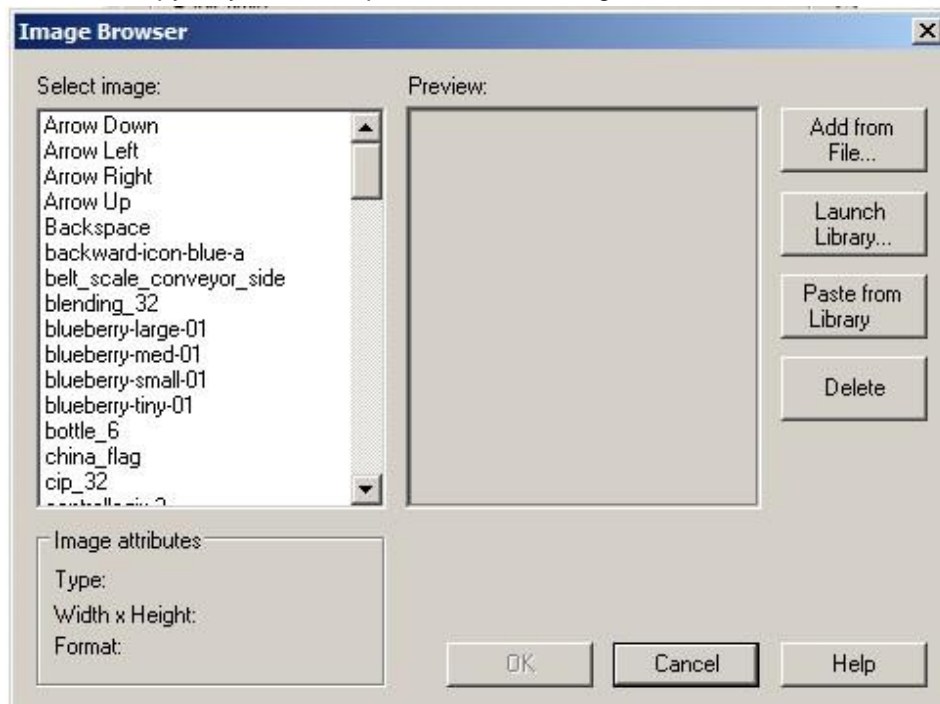


3. Select the **Up Appearance** Tab.
4. In the Image settings section, select the **Use Image reference** radio button.

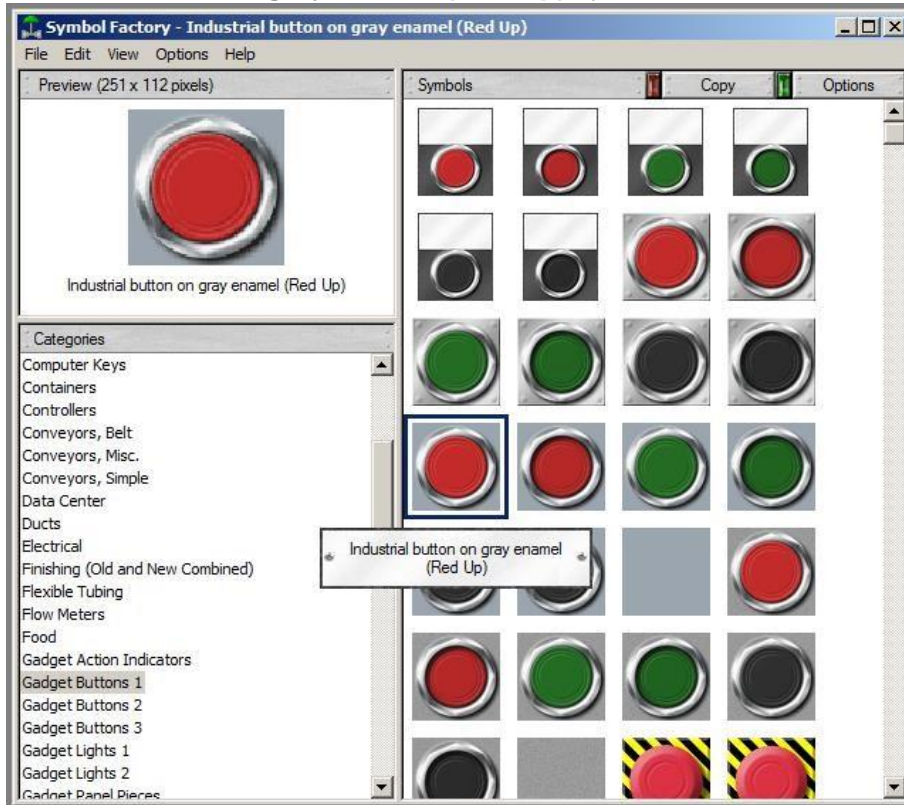



- Select  next to Image:. This will launch the Image Browser dialog.
5. next to Image:. This will launch the Image

The **Image Browser** dialogue has been modified to allow you to launch the Symbol Factory Library, select and copy a symbol, and paste it into the images folder for use on buttons.



- Select
6. **Launch Library...** to launch the Symbol Factory library.
  7. Select the **Gadget Buttons 1** category in the library and select the **Industrial button on gray enamel (Red Up)** symbol.



8. Select  above the Symbols Thumbnail window in library. Symbol Factory Library will automatically minimize.
9. Select **Paste from Library** on Image Browser dialogue.

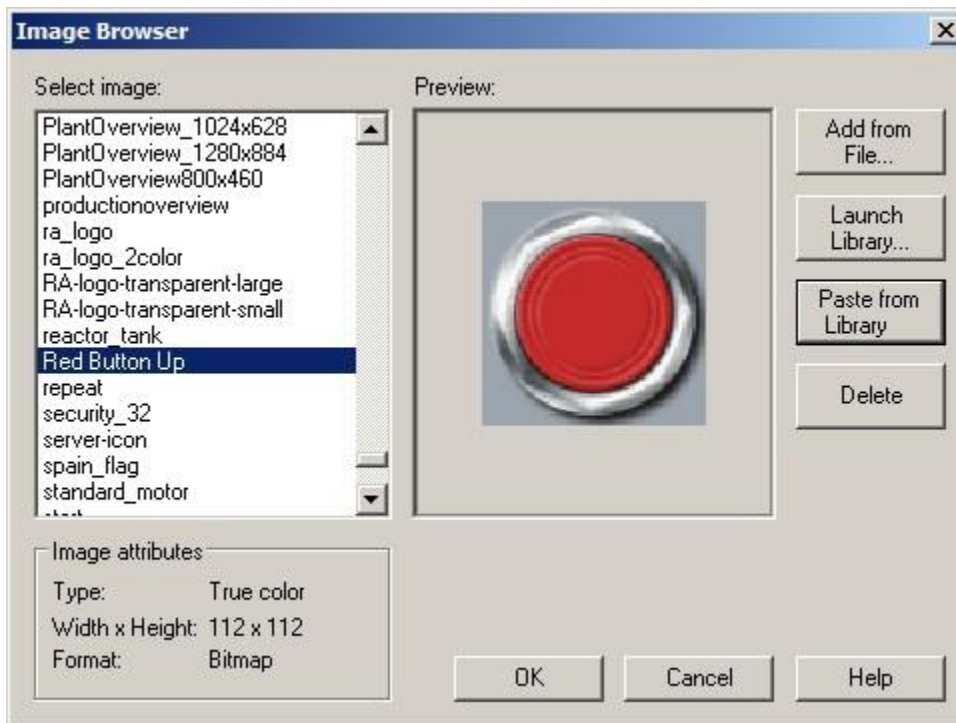


10. You will be prompted to enter a unique name for the image.

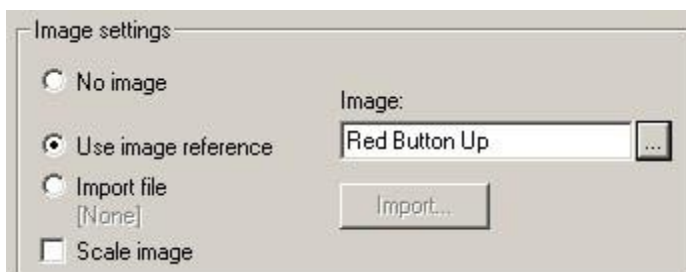


11. Enter **Red Button Up** in Image Name Entry dialogue and select **OK**.

This will paste symbol into **Preview** area of the Image Browser as well as add it to the images folder for future use.



12. Select **OK**. The Image Browser will be closed and **Red Button Up** will be listed in the Image: entry on the Button Properties.



13. **OK** to close Button Properties. The image is now displayed on the button face.

Select



14. Note that the image has been added under **Images** in the **Project Explorer**



15. Close this screen without saving.

### Color Animation Enhancements

In addition to Symbol Factory graphics library, FactoryTalk View SE also provides improved color/shading animation. In this section you will review the color animation capability.

In FactoryTalk View SE 7.0, gradient coloring was added. Gradient coloring allows a variable or static blending of two colors.

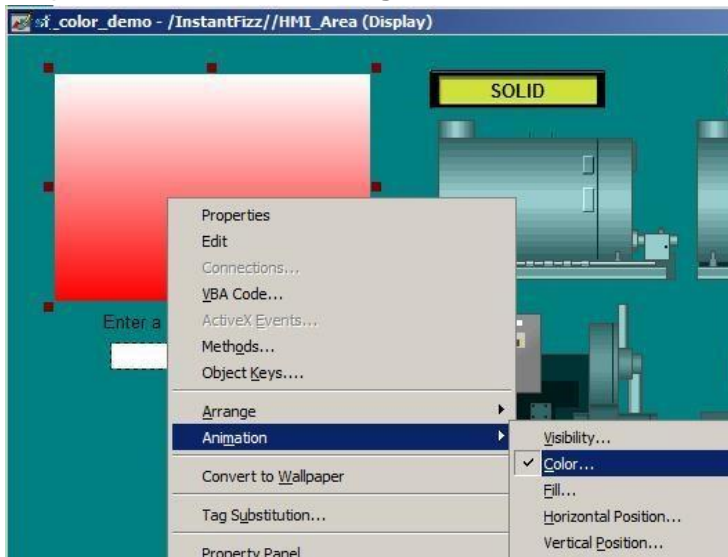


### Symbol Factory Color Animation Demo Screen

1. From the FactoryTalk View Studio explorer, open the display **sf\_color\_demo**.



2. the Gradient colored Red Square, on the left. Right-click and select **Animation > Color** to open the **Animation dialog**.



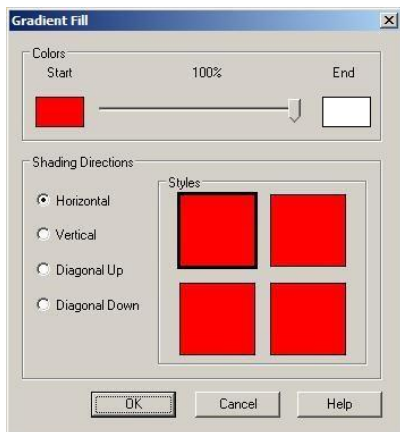
Notice the Fill Style is set to **Gradient** for each configured state.

3. Left click the small red square to the right of Fill: to open the Gradient Fill dialog.





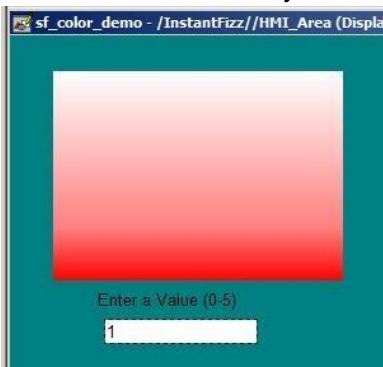
## Select




The Colors slider adjusts the percentage of each color to be displayed. To select different colors Left click the **Start** or **End** Color and select a new color. The percentage of Gradient can be different for each value of the **Color Animation**.

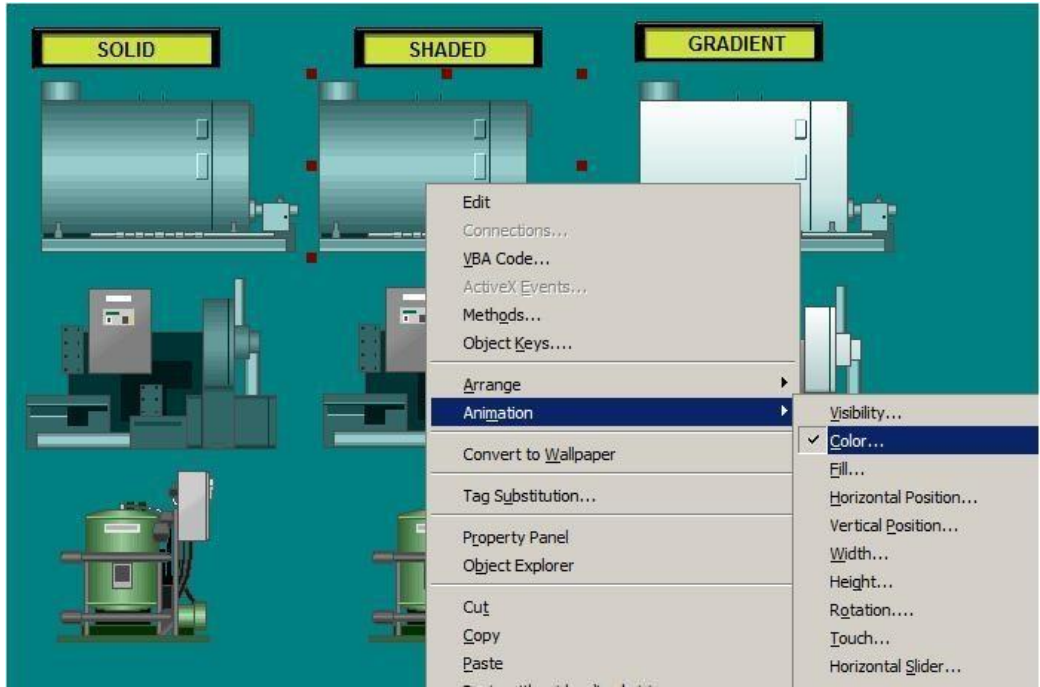
4. When you are finished looking at the Gradient Fill and Color animation configuration, click OK or **Cancel** to close the Gradient Fill dialog, **Close** to close the Animation dialog and select **No** when prompted to save changes.

5. Test run this screen by selecting **Test Display** button  on the Graphics toolbar.

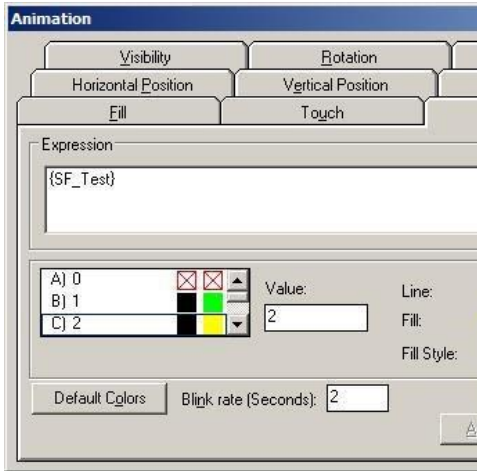


Enter Values from 0 to 5 to see the different Percentage of Gradient fill. Remember to hit the **Enter** key after you type in a new value.

6. Stop the running display by selecting **Edit Display** button  on the Graphics toolbar
7. Right-click on the Boiler symbol under the middle Shaded area. Select **Animation > Color** to open the Animation dialog.





The Color tab shows the color animation configuration for the Boiler. Note that for the '**A) 0**' state, the **Original** Fill Style is selected. This means that when the expression evaluates to a value of '0', the boiler graphic will be displayed in its original color.

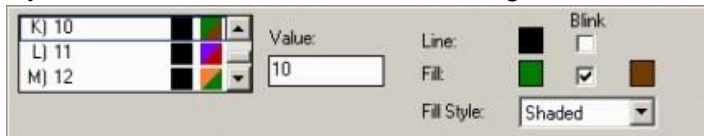


Fill Styles include Original, Shaded, Solid and Gradient.

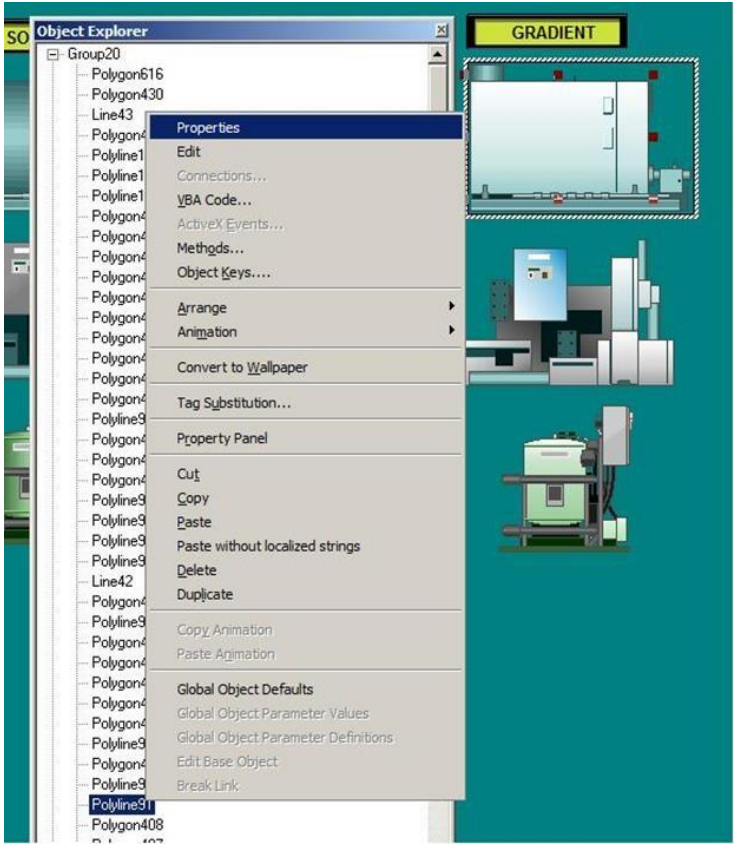
An HMI Memory Tags, **SF\_Test**, has been configured for this example.

When Fill Style is set to Original, the color indicators reflect this selection by displaying the  symbol.

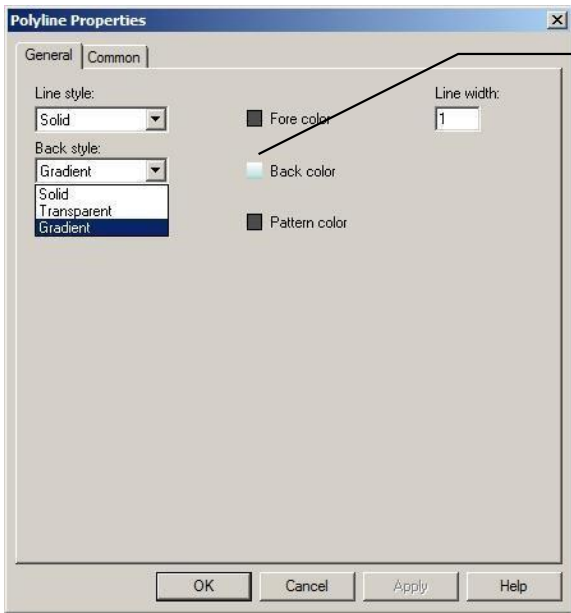
8. Scroll through the A to P thresholds on the left.  This graphic symbol has been configured for a color selection and value for each threshold. Threshold A (Value=0) has a Fill Style of Original. The remaining thresholds (B to P) are configured for the Shaded Fill Style. Thresholds K, L, and M are configured to blink the Fill Color.



9. Click **Close** to close the Animation dialog.
10. Select **No** when asked to save changes to the Animation dialog.
11. Right-Click on the **Boiler** under the Gradient Column, and Select the **Object Explorer**. Expand Group20 and Right Click on **PolyLine91**. Select **Properties**.



The Polyline Back Style is set to Gradient. Left Click on the Back Color

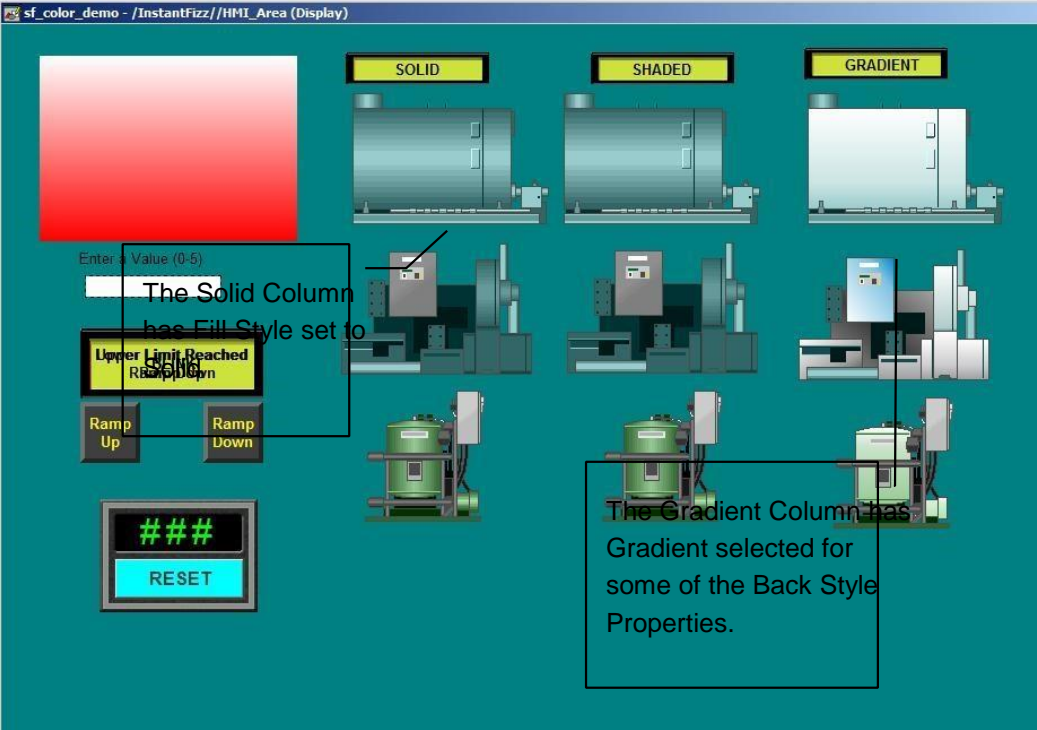


Left clicking on the Colored square brings up the Gradient Fill Dialog.



12. Click **OK** to close the Gradient Fill window
13. Click **OK** to close the Polyline Properties Window

14. The machines below the **Boiler** have Polylines with their back Styles set to Gradient and the Gradient fill properties set to enhance the coloring of the machines. Explore each machine, setting the Back Style and Gradient Fill coloring percentage and shading direction.




15. Test run this screen by selecting the **Test Display** button on the Graphics toolbar. The screen will appear like this when running:



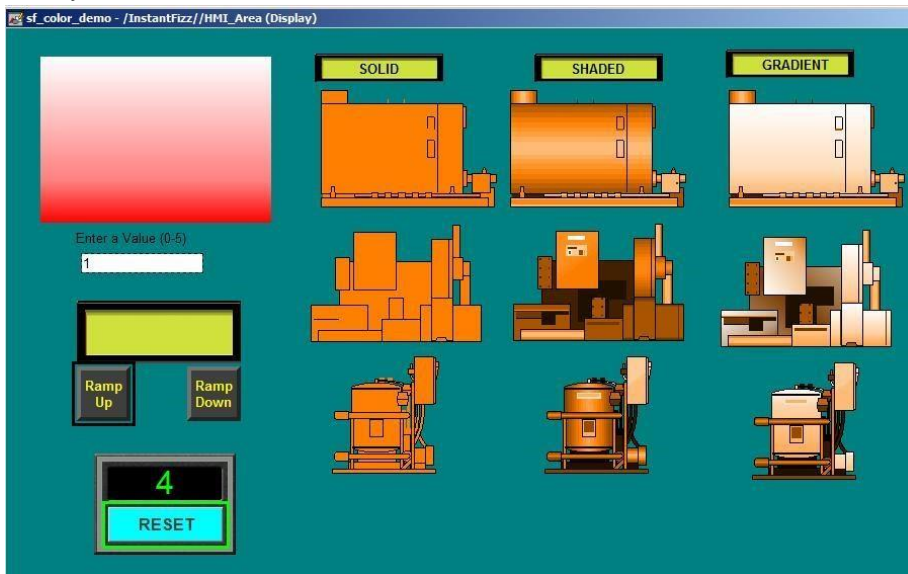


The starting value is zero (0). If the value shown in the indicator is not zero, then click **RESET** to set it back to zero (0). Each graphic symbol on this display is configured to maintain the original library color at value zero (0).


16. To show the color changes on each symbol, increment the analog value by selecting the

**Ramp Up** button . The value will increase by 1 and each symbol will change to the next color. Notice that even with the color change, the shading of the objects in the middle column has been maintained.

When the value is greater than zero, notice the difference between the Solid, Shaded and Gradient Fill Styles.



17. Ramp the value to 11, 12, and 13 to see the object blink between two colors.

18. Stop the running display by selecting **Edit Display** 

19. Close the display without saving.

## Find and Replace

As of version 7.0, FactoryTalk View includes a new feature called Find and Replace. This feature allows for finding a specific tag or a text string within specific product component and location. The product components that can be searched within are Graphic Displays and Global Objects. The location (the scope) of the search can be one or more HMI servers that belong to the same application.

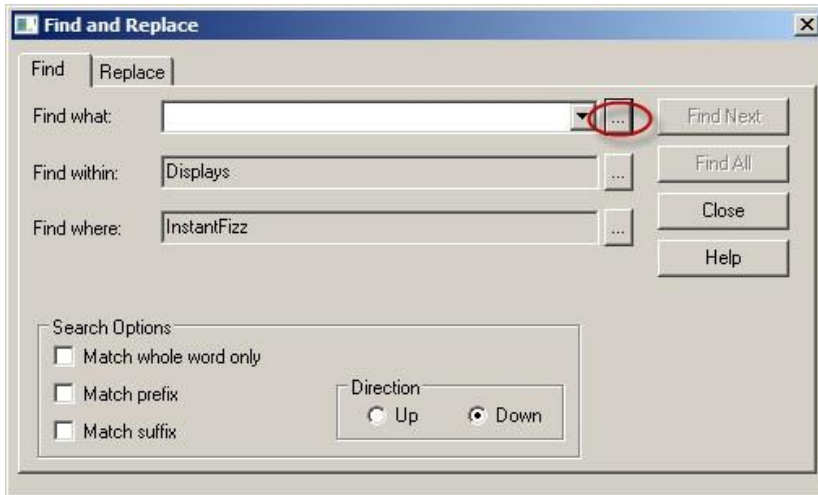
We will explore this feature in more detail in the following exercise.

1. In the Standard toolbar in View Studio click on the **Find** icon:

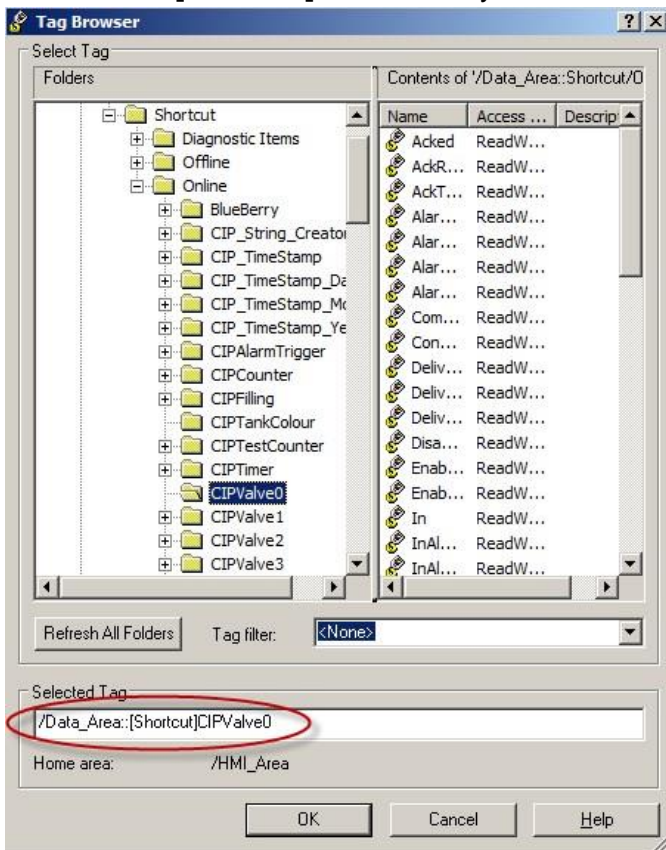




2. In the **Find and Replace** dialog click on the ellipsis button next to **Find what** field to open Tag Browser:

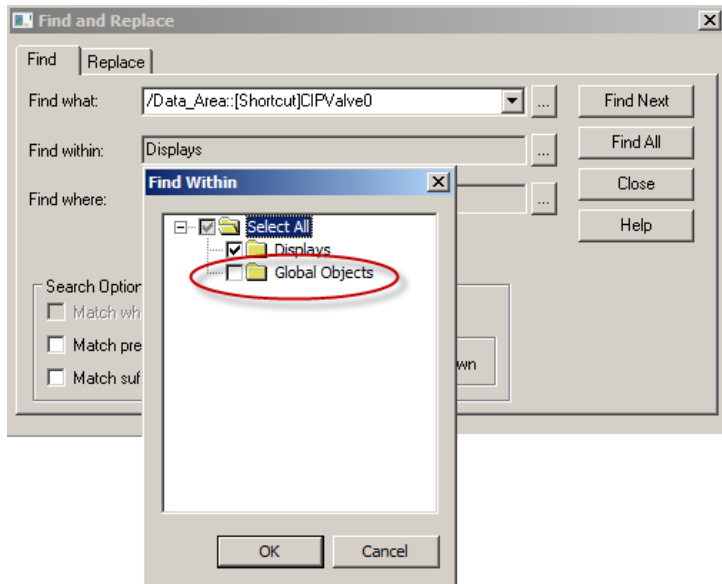


3. Select **CIPValve0** item, as shown below making sure that **/Data\_Area::[Shortcut]CIPValve0** syntax is shown in the Selected Tag field:

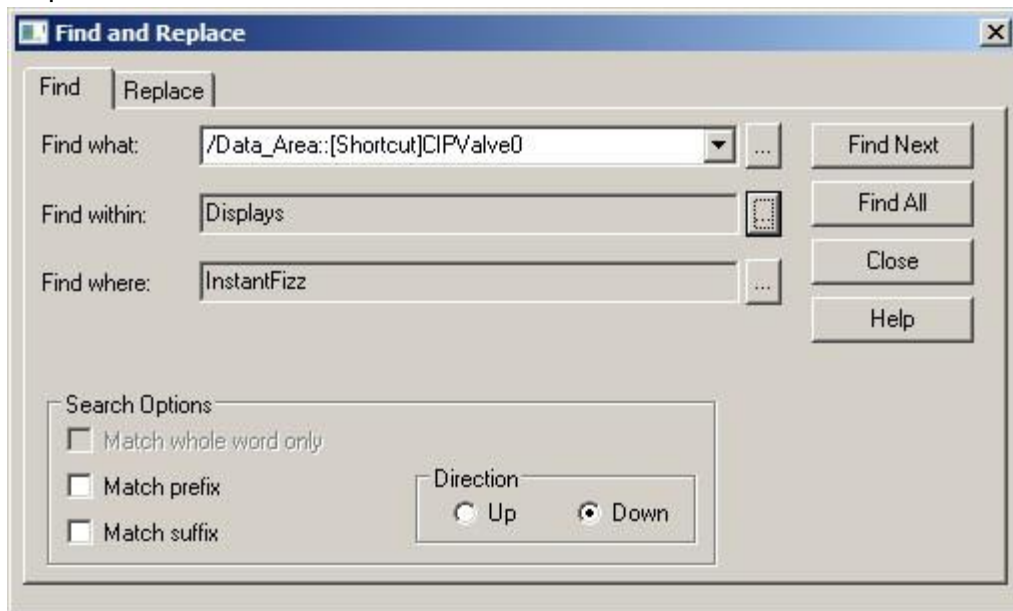


4. Click **OK** to close Tag Browser .
5. In the **Find and Replace** dialog, click on the ellipsis button next to **Find within** field and unselect the checkbox next to **Global Objects**. Notice we limited the component search scope to **Displays** only.



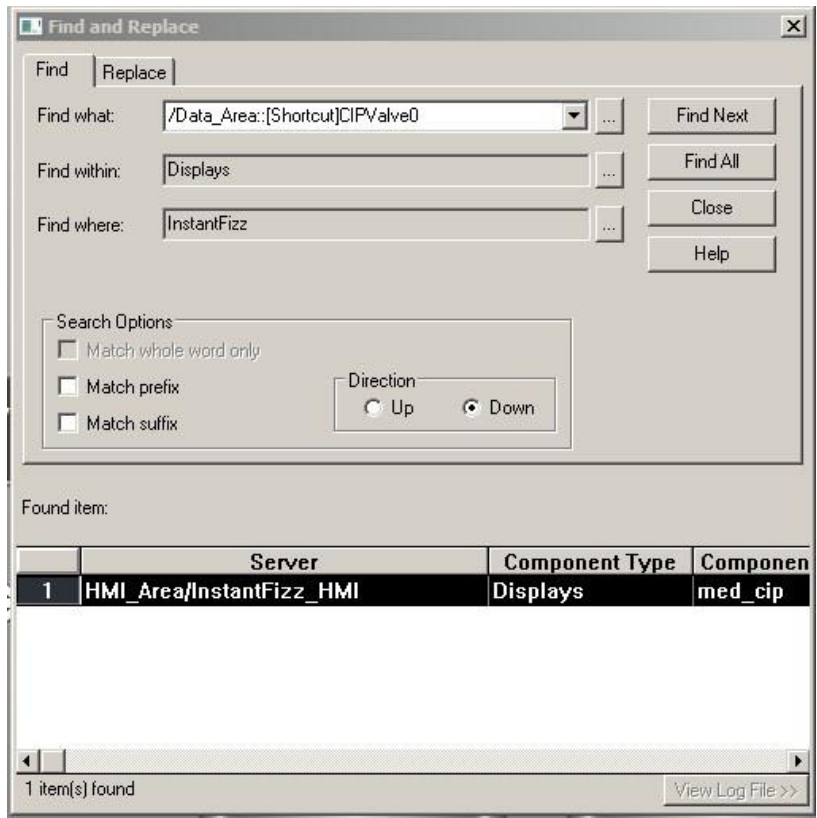


6. Click **OK** to close Find Within dialog - the Find and Replace window should look like in the snapshot below:



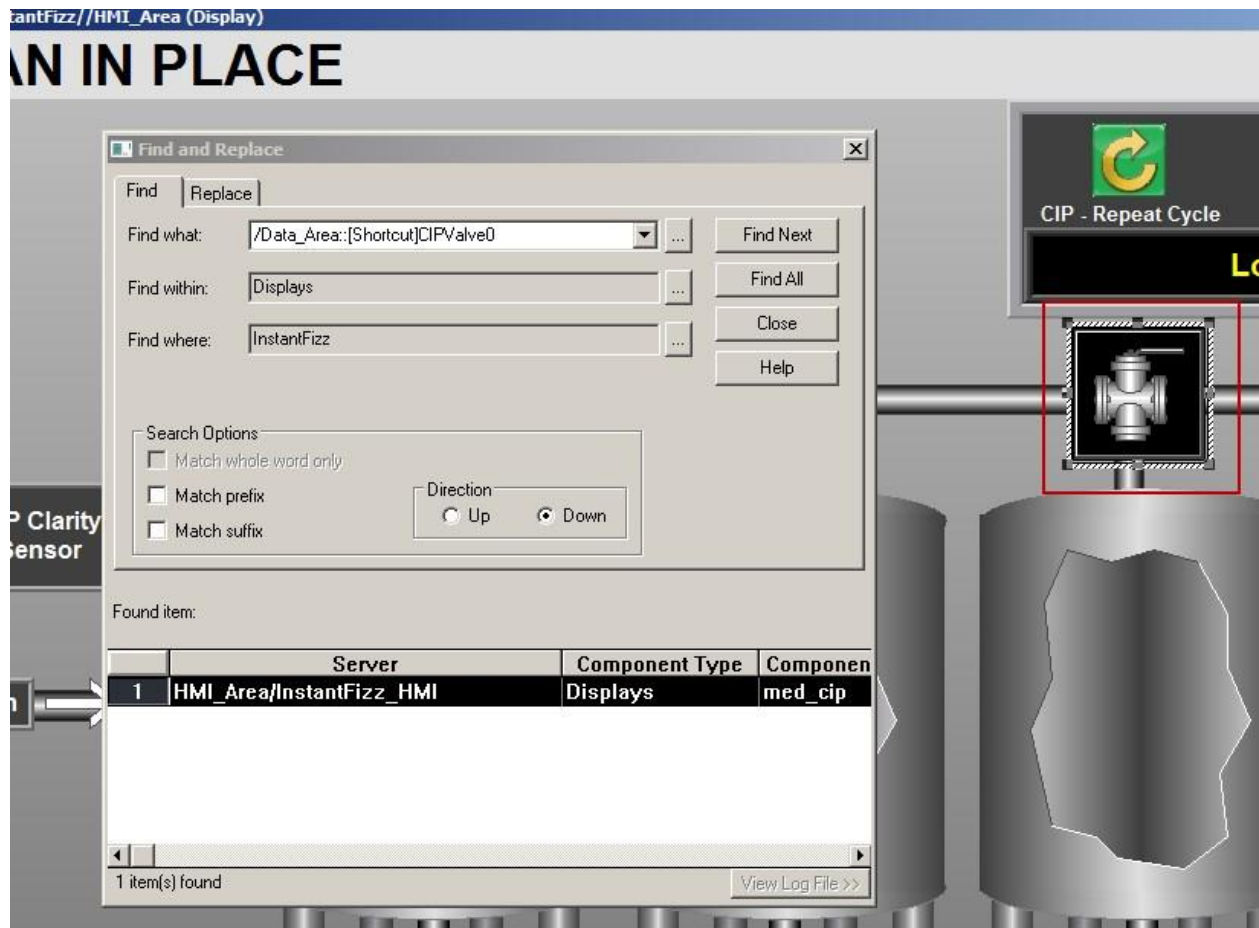
Note that we didn't need to modify the **Find where** field since the application contains only one HMI server which by default becomes the scope of the search.

7. Click **Find All** button to find all instances of the data item **CIPValve0** in all graphic displays. After the search is done, the search results will be presented in table format as shown below:



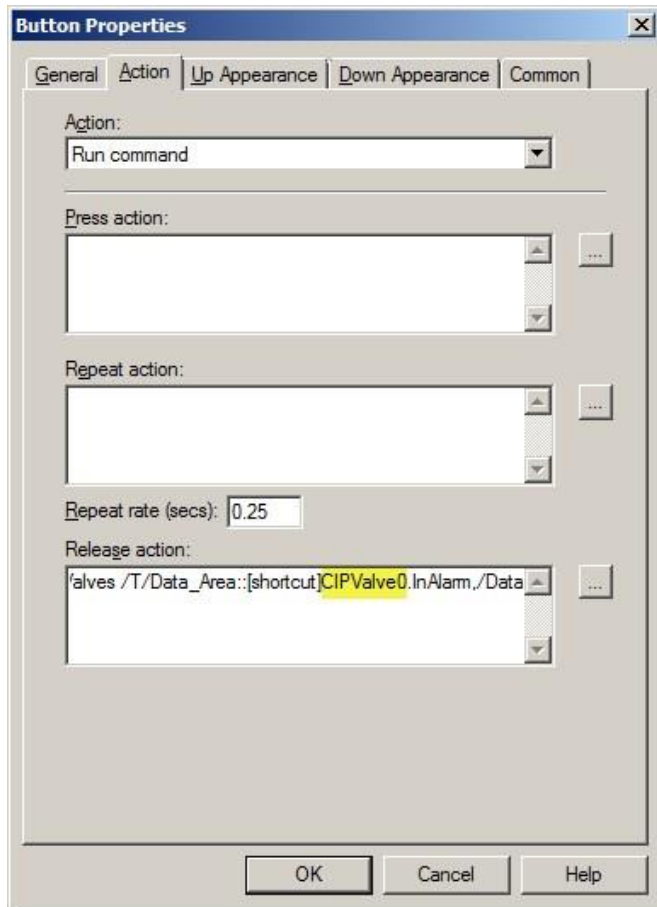
The search found one graphic display containing the searched data item: **med\_cip**.

8. Double-click on the found item in the table - this action will open the hosting display **med\_cip**.
9. Using horizontal scroll bar at the bottom of the **med\_cip** display, scroll to the right until you see the highlighted object as shown below:



In addition to automatically opening the display hosting the searched item, the new feature also highlights the object on a display associated with the searched item (CIPValve0). In the next step, let's edit the highlighted object:

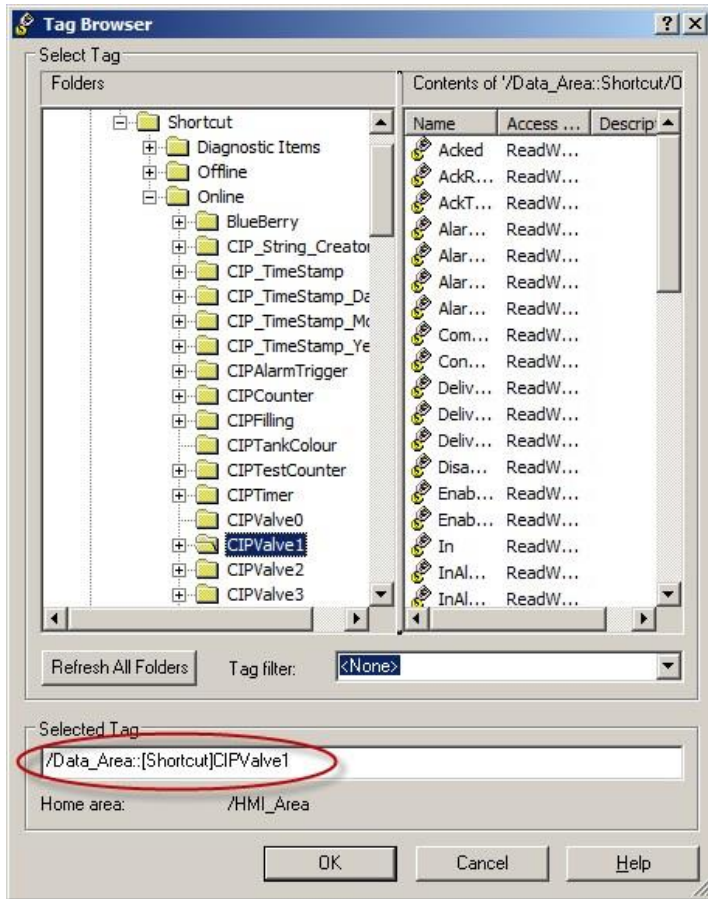
10. Double-click on the highlighted object and in the **Button Properties** click on the **Action** tab.
11. Position the cursor in the **Release Action** field and scroll a few characters to the right – you will find that the item **CIPValve0** indeed exists in the command syntax:



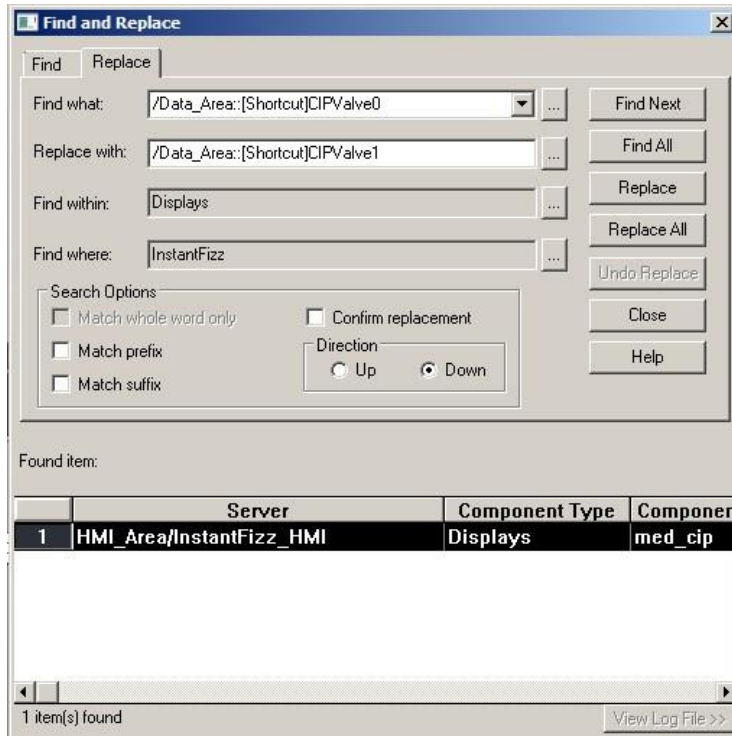
12. Click **OK** to close Button Properties window. Keep the **Find and Replace** window open.

In the next part of this exercise we will explore Replace function of the Find and Replace feature

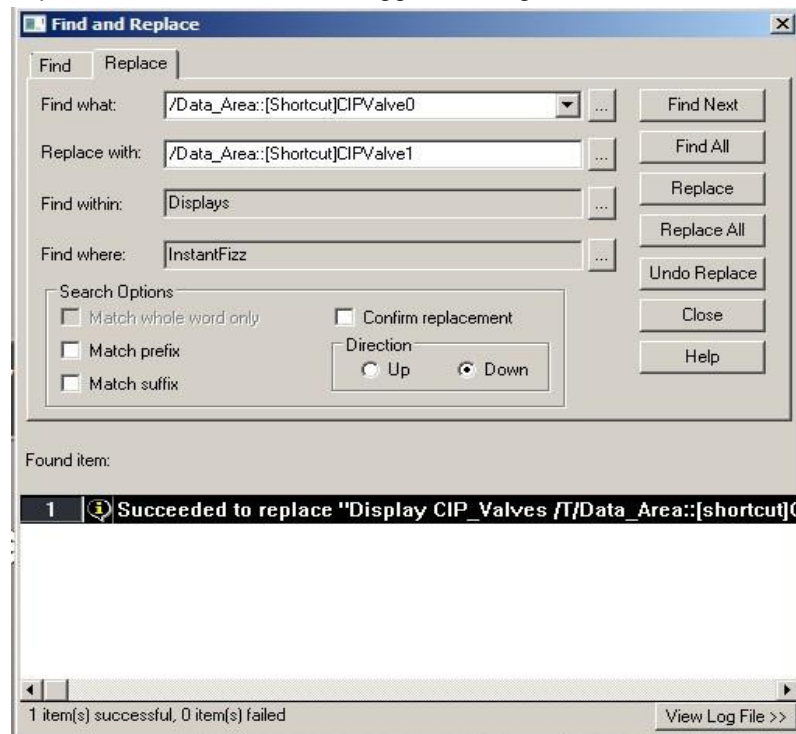
1. In the Find and Replace window click on **Replace** tab.
2. Click on the ellipsis button next to **Replace with** field.
3. In the Tag Browser select the item **CIPValve1** as shown below:



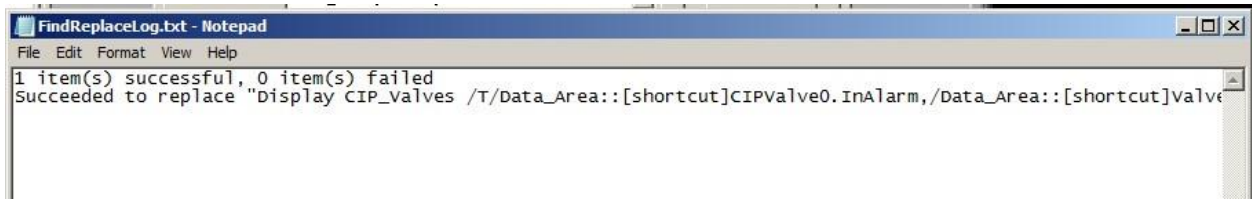
4. Click **OK** to close Tag Browser.
  5. Back in the Find and Replace window, unselect **Confirm replacement** checkbox.
- Find and Replace window should look like in this screenshot:



- Click on **Replace All** button to replace all found instances of CIPValve0 with CIPValve1. After the replacements are completed, the results will again be presented in a table format. Replacement results are also logged to a log file in text format:



- Click on the **View Log File** button at the bottom right corner – the log file will open in Notepad:

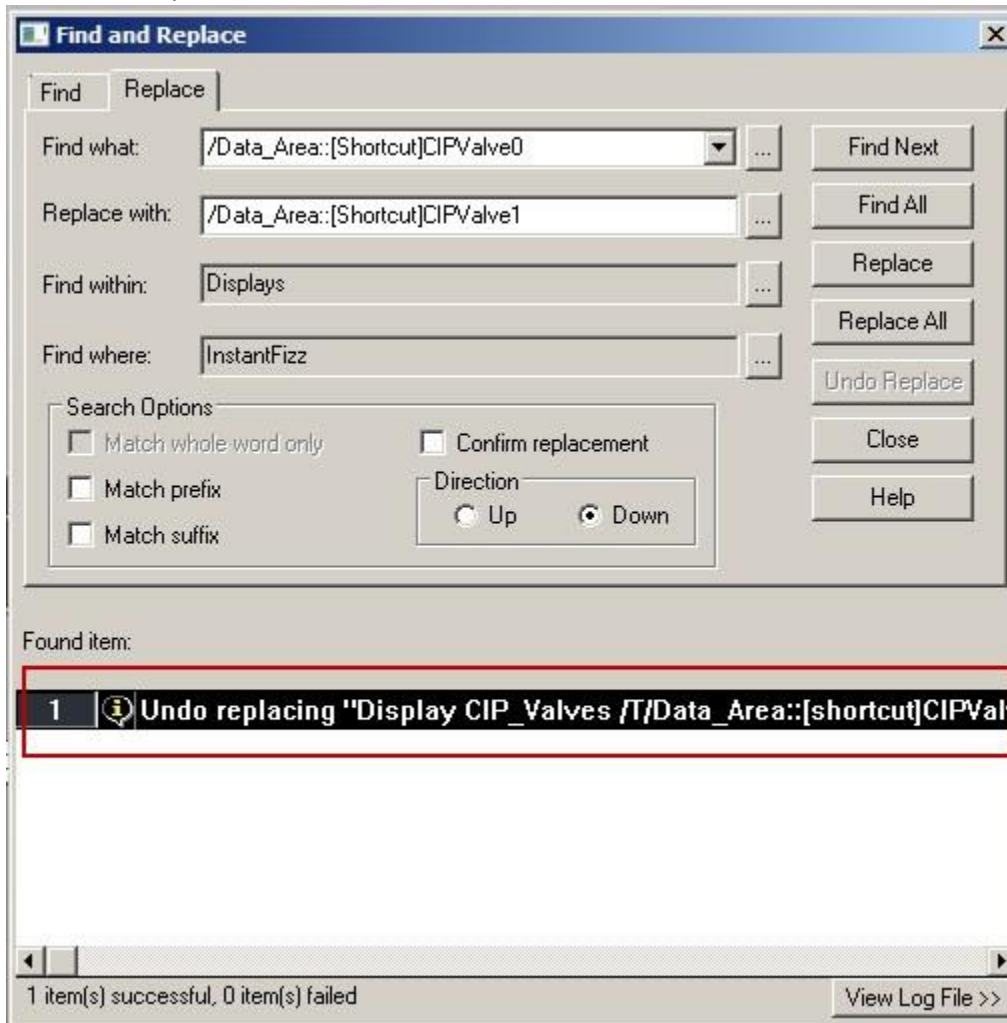


After reviewing the log file, close the Notepad.

For the rest of the lab we don't want the above replacement to be in effect so let's undo them all:

8. Click on **Undo Replace** button.

After all the replacements have been undone, the undo results will be shown in the table:



Again, you can click on the **View Log File** button to open the log file and see the undo replacing results in Notepad.

Notice that there are number of search options that you can use for your custom searches such as **Match whole word only**

(if you are searching for a text string, as opposed to a data item), **Match prefix**, **Match suffix** as well as the search **Direction**. For more details on these search options click on the **Help** button.

9. Close the Find and Replace dialog.



## Cross Reference

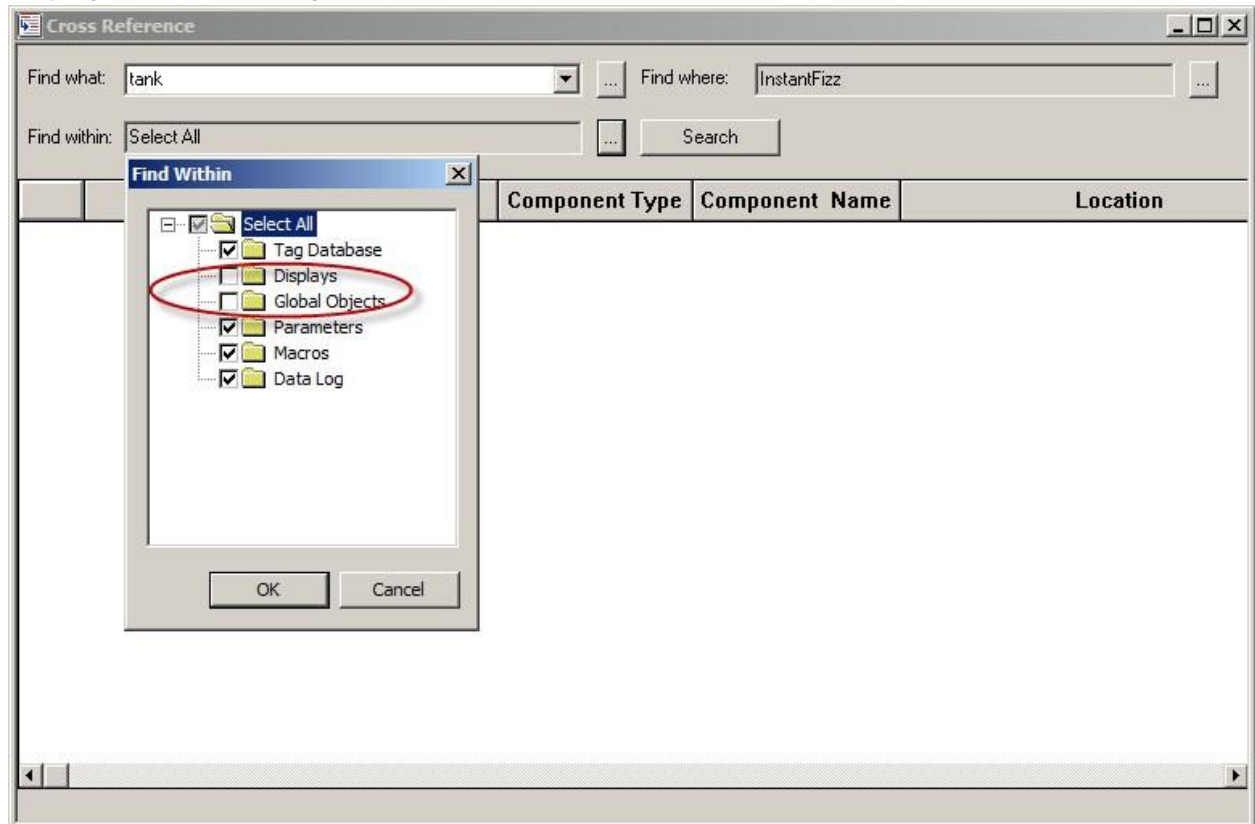
Cross Reference is yet another new feature introduced in version 7.0 of FactoryTalk View. This feature adds ability to find tags and text strings in the following HMI project components: Tag Database, Displays, Global Objects, Parameters, Macros and Data Logs. Same as Find and Replace feature, it operates across multiple HMI servers and presents results in an interactive spreadsheet format.

1. In the Standard toolbar in View Studio click on the **Cross Reference** icon:



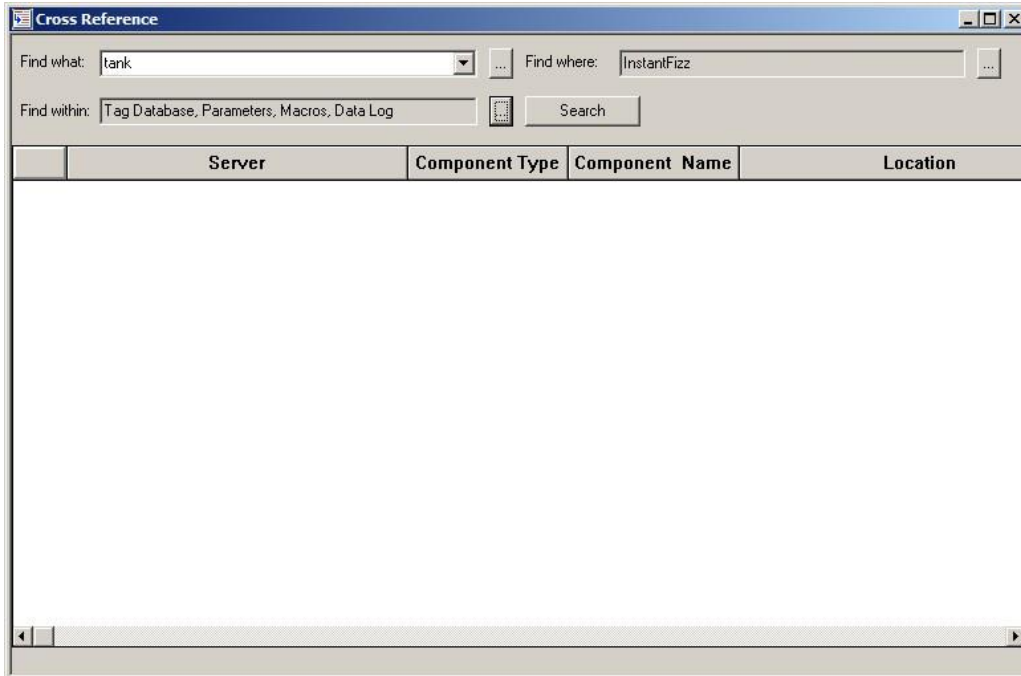
This time we will search for a specific text string within selected components:

2. In the Cross Reference window, type in string **'tank'** in the **Find what** field.
3. Click on the ellipsis button next to **Find within** field and uncheck the checkboxes next to **Display** and **Global Objects**:



In this example we are interested in finding all instances of the text sting 'tank' but we want Displays and Global Objects excluded from the search.

4. Click **OK** to close Find Within dialog - the Cross Reference window should look like in the snapshot below:



Note that we didn't need to modify the **Find where** field since the application contains only one HMI server which by default becomes the scope of the search.

5. Click **Search** button.

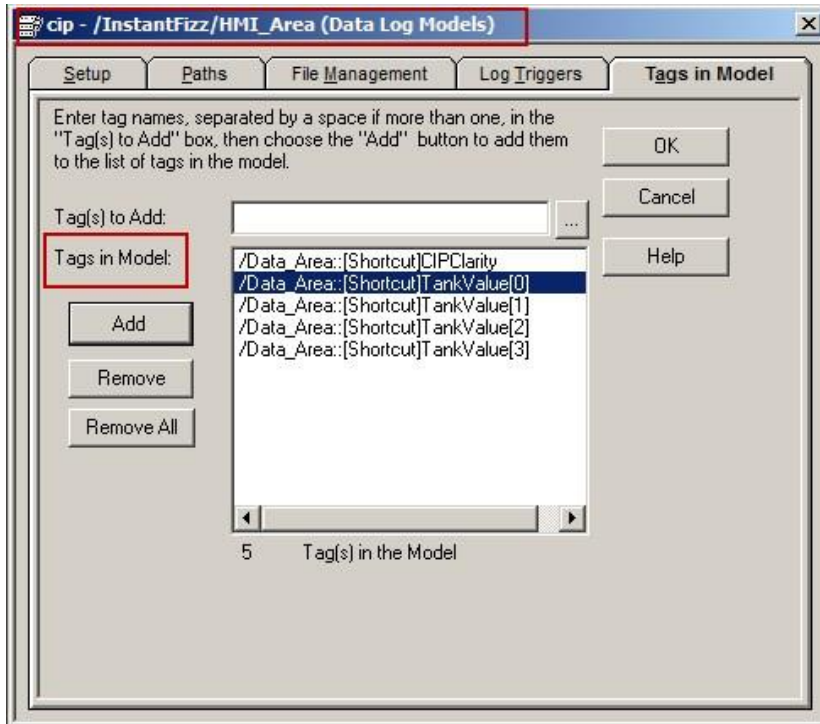
After the search is done, the search results will be presented in table format as shown below:

	Server	Component Type	Component Name	Location	Reference
1	HMI_Area/InstantFizz_HMI	Data Log	cip	/Data_Area:[Shortcut]TankValue[0]	/Data_Area:[Shortcut]TankValue[0]
2	HMI_Area/InstantFizz_HMI	Data Log	cip	/Data_Area:[Shortcut]TankValue[1]	/Data_Area:[Shortcut]TankValue[1]
3	HMI_Area/InstantFizz_HMI	Data Log	cip	/Data_Area:[Shortcut]TankValue[2]	/Data_Area:[Shortcut]TankValue[2]
4	HMI_Area/InstantFizz_HMI	Data Log	cip	/Data_Area:[Shortcut]TankValue[3]	/Data_Area:[Shortcut]TankValue[3]

The search found four instances of the text string 'tank', all of which belong to the Data Log component called 'cip'

6. Double-click on the first item in the table.

This action will open the data log model **cip** and the tab Tags in Model. In the list of tags you will see a highlighted tag containing the first found instance of the text string 'tank':



In the above list, you will also notice three other instances of the string 'tank' found by Cross Reference.

7. Click **OK** to close Data Log Model window.
8. Close the Cross Reference window.

The Cross Reference function exercise demonstrated how easy it is to find and locate any tag or a text string in the HMI project.

## PNG Support

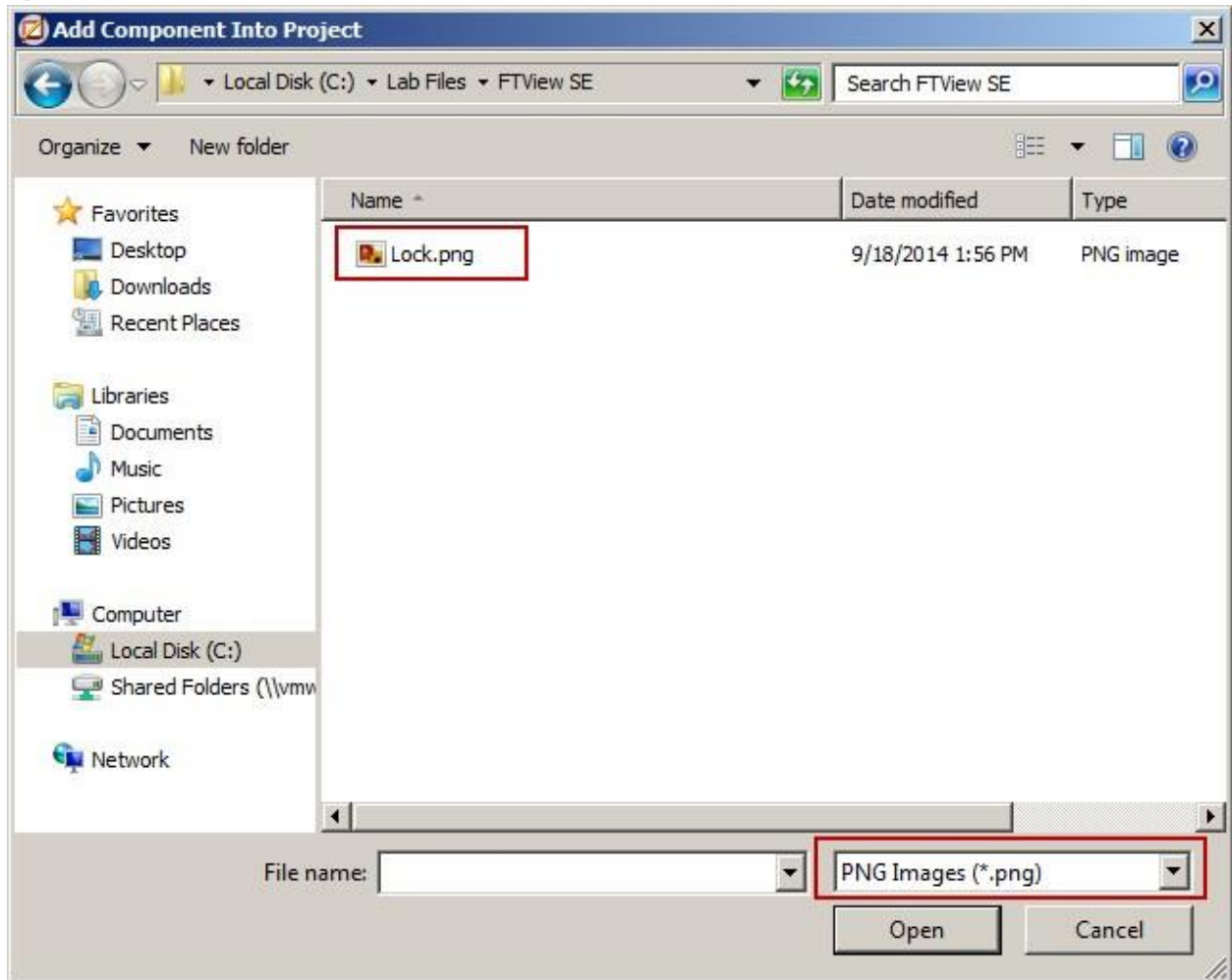
As of version 7.0, FactoryTalk View supports the PNG image type. PNG is a lightweight graphic format that is becoming popular in the industry. One of the nice features of PNG is that it supports transparency which allows the background color to show through. In addition to PNG file type, JPEG and BMP image types are also supported.

### Importing PNG Files

1. To import a **PNG** file in **Studio**, right click on Images and select the **Add Component Into Application** menu.



2. Navigate to **C:\Lab Files\FTView SE** folder.
3. Select **PNG Images (\*.png)** in bottom right of dialog and select **Lock.png** file and press **Open** button.



The Lock.png file will be imported into application.

4. Find the **Lock** image in the application tree and double-click it to open it.  
You can find the image under the **Images** folder from the Project Explorer.



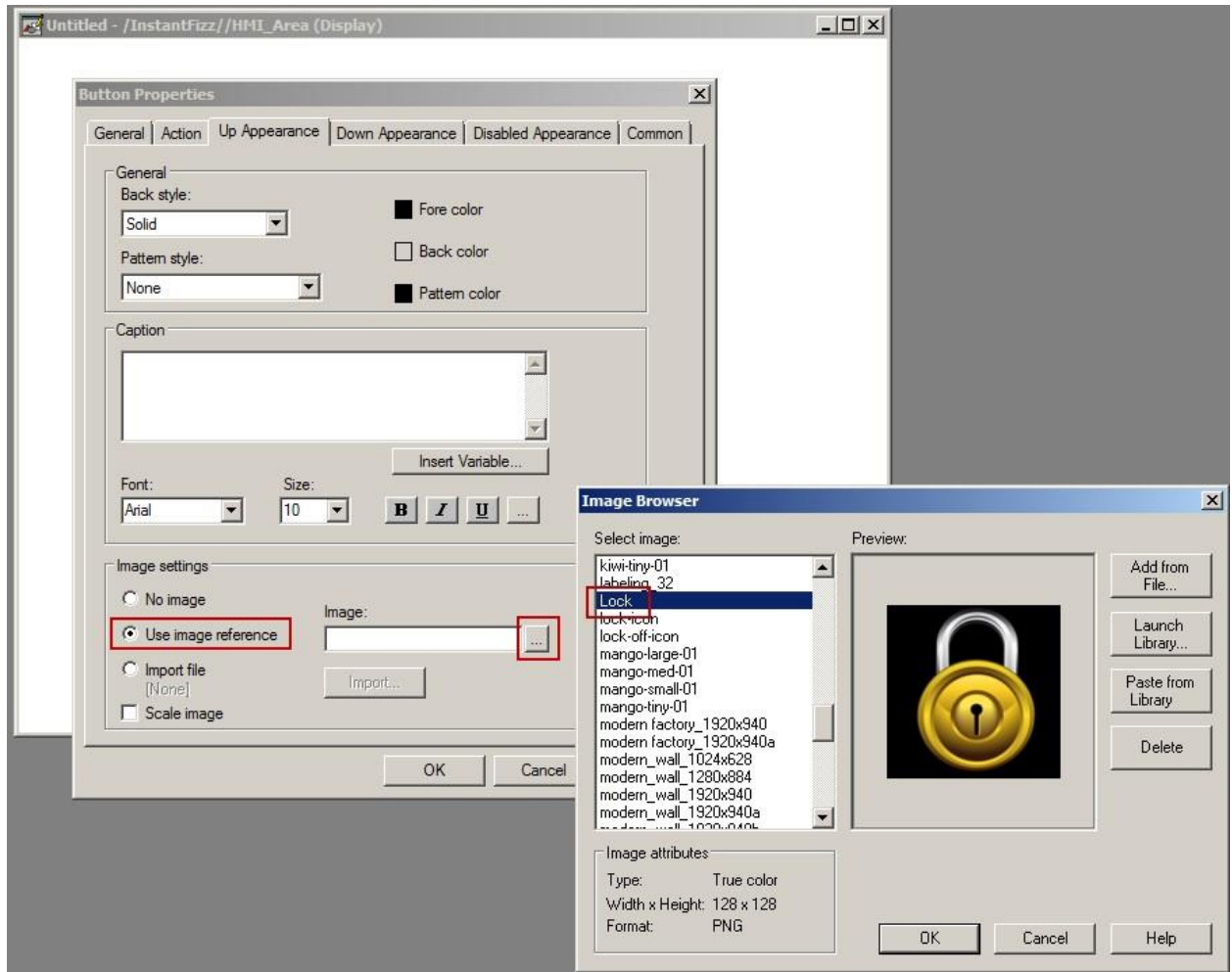
Notice the format is PNG.

5. Close the PNG file by pressing the **Close** button.

PNG files can be used anywhere an **Image** object can be used.

#### Using PNG Image on Button Object

1. Expand the **Graphics** folder in the Explorer pane. Right-click **Displays** and select **New** to create a new Display.  
A new display is opened in the workspace area.
2. Select the **Objects > Push Button > Button** menu or click the **Button** tool on the graphics tool bar.
3. Left click the mouse and drag to make a fairly large button on any empty area in the display. When you release the mouse button, the **Button** property page will open.
4. Select the **Up Appearance** tab.
5. Under **Image settings**, select the **Use image reference** option and then press the ... button to open the **Image** browser.
6. Select the **Lock** image and press OK.



7. Press **OK** to save the Button properties.



Notice how the Lock image is transparent and you can see the button color behind it.

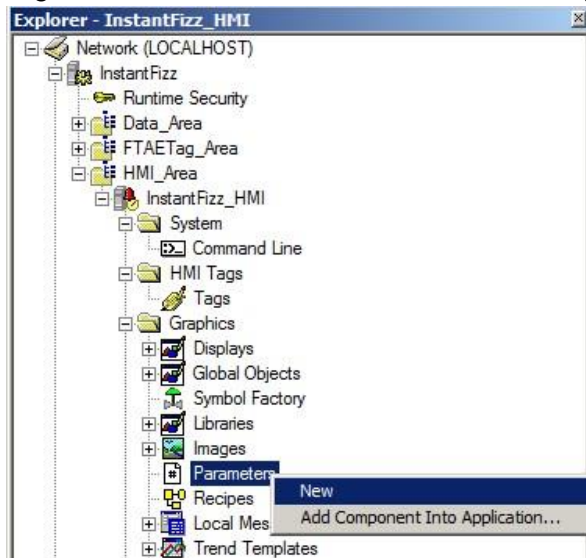
8. Close the display without saving.

## Parameter File Supports Tag Browsing

When you create a graphic display that uses tag placeholders, you can specify a parameter file that will supply tag or folder names, or parts of tag names, for the tag placeholders at run time. This allows you to use different sets of tags with a single graphic display.

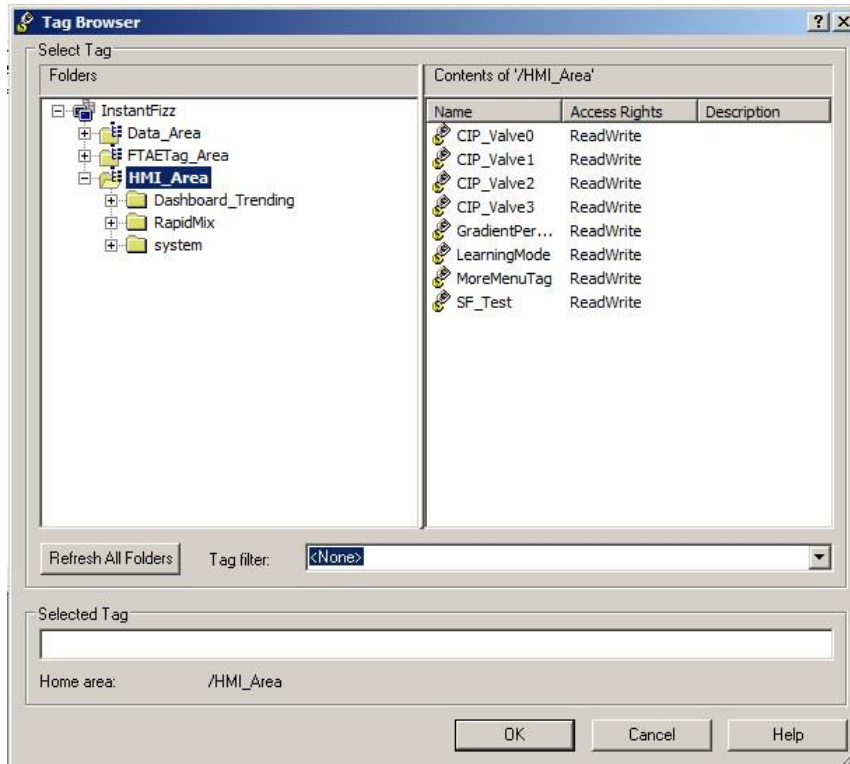
As of version 7.0, FactoryTalk View supports the ability to launch the Tag Browser from the Parameters editor.

1. Right click on the **Parameters** editor in the application tree and select **New** menu.



2. In the new Parameters file that opens, type in the string '#1=' and then double-click to open the Tag Browser.





3. Select any tag and press **OK**.

You have just created a single Parameter definition. Notice how the full tag path is added to the Parameter definition

4. Close the Parameters file without saving.

## Digit Grouping

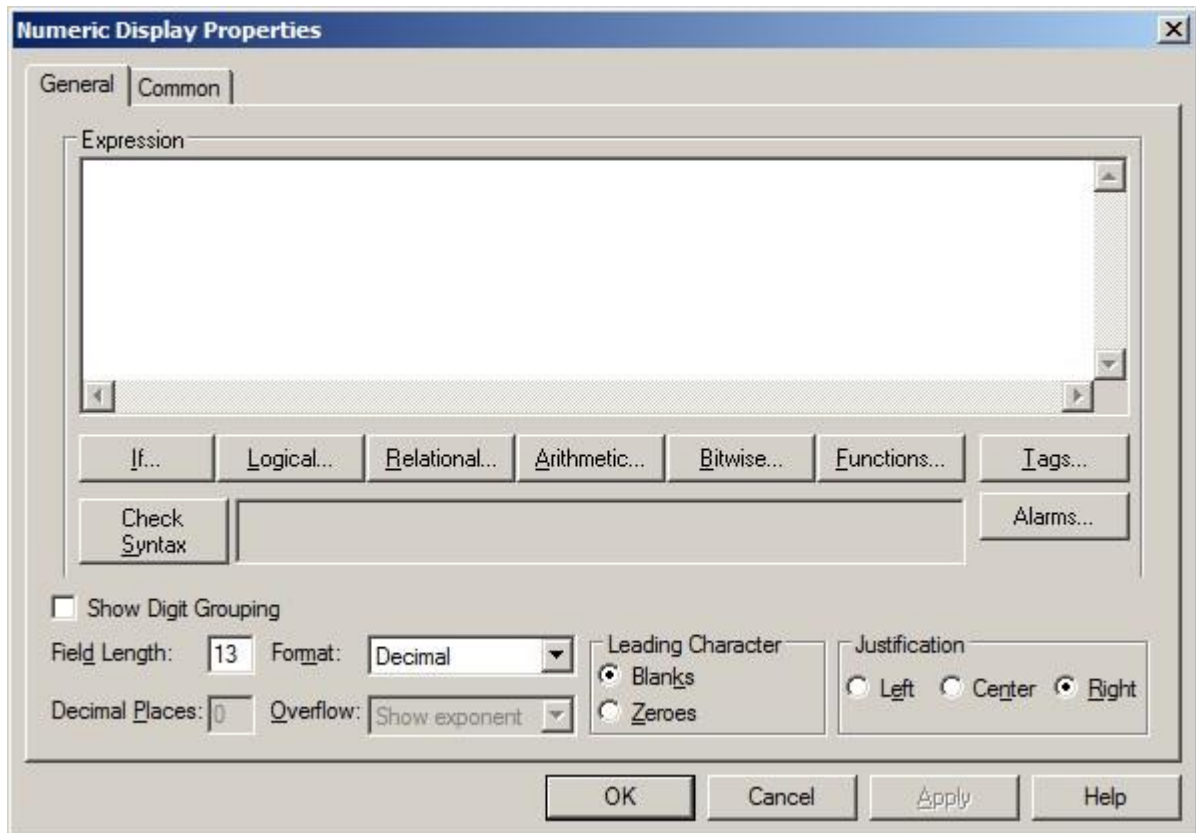
The Numeric Display graphic object can show the digit grouping of the tag value at runtime. The digits are divided in groups of three by the Windows digit grouping symbol. The digit grouping symbol uses the numeric format of the current runtime language.

The format can be configured in the Windows 'Region and Language' settings of Control Panel.

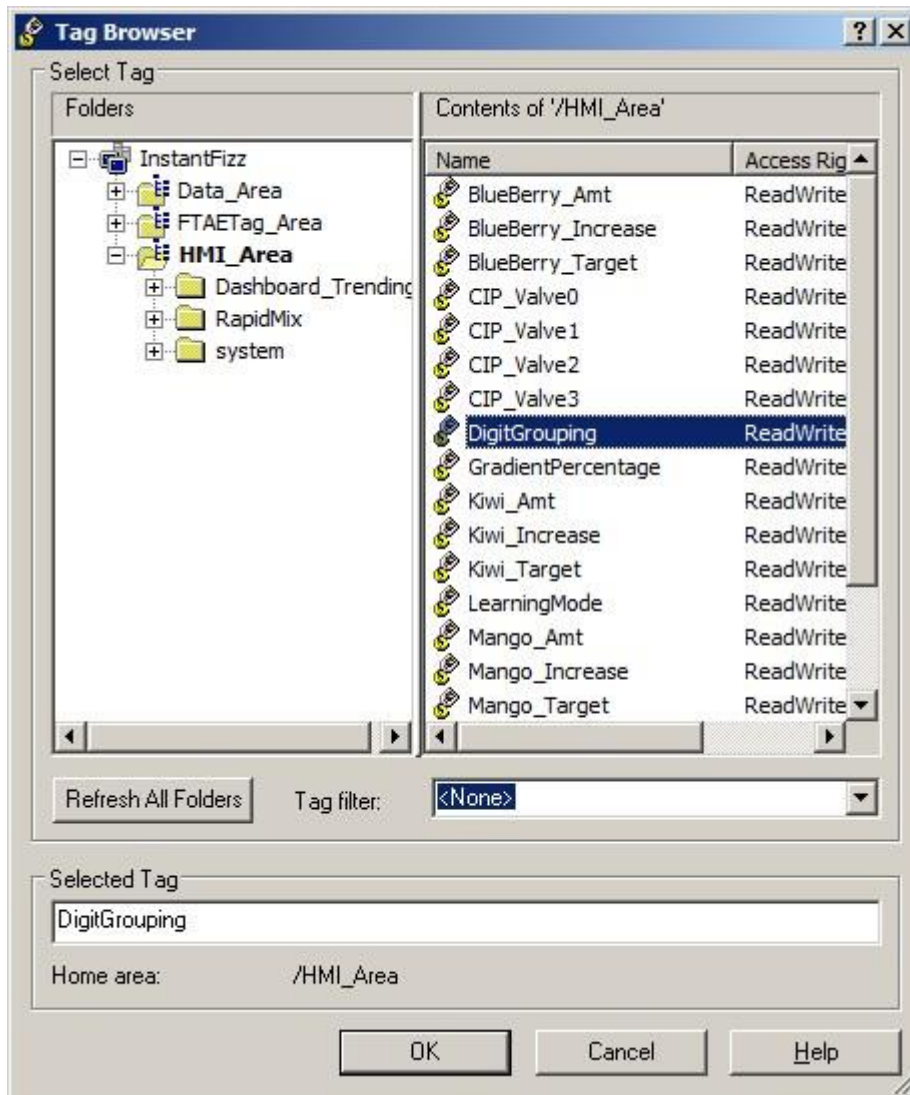
Let's explore this feature.

1. Open the **languages** display in **FactoryTalk View Studio**.
2. Select the **Objects > Numeric and String > Numeric Display** menu.
3. On the blank area at the bottom of the display, click and drag the mouse to draw the **Numeric Display** object and then release the mouse button to create the object.

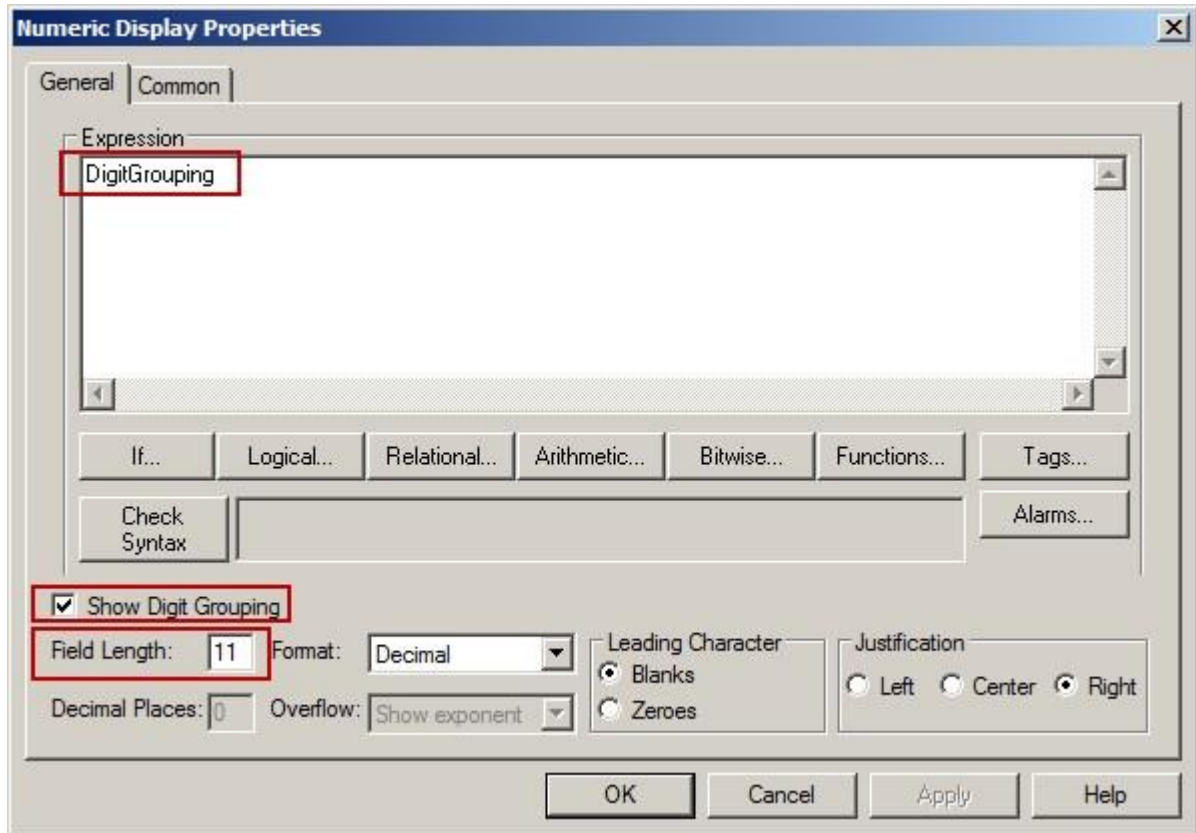
The **Numeric Display** properties dialog will appear.



4. Click on the **Tags...** button to open the Tag Browser.
5. Select the **HMI\_Area** item on the left side to display the HMI tags on the right side.
6. Select the **DigitGrouping** tag on the right side and press **OK**.



7. Set the **Field Length** to 11.
8. Check the **Show Digit Grouping** option and press **OK** to save the changes.



We have just created a Numeric Display object that is using **Digit Grouping**. Now let's duplicate this object and turn off the **Digit Grouping** so that you can see the difference.

9. Right-click the **Numeric Display** object and select the **Duplicate** menu item. (Alternatively, you could have selected the object and pressed **Ctrl-D** to duplicate it.)



10. Click the duplicated object and drag it below the original object.

11. Double-click the duplicated object to open the **Numeric Display** Properties dialog.

12. Uncheck the **Show Digit Grouping** option and press **OK** to save the changes.

Now we are going to need a **Numeric Input** object so that we can modify the value of the **DigitGrouping** tag.

13. Select the **Objects > Numeric and String > Numeric Input** menu.

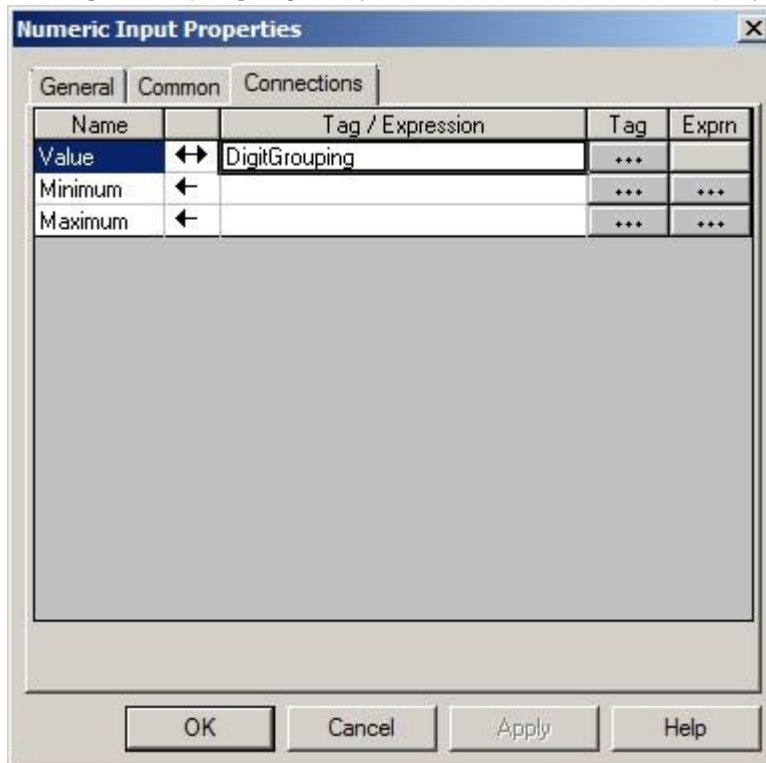
14. To the right of the **Numeric Display** object, click and drag the mouse to draw the **Numeric Input** object and then release the mouse button to create the object.

The **Numeric Input** properties dialog will appear.

15. Set the **Field Length** to 11.

16. Select the **Connections** tab.

17. Press the ... button under the **Tag** column for the **Value** connection and browse and select the **DigitGrouping** tag like you did for the Numeric Display object.



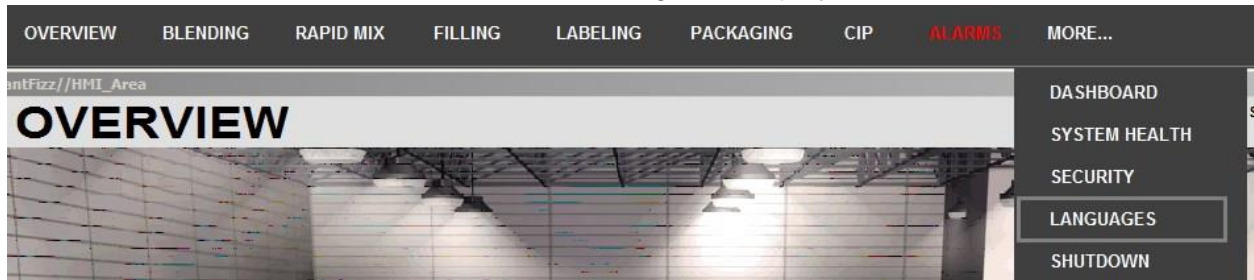
18. Press **OK** to save the dialog changes.

19. **Close** and **save** the languages display.



Let's see how this **Digit Grouping** feature looks at runtime.

- Switch to the running **client** window and click on the **MORE...** button on the right side and then the **LANGUAGES** button in the navigation display.



- In the Numeric Input object, enter the number **123456789** and press the **ENTER** key.

Notice that the Numeric Display object with **Digit Grouping** has commas to separate the digits. Also notice that the Numeric Display object without **Digit Grouping** has no separator character in the digits.

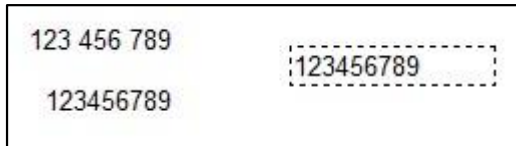


Now, let's see how switching the language affects the **Digit Grouping** display.

- Press the **French** button to switch the application to use the French language.

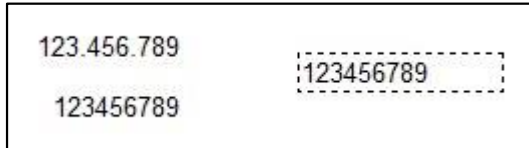
Notice that the Numeric Display object with **Digit Grouping** has spaces to separate the digits. This is the French standard.





23. Press the **German** button (third button from left) to switch the application to use the German language.

Notice that the Numeric Display object with **Digit Grouping** has periods to separate the digits. This is the German standard.

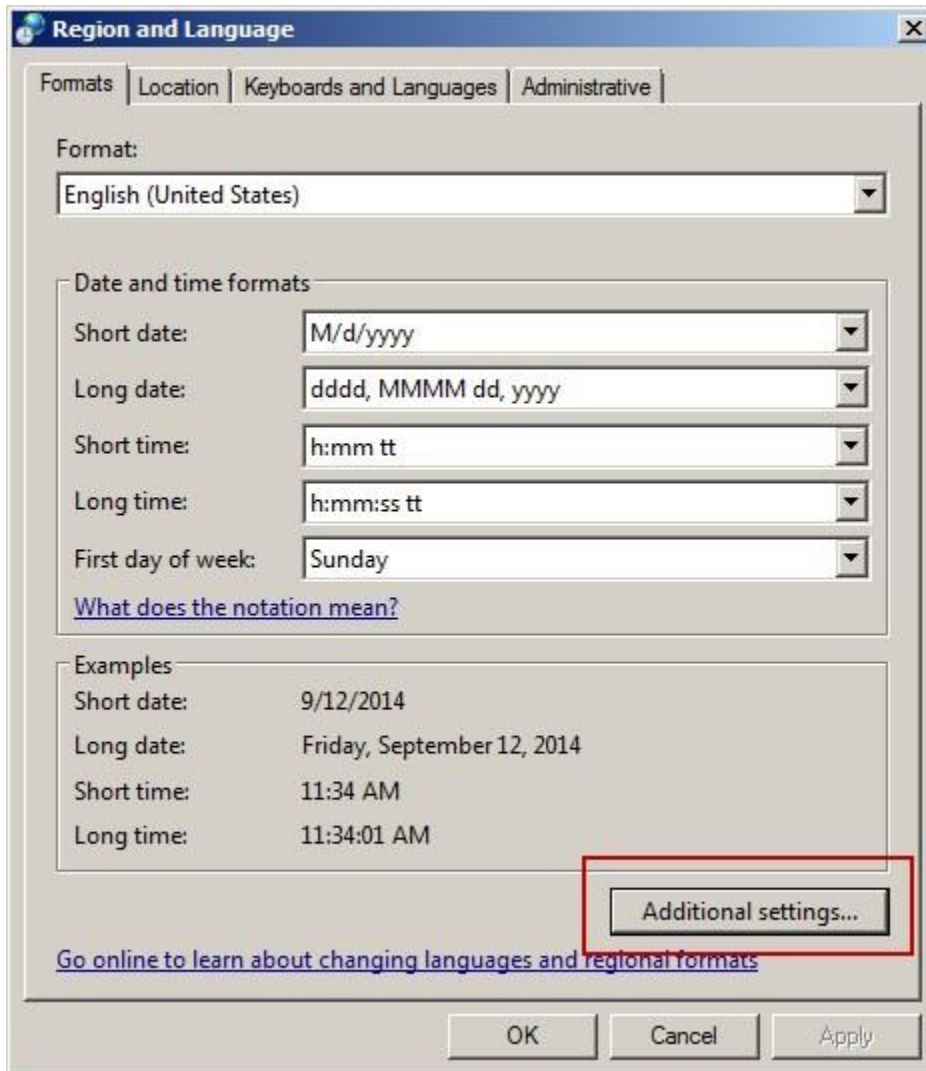


24. Press the **English** button (left most button) to switch back to English and then close the display.

Now, we're going to see how the different languages define the different standards for **Digit Grouping** and what character is used to separate the digits. This is defined in the Windows operating system.

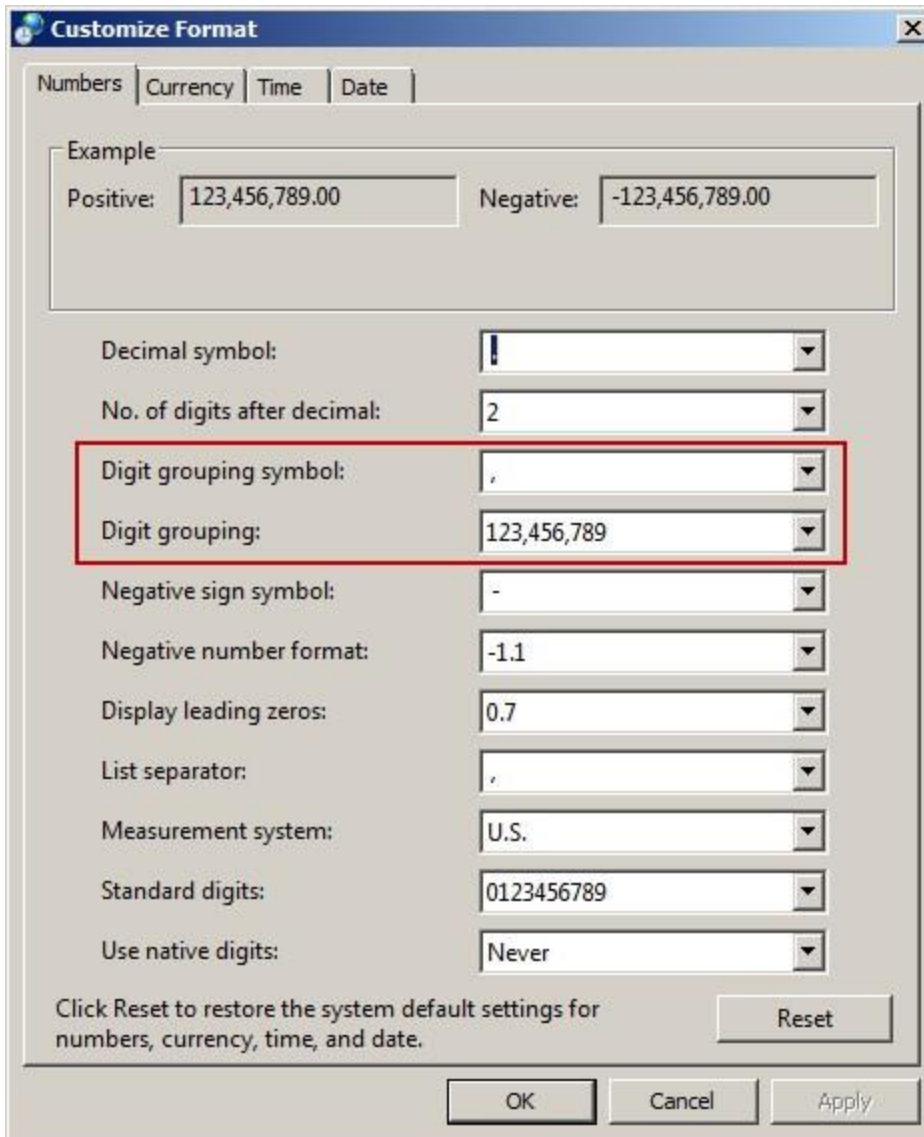
25. From the Windows **START** menu, select the **Control Panel > Region and Language** option.

This will open the region and language options for English (United States).



26. Click on the **Additional settings...** button.

This will display the **Digit Grouping** format and the **Digit Grouping symbol** used to separate the digits.



27. Press **Cancel** to close the dialog.

28. Explore the **Digit Grouping** for other languages by selecting a different language in the **Format** list at the top of the dialog and then opening the **Additional settings...** button again.



29. When you are done exploring, press **Cancel** on both these dialogs so that the default language for the computer remains at English.

---

## Section 4: Data logging and trending

### About this section

In this section, you will learn about:

- Runtime Exploration
  - Data Logging
  - Trending
- Configuration
  - Exploration
    - Historical
  - Trending
    - Runtime
  - Trending
    - Data
  - Log Models
- Configuring
  - Methods using the Invoke command

You will be using the **cip\_sensortrend** display and CIP data log model.

### Runtime exploration

#### Data logging

Data logging is a FactoryTalk View component that collects and stores tag values. You specify which tag values to collect, when to collect them, and where to store them by defining a data log model.

The HMI tag database does not store actual tag values; it only defines which values are to be collected. The values themselves are stored on a given HMI server. When the HMI server is turned off, the value table is cleared (excluding retentive tags). To have a permanent record of tag values, log them to the data log file on disk. Remember: controller tags will keep their value when an HMI server is powered down because they're located within the controller; they're not HMI (or memory) tags stored on the server itself.

To log tag values to disk, you create a data log model and specify the tags that are to be logged. This is done in the Data Log Model editor. The values can also be logged to an ODBC-compliant database.

An application can have up to 20 data log models running at a time. The maximum number of tags that can be logged by one data log model is 10,000.

#### Trending

A trend is a visual representation, or chart, of current or historical tag values. A trend provides operators with a way of tracking plant activity as it is happening. The trend object displays real-time data and historical data from the FactoryTalk View Site

Edition data logs. Pens on the run-time chart represent data from the tags and expressions that you add to the trend object. The trend object provides extensive, flexible run-time control. You can add pens,

toggle between isolated and non-isolated graphing, specify unique line settings, plot one variable against another in XY plots, and print chart data.

There are two different types of Trending:

### Historical Trending

- Historical Trending is when a trend polls data from a data log that is previously configured. This will allow the user to browse through a timeline to look at the data over a given period of time.

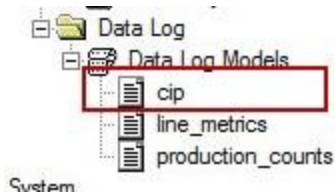
### Runtime trending

- A Runtime trend displays data trends directly from the processor. The trend will start trending its runtime data from the time it's first loaded on the display.

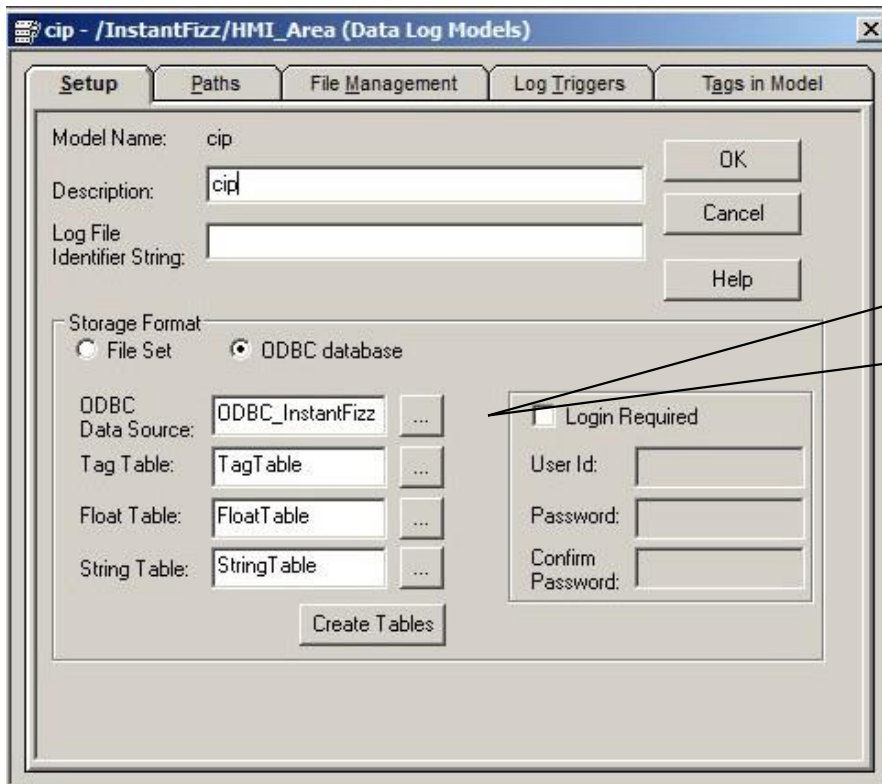
## Configuration exploration

### Data log models

1. Double-click the **cip** datalog model to open.



The **CIP Data Log Model** dialog opens to the **Setup** tab.



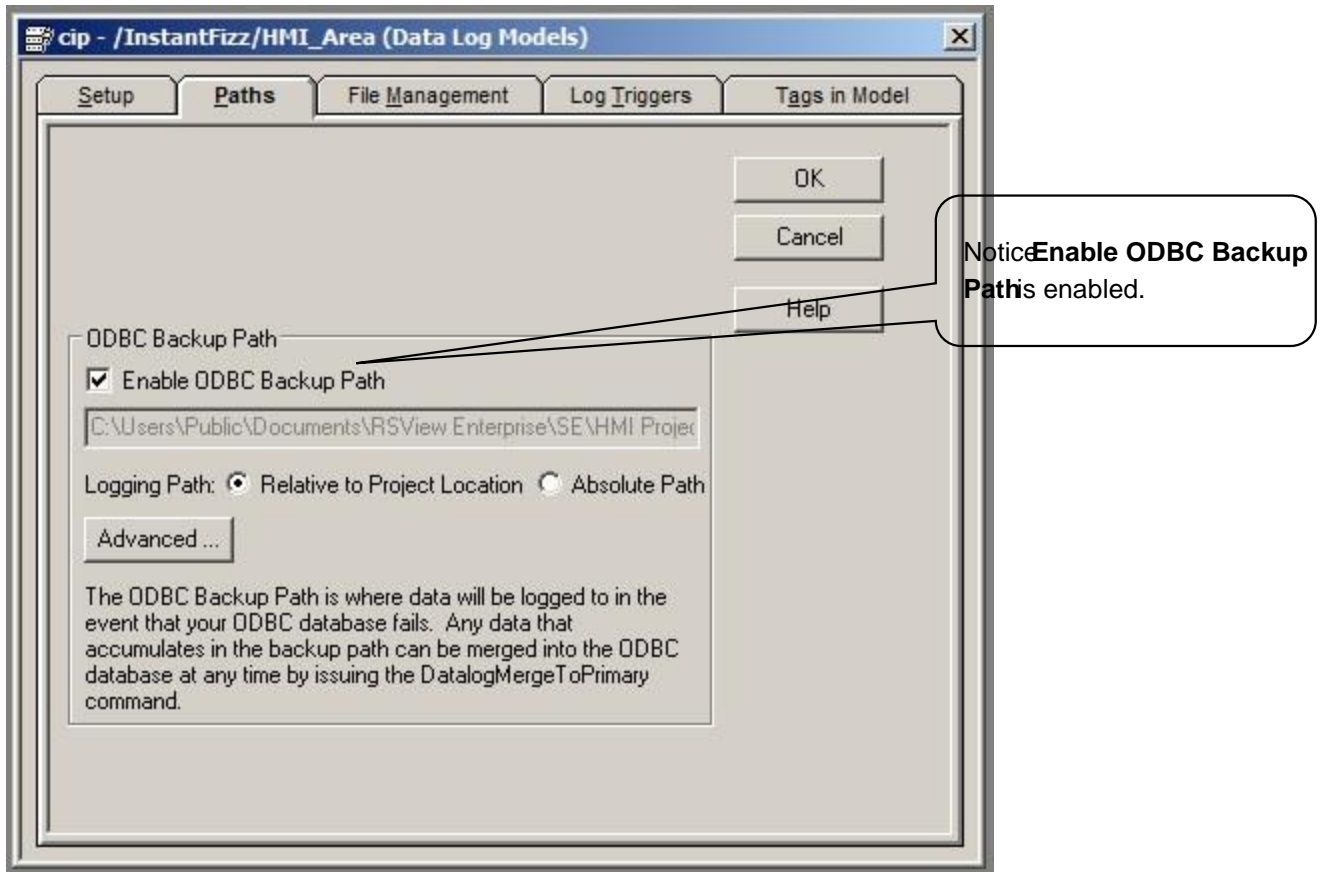
You can set up logging to log to a file set or to log to any database that you can connect to with ODBC.

The data log model can be configured to be stored as a file set or an ODBC database. Notice that this data model is being stored in an ODBC database. There is a System Data Source Name (DSN) called ODBC\_InstantFizz that points to a Microsoft Access database called ODBC\_InstantFizz.

An ODBC System DSN stores information about how to connect to the indicated data pointer. If you would like to see how the System DSN was configured, double click **C:\Windows\SysWOW64\odbcad32.exe**. The ODBC Date Source Administrator window will open. Select the System **DSN tab**. Double click the **ODBC\_InstantFizz** System Data source to see how the System DSN was setup.

2. Click the **Create Tables** button if you haven't already done so. You should get a FactoryTalk View Datalog Editor message saying ODBC tables were successfully created. Click the **OK** button.
3. Click on the **Paths** tab.





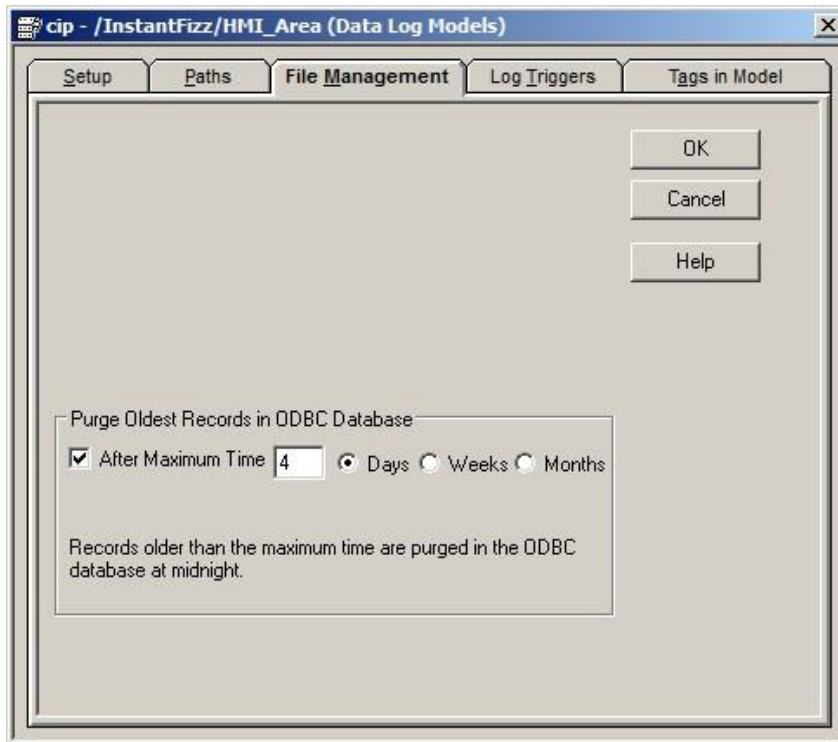
This Paths tab is used to set the file location of the data logs. This is useful if you have a distributed network set up; you will want to store the logs to one common location instead of on each individual HMI server on the network.

Notice that the **Enable ODBC Backup Path** was selected and a logging path was specified. If the connection to the ODBC database fails (Microsoft Access) you will be able to see the historical data in the trend display because of the secondary Backup Path.

Click the

4. **File Management** tab.

This configuration tab is used to configure how long you want to keep logging to individual files until you either create a new file or delete older files.

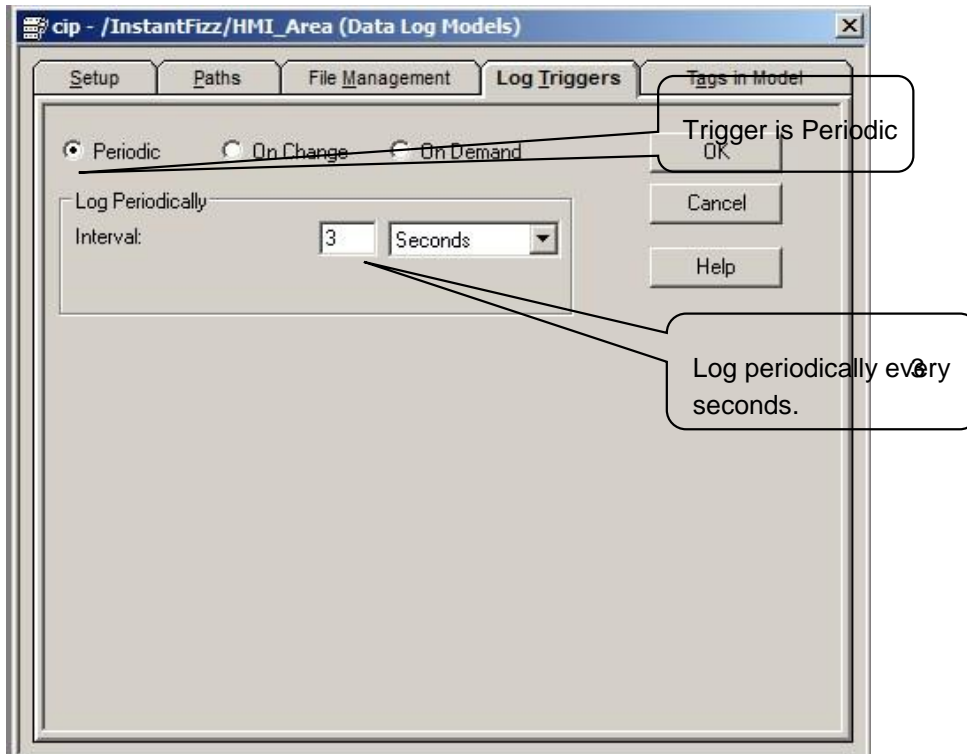


5. **Log Triggers** tab.

This configuration tab is used to decide how often and based upon what event the log should be updated.

Notice that the trigger for logging is **Periodic**.

Click the



The data log can be configured to log Periodically, On Change, or On Demand.

### **Periodic**

To log tag values periodically, select Periodic, then type a time in the Interval field and select a time unit to specify how often tag values will be logged. All tags will be logged each time this interval expires.

### **On Change**

On Change logging only logs tags whose values have changed. Use the On Change trigger to log tag values once a certain percentage of change in the value has occurred. The percentage is based on the tag's minimum and maximum (or High EU and Low EU) values. For example, specifying 10 means a tag's value must change by 10 percent to be logged. This applies to analog HMI tags only. For data server tags and digital and string HMI tags, which do not have minimum/maximum properties, every change is logged.

### **On Demand**

Choose On Demand as the trigger, to log data only when the FactoryTalk View command DataLogSnapshot is issued. This command can be given anywhere that other FactoryTalk View commands and macros can be. For example, it could be typed in the command line, or specified as the action for an event. If you specify Periodic or On Change, you can still use on demand logging whenever it is appropriate.

6. Tags in Model tab.

Click the

This configuration tab is used to select what information is logged. You can add or remove different tags from this menu using the tag browser.

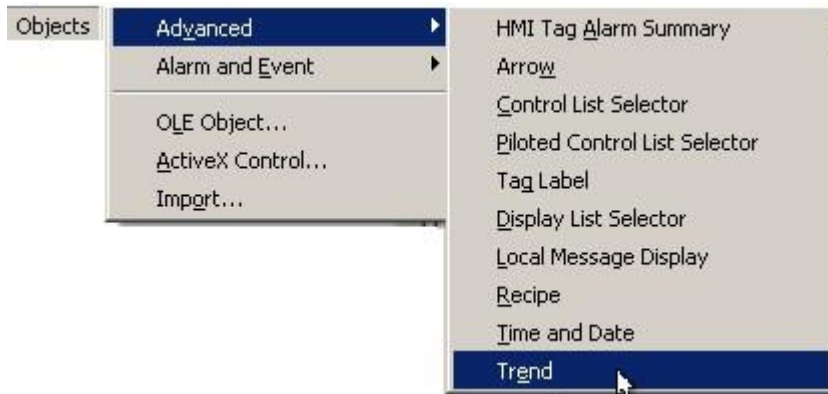
Notice 5 tags have been added to the model.

7. Click the **CANCEL** button to close the **CIP Data Log Model** dialog.

## Configuring trends

**Do not** add a new trend or change this one. You can do so at the end of this section.

1. From the **Explorer**, double-click on the **cip\_sensortrend** display to open it.  
A new Trend object can be created by selecting the **Objects > Advanced > Trend** menu item



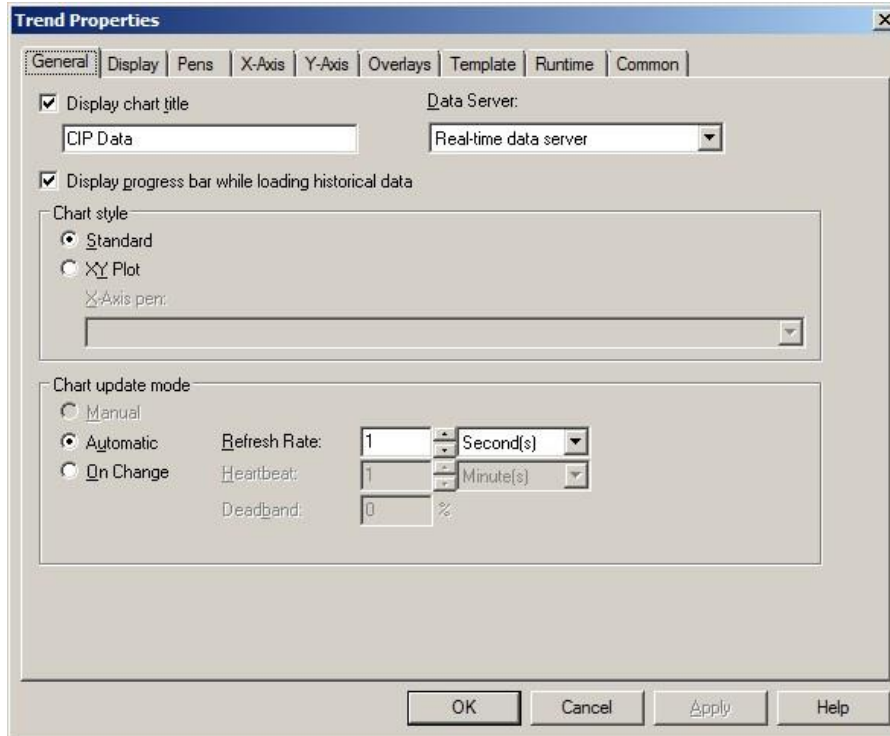
or by selecting the **Trend** button on the menu bar



2. Double-click on the grid of the existing **Trend** object on the display.  
The **Trend Properties** dialog will open. Let's get familiar with the trend properties and capabilities.

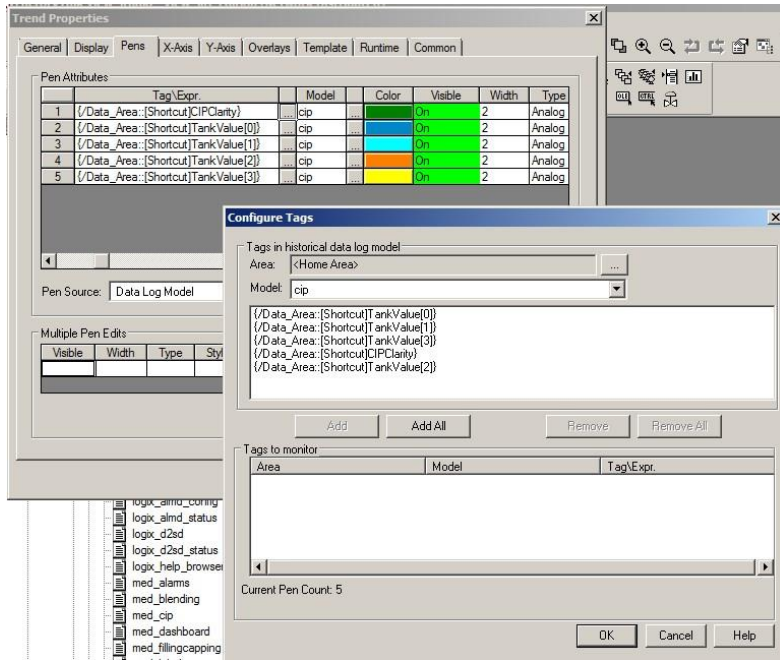
3. Click on the **General** tab.

The general tab contains several configurable properties. The Chart Style determines the plotting style of the chart. A trend chart can either plot values against time or against a selected pen.



4. Click the **Pens** tab.

Notice that the **Model** column has **CIP** which is the **data model** that was created.



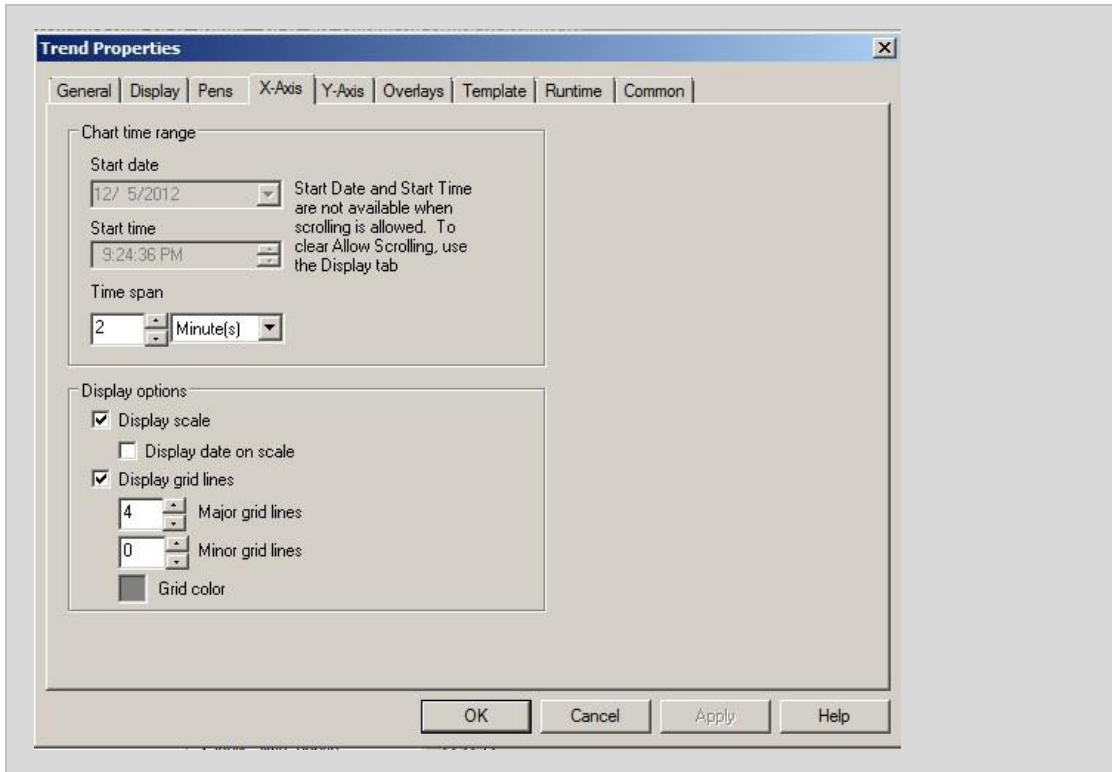
These pens were added by setting the Pen Source to 'Data Log Model' and then selecting the Add Pen(s) button. The Configure Tags dialog opened. The Add All button was selected to add all the tags. Then the OK was clicked to accept the changes.



Click on the

5. **X-Axis** tab.

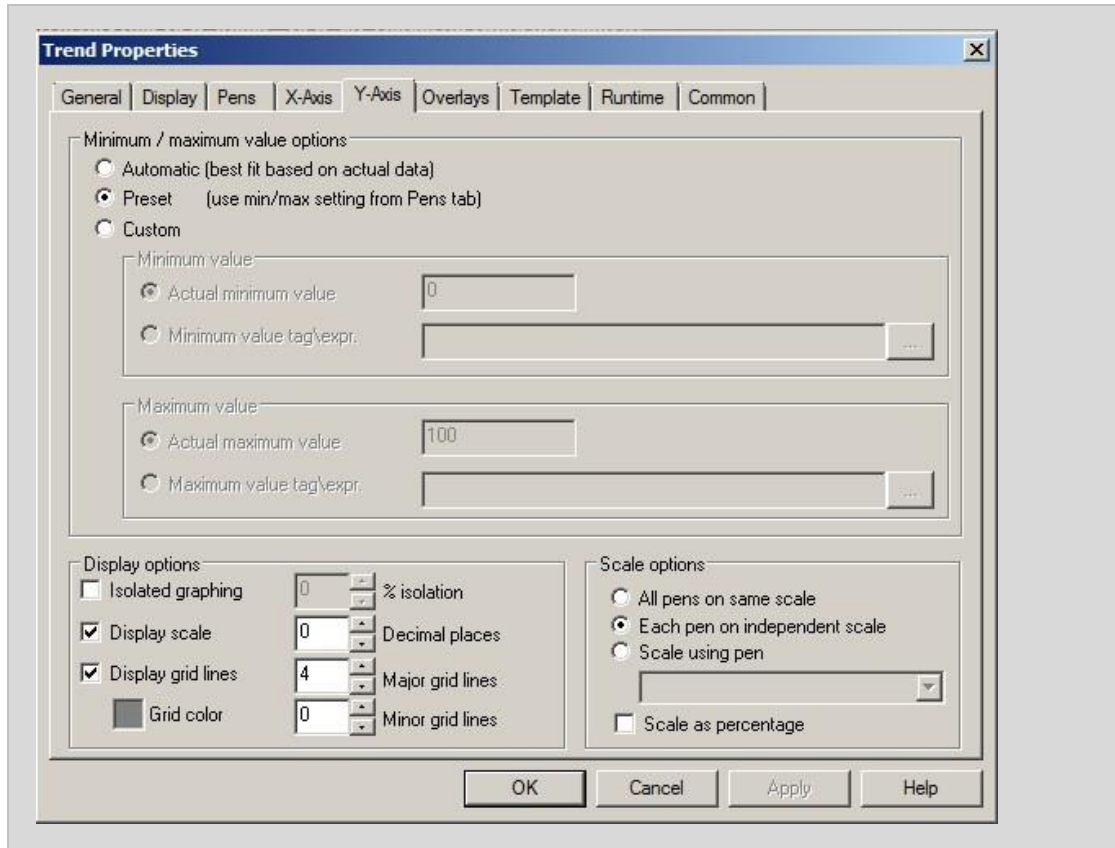
Use this tab to set up the time range and display options for the chart's run-time horizontal axis. The Time Span controls the amount of data that displays on the run-time chart. For a standard line chart, the time span controls the chart's horizontal scale. For an XY plot, the Min and Max properties (set up on the Pens tab) of the selected pen control the horizontal scale, and the time span controls the number of data points plotted.



6. **Y-Axis**.

Use this tab to set up the minimum/maximum value options, display options, and scale options for the chart's run-time vertical axis. For example, **Isolated graphing** places each pen in a separate band of the chart. To allow pens to overlap, you would need to clear the check box.

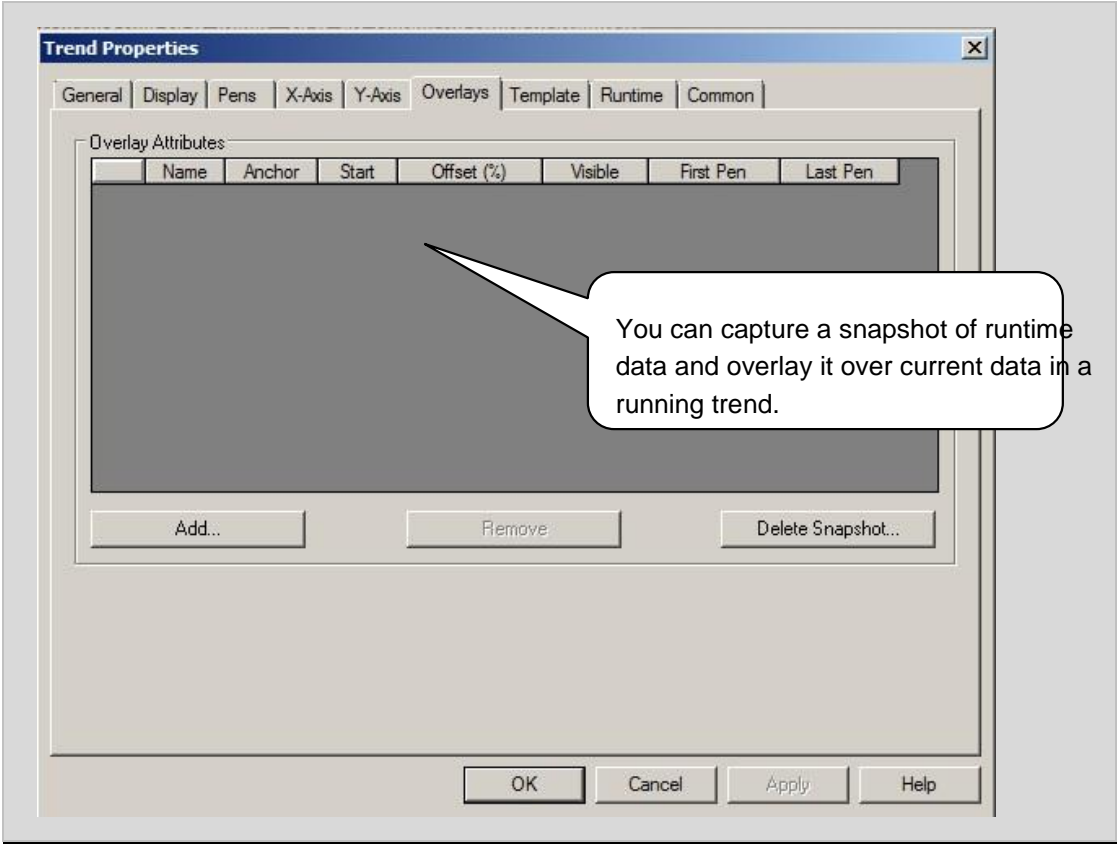
Click on the



## 7. Overlays

With the trend object, you can capture and save a snapshot of graphed data at run-time and then layer that saved data over current data in a trend. This feature allows you to overlay and compare historical data with current data or any multiple sets of data.

Click on the **Overlays** tab.

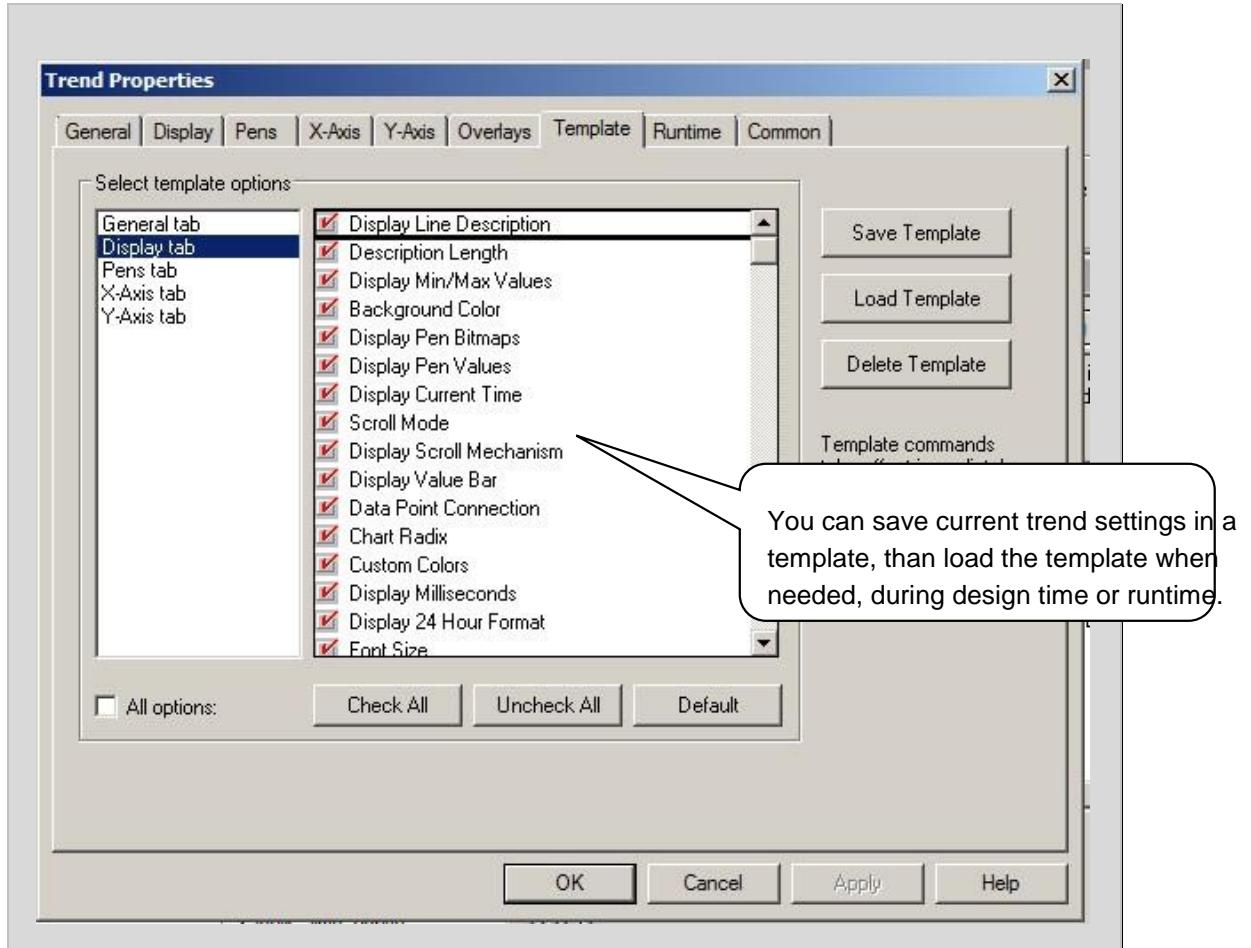


Click on the **Template** tab.

## 8. **Template**

Use this tab to save the settings for the current trend as a template, to load another template for the current trend to use, or to delete a template. The settings that are saved for the template are those from the General, Display, Pens, and X and Y Axis tabs.

A template can be applied to all trends in an application to create a consistent appearance. You can load templates during design time and runtime.

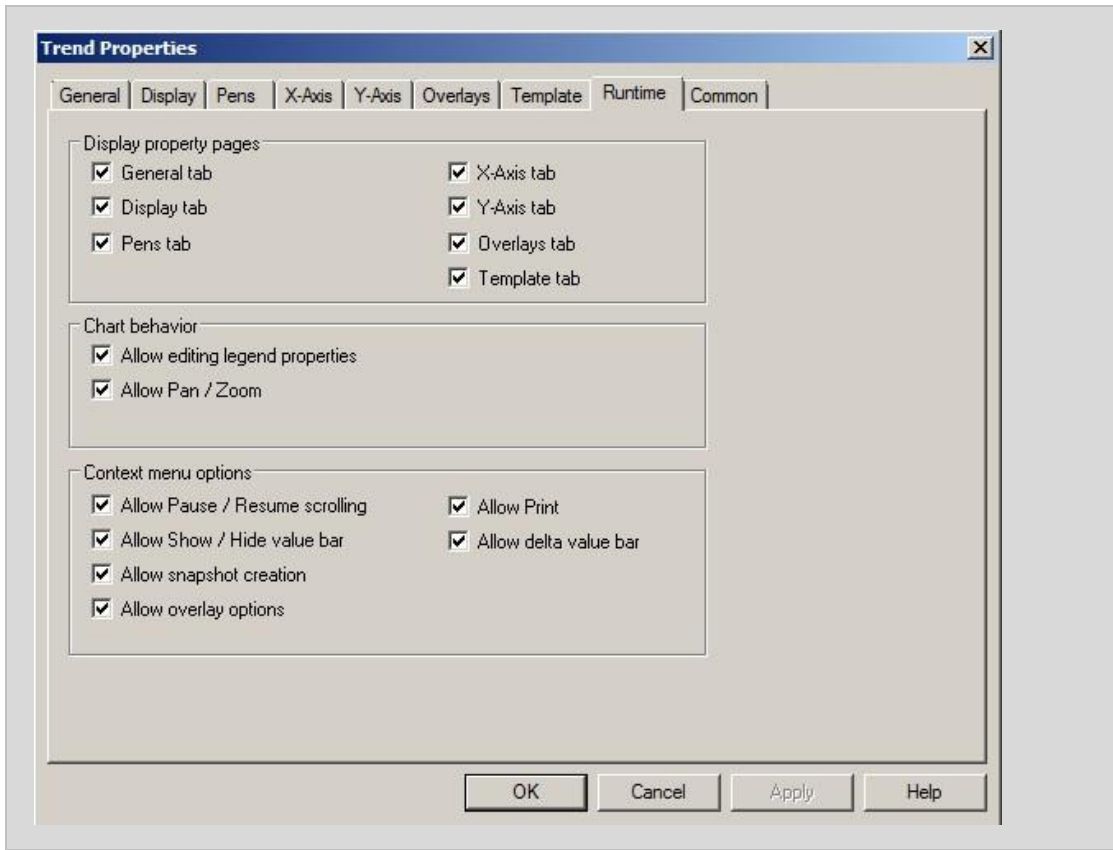


We will import several pre-created templates into the application. But first we will complete reviewing the Trend properties and capabilities.

## 9. **Runtime**

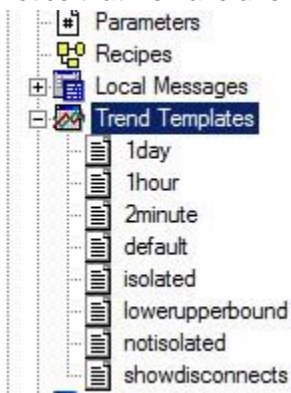
Use this tab to determine which trend options are available to operators at run time. To prevent operators from changing these options, the Runtime tab is not available at run time.

Click on the **Common** tab.



10. Click the **Cancel** button on the **Trend Properties** dialog to close the dialog.

Notice that we have a few **Trend Templates** already added to the project.



11. In the **cip\_sensortrend** display, right-click on the **Start** group object to display the context menu and then select the **Animation > Touch** menu to open the Animation properties.

12. The Release Action command **DataLogOn cip** starts the data logging for the model.



The datalogon command can be placed in a start-up macro as well. When the client is launched, the datalogging is started automatically from the start-up macro.

13. Notice the **Confirm Action** check box is checked.



You can choose to show a confirmation popup dialog box and configure the appearance when executing commands or actions on buttons, numeric inputs, string inputs, or objects with touch animation. The benefits of using this feature:

- Enable operations to confirm they want to take requested action before it occurs.
- Remove the need to use a display graphic to perform this native action.

14. Click the **Configure** Button.

The Confirmation configuration dialog box opens up.



Notice that there are several properties here that can be customized in this dialog box.

**Title Bar** - Select this check box to show the title bar of the confirmation dialog box. You can enter up to 255 characters.

**Insert Variable** - Adds an embedded variable to the title bar. Click Insert Variable and then select the type of variable to be inserted in the caption. In the Variable dialog box, specify the details. After you have set up the embedded variable, the syntax appears in the Title Bar field.

**Window position** - Select a position where the confirmation dialog box will be shown.

**Caption** - Enter the message that will be shown in the confirmation dialog box. If you have added an embedded variable to the caption, the embedded variable syntax appears in the Caption field, but the text of the variable appears on the dialog box.

**Insert Variable** - Adds an embedded variable to the message. Click Insert Variable and then select the type of variable to be inserted in the caption. In the Variable dialog box, specify the details. After you have set up the embedded variable, the syntax appears in the Caption field.

**Font** - Select a font for the message, or click the browse button (...) to open the standard font dialog box that lets you select a font and specify how it is to look (except for font color).

**Size** - Select a font size from the list.

**Image settings** - You can choose a pattern and color for the confirmation dialog box. You can also import a bitmap or icon to be displayed on the dialog box.

**No image** - Select this if no bitmap, jpeg, png, or icon is to be displayed on the dialog box.

**Use image reference** - Select this to use an image from the current HMI project.

**Image** - Enter the name of the image file or click the browse button (...) to select an image to reference using the Image Browser. If you add an image to the Image Browser, it is also added to the Images folder in the Explorer.

**Import file** - Select this and then click Import to locate and select a bitmap or icon to display on the confirmation dialog box. The imported graphic is placed in the center of the dialog box and saved with it. The caption is displayed on top of the imported graphic. This does not add the image to the Images folder in the Explorer.

**Scale image** - Select this check box to scale the image to fit within the dimensions of the dialog box. This applies to both referenced and imported images. The recommended dimensions are 48 x 48 pixels.

**Button Settings** - Select buttons that will be shown in the confirmation dialog box, including OK and Cancel, OK only, Yes and No, and Yes only.

15. Notice the **Caption** property that has a custom text that will be displayed in a dialog box at runtime in the client to the user with the buttons **OK** and **Cancel** that are clickable.
16. Click the **Cancel** button to close the Confirmation properties.
17. In the Animation properties window, notice that the **Touch Style** is specified as Irregular.



Since we are using a circular object to execute our DataLogOn command with touch animation, we would like for the touch area to be circular as well. An irregular touch style is selected when the object outline is the desired touch area versus the default rectangular bounding box.

18. Click the **Close** button to close the Animation properties.

We will now configure the Confirmation pop up box for the Stop button.

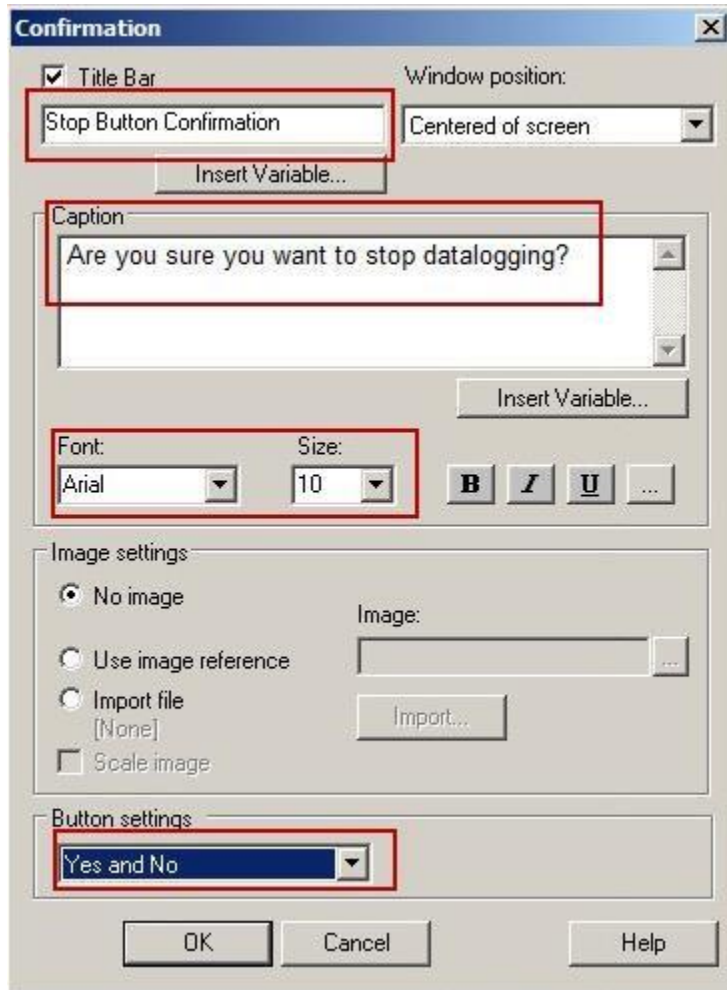


17. Right-click on the **Stop** group object and navigate to **Animation > Touch** menu to open the Animation properties.
18. Check the **Confirm Action** checkbox.



19. Click the **Configure** button.

20. Configure the Confirmation window as shown below and Click **OK** to close the dialog.




21. Click **Apply** then **Close** to save the animation changes.

22. Close the **cip\_sensortrend** display. When prompted to save changes, select the **Yes** button.

### Viewing the trend at runtime

1. Go to the running client. If you have any open pop-up displays, please close them. If

you do not have a running client, click on the  to launch the client.

2. Click the **CIP** button from the navigation display. Then click on **CIP Trend** button in the top right corner .

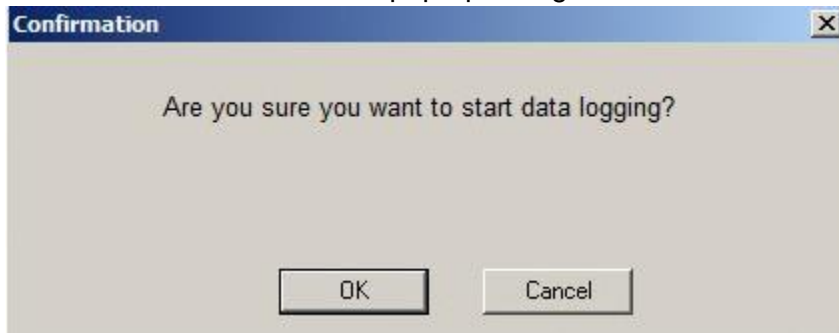
The **cip\_sensortrend** will be displayed.

3. Hover the cursor over the **Start** button and notice that the highlight is confirming that we have an irregular touch area configured since it is the shape of the object and not a rectangular box.



Click the **Start** button to start historical logging.

4. You will see the Confirmation pop-up dialog.



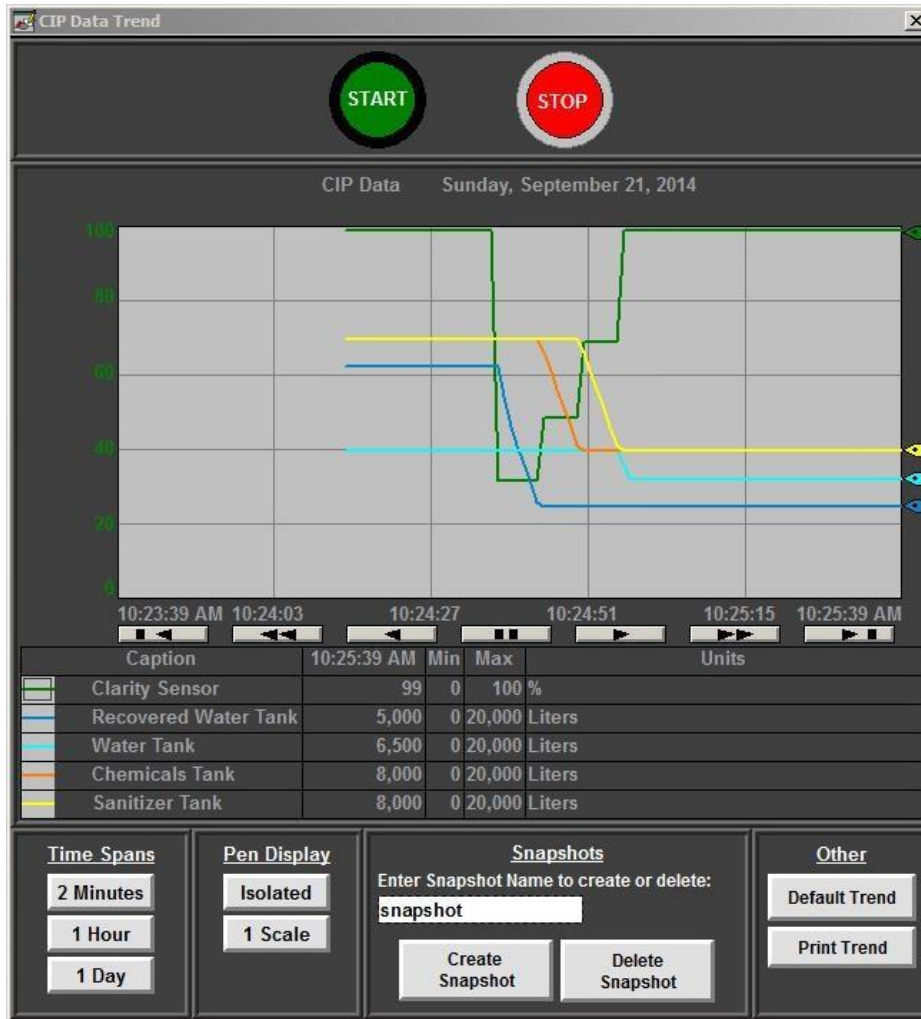
Click on the **OK** button to start logging historical data.

5. While the **cip\_sensortrend** display is open, click on the **CIP – Repeat Cycle** button



on the **CIP** display to start the CIP clean.

6. The CIP data is being logged. Observe the trend in the **cip\_sensorTrend** display. To view the data of another CIP cleaning cycle, click the **CIP – Repeat Cycle** button again.



- Earlier we have executed the DataLogOn command with the START button. The trend historical data is stored in the Microsoft Access database. If we close the trend and then re-open it, we will still be able to view our CIP cycle from a few minutes ago. You can try it. Click the close button on the **cip\_sensortrend** display by clicking the **X** button. Then re-open the trend display from the **CIP** display. You may need to **Scroll back** on the trend to see the CIP cycle data.



- The **cip\_sensortrend** display contains several buttons to demonstrate trend functionalities and capabilities.



9. As seen earlier, the trend can be configured to display data in different time spans, for example 2 Minutes, 1 Hour, 1 Day etc. To view the different time spans at runtime, the trend configurations can be saved to templates, such as the templates we had imported earlier. The templates can then be loaded at runtime using the **Invoke** command.
10. Go back to **FactoryTalk View Studio** and open the **cip\_sensortrend** display (if it is not already open). Then double-click on the **2 Minutes** button.
11. The **Button Properties** window will open. Select the **Action** tab.  
**Invoke** command is used to launch the template called '**2Minute**' at runtime.



12. Click the **Cancel** button to exit the button properties.
13. Go back to the running client. Click on the different **time spans buttons** to see the trend object **X-Axis** change.



14. You can also use templates to switch between different pen views: Isolated pens or all pens on one scale. Click on the **Pen Display buttons** to see the pens change.

Pen Display

Isolated

1 Scale



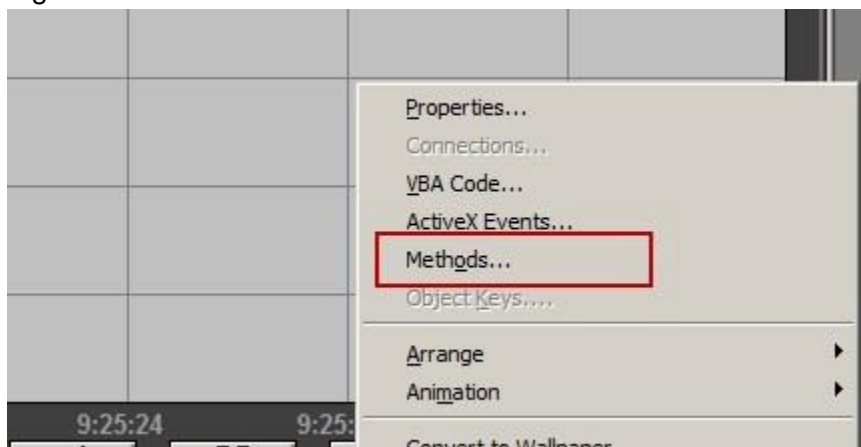
Isolated pens. Each pen has a separate trend band. Notice the Y-axis.



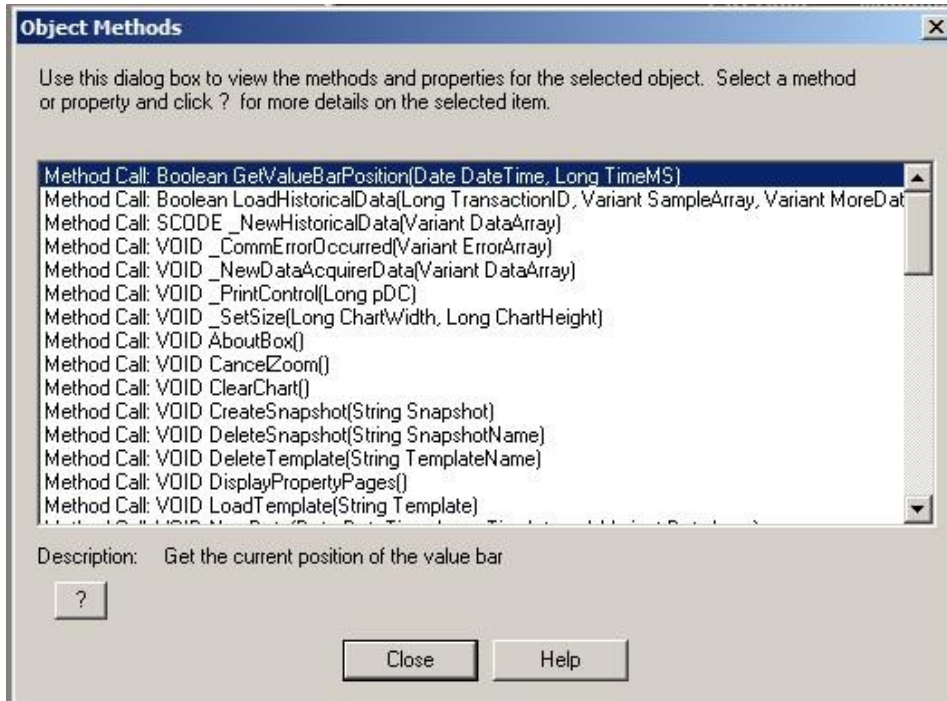
All pens on 1 Scale. Notice the Y-axis.

Many objects in FactoryTalk View SE have **Object Methods** for additional object functionality. You must use **Invoke** commands to call a **Method**. Let's explore some **Object Methods** for the trend object.

15. Go back to **FactoryTalk View Studio** and the **cip\_sensortrend** display.
16. Right-click on the trend and select **Methods**.

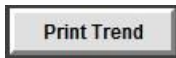


The **Object Methods** dialog box appears. It displays methods and properties implemented in the selected trend object. A method is a function that is part of an object.



17. Explore the different methods available and then click **Close**.

18. Let's see an example where the Object Methods can be used. We have already used the **LoadTemplate** method to load our Time Span templates. Double-click on the **Print Trend** button



on the **cip\_sensortrend** display.

19. In the Button Properties, select the Action tab.

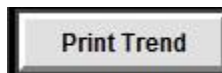
The **Invoke** command was used to launch the **PrintChart Method**.



Always use the Invoke command to call a method. When you use the Invoke command to call a method, you need to specify the name of the object in which the method is implemented. In our case the name of the object is **Trend1**. For help on how to use the Invoke command, go to FactoryTalk View Help and search for Invoke.

20. Click **Cancel** to exit the Button Properties.

21. Let's see our **PrintChart** method in action. Return to the running client. On the **cip\_sensortrend** display click



on the **Print Trend** button.

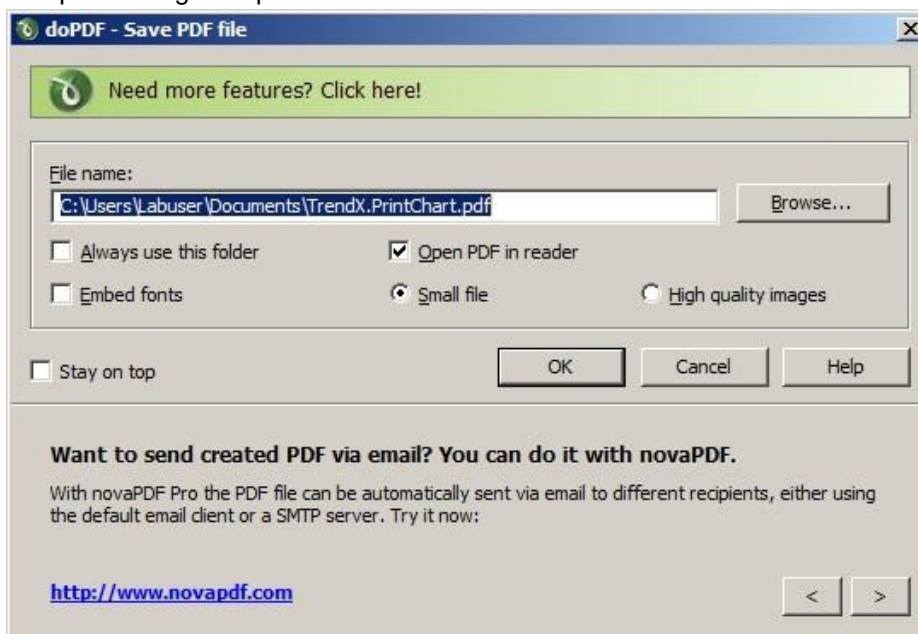
22. The **Print** properties will open. In this example, we are printing to a .PDF file – select the **doPDF v7** printer.





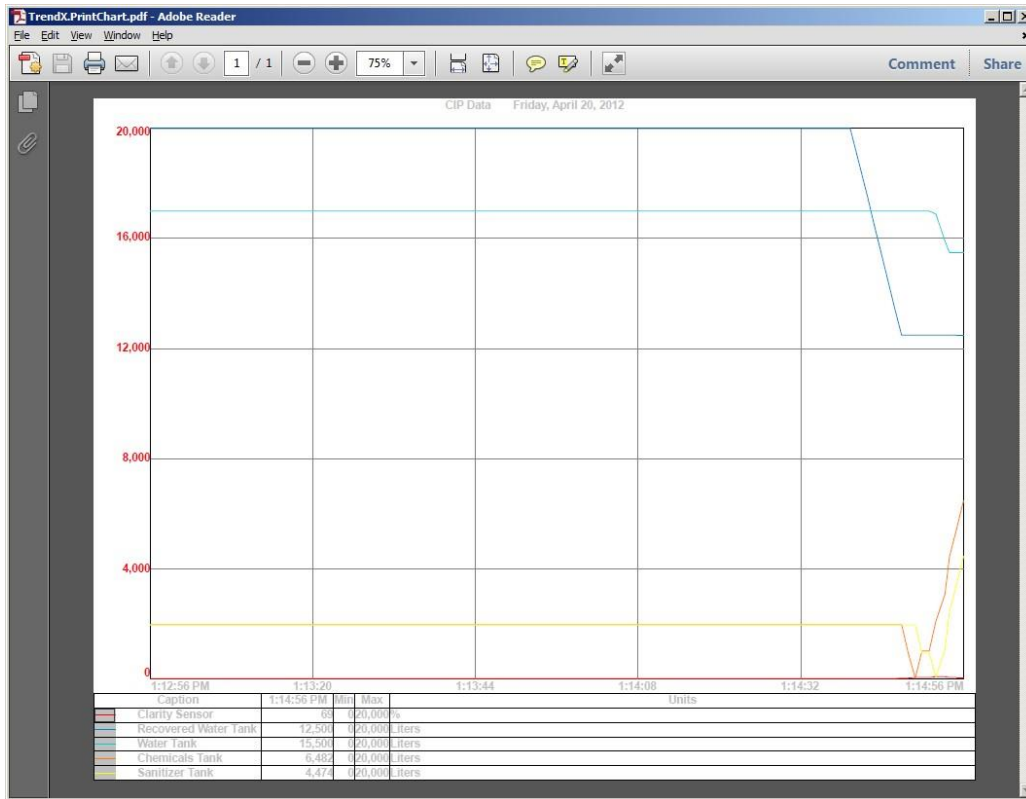
23. Click **OK**.

The print dialog will open.

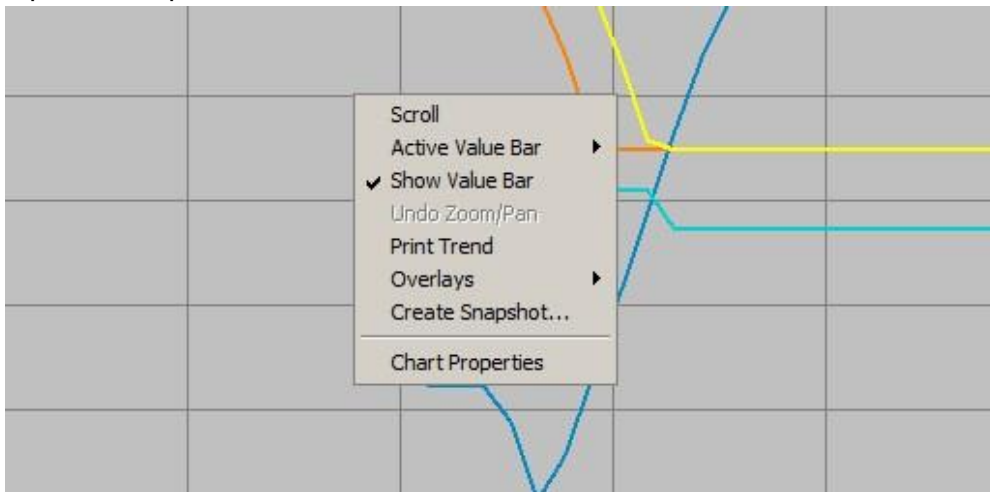


24. Press **OK**. This will save and then open the PDF file in **Adobe Reader**. Minimize the Adobe Reader window when done.





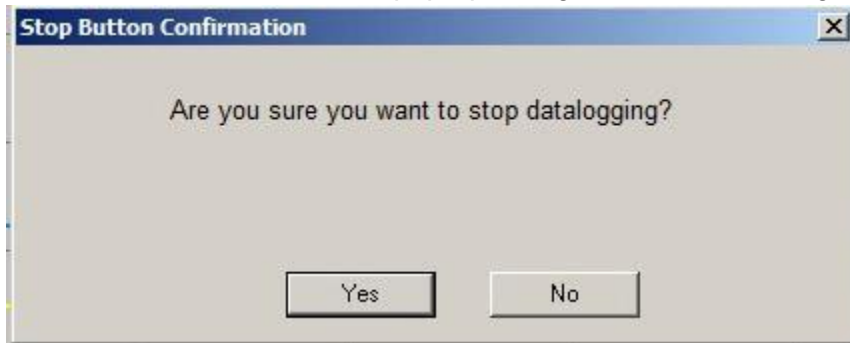
25. As discussed in **Configuring Trends** section, many of the trend properties can be configured to be accessible during run-time. Right-click on the trend in the client and explore the options.





26. We will now finally stop the data logging by clicking on the **Stop** button on the CIP Data Trend screen.

27. You will see the Confirmation pop-up dialog box that was configured earlier:



28. Click the **Yes** button to stop logging data and close the CIP Data Trend screen.

Congratulations! You have learned how to set up data logging and trending and understood the different properties of the two features at design time and runtime. We also learnt how to configure Methods using the Invoke command.

## Section 5: FactoryTalk Alarms and Events

### About this section

In this section, you will be doing the following:

- Learn about types of Alarming Systems
- Learn about FactoryTalk Alarms and Events components
- Subscribing to Alarms and Events
- Enabling Alarms and Events on an existing data server
- Testing the Alarms and Events connection using the default alarm summary object in FactoryTalk View Studio.
- Exploring RSLogix 5000 alarm instructions: ALMD and ALMA
- Acknowledging an alarm from RSLogix 5000
- Learning about Associated Tags
- Embedding variables in alarm messages
- Exploring Tag-based Alarms and Events
- Viewing alarms as tags

- 
- Changing tag update rates
  - Acknowledging alarms from the alarm summary object
  - Acknowledging and shelving alarms remotely
  - Viewing historical data and creating filters

## Introduction to alarming

Alarms are an important part of plant control applications because they alert operators when something goes wrong. Often, it is also important to have a record of alarms, when they occurred, and if they were acknowledged and by whom.

### Types of Alarm System

FactoryTalk View SE supports the existing traditional HMI tag alarm system and the FactoryTalk Alarms and Events system. The choices you make will depend on factors such as the design of your application, the processes you need to monitor for alarms, the types of devices used in the application, and whether you want to build alarm detection into those devices.

### Traditional HMI Tag Alarm System

In FactoryTalk View Studio, you can set up a complete alarm system. At run time, alarm monitoring occurs at the HMI server. If an alarm is detected (tag values outside the configured limits), a notification is sent to the connected FactoryTalk View SE clients, where operators can view and acknowledge the alarms. This is a traditional HMI tag alarm system.

The HMI tag alarm system only detects alarms set up for tags in an HMI server's tag database. HMI tag alarm detection does not include FactoryTalk alarms. FactoryTalk View SE Clients receive HMI tag alarm information by way of the FactoryTalk View SE Servers (also called HMI servers) that contain the HMI tags.

Traditional HMI tag alarm data is not managed by FactoryTalk Alarms and Events services. To monitor and respond to HMI tag alarms, you must use the HMI tag alarm displays and logs available in FactoryTalk View SE. This section of the lab will **not** demonstrate any features of the traditional HMI Tag Alarm system.

### FactoryTalk Alarms and Events system

FactoryTalk Alarms and Events allow multiple FactoryTalk products to participate together in a common, consistent view of alarms and events throughout a FactoryTalk application.

#### ***FactoryTalk Alarms and Events supports two types of alarm monitoring:***

- **Device-based alarm monitoring.** Built-in alarm instructions that are available in RSLogix 5000 v.16 or later, are programmed in a logic project and then downloaded to a Logix5000 controller. The controller detects alarm conditions and publishes event information, which can be displayed and logged.
- **Tag-based alarm monitoring.** If you are not using Logix5000 controllers, or if you do not want to use the built-in alarm instructions that are available with RSLogix 5000, tag-based alarm monitoring offers the equivalent of HMI tag alarm monitoring, but with an expanded feature set. Software-based Tag Alarm and Event servers monitor controllers for alarm conditions through data servers and publish event information that can be displayed and logged. Tag-based alarm monitoring is supported for Logix5000 controllers, PLC-5, and SLC 500 devices communicating through Rockwell Automation

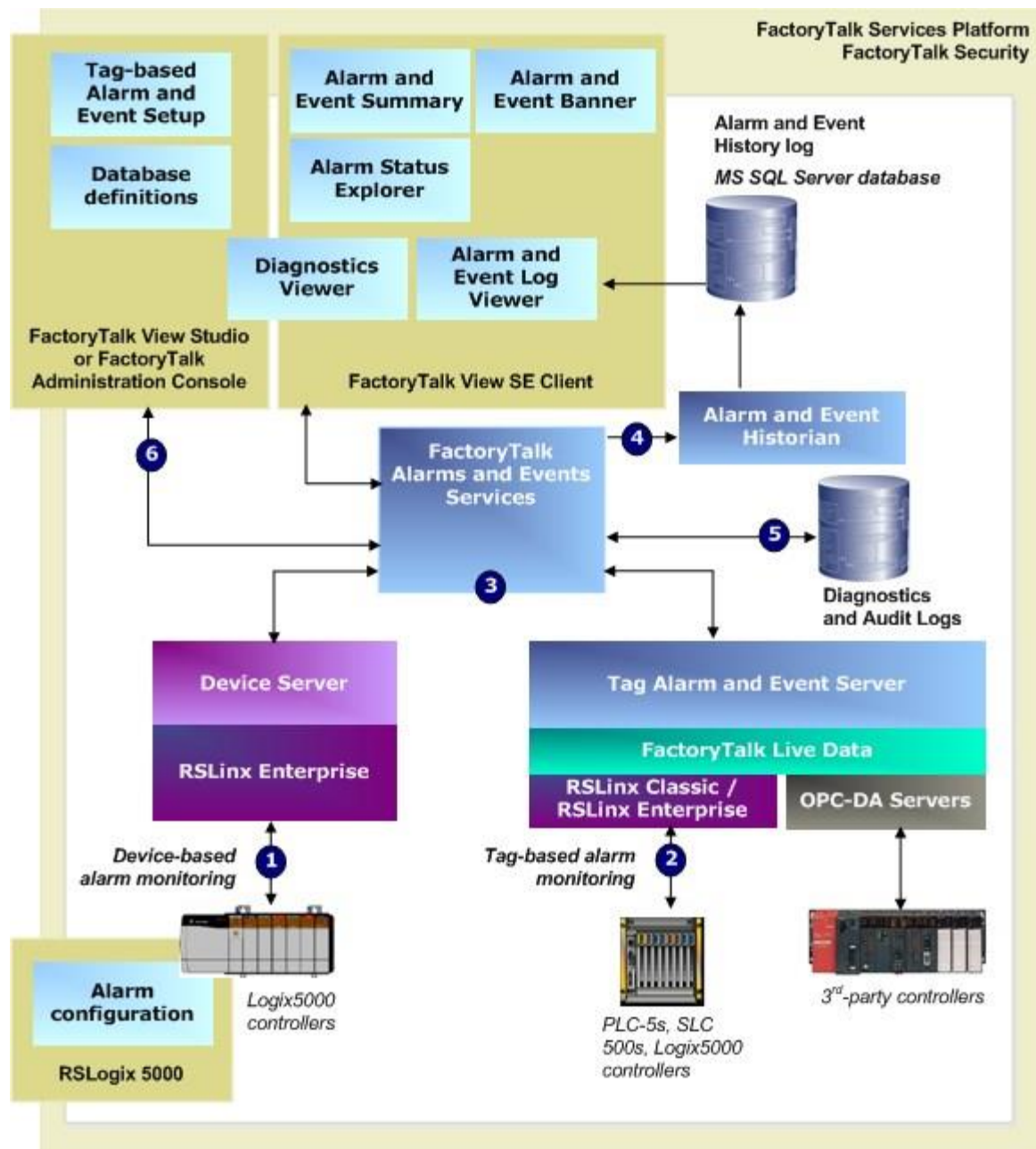
device servers (RSLinx Enterprise), or for third-party controllers communicating through OPC Data Servers.

#### **Features of FactoryTalk Alarms and Events:**

- **Provides a single, integrated set of alarm information.** All participating FactoryTalk products work together to provide a consistent way to define, manage, log, and view alarm and event information across a FactoryTalk application.
- **Streamlines alarm programming and eliminates polling with device-based alarm monitoring.** If your automation system includes Logix5000 controllers, you can use pre-built alarm instructions, available in RSLogix 5000 v. 16 or later, to simplify coding, and then download them to the controller. Device-based alarm monitoring eliminates the need for duplicating alarm tags in an HMI server and requires fewer controller communication resources by eliminating polling.
- **Supports other controllers in the integrated system with tag-based alarm monitoring.** If your automation system includes other Rockwell Automation controllers, such as PLC-5s or SLC 500s, or if you prefer not to use the alarm instructions with Logix5000 controllers, software-based tag servers monitor controllers for alarm conditions and publish event information.
- **Monitors alarms and events from third-party controllers.** Tag-based alarm monitoring also makes it possible to monitor alarm conditions from third-party controllers, which communicate through OPC Data Servers.
- **Provides accurate time stamps on alarm conditions** that are generated from Logix5000 controllers using device-based alarm monitoring. When you use device-based alarm monitoring, timestamps are applied immediately in the controller and are not delayed until alarms reach an HMI server. To make sure that the timestamps on device-based alarms are accurate, synchronize the clocks of all controllers that produce alarms. The event time is propagated throughout the FactoryTalk Alarms and Events system, so inaccurate timestamps can affect where alarms are displayed in the Alarm and Event Summary or the Alarm and Event Banner as well as reports about the alarm and event history.
- **Sends process data with events and messages.** You can associate up to four tags with each alarm to include process data with event information and alarm messages.
- **Secures access to alarm and event operations** through integration with FactoryTalk Security.
- **Generates messages for logging**, including audit messages that track operator actions, system-related diagnostic messages, and historical alarm and event messages.
- **Displays alarm messages and status information** at run time, from FactoryTalk View graphic displays.

#### **FactoryTalk Alarms and Events components**

The diagram below shows a high-level view of the components of the FactoryTalk Alarms and Events system. Each component is further explained after the diagram.



## 1. Device-based alarm monitoring

Device-based alarm monitoring is setup by programming alarm instructions that are available with RSLogix 5000 v. 16 or later, and then downloading them to Logix5000 controllers. The controller detects alarm conditions and notifies RSLinx Enterprise of alarm states. A Rockwell Automation Device Server (RSLinx Enterprise) extracts the alarm information and publishes it to FactoryTalk Alarms and Events services.

Use device-based alarm monitoring with:

- Logix5000 controllers, that you have programmed with RSLogix 5000 v. 16 or later software, and Rockwell Automation device servers (RSLinx Enterprise).

## 2. Tag-based alarm monitoring

The Tag Alarm and Event Server uses tags to monitor programmable controllers for alarm conditions. When an alarm condition is detected, the server publishes the information to FactoryTalk Alarms and Events services.

Use tag-based alarm monitoring with:

- PLC-5 or SLC 500 controllers, and Rockwell Automation device servers (RSLinx Enterprise).
- RSLinx Classic and RSLinx Gateway.
- Third-party controllers and OPC Data Servers.
- Logix5000 controllers.

## 3. FactoryTalk Alarms and Events services

Both device-based and tag-based alarms and events are published to FactoryTalk Alarms and Events services, which then routes the information to FactoryTalk Alarms and Events objects hosted in FactoryTalk View, alarm and event history log, and to diagnostic logs and audit logs.

## 4. Alarm and Event Historian Log

The Alarm and Event Historian is a logging component that installs silently as part of the alarms and events software. It manages connections between alarm servers and databases and logs data from each alarm server to an alarm history database. You can use the Alarm and Event Log Viewer to view and print data from alarm history databases. Third-party database tools can also retrieve, view, analyze, and print alarm history information. To use alarm and event logging, install Microsoft SQL Server separately, or use an existing Microsoft SQL Server database.

## 5. Diagnostic and audit logs

FactoryTalk Diagnostics routes messages generated by FactoryTalk Alarms and Events to local logs on the computers hosting

FactoryTalk components, and optionally to a centralized database log. Audit messages are routed to the local log and to the FactoryTalk Audit Log if FactoryTalk AssetCentre is installed.

## 6. Alarm and event setup and monitoring

FactoryTalk Alarms and Events includes a number of software components that allow engineers and operators to define alarm conditions, set up alarm servers, view and interact with alarm conditions, and view historical alarm and event information.

### Alarm and Event Summary

Use the Alarm and Event Summary object, embedded in a FactoryTalk View graphic display, to acknowledge, disable, suppress, shelve, filter, and sort alarms at run time.

### Alarm and Event Banner

Use the Alarm and Event Banner object, embedded in a FactoryTalk View graphic display, to monitor and respond to the most serious alarms that require immediate attention.

### Alarm Status Explorer

Use the Alarm Status Explorer object, embedded in a FactoryTalk View graphic display, to enable or disable alarms, suppress or unsuppress alarms, shelve or unshelve and view operator comments.

### Database definitions

Use database definitions to define logging options from an alarm server to a Microsoft SQL Server database.

### **Alarm and Event Log Viewer**

Use the Alarm and Event Log Viewer object, embedded in a FactoryTalk View graphic display, to view and filter historical alarm information stored in Microsoft SQL Server databases.

### **Diagnostics Viewer**



Use the Diagnostics Viewer to view, filter, and export system-generated diagnostic messages. Run the Diagnostics Viewer from either FactoryTalk View Studio or FactoryTalk Administration Console.

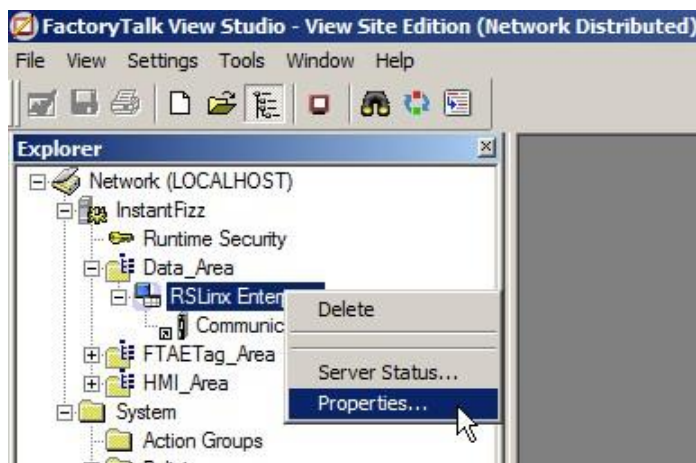
### **FactoryTalk Audit Log**

Use the FactoryTalk Audit Log to view and manage audit messages routed by FactoryTalk Diagnostics. To access the Audit Log, use FactoryTalk AssetCentre software.

## Subscribing to Device Based Alarms and Events

Your application has been pre-configured to enable Device Based alarms. Let us explore how this was set up and display some of these alarms

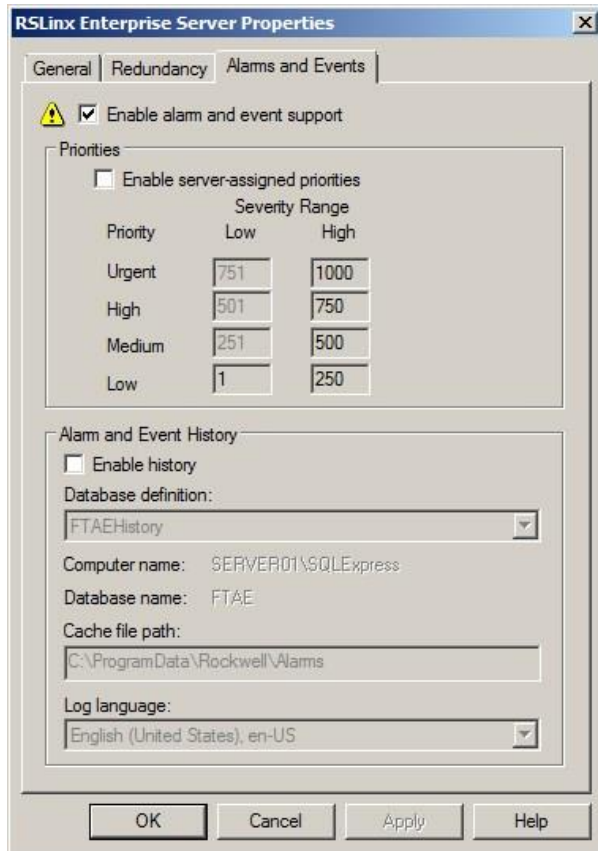
1. Open **FactoryTalk View Studio** and the **InstantFizz** application if it is not already open.
2. In the FactoryTalk View Studio Explorer window of FactoryTalk View Studio, open the application node (  InstantFizz).
3. Open the area named  Data\_Area. Right-click **RSLinX Enterprise** and select **Properties....**





In the

4. **RSLinx Enterprise Server Properties** dialog box, click the Alarms and Events tab and notice that the **Enable alarm and event support** check box is checked.

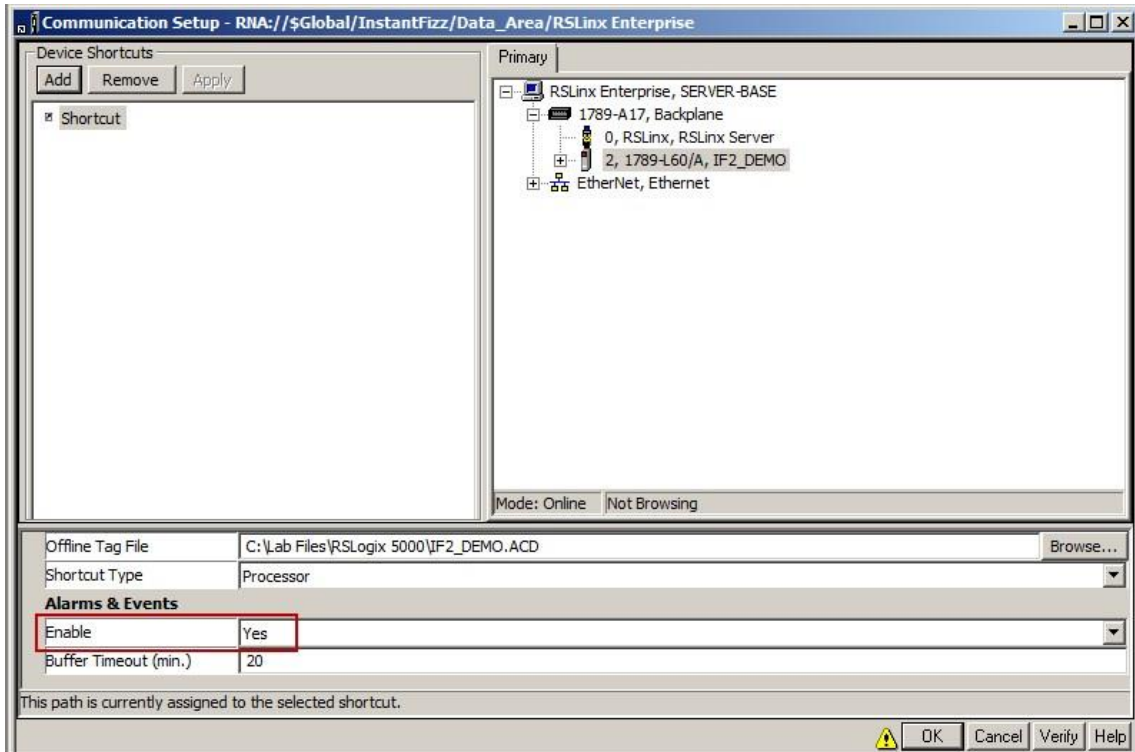


5. Click **Cancel** to close the RSLinx Enterprise Server Properties dialog box.
6. In the **FactoryTalk View Studio Explorer** window, double-click the **RSLinx Enterprise** server to expand it, and then double-click **Communication Setup**. Resize the dialog (if needed) so that you can see everything.

In the




7. **Communication Setup** dialog box, under Device Shortcuts, select the shortcut named **Shortcut**.
8. Under **Alarms & Events**, notice that the **Enable** setting is set to **Yes**.



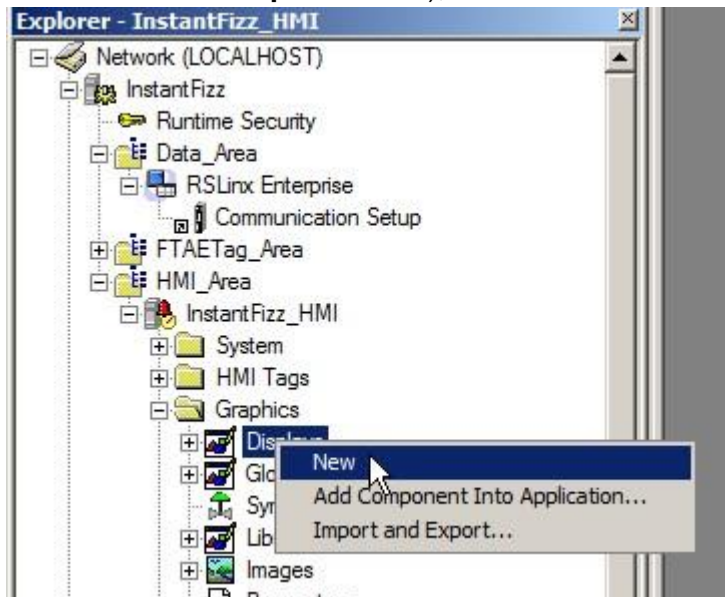
9. Click **Cancel** to close.

This confirms that device-based alarms is enabled and that the **RSLinx Enterprise** server is subscribing to alarms and events from the **SoftLogix** controller running on this computer

10. If you have not already done so, **maximize**  the FactoryTalk View Studio window.

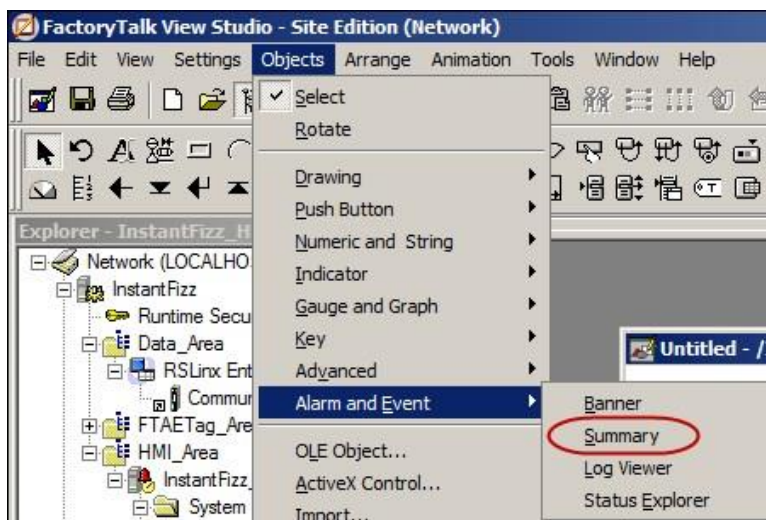
In the

11. **Explorer** window, right-click the **Displays** node (located under the HMI server node in the **Graphics** folder), and then click **New**.



A blank display appears in the workspace.

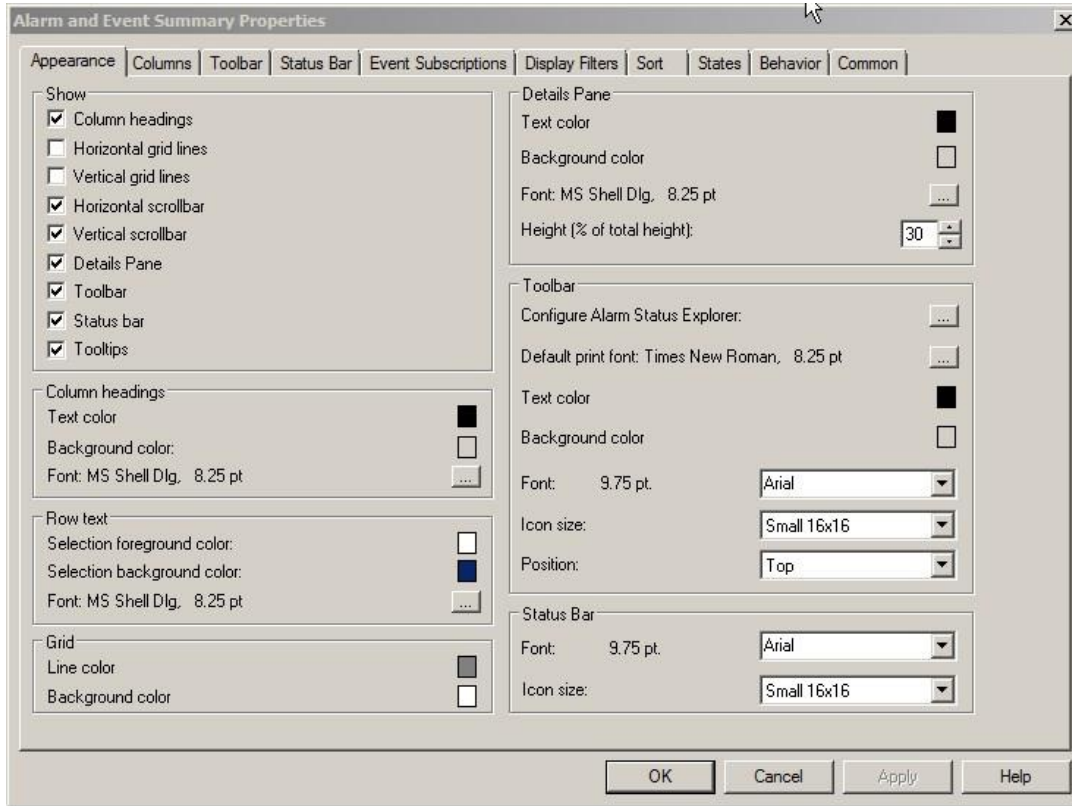
12. On the **Objects** menu, point to **Alarm and Event**, and then click **Summary** or on the toolbar, click the Summary button.

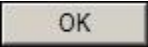


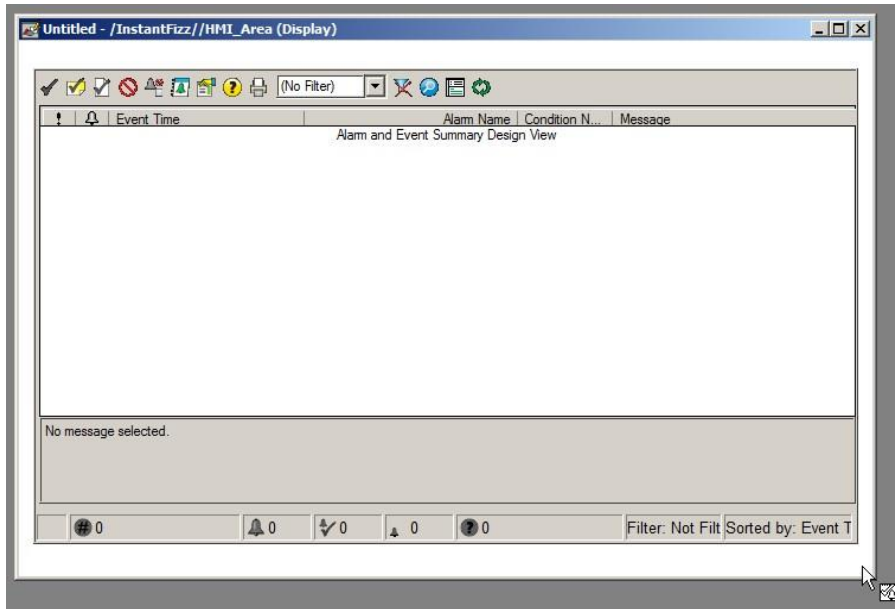
**In the**  
The mouse pointer changes to the Object drawing cursor.


13. Using the mouse, drag the rectangle to fill the display. Start in the upper left corner, drag down to the lower right and then release the left mouse button.

The **Alarm and Event Summary** object will appear in the display with the **Alarm and Event Summary Properties** on top.

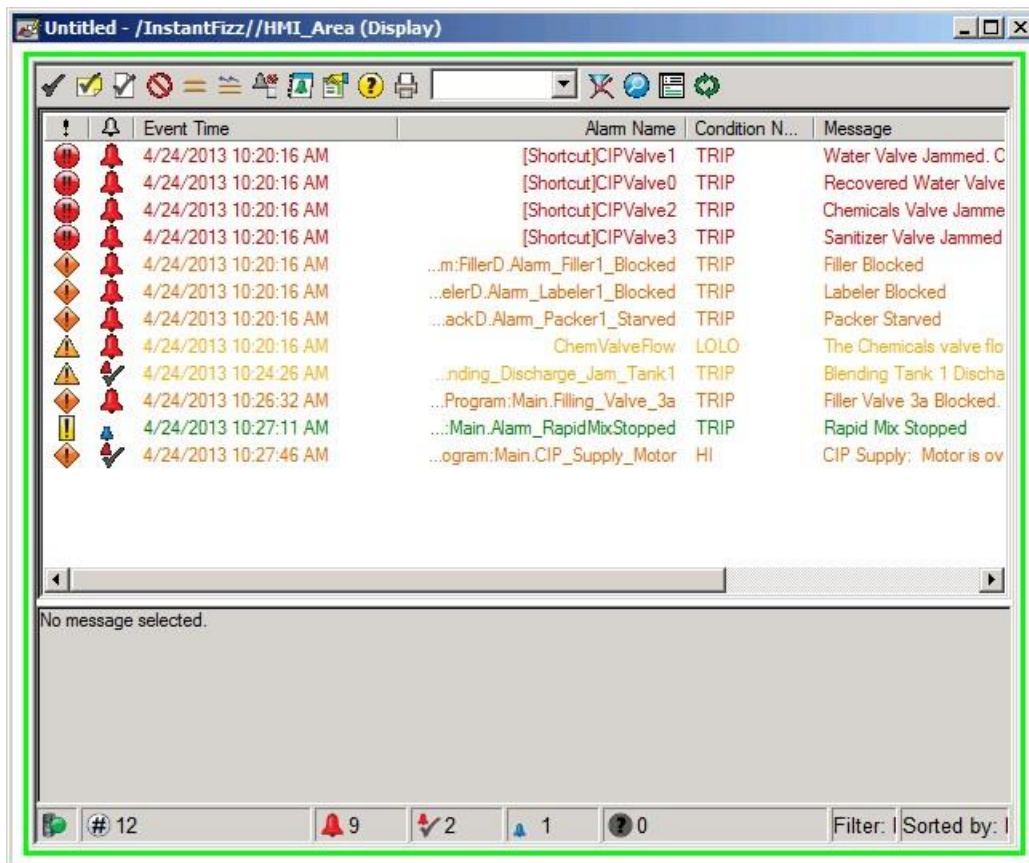


14. Press **OK**  or **Cancel** to close the Property page as we will use the default settings. (You'll have an opportunity to review some of the properties later). You will see the Alarm summary on the display.





15. On the **Graphics** toolbar, click **Test Display** button .

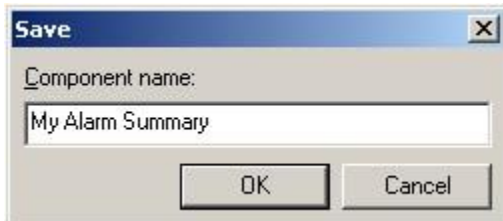
You will see alarms!



You are looking at the default configuration of alarm summary object. This control is highly configurable, so if you don't like the display choices, don't worry! Later in the lab,

you'll be learning how to configure this control and work with one that has already been modified.

16. On the **Graphics** toolbar, stop testing the display by clicking the **Edit Display** button .
17. Save the display by clicking the **Save** button  on the toolbar. In the **Save** dialog box, type **'My Alarm Summary'** for the name of the display and then click **OK**.



18. You can leave this display open and just minimize **FactoryTalk View SE Studio**. You'll come back to it in the next section.

### Working with Device-based Alarms and Events - ALMD

You will now explore device-based digital alarms with the RSLogix 5000 instruction named ALMD.

A digital alarm (ALMD instruction) is configured to monitor its input for one of the following alarm conditions:

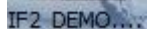
- the input value equal to one
- the input value equal to zero

When the alarm condition is true, the alarm enters the **In Alarm** state. When the alarm condition is false, the alarm enters the **Normal** or **Out of Alarm** state.

In this part of the lab, you will:

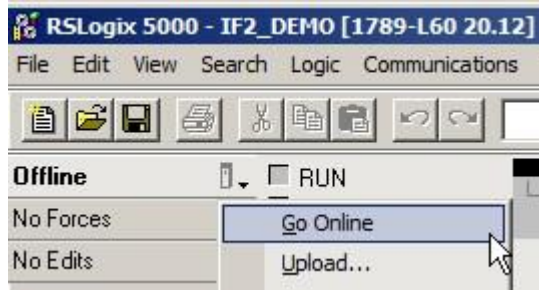
- Trigger a digital alarm within RSLogix 5000.
- Change an alarm's severity level and message while online.
- Acknowledge an alarm within RSLogix 5000.
- View the alarm status in a FactoryTalk View SE test display.



1. Click on the **IF2\_DEMO.ACD** shortcut  from the Windows **Start** button to open the project that is used in this lab.
2. Go online with the controller by clicking the pull down next to the Controller Status display and selecting **Go**

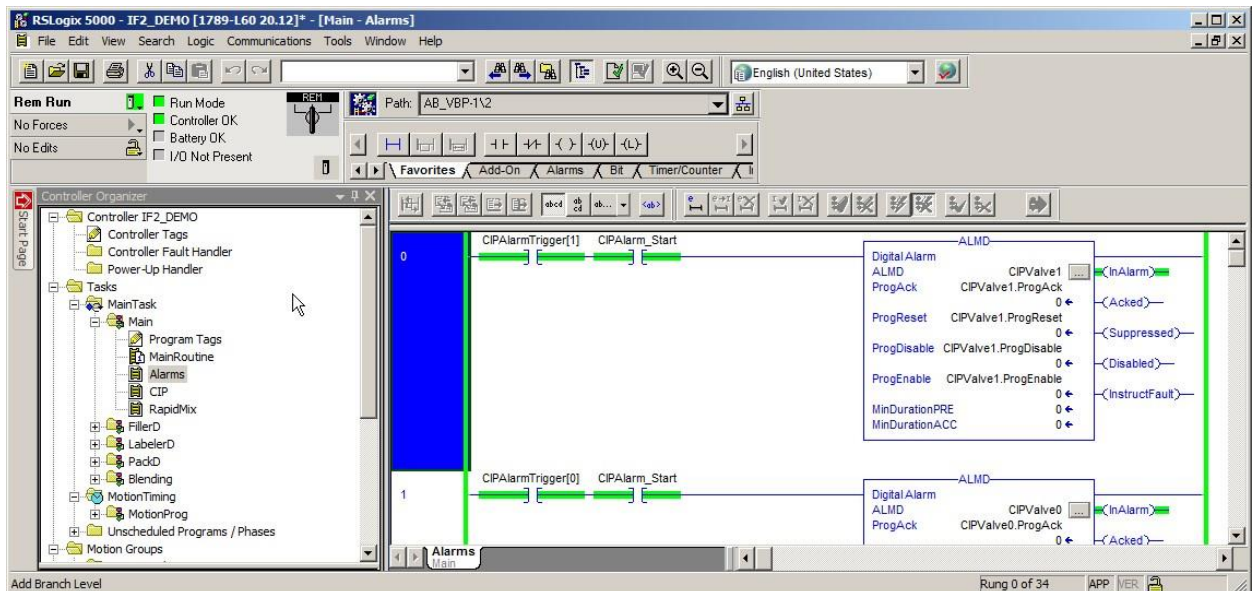


## Online...



If you are prompted with the Connected To Go Online dialog, choose **Download** twice and select **Yes** when prompted to change to the controller back to Remote Run.

3. In the left pane, expand the **Main** folder under **Tasks**, and then double-click the **Alarms** routine.




Look at **Rung 0** and notice that the **ALMD** alarm instruction named **CIPValve1** is active. You can tell by looking at the instruction's output parameter **InAlarm**. Notice that it is highlighted green.



The **InAlarm** parameter is active because the input condition is true. In this case, the input conditions are the BOOLEAN tags named **CIPAlarmTrigger[1]** and **CIPAlarm\_Start**. You can tell this condition is true because it is also highlighted in green.




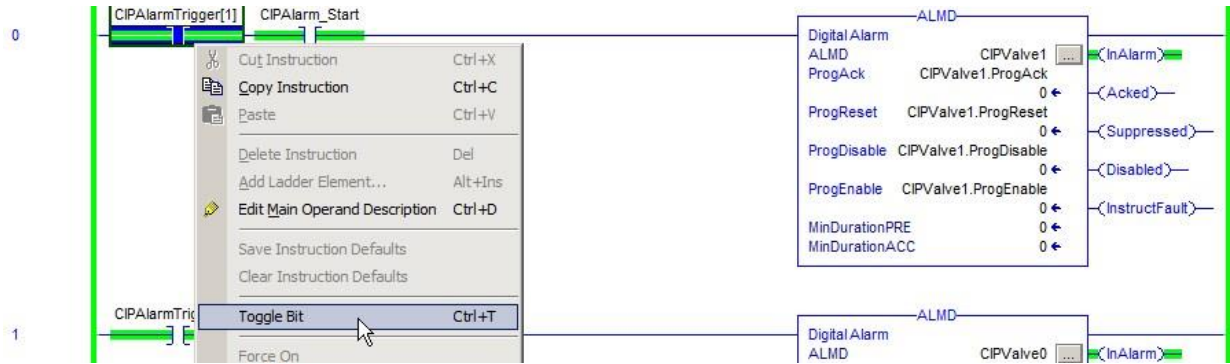
4. Return to **FactoryTalk View SE Studio** – you should have it minimized in your Windows task bar from the previous section.
5. Select the **My Alarm Summary** screen you created earlier and click the **Test Display** button  from the **Graphics** toolbar.



Notice that the alarm in the list, with the Alarm Name of **[Shortcut]CIPValve1**, corresponds to the ALMD instruction we had just observed in Rung 0 of RSLogix 5000. The Alarm and Event Summary object is indicating that the alarm is active just as we would expect.

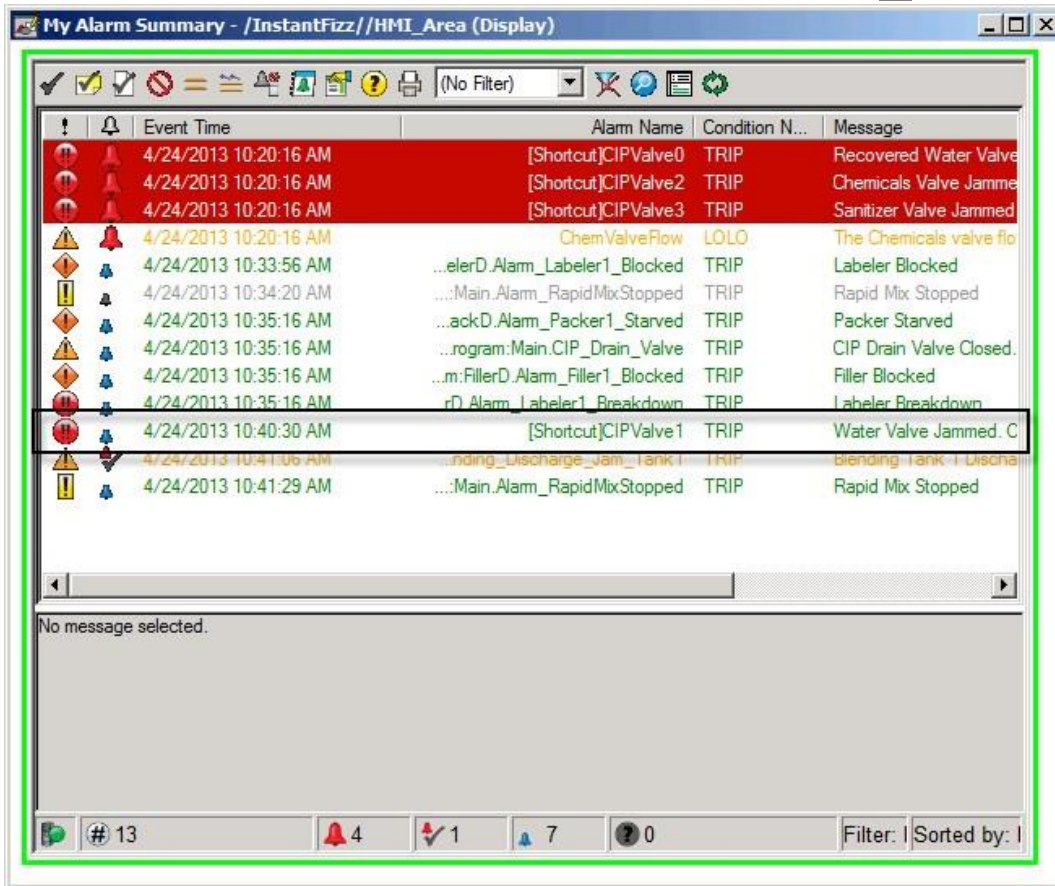


- Return to **RSLogix 5000**. In Rung 0, select the **Examine On**  instruction for **CIPAlarmTrigger[1]**. Rightclick and select **Toggle Bit** or select **Control+T** to turn the bit off.



Notice now that the input condition of CIPAlarmTrigger[1] is off and the InAlarm parameter of the CIPValve1 ALMD instruction is no longer active.

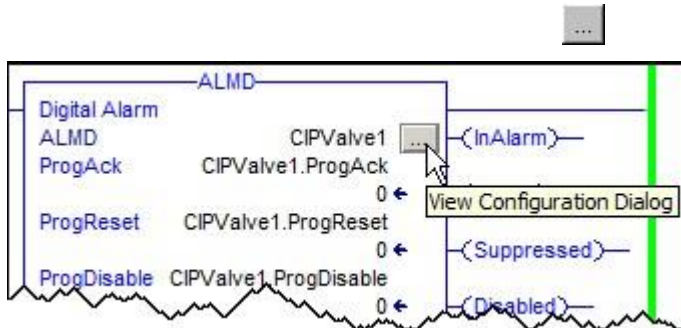




- Return to **FactoryTalk View SE Studio** and **select** the **My Alarm Summary** screen you created earlier. If it is not still running in test mode, on the **Graphics** toolbar, click **Test Display** button.

You will see in your test display that the **CIPValve1** alarm is still in the alarm list, but it is no longer active. You may have to scroll down your alarm list to find it.

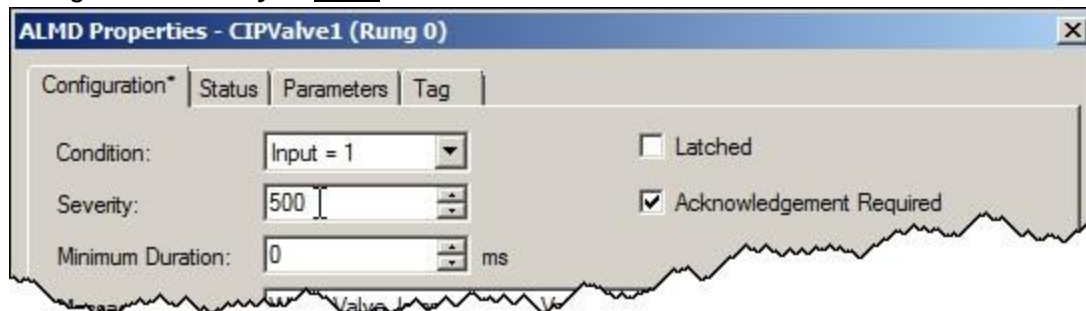
Inside button box.



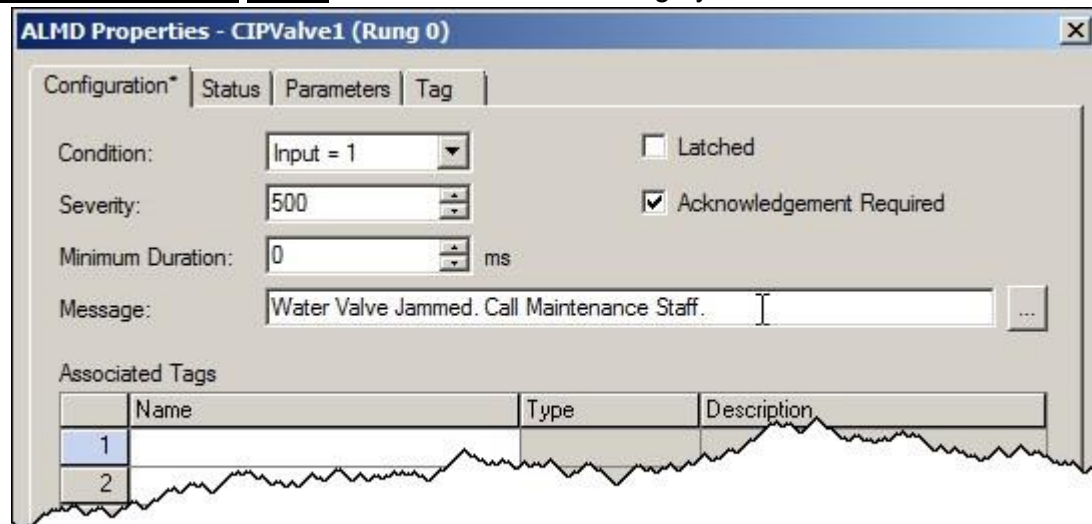
8. Return to **RSLogix 5000**. Now we will see how the ALMD instruction is configured.

the alarm instruction, click the Browse to open the **ALMD Properties** dialog

9. Change the **Severity** to **'500'**.



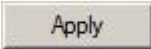
10. Change the Message from **Water Valve Jammed. Open Valve.** to **'Water Valve Jammed. Call Maintenance Staff.'** ... or to whatever message you want. Feel free to be creative!

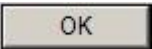



There are many properties of an **ALMD** instruction that you can configure within this dialog box. If you'd like them and what they do, click **Help** on this dialog

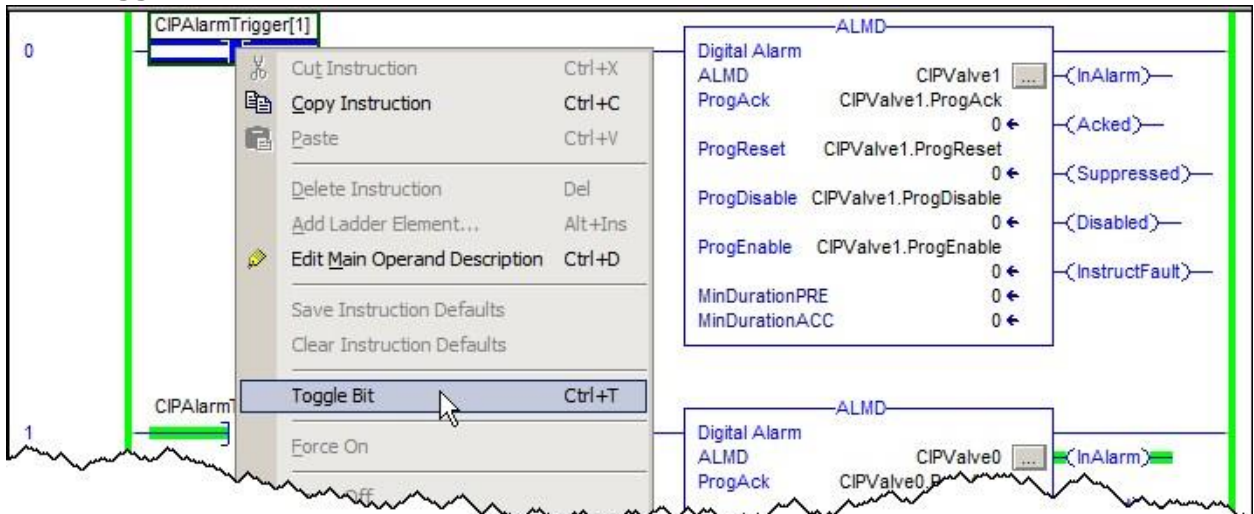
Help

instruction that you can more information about box.

11. Click **Apply**  to apply changes.

12. Click **OK**  to close the **ALMD Properties** dialog box.


13. Select again the **Examine On**  instruction for **CIPAlarmTrigger[1]**. **Right-click** and select **Toggle Bit** or select **Control+T** to turn the bit to on.



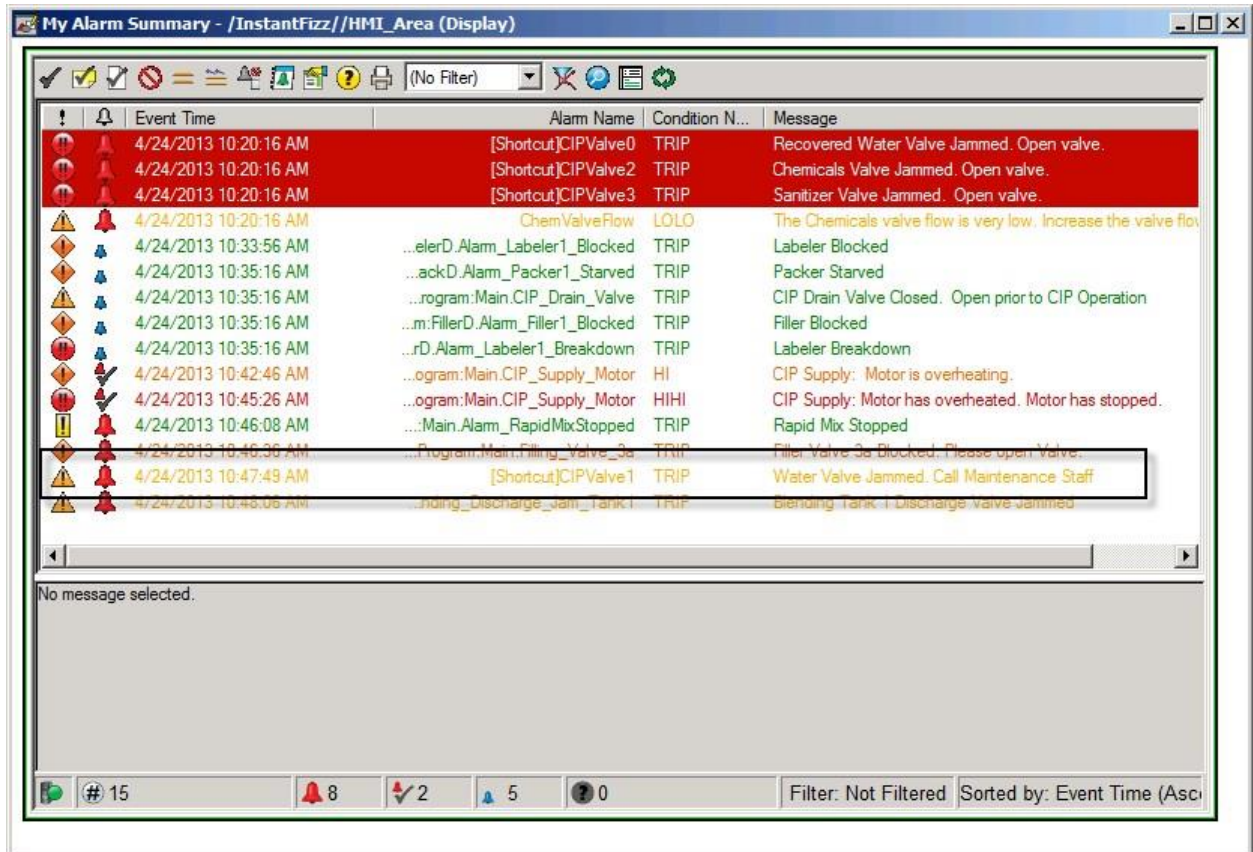
Notice now that the input condition of **CIPAlarmTrigger[1]** is on, the **InAlarm** parameter of the **CIPValve1 ALMD** instruction is active.








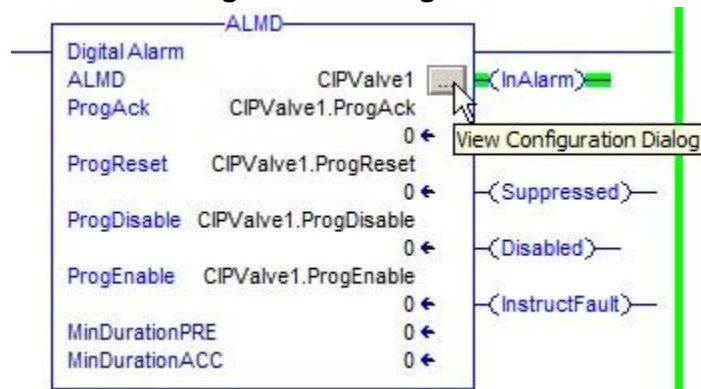
- Return to **FactoryTalk View SE Studio** and select the **My Alarm Summary** screen you created earlier. If it is not still running in test mode, on the **Graphics** toolbar, click **Test Display** button. 

Scroll down to about the middle of your alarm list and you will see the **CIPValve1** alarm active, but with different colors than previously seen and your new message.

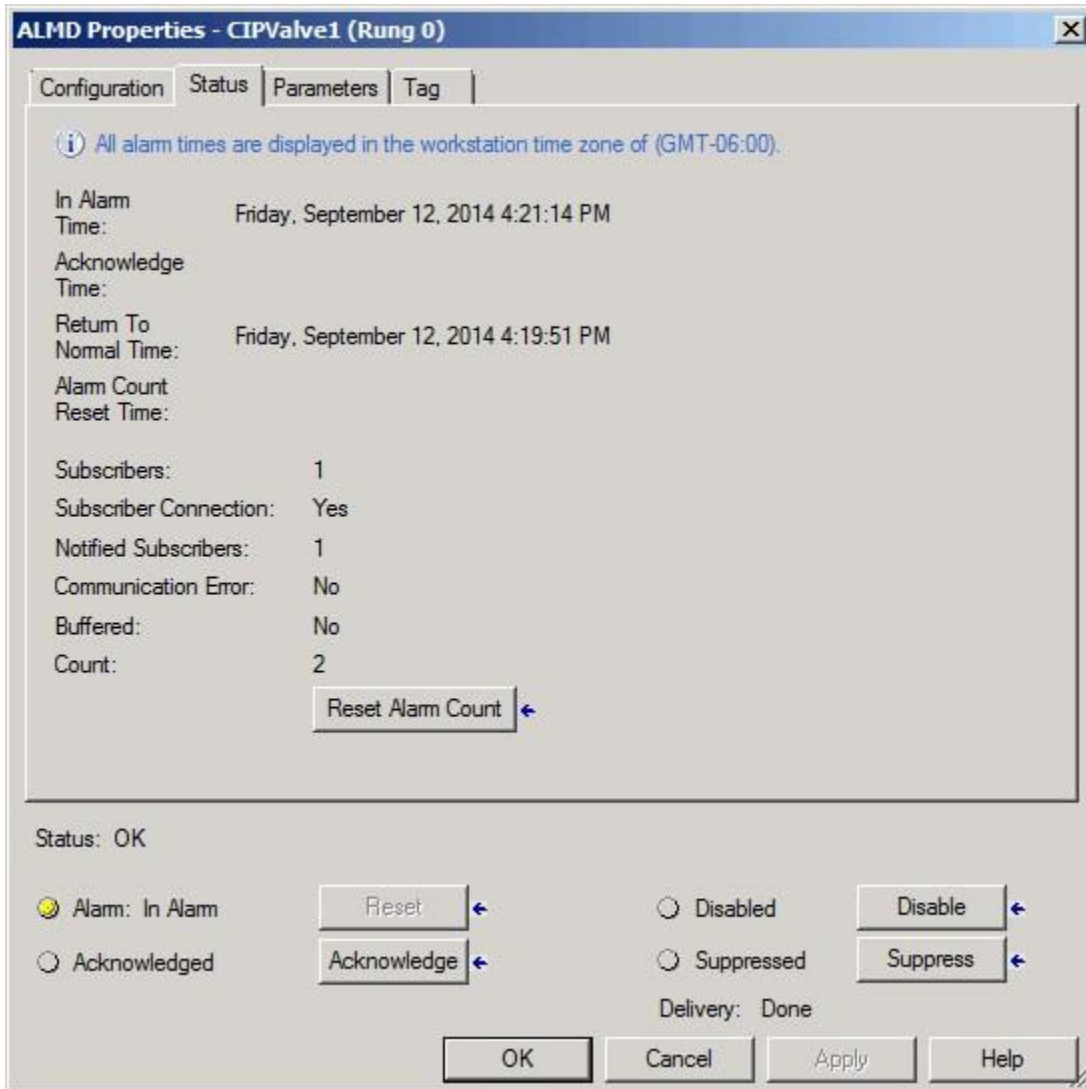


The color change occurs because we changed the Severity of the alarm. Also notice that the severity icon has changed from  (Urgent) to  (Medium).

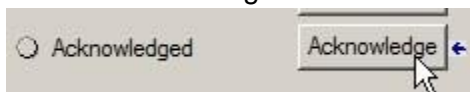
- Return to **RSLogix 5000**. Inside the alarm instruction, click the Browse button  to open the **ALMD Configuration Dialog** box.



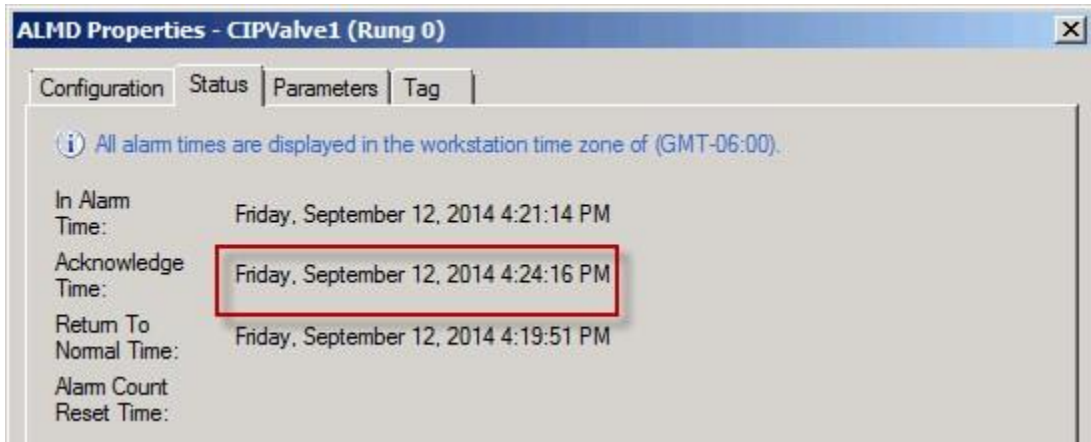
- In the **ALMD Properties** dialog box, select the **Status** tab and then observe the **In Alarm Time** and **Return To Normal Time**.



17. To acknowledge the alarm from **RSLogix 5000**, click the **Acknowledge** button near the bottom of the dialog window.

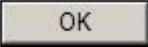


The Acknowledge Time will update on the Status tab.



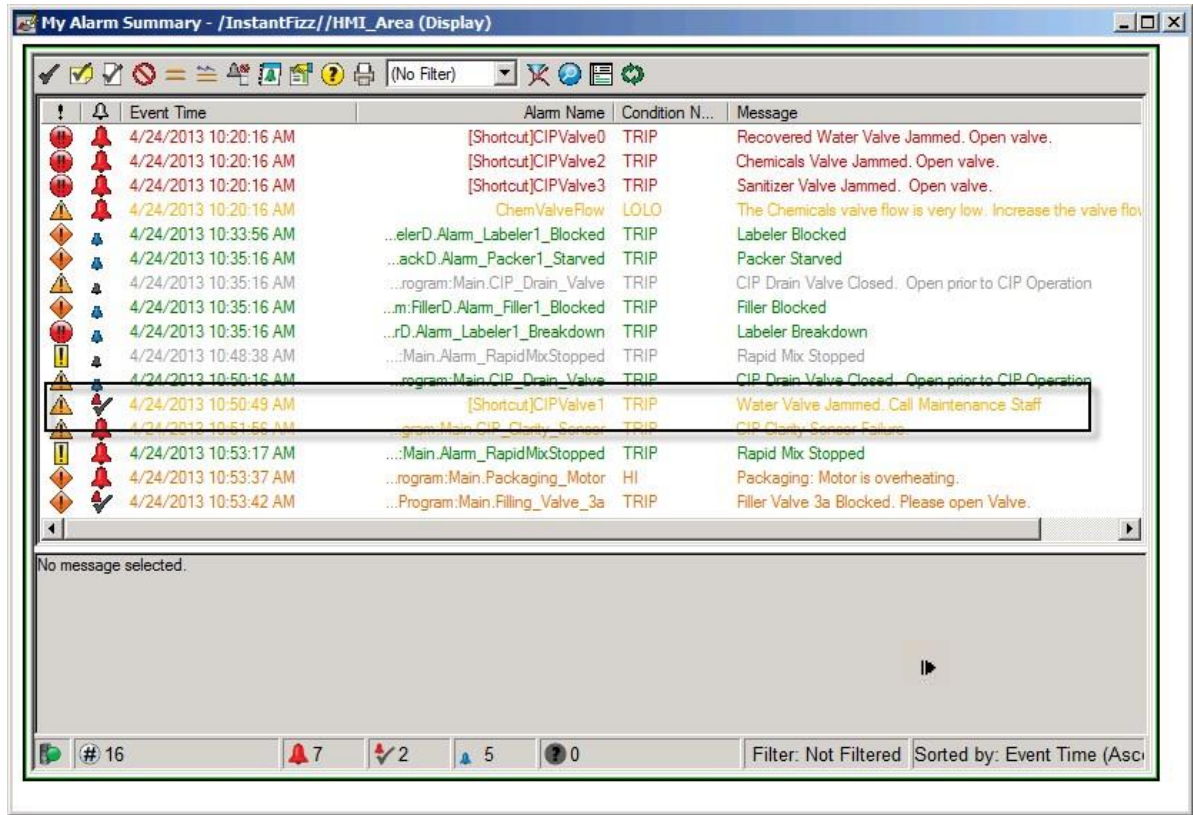
You will also see the **Acked** output of the alarm instruction becomes active (green highlight).



18. Click **OK**  to close the **ALMD Properties** dialog box.

19. Return again to FactoryTalk View SE Studio and select the **My Alarm Summary** screen you created earlier. If it is not still running in test mode, on the **Graphics** toolbar, click **Test Display** button.

Find the **CIPValve1** alarm in your list.



(In Alarm and Unacknowledged) (In Alarm and Acknowledged)

Notice that the alarm status icon has changed from Acknowledged) indicating that the alarm has been acknowledged.

20. On the **Graphics** toolbar, stop testing the display by clicking the **Edit Display** button .

Congratulations! You have just:

- Observed and manipulated an ALMD instruction online while the program is running.
- Acknowledged an alarm within RSLogix 5000.
- Viewed the alarm status in a FactoryTalk View SE test display.

Whether you realize it or not, you have also witnessed that online changes to alarm instructions take effect without having to reboot your PC or restart your client. If an active alarm has a pending edit, the update takes affect whenever the alarm changes state next.



In the next section, you will explore ALMA instructions in a similar way.

## Working with Device-based Alarms and Events - ALMA

Now we will explore analog device-based alarms using the RSLogix 5000 instruction named ALMA.

An analog alarm can be configured to monitor for two types of alarm conditions: Level and Rate of Change.

A Level alarm monitors an input for alarm conditions that go In Alarm when the input value goes above or below predefined limits. When defining a level alarm, you can configure up to four alarm level conditions each with limits (sometimes called thresholds), a severity and alarm message. The supported alarm conditions/thresholds are:

- High High (HIHI)

- High (HI)
- Low (LO)
- Low Low (LOLO)

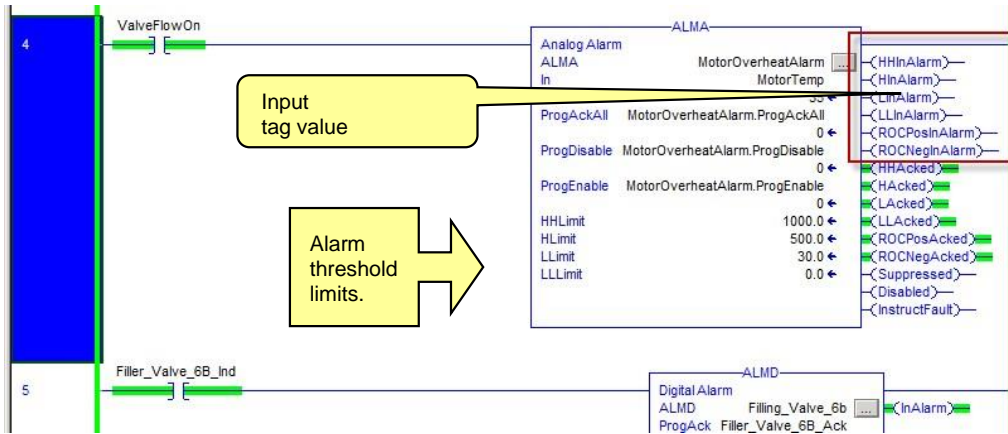
A Rate of Change alarm monitors an input for alarm conditions that go In Alarm when the input value changes faster or slower than predefined limits. When defining a level alarm, you can configure up to two rate of change conditions each with limits, a severity, and an alarm message. The supported alarm conditions are:

- Rate of Change Positive (ROC\_POS)
- Rate of Change Negative (ROC\_NEG)

In this part of the lab, you will:

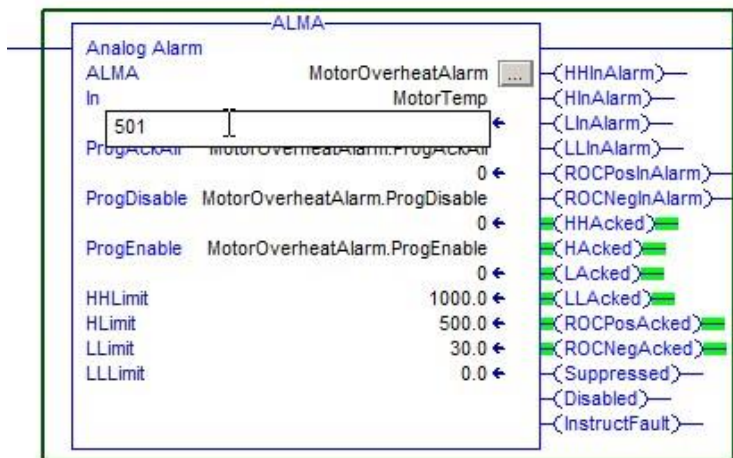
- Trigger different levels of an analog alarm within RSLogix 5000.
- Learn about associated tags.
- Learn how to embed a variable in an alarm message.
- View the alarm status in a FactoryTalk View SE test display.

1. Return to **RSLogix 5000**.
2. If the **Alarms** subroutine is not open, then in the left pane, expand the **Main** folder under **Tasks**, and then double-click the **Alarms** routine.
3. If not already online with the controller, click the pull down next to the **Controller Status** display and select **Go Online...**
4. Scroll down to **Rung 4** to see an **ALMA** alarm instruction named **MotorOverheatAlarm**. Notice that no alarm levels are active – you can tell because none of the alarm condition parameters are highlighted in green.

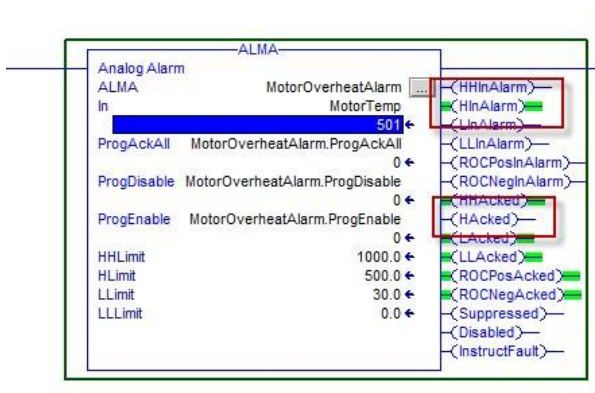


The **In** input for this alarm is the tag named **MotorTemp**. The value of this tag will trigger this alarm if it crosses any of the defined threshold limits.

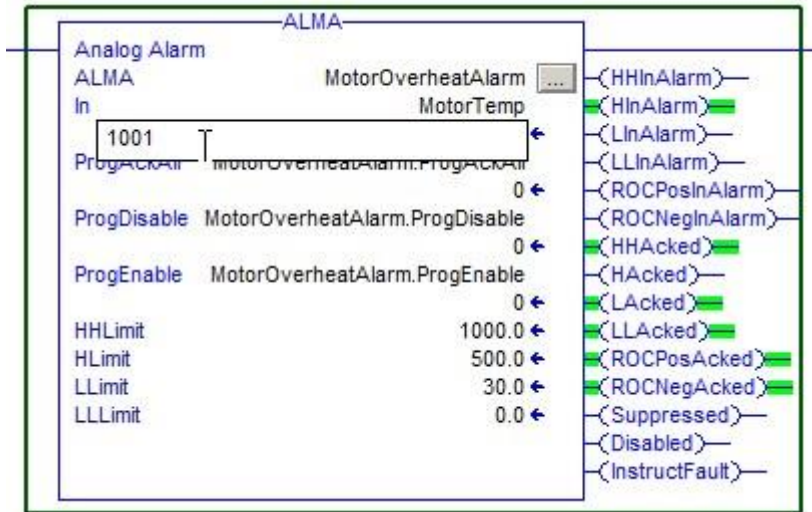
- Double-click the **35** value below **MotorTemp** and change the value to **'501'** and hit the **Enter** key



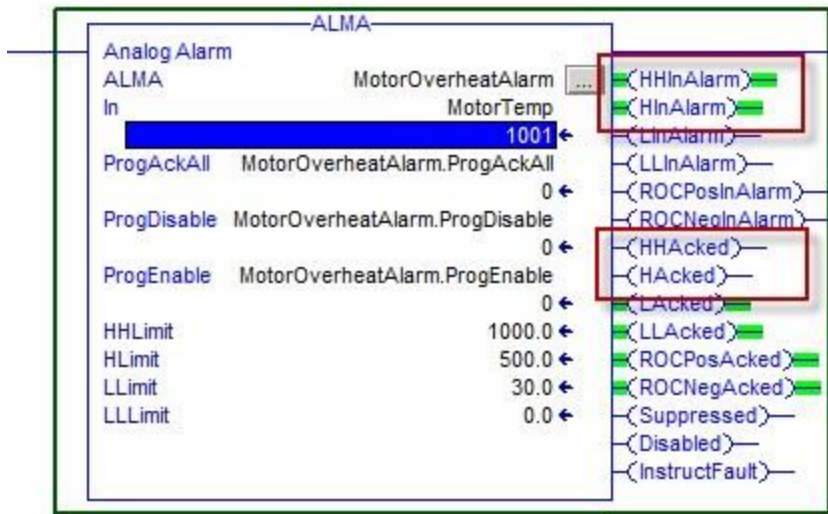
Notice how the **HInAlarm** condition is now true and the **Hacked** condition is now false.




- Double-click the **501** value below **MotorTemp** and change the value to **'1001'**.



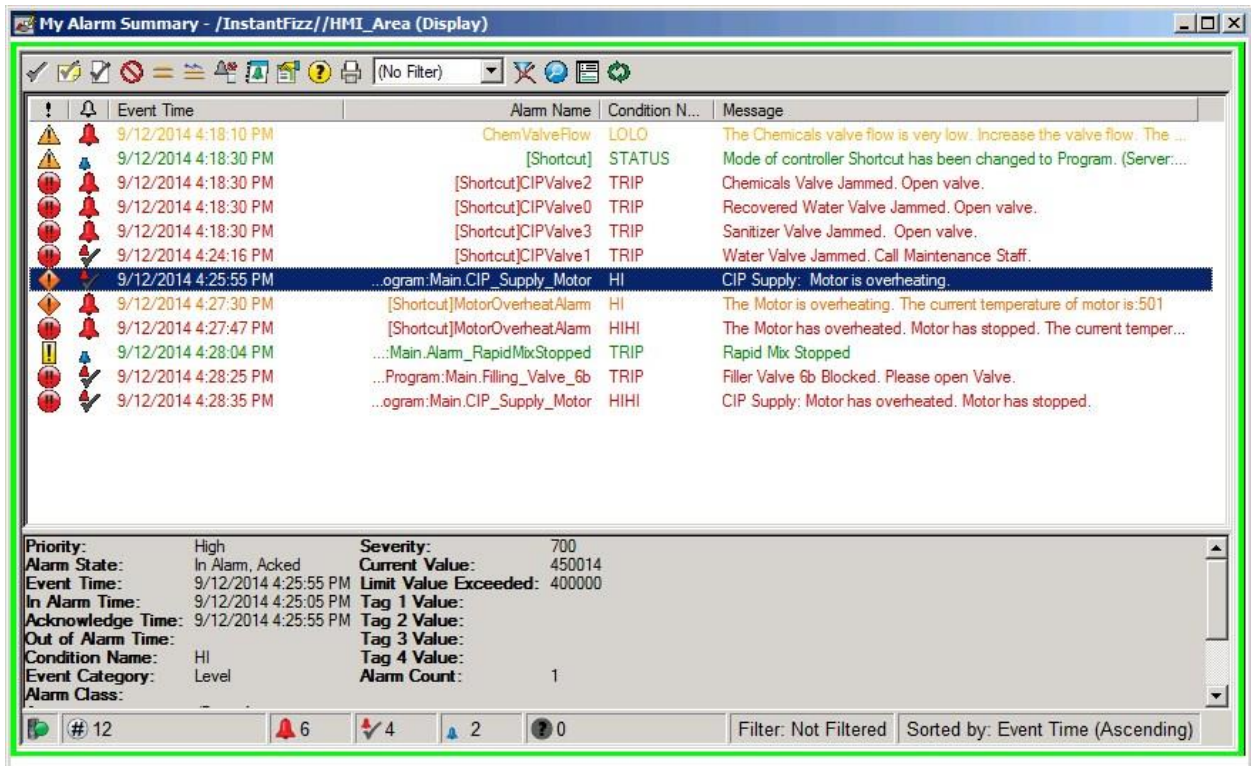
Notice now that both the **HInAlarm** and **HHInAlarm** conditions are true, while both the **Hacked** and **HHacked** conditions have turned false.



If a High High (HHI) alarm level condition is met, the High (HI) alarm level condition is always true as well. Therefore, whenever a HH alarm is active the H alarm is as well. The same case applies to the two low level alarms. If the Low Low (LOLO) alarm condition is met, the Low (LO) alarm is also triggered.

- Return to **FactoryTalk View SE Studio** and select the **My Alarm Summary** screen you created earlier. If it is not still running in test mode, on the **Graphics** toolbar, click **Test Display** button . 

Find the 2 entries (**HI**, **HHI**) for the **[Shortcut]MotorOverheatAlarm**. You may need to scroll down the list.



While both conditions are active, they each have a different severity level as designated by the different display colors and the different severity icons: (Urgent) and (High).

You can see on the alarm summary screen that the alarm message for the **HI** condition is **The Motor is overheating. The current temperature of the motor is 501**. Recall that 501 is the first value we set to **MotorTemp** to trigger the **HI** condition.

In many cases it is useful to have additional process information associated with an alarm. When an alarm is defined, you can associate up to four tags with the alarm. At run time, the tag values are recorded in the alarm and event history log and can also be displayed in the Alarm and Event Summary or Alarm and Event Log Viewer. The values of associated tags can also be embedded in alarm messages and the contents of the Alarm and Event Summary or Alarm and Event Log Viewer can also be filtered based on the value of an associated tag.

8. Notice how the alarm message for the **HIHI** condition is too long to be properly displayed on the summary screen. Fear not, however, the complete alarm message is not lost! Use the mouse to **click** on the **HIHI** alarm in the summary display. The alarm details will appear in the **Details Pane** below the summary display. Use the scrollbar on the right to scroll to the end of the Details Pane where you will see the complete message: **The Motor has overheated. Motor has stopped. The current temperature of the motor is 1001.**

	9/12/2014 4:27:30 PM	[Shortcut]MotorOverheatAlarm	HI	The Motor is overheating. The current temperature of motor is:501
	9/12/2014 4:27:47 PM	[Shortcut]MotorOverheatAlarm	HIHI	The Motor has overheated. Motor has stopped. The current temper...
	9/12/2014 4:28:04 PM	...Main.Alarm.RapidMixStopped	TRIP	Rapid Mix Stopped
	9/12/2014 4:28:35 PM	...ogram:Main.CIP_Supply_Motor	HIHI	CIP Supply: Motor has overheated. Motor has stopped.
	9/12/2014 4:30:55 PM	...nding_Discharge_Jam_Tank1	TRIP	Blending Tank 1 Discharge Valve Jammed
	9/12/2014 4:31:45 PM	...rogram:Main.CIP_Drain_Valve	TRIP	CIP Drain Valve Closed. Open prior to CIP Operation
	9/12/2014 4:31:45 PM	...rogram:Main.CIP_Clarify_Sensor	TRIP	CIP Clarify Sensor Failure.
	9/12/2014 4:31:46 PM	...rogram:Main.Packaging_Motor	HI	Packaging: Motor is overheating.
	9/12/2014 4:31:47 PM	...Program:Main.Filling_Valve_3a	TRIP	Filler Valve 3a Blocked. Please open Valve.


  

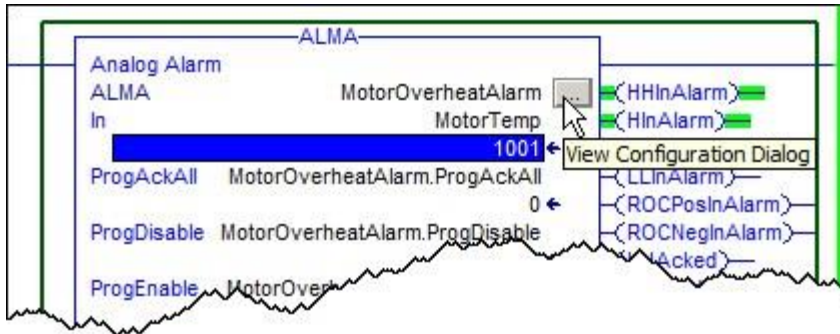
<b>Out of Alarm Time:</b>		<b>Tag 3 Value:</b>	
<b>Condition Name:</b>	HIHI	<b>Tag 4 Value:</b>	
<b>Event Category:</b>	Level	<b>Alarm Count:</b>	1
<b>Alarm Class:</b>			
<b>Area:</b>	/Data_Area		
<b>Server Name:</b>	RSLinx Enterprise		
<b>Alarm Name:</b>	[Shortcut]MotorOverheatAlarm		
<b>Message:</b>	The Motor has overheated. Motor has stopped. The current temperature of motor is:1001		

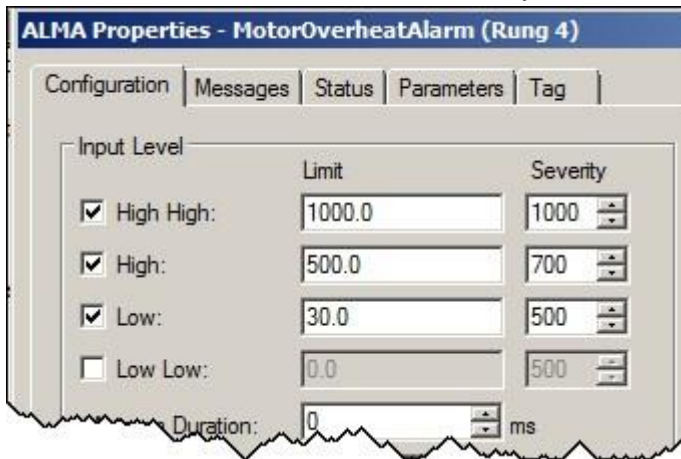
	# 16		9		5		2		0	Filter: Not Filtered	Sorted by: Event Time (Ascending)
--	------	--	---	--	---	--	---	--	---	----------------------	-----------------------------------

Recall that 1001 is the second value we set to **MotorTemp** to trigger the **HIHI** condition.

- Return to **RSLogix 5000**. Inside the alarm instruction, click the Browse button  to open the **ALMA Configuration Dialog** box.

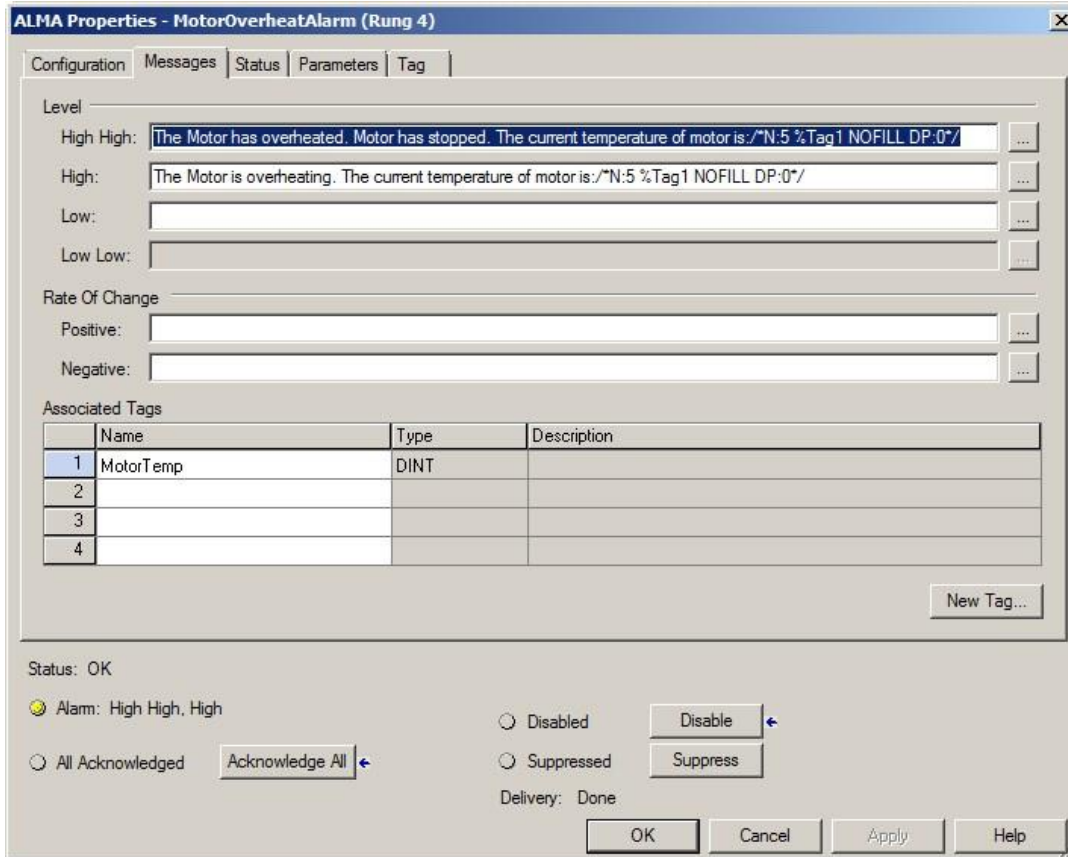



Notice on the **Configuration** tab the three levels that are enabled: **High High**, **High**, and **Low**. The **Low Low** limit is not enabled, as indicated by the unchecked check box.

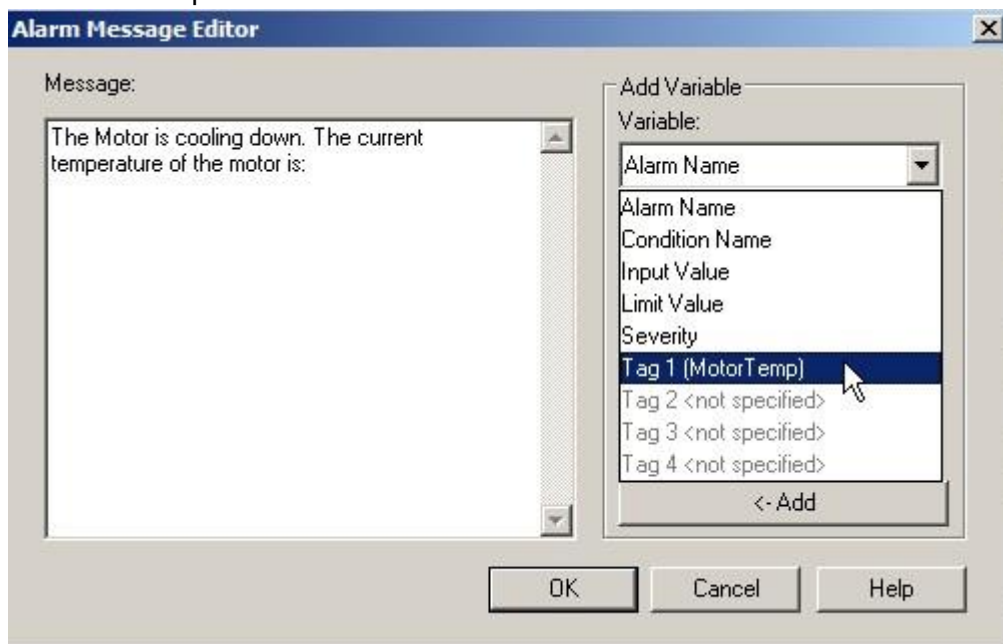


- Select the **Messages** tab to see the messages defined for each level, as well the **Associated Tag (MotorTemp)** that we observed was included in each of the messages.





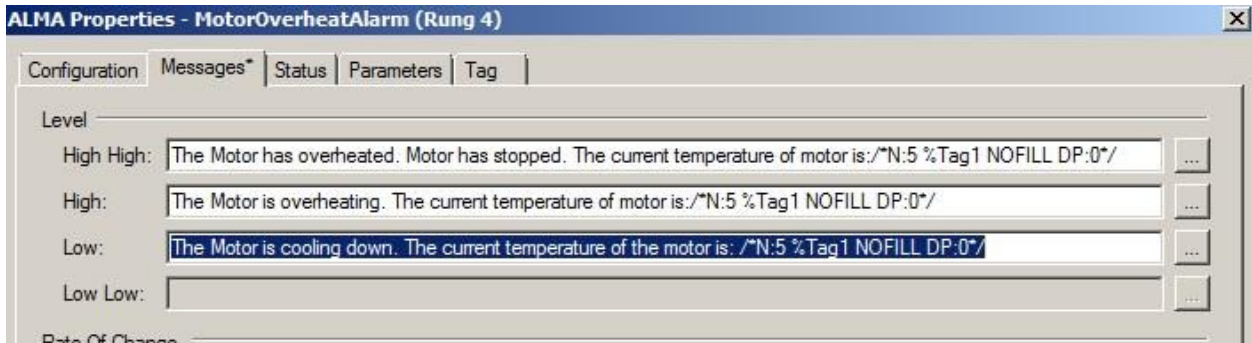
11. In the **Low** level field, type **The Motor is cooling down. The current temperature of the motor is:** then click the **Browse** button  to open the Alarm Message Editor.
12. Select the variable to add to the alarm message by selecting **Tag 1 (Motor Temp)** from the **Add Variable** pull down menu.



13. Click **<- Add**  to add the variable to the message.

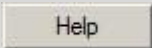
14. Click **OK**  to close the Alarm Message Editor and add your changes.

The resulting message will look like this:

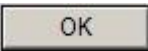


Notice the **Messages** tab now says **Messages\*** which indicates there are unsaved configuration changes to the instruction.

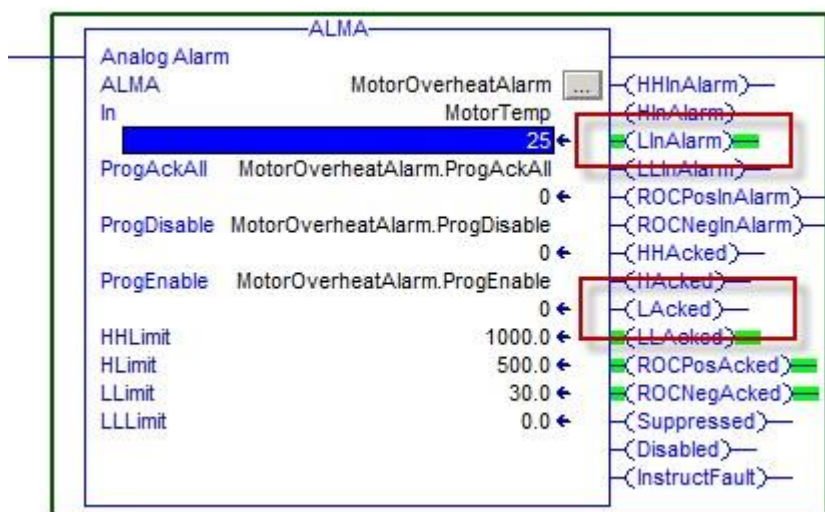
Like the ALMD instruction, there are many properties of an **ALMA** instruction that you can configure within this dialog box. If you'd like more information about them and what they do, click **Help** on this dialog box.




15. Click **Apply**  to apply changes.

16. Click **OK**  to close the **ALMA Properties** dialog box.

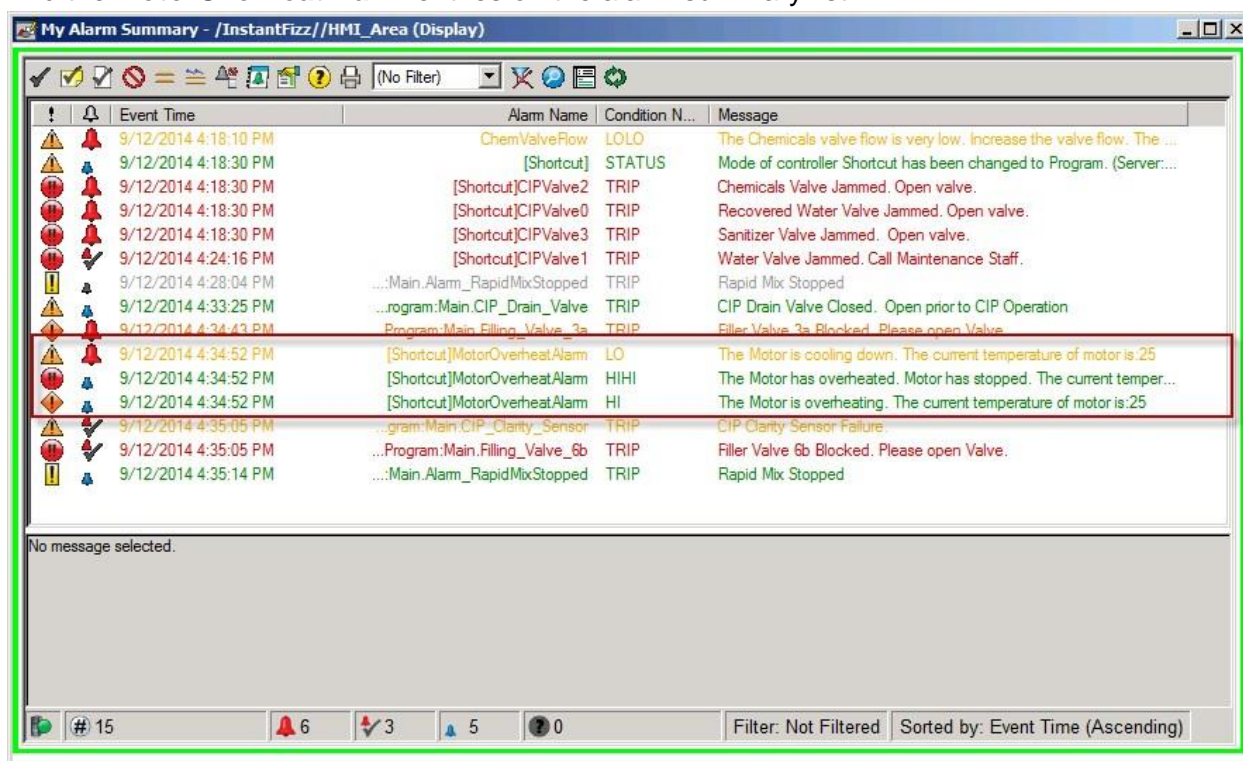
17. Double-click the **1001** value below **MotorTemp** and change the value to **'25'**. The **LInAlarm** condition will now be true and **LAcked** parameter will turn false.



18. Return to **FactoryTalk View SE Studio** and **select** the **My Alarm** Summary screen you created earlier. If it is not still running in test mode, on the **Graphics** toolbar, click **Test Display** button .





19. Find the **MotorOverheatAlarm** entries on the alarm summary list.




The **LO** alarm condition is now acknowledged and the **HIHI**/**HI** conditions are now not acknowledged but unacknowledged. Notice also the alarm messages displayed for the **HIHI** and **HI** conditions. Both say **The current temperature of the motor is: 25**. What happened to the tag values 501 and 1001?

The messages now display the current temperature of the motor since it was that tag input value that cleared the alarm state for that condition. The original messages containing 501 and 1001 would already be saved in the historical database. (We'll configure our historical database later.)

20. On the **Graphics** toolbar, stop testing the display by clicking the **Edit Display** button .

21. Close the **My Alarm Summary** display by clicking the  in the corner. Press **No** when asked to save changes.

22. Return to **RSLogix 5000** and save the program by clicking the **Save** button  on the toolbar and then click **Yes** on the confirmation dialog.

Congratulations! You have just:

- observed and made online edits to an ALMA and ALMD instruction.
- learned about Associated Tags and how to embed variables in alarm messages.

In the next part, we will explore Tag-Based FactoryTalk Alarms and Events.

## Working with Tag-based Alarms and Events

In this part of the lab, you will:

- Observe tag-based FactoryTalk alarms that have already been configured.
- Learn about the new Shelf / Unshelf feature that aligns with ISA Standard 18.2
- Learn how to configure tag-based alarms for remote acknowledgement, allowing alarms to be acknowledged from faceplates or pushbuttons.
- View the alarm status in a FactoryTalk View SE test display.

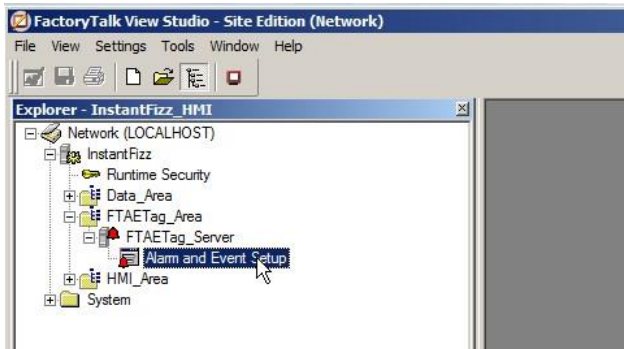
With tag-based alarm detection, a Tag Alarm and Event Server uses tags to monitor programmable controllers for alarm conditions. When an alarm condition is detected, the server generates an alarm to the FactoryTalk Alarms and Events system.

Use tag-based alarm monitoring with:

- PLC-5 or SLC 500 controllers, communicating through RSLinx Classic or RSLinx Gateway.
- Third-party controllers, communicating through OPC-DA servers (e.g., KEPWare).
- Logix5000 controllers that have not been programmed with pre-built alarm instructions available in RSLogix 5000 v. 16 or later. Tag-based alarms are also useful for monitoring alarm conditions in instructions that do not currently produce devicebased alarms (e.g., PIDE).

1. Switch back to **FactoryTalk View Studio**.

In the **Explorer** window, under the application file ( **Instant Fizz**), expand the **FTAETag\_Server**. Double-click the **Alarm and Event Setup**. In the **FTAETag\_Area**, expand



3. Notice in the Alarm and Event Setup, there are 5 alarms pre-configured – one Level alarm and four Digital alarms.

Name	Type	Input Tag	Ack Req'd	Alarm as a Tag	Alarm Class	FactoryTalk View Command
ChemValveFlow	Level	/Data_Area::[Shortcut]chemValveFlow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		display CIP Overview
LabelJam	Digital	/Data_Area::[Shortcut]AlarmLabelJam	<input checked="" type="checkbox"/>	<input type="checkbox"/>		display Labeling
PaperOut	Digital	/Data_Area::[Shortcut]AlarmPaperOut	<input checked="" type="checkbox"/>	<input type="checkbox"/>		display Labeling
Tank1Jam	Digital	/Data_Area::[Shortcut]AlarmTank1Disc	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		display Blending
Tank2Jam	Digital	/Data_Area::[Shortcut]AlarmTank2Disc	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		display Blending

4.

Select and double-click the first alarm in the list, which is a **Level** alarm, named **ChemValveFlow**.

**Level Alarm Properties**

Level | Messages | Status Tags | Control Tags

Name: ChemValveFlow

Input Tag: /Data\_Area::[Shortcut]chemValveFlow

	Limit - Value or Tag	Severity	Apply Duration
<input checked="" type="checkbox"/> High High	1000	500	<input type="checkbox"/>
<input type="checkbox"/> High			<input checked="" type="checkbox"/>
<input type="checkbox"/> Low			<input checked="" type="checkbox"/>
<input checked="" type="checkbox"/> Low Low	100	500	<input type="checkbox"/>

Minimum duration: 0 Seconds

Deadband: 0

Acknowledge required

Show Alarm as a Tag

Alarm Class: [Dropdown]

FactoryTalk View Command: display med\_cip

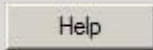
OK Cancel Help

Notice that the configuration of the tag-based Level alarm is very similar to the device-based ALMA with the following exceptions:

- **Input Tag** – This is the tag that is monitored for the alarm condition.
- **Show Alarm as Tag** – This option allows you to monitor the status and operate on alarms programmatically at run time using tags that are exposed by the Tag Alarm and Event server. Live Data clients can read and write to the alarm tags to monitor and change alarm states.
- **Status Tags** – Status tags allow you to keep alarm state changes synchronized with a programmable controller. For example, if you assign a tag to the Acknowledged status tag, when an operator acknowledges the alarm from the Alarm Summary, the status tag is set to 1.
- **Control Tags** – Control tags allow you monitor and control alarms by using tags in the controller. For example, if you assign a tag to the Suppress control tag, when the value of the tag changes

to positive, the Tag Alarms and Events Server automatically Suppresses alarms associated with the control tag.

Like the device-based alarms, there are many properties of a **Level Alarm** that you can configure within



**Level Alarm Properties** dialog box. If you'd like more information about them and what they do, click **Help**

on this dialog box.

5.

Click **OK**  to close the **Level Alarm Properties** dialog box.

6.

Select and double-click the **Digital** alarm in the list named **PaperOut**.

**Digital Alarm Properties**

Digital | Status Tags | Control Tags

Name: PaperOut

Input Tag: /Data\_Area:([Shortcut])AlarmPaperOut

Condition: Input <> 0  Latched

Severity: 500  Acknowledge required

Minimum duration: 0 Seconds  Show Alarm as a Tag

Message: Labeler Out of Papers

ID: 4 New... Edit... Browse...

Tag Name
Tag1
Tag2
Tag3
Tag4

Alarm Class:

FactoryTalk View Command: display med\_labeling

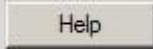
OK Cancel Help

Notice that the configuration of the tag-based Digital alarm is very similar to the device-based ALMD with the following exceptions:

- **Input Tag** – This is the tag that is monitored for the alarm condition.
- **Show Alarm as Tag** – This option allows you to monitor the status and operate on alarms programmatically at run time using tags that are exposed by the Tag Alarm and Event server. Live Data clients can read and write to the alarm tags to monitor and change alarm states.
- **Status Tags** – Status tags allow you to keep alarm state changes synchronized with a programmable controller. For example, if you assign a tag to the Acknowledged status tag, when an operator acknowledges the alarm from the Alarm Summary, the status tag is set to 1

- **Control Tags** – Control tags allow you monitor and control alarms by using tags in the controller. For example, if you assign a tag to the Suppress control tag, when the value of the tag changes to positive, the Tag Alarms and Events Server automatically Suppresses alarms associated with the control tag.

Like the device-based alarms, there are many properties of a **Digital Alarm** that you can configure within the **Digital Alarm Properties** dialog box. If you'd like more information about them and what they do, click **Help**



on this dialog box.

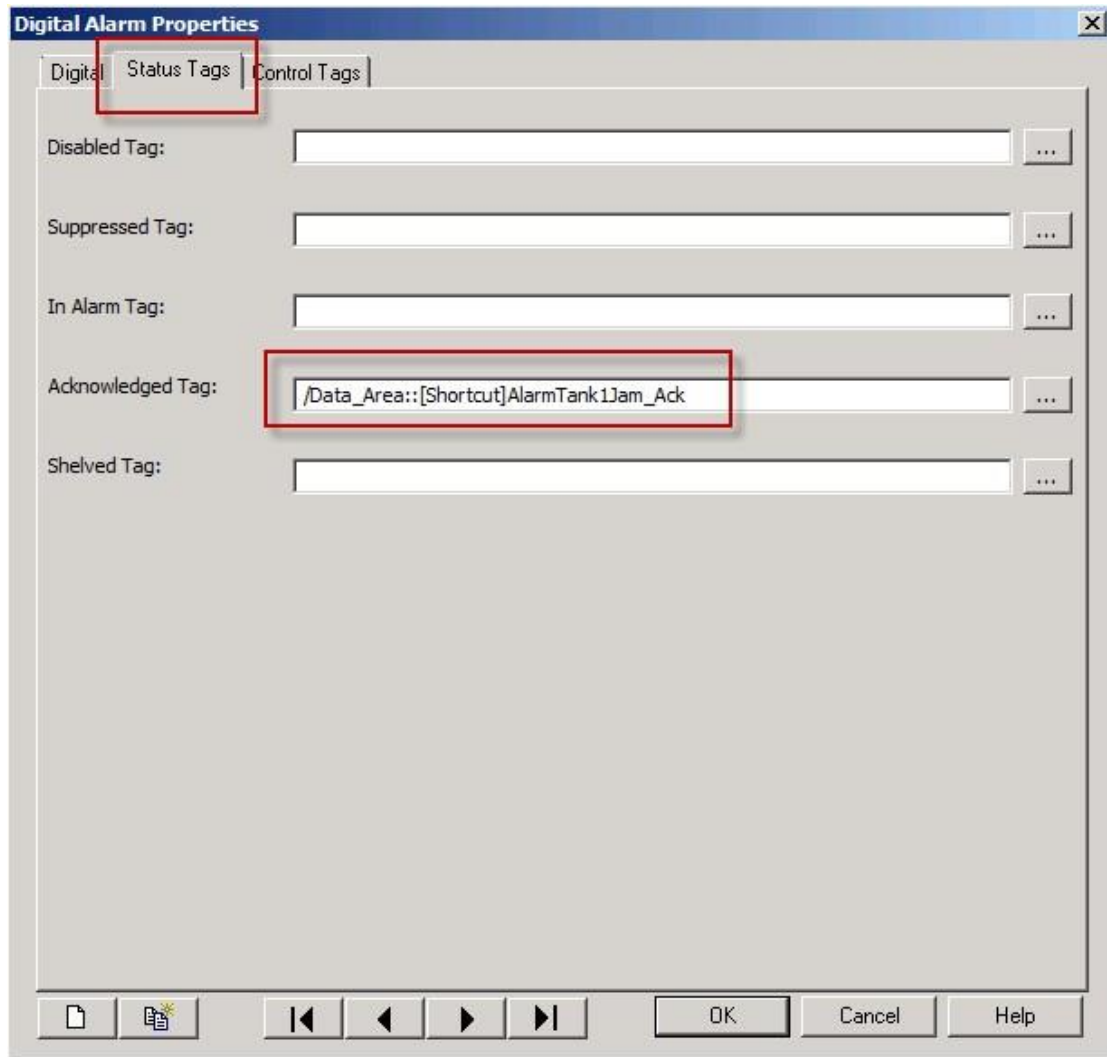
7. Select and enable the **Show Alarm as a Tag** checkbox.



This will allow us to browse the alarm parameters in the tag browser later on in the section.

8. Click **OK**  to close the **Digital Alarm Properties** dialog box.

9. In the **Alarm and Event Setup** window select and double-click the **Digital** alarm in the list named **Tank1Jam** and select the **Status Tags** tab.



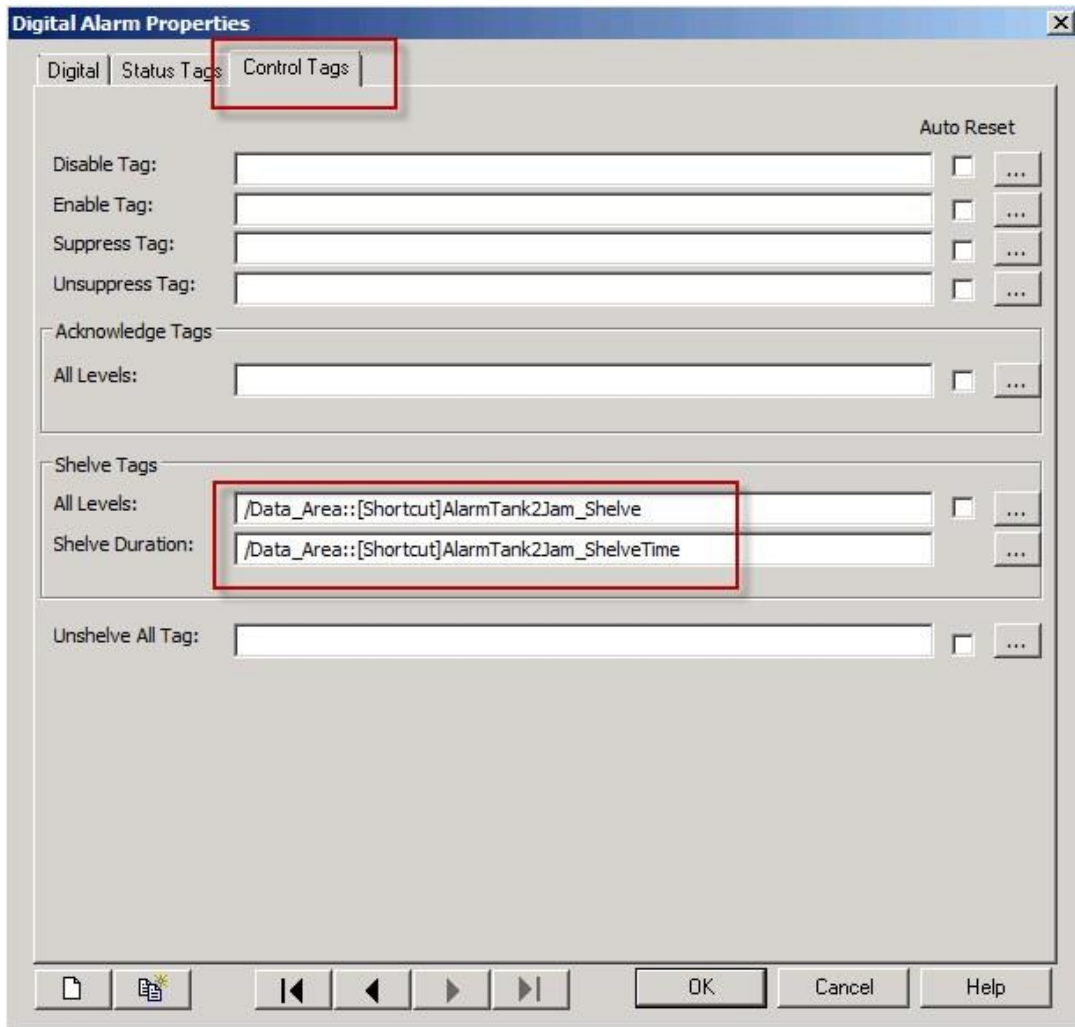
Notice the tag assigned to the Acknowledged Tag is a Controller tag. We'll be monitoring this tag on the Alarms display later on

10.

Click **OK**  to close the **Digital Alarm Properties** dialog box.

11.

In the **Alarm and Event Setup** window, select and double-click the **Digital** alarm in the list named **Tank2Jam** and select the **Control Tags** tab.



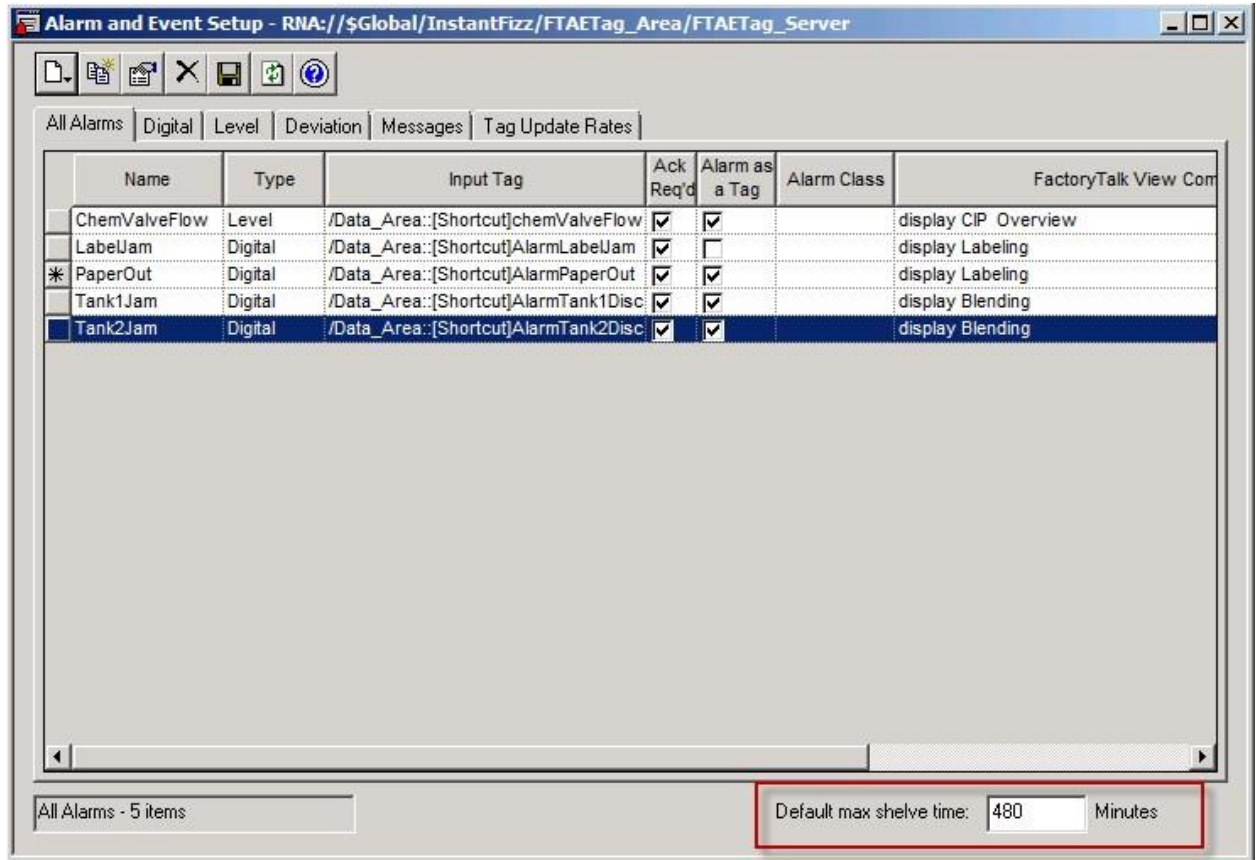
Alarm control tags allow you to monitor alarms by using tags in the controller. If you assign a tag as the control tag, when the value of the tag changes to positive, the Tag Alarms and Events Server automatically acknowledges, disables, enables, suppresses, unsuppresses, shelves or unshelves all alarms associated with the control tag.

When using Shelve Control tags, you must also define the Shelve Duration tag and set it to the amount of time in minutes for the alarms to be shelved. For this example, the ShelveTime tag is set to 2 as shown in the Logix controller tags

AlarmTank2Jam	0
AlarmTank2Jam_Shelve	0
AlarmTank2Jam_ShelveTime	2

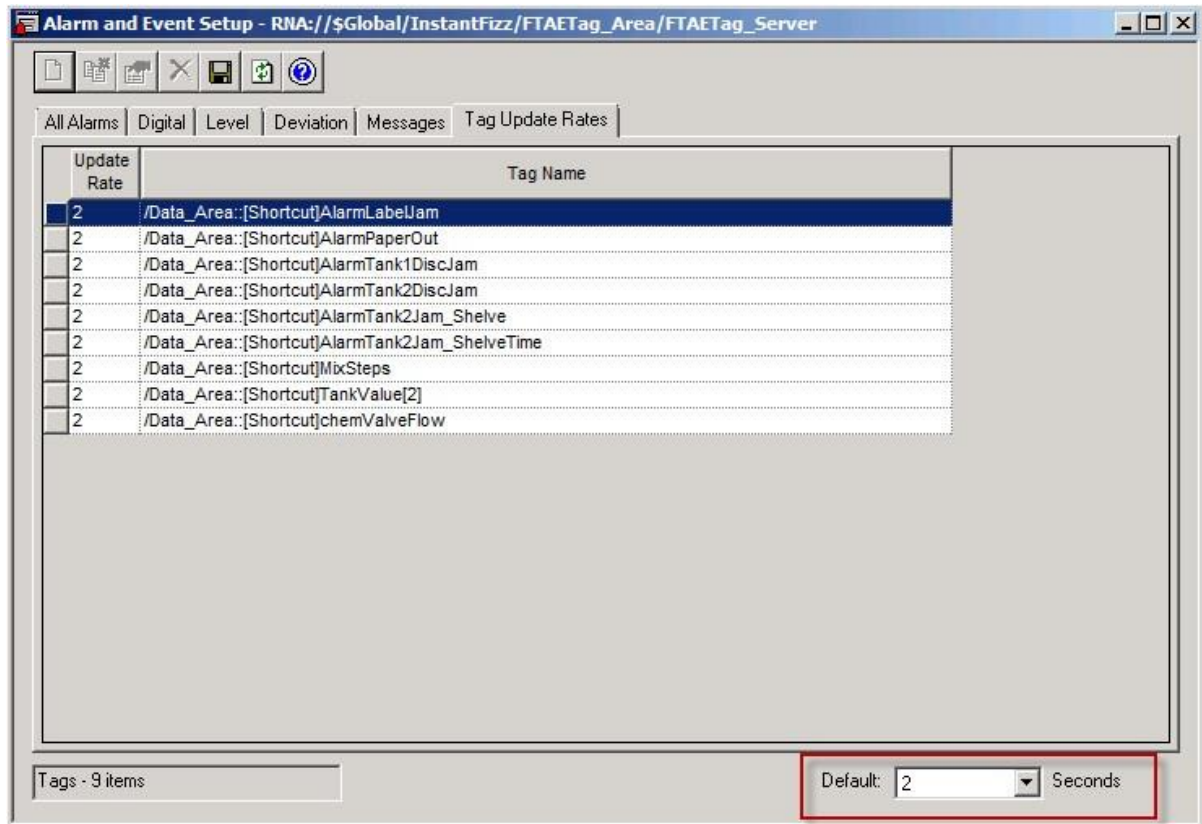


12. Click **OK** to close the **Digital Alarm Properties** dialog box.



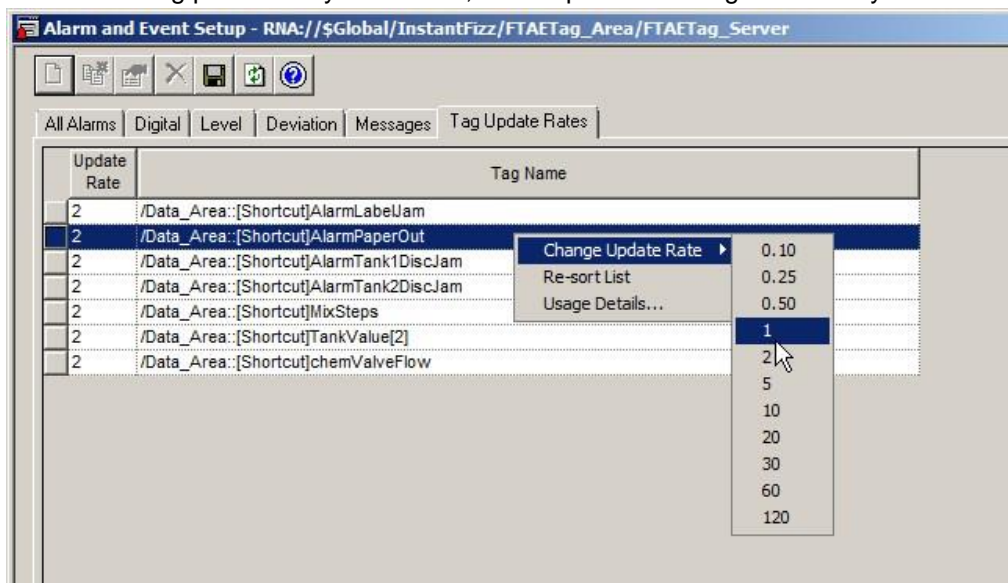
The global default max shelf time for all tag-based alarms is 480 minutes. This value can be set from 1 to 2147483647. You'll be interacting with the shelf time a litter later.

13. In the **Alarm and Event Setup** window, select the **Tag Update Rates** tab.



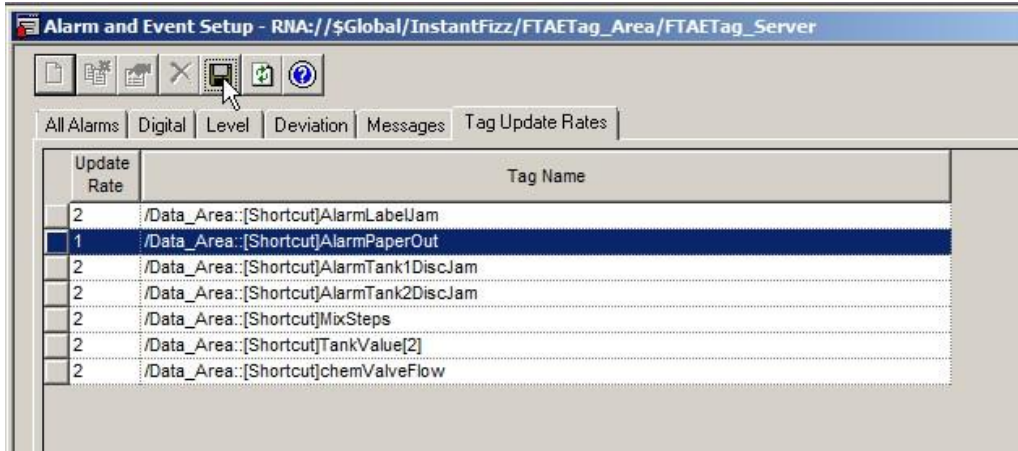
The default update rate for the tag-based alarms is 2 seconds. This means that tag values are polled and evaluated for alarm status every 2 seconds. The default rate can be modified to a value between .10s – 120s and will affect all new alarms defined afterwards. The update rate for alarms already configured can be changed individually as desired.

14. Select the **AlarmPaperOut** alarm and right-click. Select **Change Update Rate** and select **1**. Instead of being polled every 2 seconds, we will poll for this tag value every second.




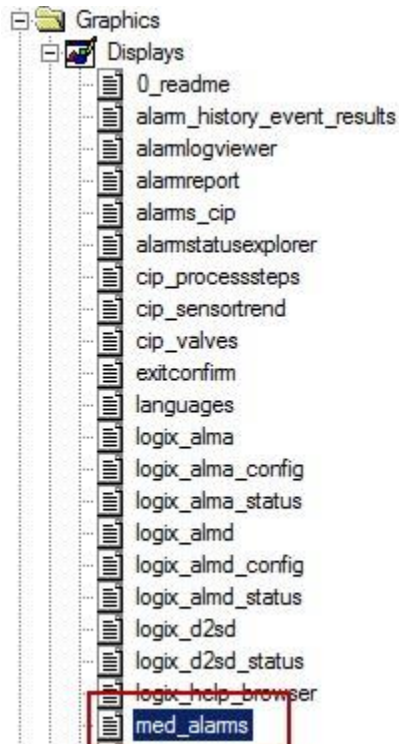
Be cautious when selecting higher-speed tag update rates such as 0.10 and 0.25. It does not mean that new alarms will appear on the alarm summary screen faster. Alarms can only be polled as fast as the data server allows, depending on its load with the rest of the application. Additionally, the alarm summary screen refreshes at its own update rate; tag-based alarm server Tag Update Rates affect only the time stamp of the alarm.

15. Save the changes to the Alarm and Event Setup by clicking the **Save** button,  on the toolbar.

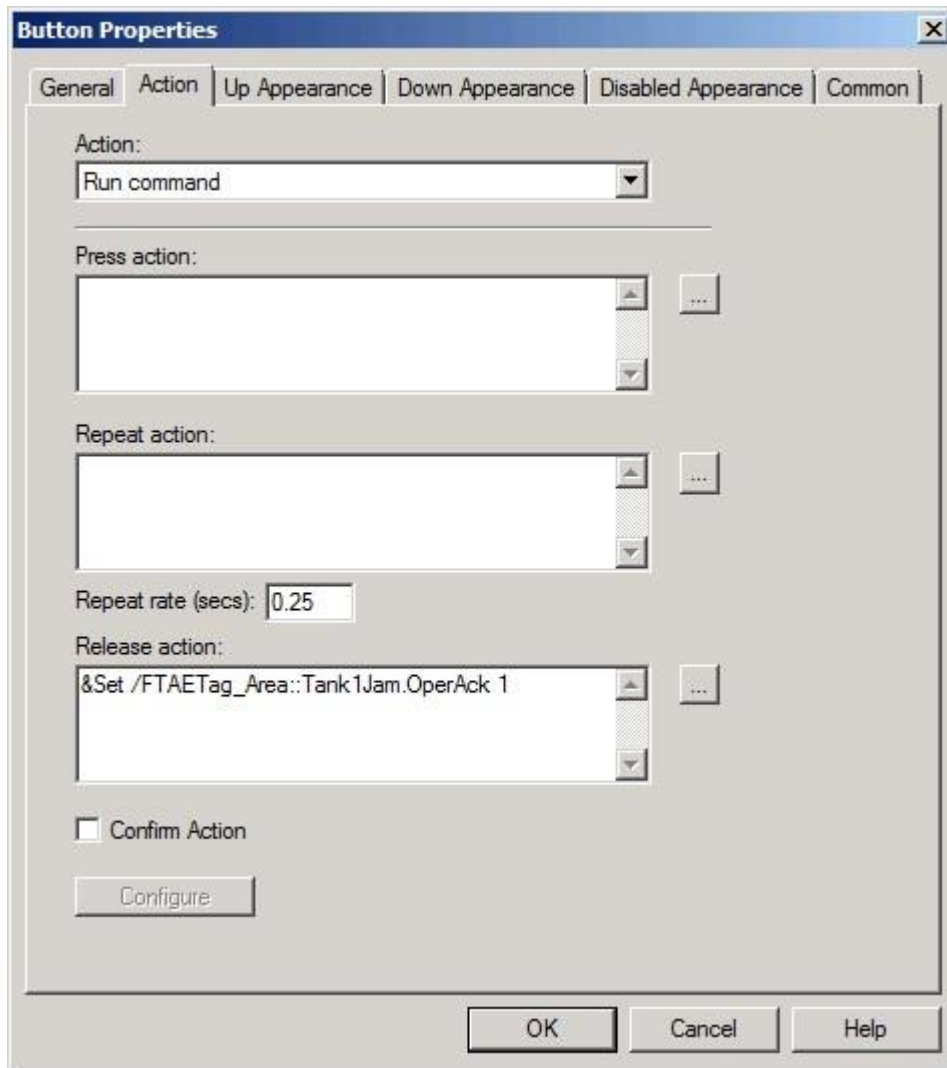


Clicking Save immediately loads any alarm edits or new alarms to the Alarms and Events server.

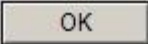

16. Click the **X**  button in the corner to close the **Alarm and Event Setup**.
17. In the **Explorer** window of Factory Talk View Studio, double-click the **med\_alarms** screen located under the HMI server node in the **Graphics > Displays** folder.





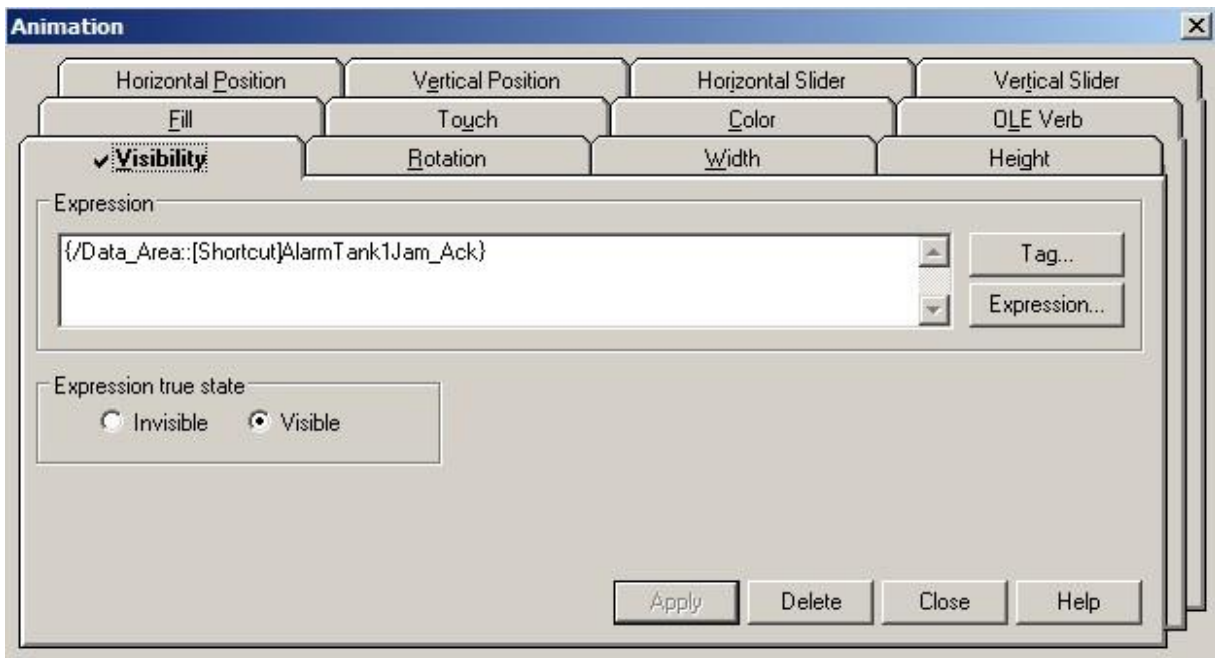


Notice the **Release action**: When the button is pressed and released, the **Tank1Jam** tag based alarm will be acknowledged without interacting with the Alarm and Event Summary!

21. Click **OK**  to close the **Button Properties** dialog box.
22. On the **med\_alarms** screen, right-click the **state\_activeacked** image  on the **Acknowledge** button and select **Animation > Visibility** from the context menu.



Note this is the same tag used as a **Status Tag** for **Tank1Jam** digital alarm in the **Alarm and Event Setup**. This image will be visible when Tank 1 is acknowledged and the Status tag associated with **Tank1Jam** alarm is set to 1



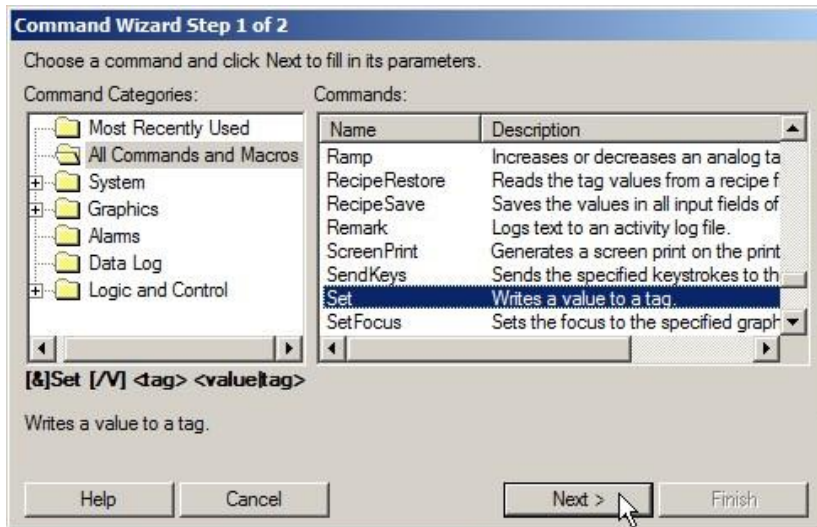
23. Click the **Close** button to close the **Animation** dialog.

Next, we are going to configure a new Alarms and Events feature, **Shelving**.

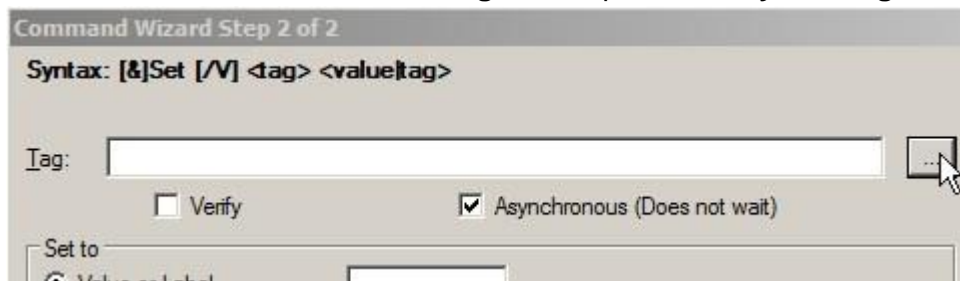
24. On the alarms screen, select and double-click the **Shelve** button below the **Tank 2 Discharge Valve** button. We are going to configure this button to shelve the **Tank2Jam** alarm from the Control tag we observed in the Alarm and Events setup





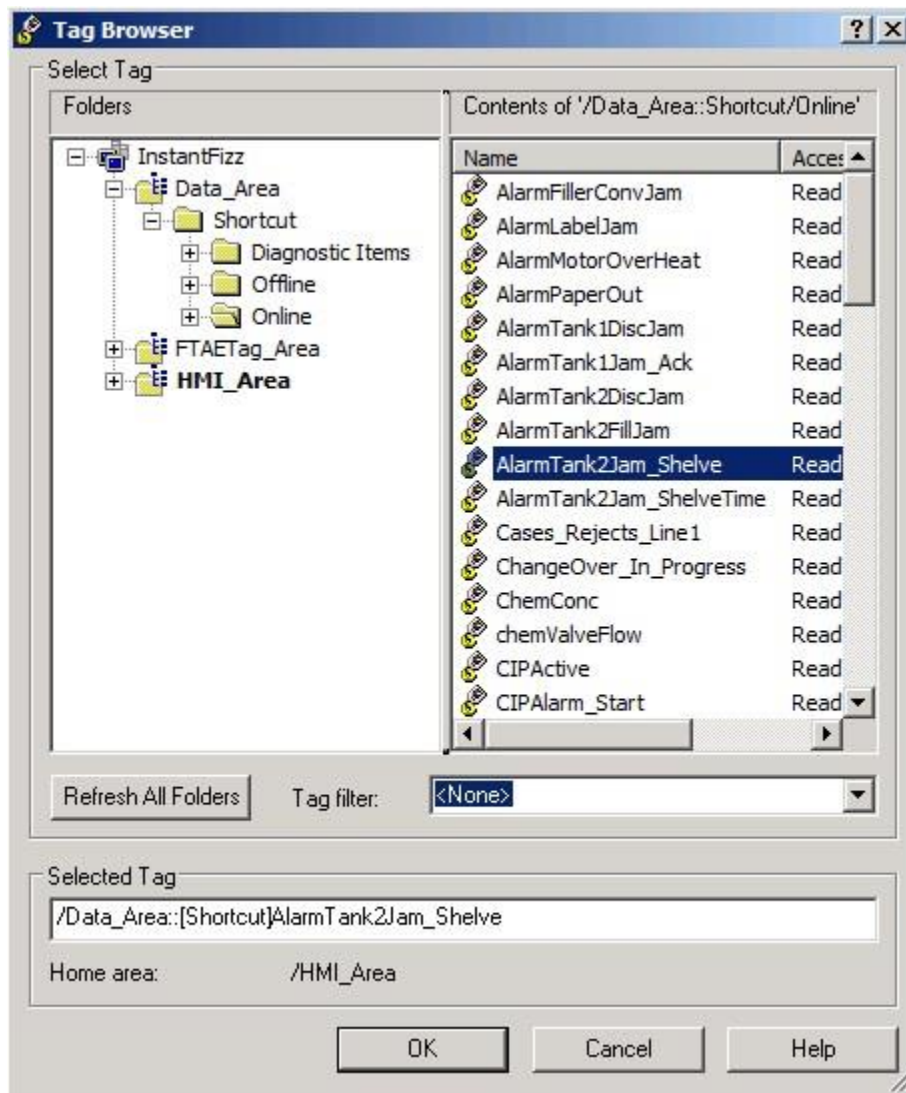


27. Click the **Browse** button  next to the **Tag** field to open the **FactoryTalk Tag Browser**

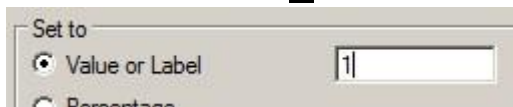


28. Right-click on the **InstantFizz** application name and select **Refresh All Folders**.

29. Browse under the **Data\_Area** folder to select **Online** and select **AlarmTank2Jam\_Shelve** from the **Contents** panel.

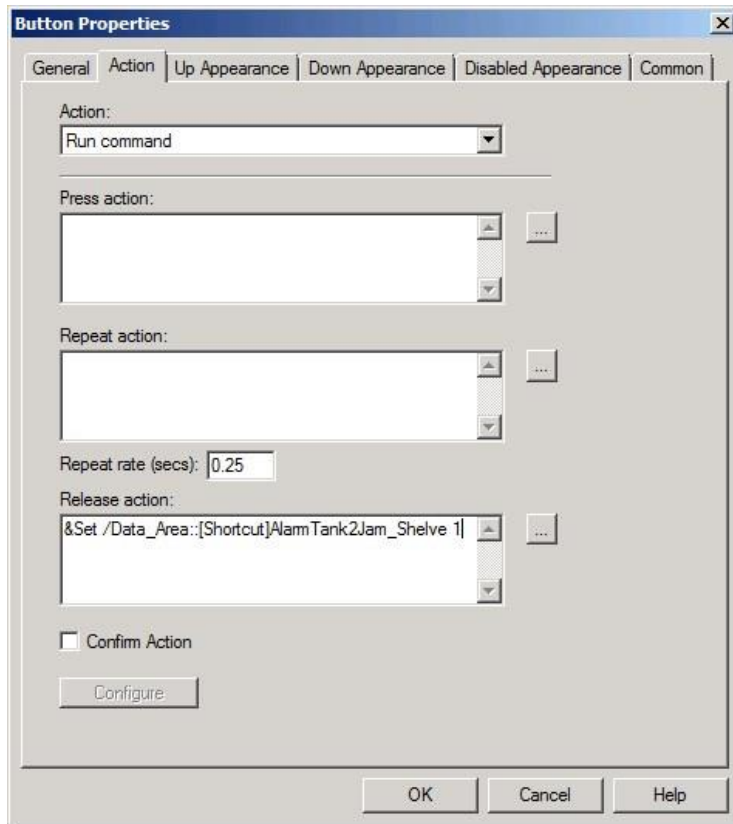





30. Set Value or Label to **'1'**.



31. Click **Finish**  to exit the Command Wizard.

The resulting Button Properties window will look like this:



32. Click **OK** to close and save the Button properties.
33. Save the display by clicking the **Save** button  on the toolbar.
34. On the **Graphics** toolbar, click **Test Display** button . Since a simulation program is running, the display will be filled with alarms.
35. Select the **Acknowledge All** icon  from the Alarm Summary toolbar and select **Acknowledge** on the following **Acknowledge Alarm with Comment** confirmation dialog.  
The resulting display will probably look something like this:

The screenshot shows the 'ALARMING' window with the following sections:

- Turn on digital device-based CIP Valve alarms** (with a red alarm bell icon)
- Turn off digital device-based CIP Valve alarms** (with a blue alarm bell icon)
- Simulate Motor Temperature alarm (analog device-based alarm)** with buttons for HIHI Limit, HI Limit, LO Limit, and Turn off.
- Simulate tag-based alarms** with buttons for Labeler out of paper, Tank 1 Discharge Valve, and Tank 2 Discharge Valve. Each has an Acknowledge and Turn off button.
- Alarm and Event Log Viewer**, **Alarm and Event Status Explorer**, and **Alarm and Event History Results** buttons.
- Alarm Summary List:**

Event Time	Alarm Name	Message
4/29/2014 3:53:58 PM	[Shortcut]CIPValve1	Water Valve Jammed. Call Maintenance Staff.
4/29/2014 7:11:33 PM	[Shortcut]Program:Main.Blending_Discharge_J...	Blending Tank 1 Discharge Valve Jammed
4/29/2014 7:13:26 PM	[Shortcut]Program:Main.Filling_Valve_3a	Filter Valve 3a Blocked. Please open Valve.
4/29/2014 7:14:44 PM	ChemValveFlow	The Chemicals valve flow is very low. Increase the valve flow. The chemicals tank level is: 8000
4/29/2014 7:14:44 PM	[Shortcut]CIPValve0	Recovered Water Valve Jammed. Open valve.
4/29/2014 7:14:44 PM	[Shortcut]CIPValve2	Chemicals Valve Jammed. Open valve.
4/29/2014 7:14:44 PM	[Shortcut]CIPValve3	Sanitizer Valve Jammed. Open valve.
4/29/2014 7:14:44 PM	[Shortcut]MotorOverheatAlarm	The Motor is cooling down. The current temperature of the motor is 25
4/29/2014 7:14:44 PM	[Shortcut]Program:Main.Filling_Valve_6b	Filter Valve 6b Blocked. Please open Valve.
4/29/2014 7:14:44 PM	[Shortcut]Program:Main.Labeler_Motor	Labeler Motor is overheating.

Note: The simulation is still running so more alarms will keep appearing, but with the majority of the alarms already acknowledged, it will make it easier to search the list to find the alarms we will trigger in the following steps.


36. To simulate some tag-based Digital alarms, click the 3 buttons shown below, rapidly in succession.



Notice that the new alarms will appear in the alarm summary list after 3-4 seconds.

		9/12/2014 6:17:30 PM	PaperOut	Labeler Out of Papers
		9/12/2014 6:17:30 PM	Tank1Jam	Blending Tank 1 Discharge Valve Jammed
		9/12/2014 6:17:30 PM	Tank2Jam	Blending Tank 2 Discharge Valve Jammed

FactoryTalk Alarms and Events capture the time of the event on the alarm server (for tag-based alarms) or in the controller (for device-based alarms) at the moment when the alarm condition was observed. It is this time stamp that is sent with the alarm data and populated in any of the alarm objects. Although there may be a delay in displaying the alarm data, the time and date at which it is being reported is accurate.

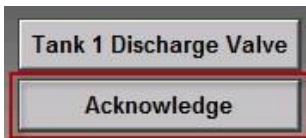
37. Select the **PaperOut** alarm and acknowledge by clicking the **Acknowledge** icon  on the **Alarm Summary** toolbar.


Event Time	Alarm Name	Message
5/8/2014 1:14:56 AM	[Shortcut]CIPValve1	Water Valve Jammed. Open Valve
5/8/2014 1:39:23 AM	ChemValveFlow	The Chemicals valve flow is very low. Increase the valve flow. The chemicals tank level is
5/8/2014 1:39:23 AM	[Shortcut]CIPValve0	Recovered Water Valve Jammed. Open valve.
5/8/2014 1:39:23 AM	[Shortcut]CIPValve2	Chemicals Valve Jammed. Open valve.
5/8/2014 1:39:23 AM	[Shortcut]CIPValve3	Sanitizer Valve Jammed. Open valve.
5/8/2014 1:39:23 AM	[Shortcut]MotorOverheatAlarm	The Motor is cooling down. The current temperature of the motor is25
5/8/2014 1:39:32 AM	Tank1Jam	Blending Tank 1 Discharge Valve Jammed
5/8/2014 1:39:33 AM	Tank2Jam	Blending Tank 2 Discharge Valve Jammed
5/8/2014 1:39:48 AM	[Shortcut]Program.Main.Filling_Valve_3a	Filler Valve 3a Blocked. Please open Valve.
5/8/2014 1:40:50 AM	[Shortcut]Program.Main.CIP_Supply_Motor	CIP Supply. Motor is overheating.
5/8/2014 1:40:53 AM	PaperOut	Labeler Out of Papers

The alarm will change state from  to . (The blue highlight is because the alarm is still selected).

The **Alarm Summary** object and its toolbar is typically how operators will interact with alarms. In some cases, however, it is desired that operators be able to Acknowledge, Disable, or Suppress alarms from a custom screen or a faceplate.

- Select the **Acknowledge** button that you viewed earlier, under the **Tank 1 Discharge Valve** button.



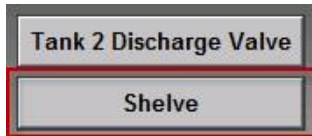
Notice that the **Tank1Jam** alarm has been acknowledged , requiring no interaction with the **Alarm Summary** object.



Also, notice that the image you viewed with Visibility animation is now shown. Status tags allow Tag based alarm states to be recognized by the controller!



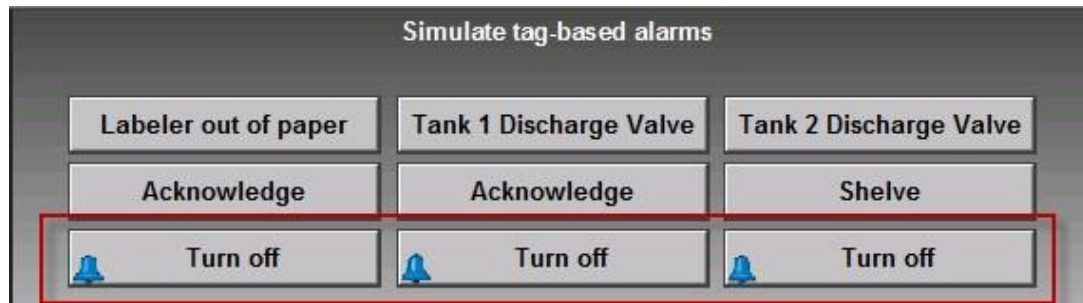
- Select the **Shelve** button that you configured earlier, under the **Tank 2 Discharge Valve** button.



Notice that the **Tank2Jam** alarm now has a **Shelved** symbol.



- Click the **Turn off** buttons for each of the three alarms to return the alarm conditions to normal.



Observe that the 2 acknowledged alarms (**PaperOut** and **Tank1Jam**) will be removed from the alarm list, while **Tank2Jam** will change from **Active Shelved** to **Normal Shelved**.

- On the **Graphics** toolbar, stop testing the display by clicking the **Edit Display** button.
- Close the **Alarms** display by selecting **File > Close** from the FactoryTalk View Studio menu. Press **Yes** when asked to save changes.

Congratulations! You have just

- Observed and made online edits to tag-based alarms.
- Enabled a tag-based alarm to be seen as a tag in the tag browser.
- Changed the update rate for a tag-based alarm.
- Configured buttons to remotely acknowledge and remotely shelve a tag-based alarm
- Acknowledged alarms at runtime via the alarm summary object and remotely via a button.
- Shelved an alarm remotely via a button.

In the next section, we will enable Alarm History and explore ways to query the historical database.

### Setting up historical alarm and event logging

FactoryTalk Alarms and Events can log alarm and event information to a database so that the information can be viewed or reported on at a later date.

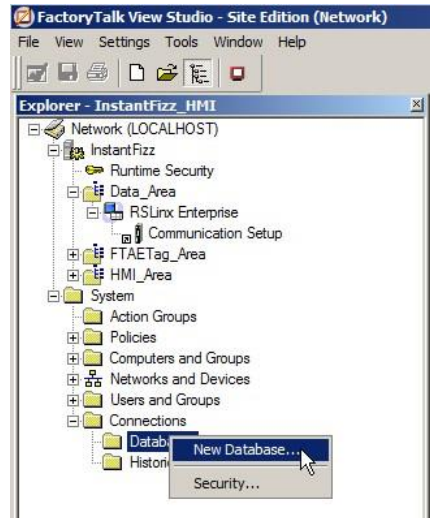
The Alarm and Event Log Viewer object is used to view the historical alarm and event information in a simple grid view.

In this section of the lab you will:

- Set up historical alarm and event logging.
- View historical events using the Log Viewer object.

## Add a database definition to the FactoryTalk system

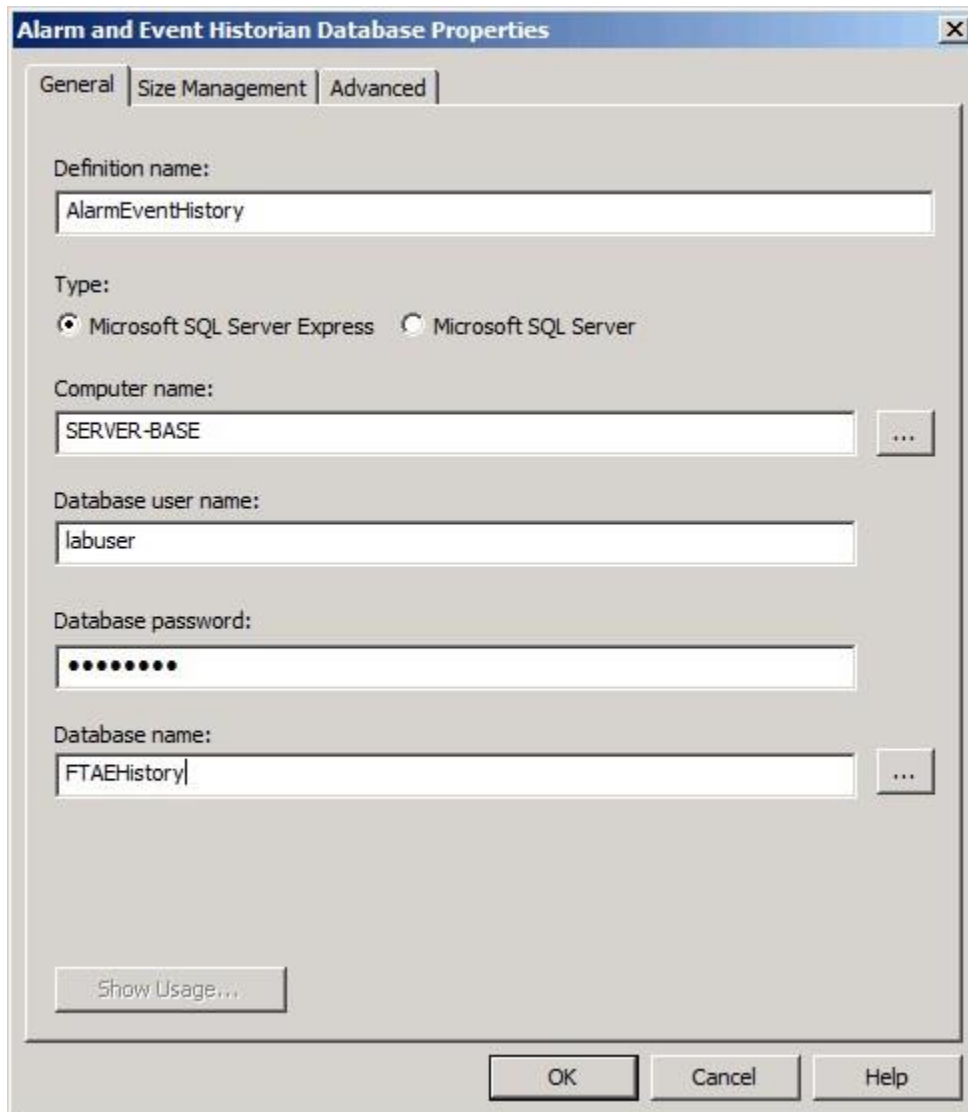
1. In the **FactoryTalk View Studio Explorer** window, expand the folders **System** > **Connections**, and then rightclick the **Databases** folder. On the context menu, click **New Database**.



2. In the **Alarm and Event Historian Database Properties** dialog box:
  - For the Definition name, type 'AlarmEventHistory'.
  - For the Database user name, type 'Labuser'.
  - For the Database password, type 'rockwell'.
  - For the Database name, type 'FTAEHistory'.

When you are finished, the **Alarm and Event Historian Database Properties** dialog box should look as follows:

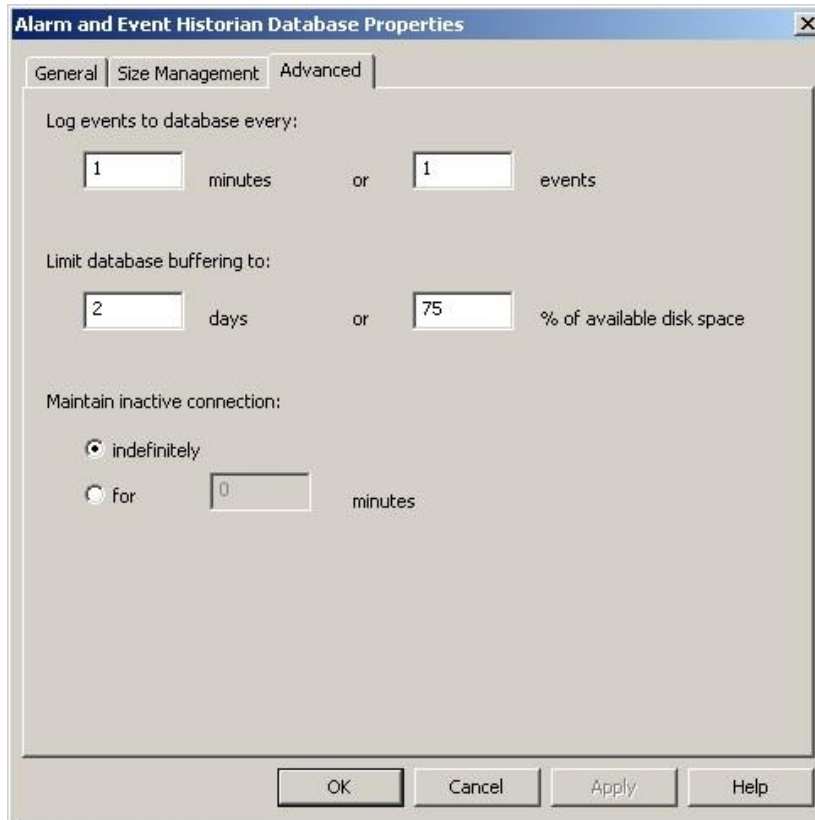




3. Click on the **Advanced** tab. The system will attempt to create the database and the user account. On the **Database does not exist** dialog box, click the **Yes** button.



4. In the **Advanced** tab, change the **Log events to database every** setting from 100 to 1. This change is just for testing purposes so that we do not have to wait for information to be logged to the database.



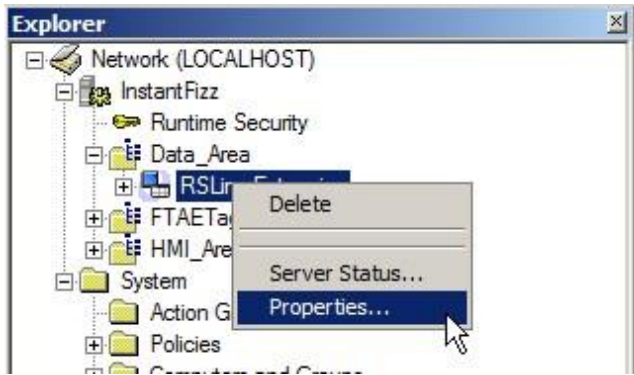
The Log events to database every settings control how many events are cached before sending them to the database. This makes writing to the database more efficient. The events are cached to a file located on the computer hosting the alarm server that is generating the alarm and event information. If the connection to the database is lost, alarm and event information will continue to be cached to files. Once the connection is restored, the cached information will be sent to the database. The Limit database buffering to settings control how much information will be cached if the database connection is lost for a long period of time. Once the limits are reached the oldest information is overwritten.

5. Click **OK** to save the Database Definition.

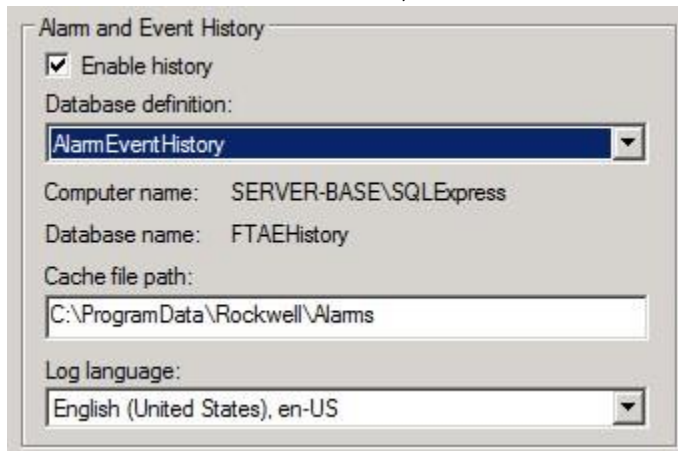
### Associate database definition with alarm servers

#### For device-based FactoryTalk Alarms and Events servers:

1. Right-click the alarm server named **RSLinx Enterprise**. (**RSLinx Enterprise** is both a data server and an alarm server). On the context menu, click **Properties...**

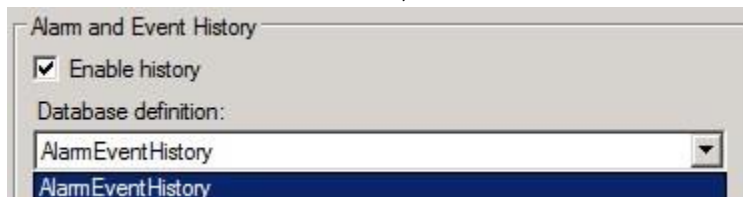


2. In the **RSLinx Enterprise Server Properties** dialog box, click the **Alarms and Events** tab.
3. Under **Alarm and Event History**, select the **Enable history** check box.
4. In the **Database definition** list, select **AlarmEventHistory**.



5. Click **OK** to save the changes.
- For tag-based FactoryTalk Alarms and Events servers:**

1. Right-click the alarm server named **FTAETag\_Server**. On the context menu, click **Properties...**
2. In the **FTAETag\_Server Properties** dialog box, click the **Priorities and History** tab.
3. Under **Alarm and Event History**, select the **Enable history** check box.
4. In the **Database definition** list, select **AlarmEventHistory**.



5. From the **Log Language** definition list, select **English (United States), en-US**.



6. Click **OK** to save the changes.

The Log language setting is related to the Language Switching feature which allows you to configure multiple languages for user defined strings that can be switched between at runtime. The combo box allows you to select the language to log alarm messages in (only one language is supported for the alarm and event log).

Congratulations! You have just configured the both FactoryTalk Alarms and Events servers to send their alarm and event information to the database. We will view this historical data in the Alarm Log Viewer in just a little while.

## Using FactoryTalk Alarm and Event Objects

To monitor and respond to FactoryTalk device-based and tag-based alarms at run time, the following FactoryTalk Alarm and Event objects are available and can be embedded into FactoryTalk View SE graphic displays.

- Alarm and Event Summary
- Alarm and Event Banner
- Alarm and Event Log Viewer
- Alarm and Event Status Explorer

HMI tag alarm monitoring and FactoryTalk Alarms and Events are two separate alarm monitoring systems that do not share alarm information with each other. FactoryTalk tag-based and device-based alarm information can only be displayed in the FactoryTalk Alarm and Event objects. FactoryTalk View HMI tag alarm information cannot be displayed in FactoryTalk Alarm and Event objects and the HMI tag alarm monitoring objects remain available for compatibility with existing applications.

In this section, you will observe all of the FactoryTalk Alarm and Event objects in both runtime via the SE client and design time via FactoryTalk View Studio. You should have both FactoryTalk View Studio and the Client opened.

## Monitor and Interact with Real time Alarms

### Alarm and Event Summary

Use the Alarm and Event Summary object, embedded in a FactoryTalk View graphic display, to acknowledge, disable, suppress, shelve, unshelve all shelved alarms, filter, and sort alarms during run time. Note: You must use the Alarm Status Explorer to unsuppress an alarm that has already been suppressed or unshelve a single alarm that has been shelved. **Alarm and Event Banner**

The Alarm and Event Banner displays only the highest priority, most severe, most recent alarms in the following order:

- Alarms whose alarm condition is In Alarm and Unacknowledged.
- Alarms whose alarm condition is In Alarm and Acknowledged.
- Alarms whose alarm condition has returned to Normal but remain Unacknowledged. (Not displayed by default.)

Use the Alarm and Event Banner object, embedded in a FactoryTalk View graphic display, to monitor and respond to the most current alarms requiring immediate attention.

### Alarm and Event Status Explorer

Use the Alarm Status Explorer object, embedded in a FactoryTalk View graphic display, to enable or disable alarms and suppress or unsuppress alarms and shelve or unshelve alarms


**Runtime Client features: Alarm and Event Summary, Alarm and Event Banner and Alarm and Event Status Explorer objects.**

### Alarm and Event Summary



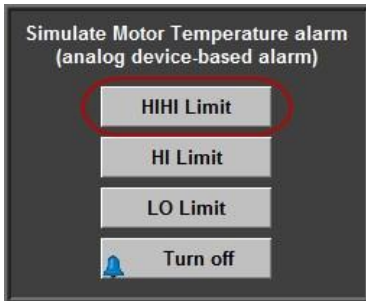
1. Double-click on the FactoryTalk View SE Client file, *IF\_Client\_1280x1024.cli* from the Desktop to launch the **InstantFizz** application. (For convenience, a shortcut to *IF\_Client\_1280x1024.cli* has also been added to the Start menu.)



2. From the FactoryTalk View SE client, click the  button in the navigation display.
3. The **ALARMING** display is opened containing an Alarm and Event Summary object.

Let's trigger some alarms that are associated with the CIP process.

4. Click on the **HIHI Limit** button to trigger some alarms. (It may take a few seconds to generate the alarms.)



5. Select a row in the summary list such as **MotorOverheatAlarm** to see details about the alarm in the details pane. Scroll down if necessary.

Priority:	Urgent	Severity:	1000
Alarm State:	In Alarm, Unacked	Current Value:	1001
Event Time:	5/8/2014 2:18:09 AM	Limit Value Exceeded:	1000
In Alarm Time:	5/8/2014 2:18:09 AM	Tag 1 Value:	1001
Acknowledge Time:		Tag 2 Value:	
Out of Alarm Time:		Tag 3 Value:	
Condition Name:	HIHI	Tag 4 Value:	
Event Category:	Level	Alarm Count:	2

Let's take a quick look at the toolbar to see how the operator can interact with the Alarm Summary

6. Mouse over the toolbar on the Summary to see what the configured buttons do (read the tool tips). These buttons can be executed on a single, selected alarm or multiple selected alarms. Here you can see we are looking at the Shelve tool tip. The Shelve and Unshelve features were added in FactoryTalk View v7.0.



Observe also the Status Bar at the bottom of the Summary. Mouse over those icons to see what information it is telling you (read the tool tips).




The Status Bar is a quick way to see you connection status and the number of alarms in different states

7. Select a row in the Summary such as **CIPValve1** and click on the run alarm command



button or doubleclick on the row in the Summary. The FactoryTalk View Command defined in the selected alarm configuration is then executed.



Note: If the button is gray  there is no FactoryTalk View command associated with that alarm or a row is not selected. Select another row and try it again. For example, the CIPValve1 alarm properties use the FactoryTalk View Command field to define Display commands and pass parameters as shown in the example below.

When you double-click on the **CIPValve1** alarm, you will then have a popup and faceplate display open with passed parameters since both were defined in the FactoryTalk View command of the alarm property.

**ALMD Properties - CIPValve1 (Rung 0)**

Configuration\* | Status | Parameters | Tag

Condition:   Latched  
Severity:   Acknowledgement Required  
Minimum Duration:  ms  
Message:  ...

Associated Tags

	Name	Type	Description
1			
2			
3			
4			

Alarm Class:

**FactoryTalk View Command:**

Status: OK

Alarm: Normal   Disabled   
 Acknowledged   Suppressed

Delivery: Done





Feel free to experiment by clicking more of the buttons in the tool bar to see what they do. Next we'll move on to using the Summary object in design time.

8. Close the two pop-up displays in order to view the Alarm and Event Summary once again.

### Alarm and Event Banner

Now, let's take a look how the **Alarm and Event Banner** displays the FactoryTalk Alarms and Events.

Observe that alarms appear both in the Alarm Banner located on the header of your client, as well as in the Alarm Summary.

**Rockwell Automation** 5/8/2014 2:18:09 AM [Shortcut]MotorOverheatAlarm HIHI The Motor has overheated. Motor has stopped. The current temperature of motor. 5/8/2014 2:25:02 AM [Shortcut]Program Main Labeler\_Motor HI Labeler\_Motor is overheating.

6 6 0

OVERVIEW BLENDING RAPID MIX FILLING LABELING PACKAGING CIP **ALARMS** MORE... VANTAGEPOINT ALARM REPORTS

## ALARMING

Thursday, May 08, 2014 2:25:21 AM SERVER01\LABUSER

Turn on digital device-based CIP Valve alarms

Turn off digital device-based CIP Valve alarms

Simulate Motor Temperature alarm (analog device-based alarm)

HIHI Limit

HI Limit

LO Limit

Turn off

Simulate tag-based alarms

Labeler out of paper	Tank 1 Discharge Valve	Tank 2 Discharge Valve
Acknowledge	Acknowledge	Shelve
Turn off	Turn off	Turn off

Alarm and Event Log Viewer

Alarm and Event Status Explorer

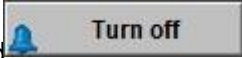
Alarm and Event History Results


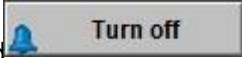
(No Filter)

Event Time	Alarm Name	Message
5/8/2014 1:14:56 AM	[Shortcut]CIPValve1	Water Valve Jammed. Open Valve
5/8/2014 1:39:23 AM	ChemValveFlow	The Chemicals valve flow is very low. Increase the valve flow. The chemicals tank level is: 14000
5/8/2014 1:39:23 AM	[Shortcut]CIPValve0	Recovered Water Valve Jammed. Open valve.
5/8/2014 1:39:23 AM	[Shortcut]CIPValve2	Chemicals Valve Jammed. Open valve.
5/8/2014 1:39:23 AM	[Shortcut]CIPValve3	Sanitizer Valve Jammed. Open valve.
5/8/2014 1:47:28 AM	Tank2Jam	Blending Tank 2 Discharge Valve Jammed

**Note:** The text and background colors have been pre-defined in this application. You may note that these colors differ from the default settings used in the

topic.

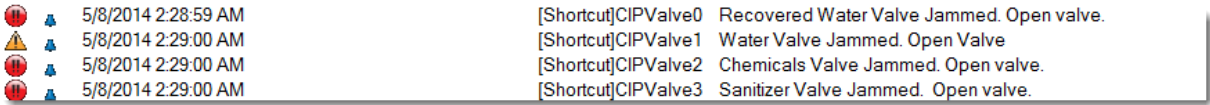



9. Now click the  and  buttons to programmatically turn off some of the alarms.

Observe how the alarm banner now shows higher priority, more severe or more recent alarms at the top of the list giving the user quick access to the most critical alarms.



The Alarm Summary object maintains returned to normal alarms in the list giving the operators an opportunity to interact with them at a later time.



10. For the next step, click the  button to enable alarms again.



11. Navigate to **CIP** display.

12. Now, select a row in the Alarm Banner in the header display and **double-click**.



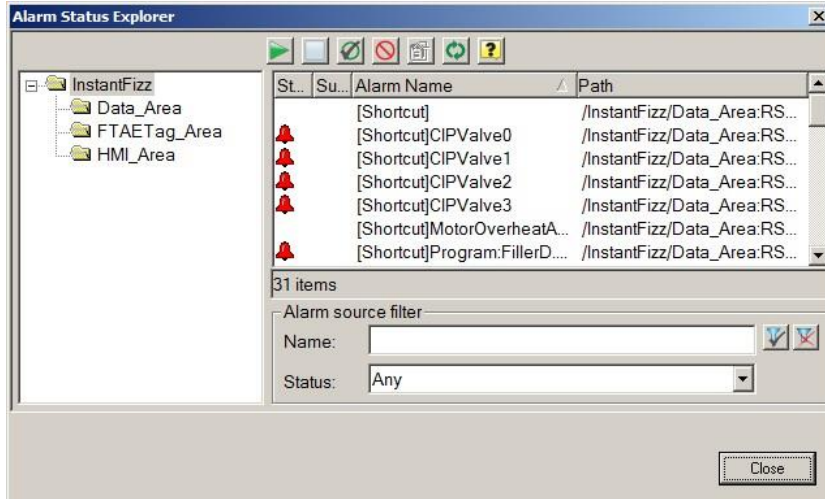
13. Observe that the Alarm and Event Summary display is automatically launched. Note that this behavior is configurable and will be discussed on the next section.

### Alarm Status Explorer

Let's take a look at the **Alarm Status Explorer** launched from the Alarm and Event Summary and view the Status Explorer placed in a graphic display.



14. Click the **Alarm Status Explorer** icon from the Alarm Summary toolbar to launch a default Alarm Status Explorer object.

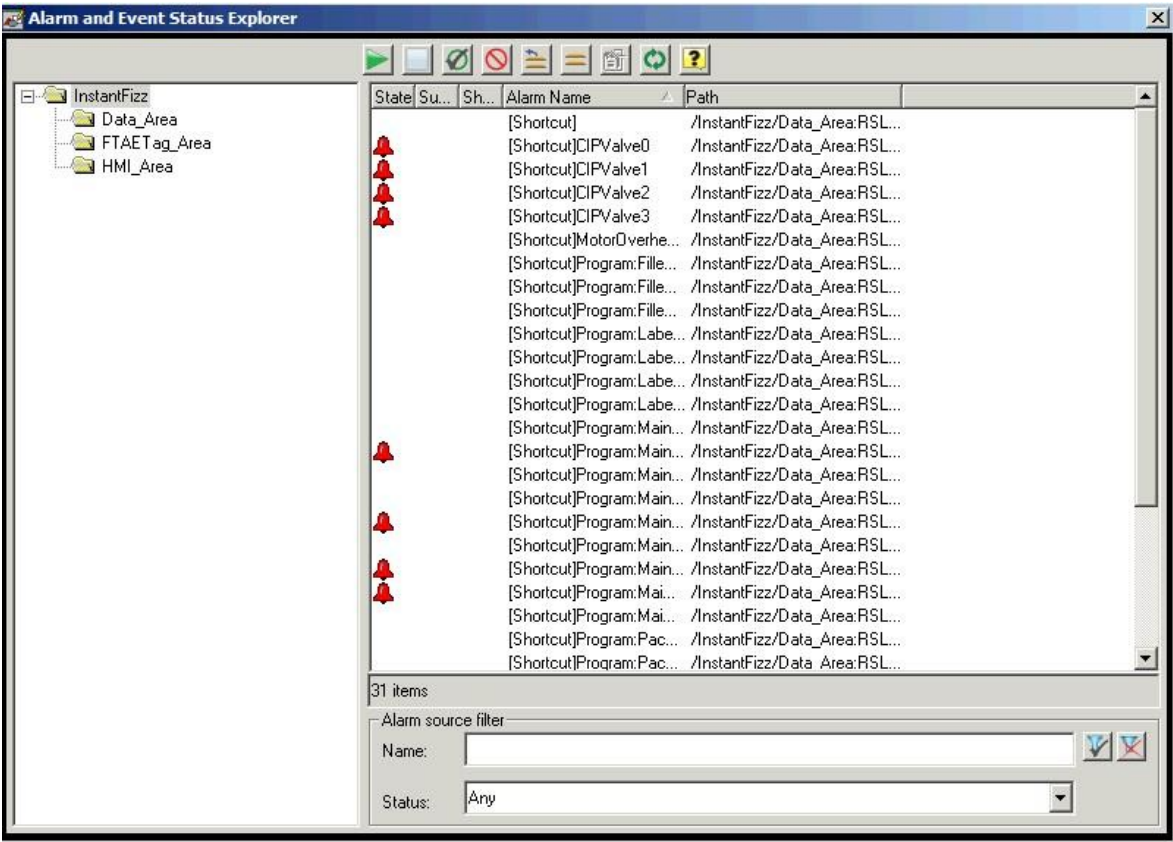


To configure the default Alarm and Event Status Explorer object launched from the Alarm Summary toolbar, use the **Toolbar** configuration settings on the **Appearance** tab of the **Alarm and Event Summary Properties** dialog.



15. Close the display.

Alarm and Event Status Explorer

16. Click the Alarm and Event Status Explorer button.



This time a display contained within the InstantFizz application was opened. The display was developed like any other display and contains an Alarm Status Explorer object within it. A separate display is not needed to use the Alarm Status Explorer object, as seen in Step 2, but it does allow you to incorporate other common features of your application if you choose.

- 17. Mouse over the icons on the Status Explorer to see what the toolbar buttons do.
- 18. Select a row in the list and select the **show details for selected alarm**  button from the toolbar. Review the details window then close.
- 19. Select a row, then hold the shift key and select another row to highlight a couple of rows and then click the **Suppress**  button. On the **Suppress Alarm** pop up type **'Suppress for maintenance'** and then press the **Suppress** button to confirm the action.

State	Su...	Sh...	Alarm Name	Path
			[Shortcut]	/InstantFizz/Data_Area:RSL...
			[Shortcut]CIPValve0	/InstantFizz/Data_Area:RSL...
			[Shortcut]CIPValve1	/InstantFizz/Data_Area:RSL...
			[Shortcut]CIPValve2	/InstantFizz/Data_Area:RSL...
			[Shortcut]CIPValve3	/InstantFizz/Data_Area:RSL...
			[Shortcut]MotorPump	/InstantFizz/Data_Area:RSL...

20. Click on a **column header** to see the list sort by the selected column. You may need to resize the columns to fully see text.

21. Click the **Suppress** column until the suppressed alarms appear at the top of the list.

22. If the selected alarms are not highlighted, select the suppressed alarms and then click the **Unsuppress** button. On the **Unsuppress Alarm** pop up press **Unsuppress** button to confirm the action.

23. At the bottom of the Alarm and Event Status explorer window, type **"\*Valve2\*"** in the **name** field for the **Alarm**

source filter and click the **Apply Filter** button.

Alarm source filter

Name:

Status:

24. Observe the list is filtered to show only the alarms that contain Valve2 in the name.

25. Click the **Cancel Filter** button.

26. Observe the list shows all alarms once again.

27. Select **FTAETag\_Area** folder in the left window to view Tag based alarms only.

28. Select an alarm that is in alarm, such as **ChemValveFlow** and click the **Shelve** button.

**Alarm and Event Status Explorer**

- InstantFizz
  - Data\_Area
  - FTAETag\_Area
  - HMI\_Area

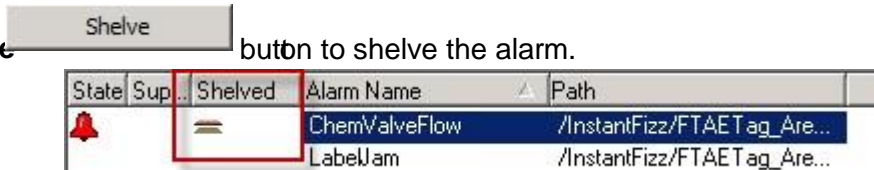
State	Su...	Sh...	Alarm Name	Path
			ChemValveFlow	/InstantFizz/FTAETag_Area...
			LabelJam	/InstantFizz/FTAETag_Area...
			PaperOut	/InstantFizz/FTAETag_Area...
			Tank1Jam	/InstantFizz/FTAETag_Area...
			Tank2Jam	/InstantFizz/FTAETag_Area...

Notice the **Shelve duration:** field. We'll leave the default shelve duration at 1 minute.



Alarm name: ...AETag\_Area:FTAETag\_9  
 Shelf duration: 1 Minutes  
 Comment:

29. Click the **Shelve** button to shelve the alarm.



Notice the shelve symbol.

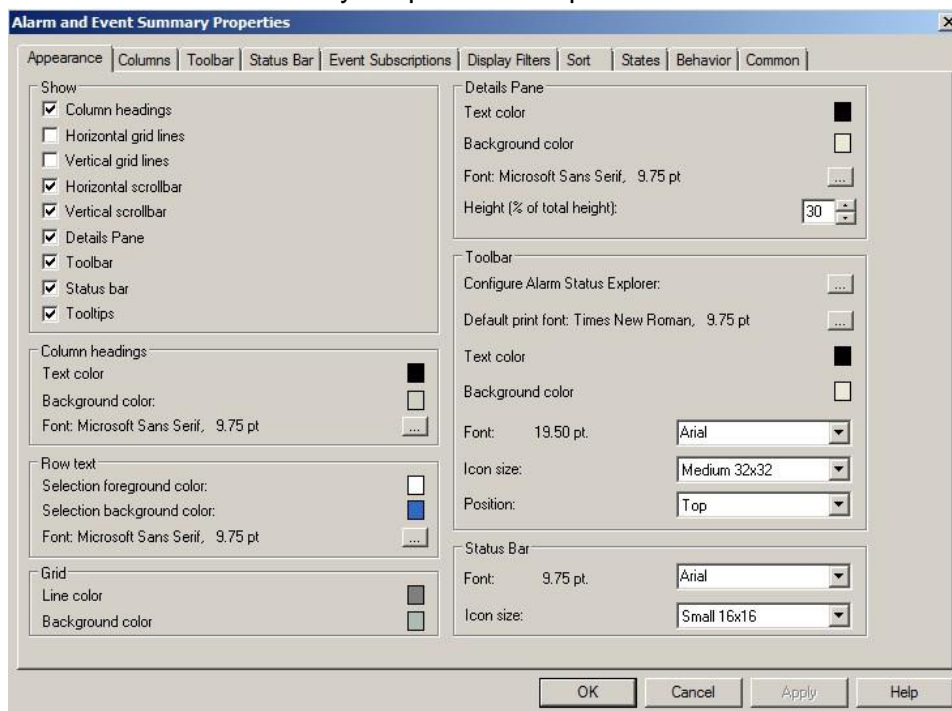
30. Use the **X** button to close the Alarm and Event Status Explorer display when finished.

31. Scroll the alarm summary object on the Alarms display to locate the alarm you shelved. The alarm state column should show the Shelved symbol. Wait until the minute duration has passed and observe the alarm returning to the unshelved state.

**Design features: Alarm and Event Summary, Alarm and Event Banner and Alarm and Event Status Explorer objects.**

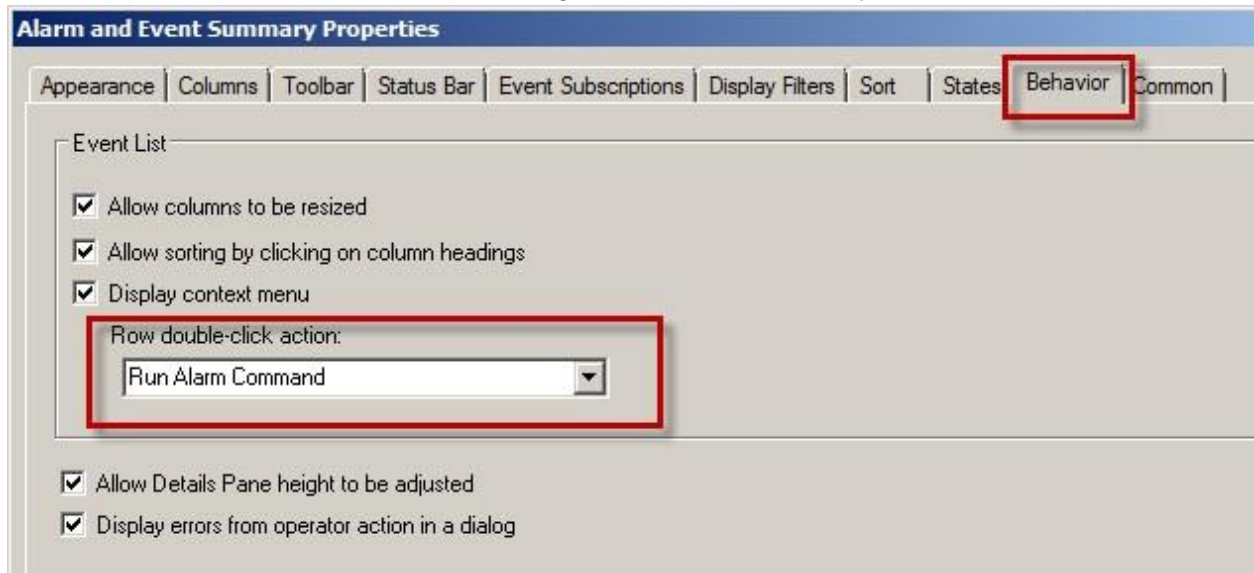
### Alarm and Event Summary

1. Return to the Studio Explorer, open the **med\_alarms** display.
2. Double-click on the **Alarm and Event Summary Design View** object in the display and the Alarm and Event Summary Properties will open.



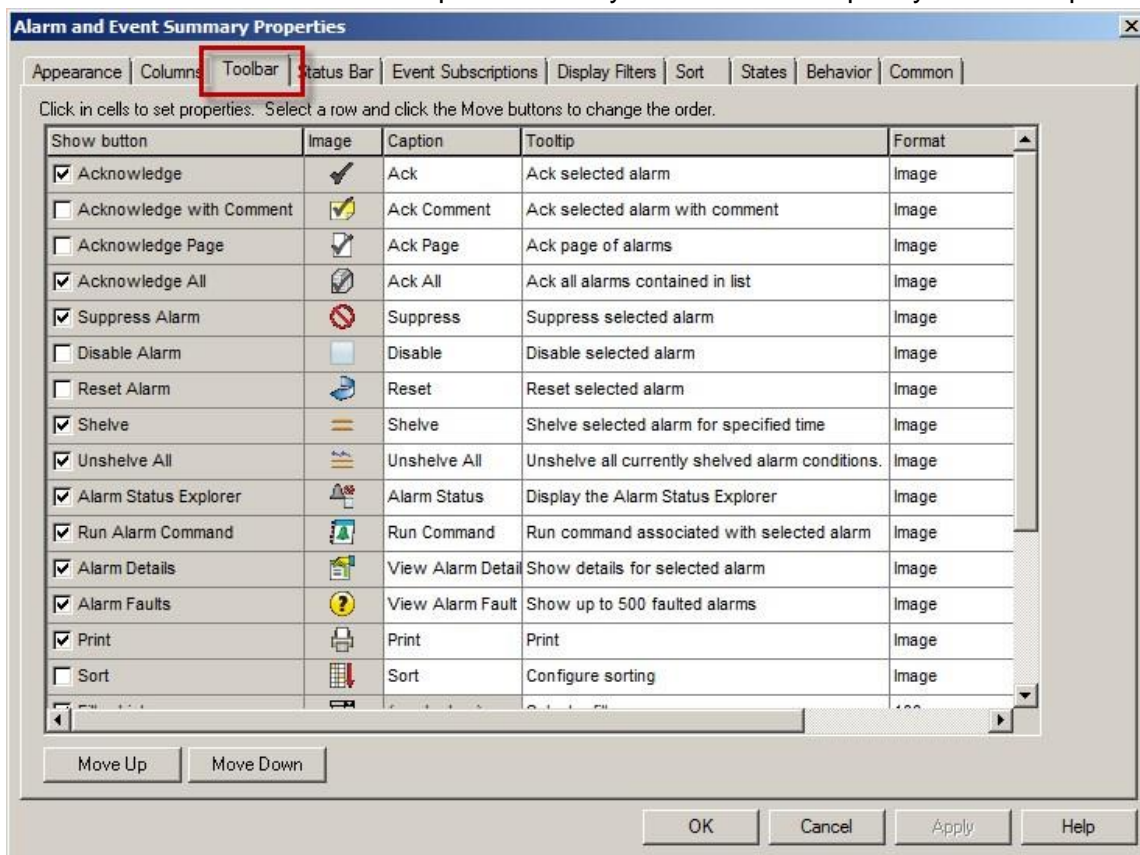


- Click on the **Behavior** tab to view the configuration of one feature you observed in the client.



The **Row double-click action:** property is set to **Run Alarm Command**. This enabled the **CIPValve1** to open the two popup displays.

- Click on the **Toolbar** tab. To save space in the Alarm and Event Summary, add only the toolbar buttons for features that operators really need to access quickly or use frequently.



A few key points are:

**Appearance**

Button sizes and text fonts can be configured

**Columns**

Hide and show columns

Resize columns

Reorder columns

**Toolbar and Status Bar**

Hide and show buttons/panes

Reorder buttons/panes

**Event Subscriptions**

This object, by default, will subscribe to all events with any priority

**Display Filters**

This object has four pre-configured filters. Each filter uses the Alarm Name field for the criteria.

**Sort**




Configure sort order

**States**

Configure text and background colors for the alarm states and

priorities **Behavior**

The **Run Alarm Command** will be invoked when the operator double-clicks on an alarm in the Alarm Summary object.

5. Feel free to make configuration changes to the **Alarm and Event Summary** properties and click **OK**. Skip to Step 8 if there are no changes made. You will get a chance to learn about more features of the Alarm and Event summary when we explore Filtering and subscriptions later in this section.
6. On the **Graphics** toolbar, click **Test Display** button . Alternately, you can save your changes and use the **Alarms**  button on the navigation display of the Client to open the display again to see your changes.
7. On the **Graphics** toolbar, stop testing the display by clicking the **Edit Display** button .
8. Close the **med\_alarms** display in FactoryTalk View Studio when you are done. Click **No** if you are prompted to save any changes.

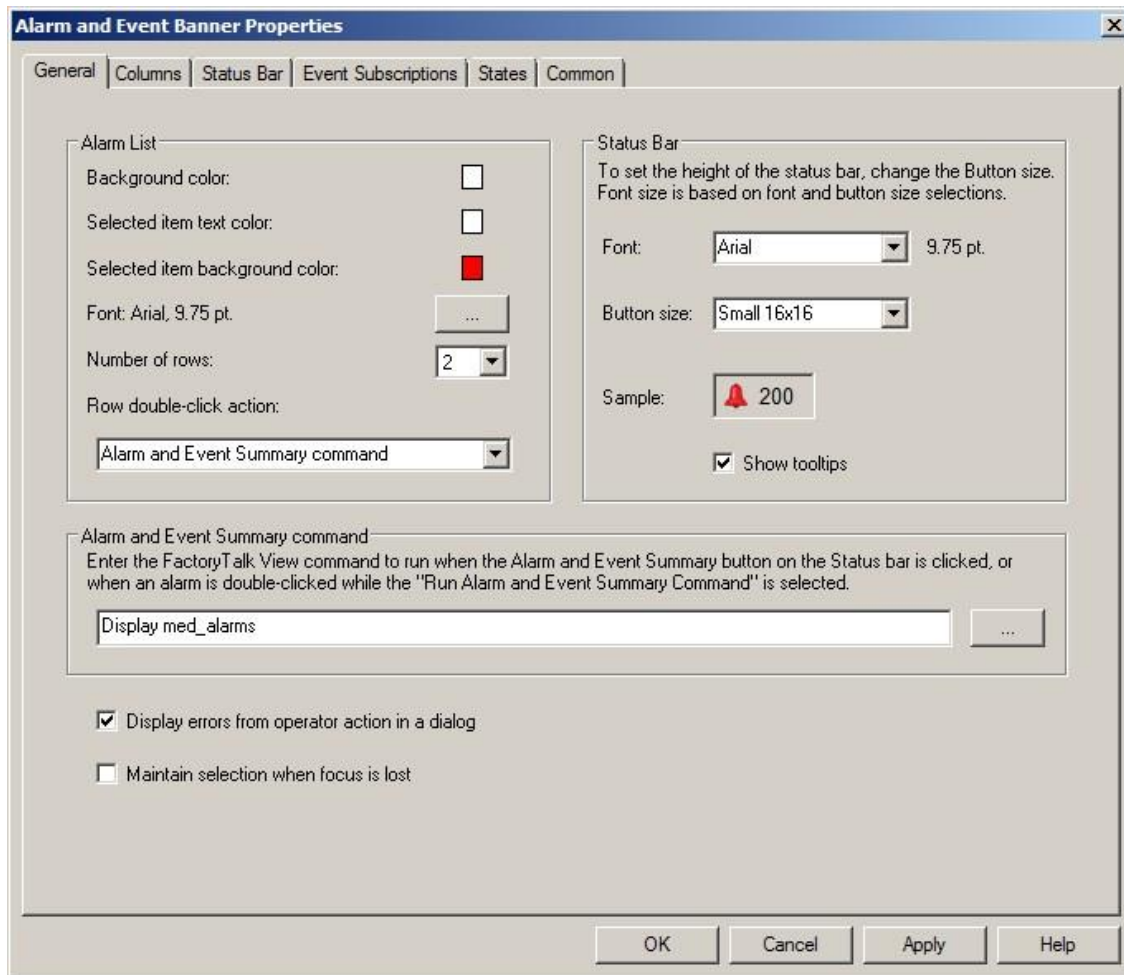
## Alarm and Event Banner

The **Alarm and Event Banner** configuration is similar to the **Alarm and Event Summary**.

9. From the Studio Explorer, double-click on the **med\_topnavbar** display to open it.



10. Double-click on the **Alarm and Event Banner Design View** object in the display and the **Alarm and Event Banner Properties** will open.



Similar to double-clicking on an alarm in the Alarm and Event Summary object, the Alarm and Event Banner can be configured to perform an action when double-clicking on an item in the object. From the properties dialog above, we see that the 'Row double-click action' is set to 'Alarm and Event Summary command' which is defined in the field as **Display med\_alarms**.

Therefore, any double-click action on any alarm in the Alarm and Event Banner will open the Alarms display, whereas the Alarm summary can be configured to run a specific command for each alarm as configured in the FactoryTalk Alarm and Event setup or Logix 5000 ALMx property.


Name	Type	Input Tag	Ack Req'd	Alarm as a Tag	Alarm Class	FactoryTalk View
ChemValveFlow	Level	/Data_Area::[Shortcut]chemValveFlow	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		display med_cip
LabelJam	Digital	/Data_Area::[Shortcut]AlarmLabelJam	<input checked="" type="checkbox"/>	<input type="checkbox"/>		display Labeling
PaperOut	Digital	/Data_Area::[Shortcut]AlarmPaperOut	<input checked="" type="checkbox"/>	<input type="checkbox"/>		display med_labeling
Tank1Jam	Digital	/Data_Area::[Shortcut]AlarmTank1Disc	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		display med_blending
Tank2Jam	Digital	/Data_Area::[Shortcut]AlarmTank2Disc	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>		display med_blending


11. Click on each tab to look at all the properties. The properties are similar to the Alarm and Event Summary object.

Note that the **Alarm and Event Banner** does not have a **Display Filters** tab.



12. Feel free to make configuration changes to the **Alarm Banner** properties, click **OK** and save any changes. Skip to Step 15 if there are no changes made.

13. On the **Graphics** toolbar, click **Test Display** button .

14. On the **Graphics** toolbar, stop testing the display by clicking the **Edit Display** button .

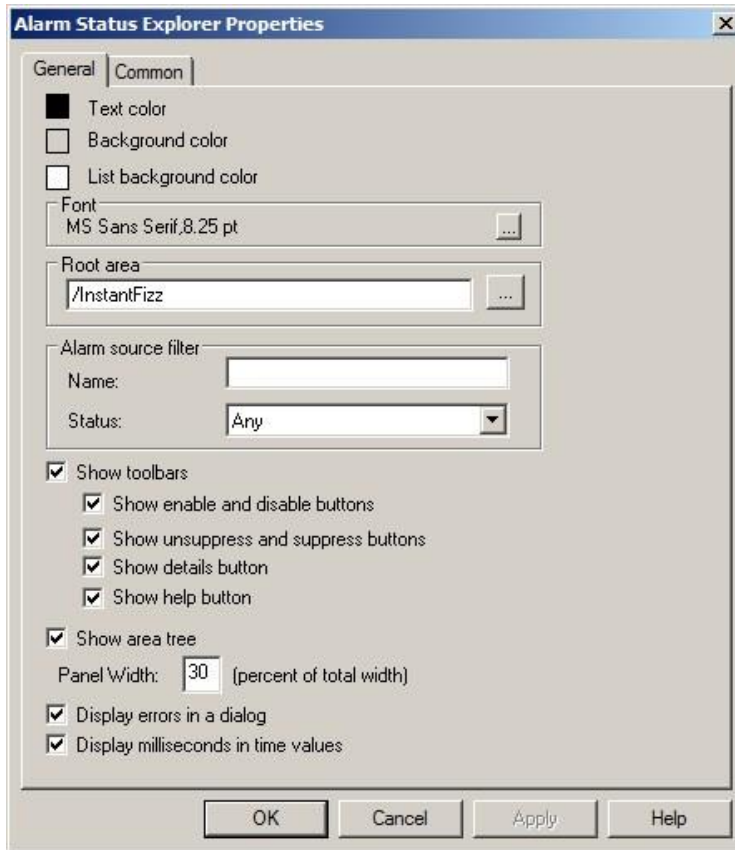
15. Close the **med\_topnavbar** display in FactoryTalk View Studio when you are done. Click **No** if you are prompted to save any changes.

### Alarm Status Explorer

The **Alarm Status Explorer** displays alarm status from each FactoryTalk Alarm and Event Server in the specified Root area.

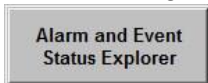
16. In the **Explorer** window of Factory Talk View Studio, double-click on the **alarmstatusexplorer** display to open it.

17. Right-click on the **Alarm Status Explorer** object in the display and then select the **Properties...** context menu item to open the **Alarm Status Explorer Properties**.




18. Feel free to make configuration changes to the **Alarm Status Explorer Properties** and click **OK**. Skip to Step 21 if there are no changes made.

19. On the **Graphics** toolbar, click **Test Display** button . Alternately, you can save your changes and use the



button on the Alarm and Event Summary display of the Client to open the display again to

see your changes.

20. On the **Graphics** toolbar, stop testing the display by clicking the **Edit Display** button .

21. Close the **alarmstaturexplorer** display in FactoryTalk View Studio when you are done. Click **No** if you are prompted to save any changes.

## Viewing Historical Alarm data with the Alarm and Event Log Viewer

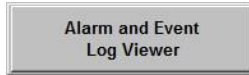
### Alarm and Event Log Viewer

Use the Alarm and Event Log Viewer object, embedded in a FactoryTalk View graphic display, to view and filter historical alarm information stored in Microsoft SQL Server databases.






## Runtime Client features: Alarm and Event LogViewer

1. Switch to the FactoryTalk View SE client and click the Alarm and Event Log Viewer button



from the Alarm Summary display.

The Alarm and Event Log Viewer consists of four parts: the toolbar, the event list, the details pane, and the status bar.

2. Select a row in the list to see details about the alarm in the Details Pane.
3. Explore the use of the toolbar buttons: Refresh , Find  and Show/Hide Details Pane .
4. View the Status bar to see the number of event records in the list, and the name of the filter (or "Not Filtered"). Select an alarm (any alarm) from the list to see details in the Details pane of the Log Viewer object.

The screenshot shows the "Alarm and Event Log Viewer" window. At the top, there is a toolbar with icons for filtering, refreshing, finding, and showing/hiding details. Below the toolbar is a table listing various alarms. The selected row is highlighted in grey. Below the table is a detailed view of the selected alarm, showing fields like Priority, Alarm State, Event Time, Condition Name, Alarm Class, Event Type, Event Category, User Name, User Comment, Area, Server Name, Alarm Name, Severity, Current Value, Limit Value Exceeded, Tag 1 Value, Tag 2 Value, Tag 3 Value, Tag 4 Value, and Computer. At the bottom, the status bar shows "Events: 23449" and "Not Filtered".

Severity	Event Time	Alarm Name	Condition Name	Message
700	9/16/2014 12:25:29 PM	[Shortcut]Program:Mai...	HI	Labeler: Motor is overheating.
1000	9/16/2014 12:25:26 PM	[Shortcut]Program:Mai...	TRIP	Filler Valve 6b Blocked. Please open Valve.
700	9/16/2014 12:25:15 PM	[Shortcut]Program:Mai...	HI	CIP Supply: Motor is overheating.
700	9/16/2014 12:24:39 PM	[Shortcut]Program:Mai...	HI	Labeler: Motor is overheating.
500	9/16/2014 12:23:45 PM	[Shortcut]Program:Mai...	TRIP	Blending Tank 1 Discharge Valve Jammed
700	9/16/2014 12:23:30 PM	[Shortcut]Program:Mai...	TRIP	Filler Valve 3a Blocked. Please open Valve.
1000	9/16/2014 12:23:28 PM	[Shortcut]Program:Lab...	TRIP	Labeler Breakdown
600	9/16/2014 12:23:28 PM	[Shortcut]Program:Filler...	TRIP	Filler Blocked
600	9/16/2014 12:23:28 PM	[Shortcut]Program:Pac...	TRIP	Packer Starved
700	9/16/2014 12:22:59 PM	[Shortcut]Program:Mai...	HI	Labeler: Motor is overheating.
700	9/16/2014 12:22:58 PM	[Shortcut]Program:Mai...	HI	Packaging: Motor is overheating.
600	9/16/2014 12:22:48 PM	[Shortcut]Program:Filler...	TRIP	Filler Blocked
600	9/16/2014 12:22:48 PM	[Shortcut]Program:Pac...	TRIP	Packer Starved
1000	9/16/2014 12:22:48 PM	[Shortcut]Program:Lab...	TRIP	Labeler Breakdown
700	9/16/2014 12:22:15 PM	[Shortcut]Program:Mai...	TRIP	Filler Valve 3a Blocked. Please open Valve.
700	9/16/2014 12:22:09 PM	[Shortcut]Program:Mai...	HI	Labeler: Motor is overheating.
700	9/16/2014 12:22:08 PM	[Shortcut]Program:Mai...	HI	Packaging: Motor is overheating.
1000	9/16/2014 12:22:06 PM	[Shortcut]Program:Mai...	TRIP	Filler Valve 6b Blocked. Please open Valve.

Priority:	Urgent	Severity:	1000
Alarm State:	Normal, Acked	Current Value:	0.0
Event Time:	9/16/2014 12:25:26 PM	Limit Value Exceeded:	0.0
Condition Name:	TRIP	Tag 1 Value:	
Alarm Class:		Tag 2 Value:	
Event Type:	Condition	Tag 3 Value:	
Event Category:	Discrete	Tag 4 Value:	
User Name:		Computer:	
User Comment:			
Area:	RNA://\$Global/InstantFizz/Data_Area		
Server Name:	RSLinux Enterprise		
Alarm Name:	[Shortcut]Program-Main Filling Valve 6b		
Events:	23449	Not Filtered	

The detailed information documents information about that particular event such as: **Alarm State**, **Priority**, **Severity**, **Current Value** (triggering condition), any **User Comment**, and much more.

The events that are captured for each alarm are described in the following Alarm state table:

This icon...	Represents this alarm state...
	In Alarm, Unacknowledged
	In Alarm, Acknowledged
	<ul style="list-style-type: none"> <li>In Alarm, Unacknowledged, Suppressed</li> <li>In Alarm, Unacknowledged, Suppressed, Shelved</li> </ul>
	<ul style="list-style-type: none"> <li>In Alarm, Acknowledged, Suppressed</li> <li>In Alarm, Acknowledged, Suppressed, Shelved</li> </ul>
	In Alarm, Unacknowledged, Shelved
	In Alarm, Acknowledged, Shelved
	Normal, Unacknowledged
	Normal, Acknowledged
	<ul style="list-style-type: none"> <li>Normal, Unacknowledged, Suppressed</li> <li>Normal, Unacknowledged, Suppressed, Shelved</li> </ul>
	<ul style="list-style-type: none"> <li>Normal, Acknowledged, Suppressed</li> <li>Normal, Acknowledged, Suppressed, Shelved</li> </ul>
	Normal, Unacknowledged, Shelved
	Normal, Acknowledged, Shelved

An icon may represent different alarm states. This is because the states Disabled, Suppressed, and Shelved take different orders of precedence:

- Disabled takes precedence over Suppressed
- Suppressed takes precedence over Shelved

5. Select the new **Edit Filter** button

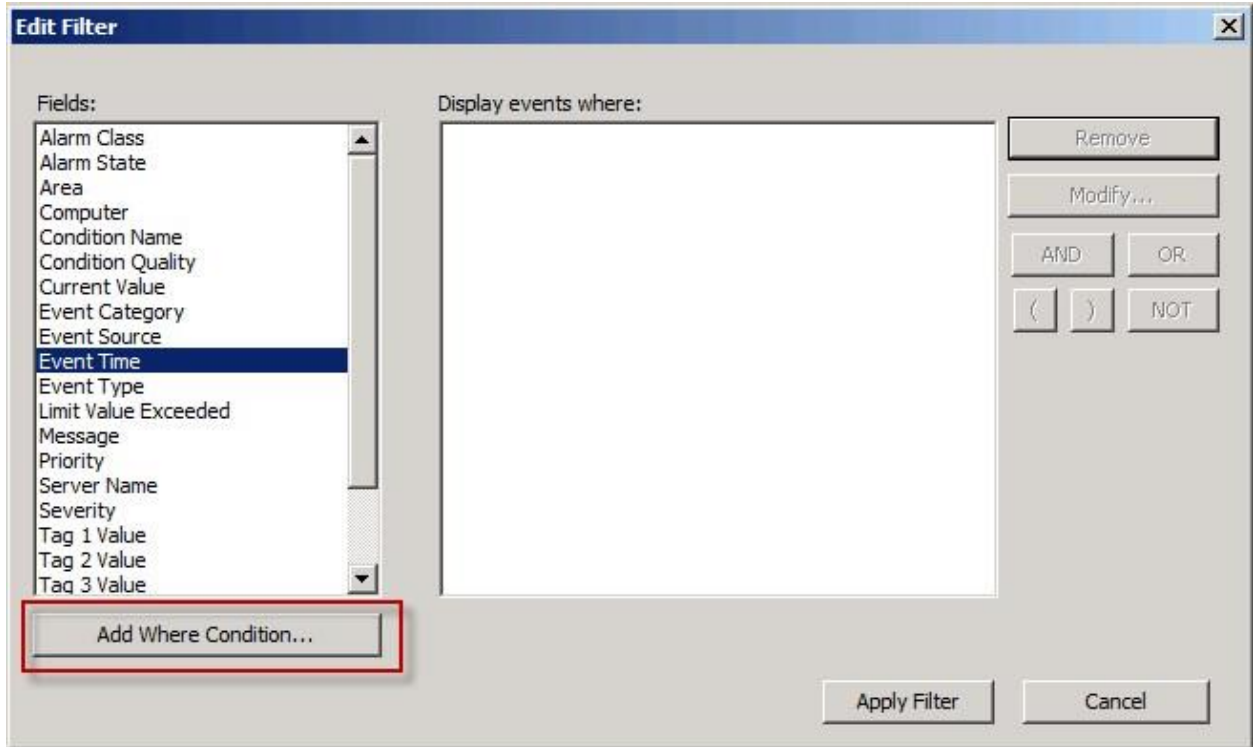


**Create or Edit filters at runtime** - A new button (Edit Filter) is added in the toolbar of Alarm and Event Log Viewer. This allows the user to create a custom filter for the events displayed in the Alarm Log at runtime.

**Filter by Event time** - Shows the events that match your defined time criteria in the Alarm and Event Log Viewer.



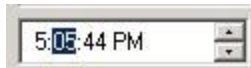
6. Select **Event Time** and click **Add where condition**



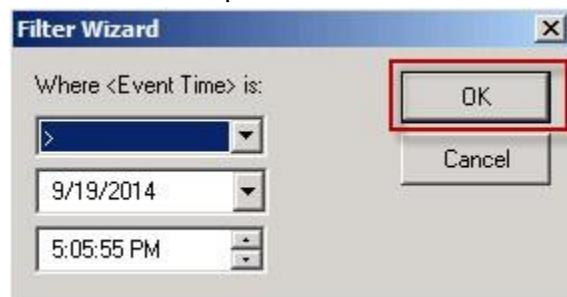
7. In the Filter Wizard dialog, change the condition to be 'Greater than' > and change the time to be ten minutes previous.



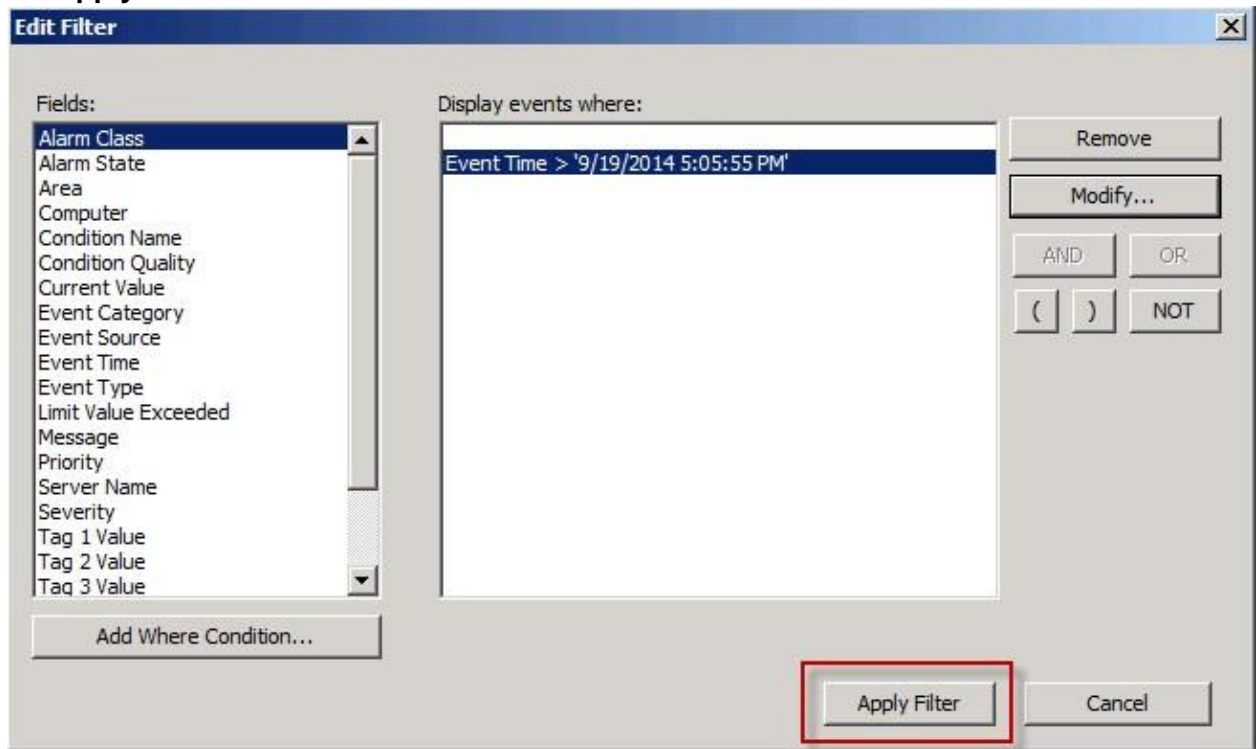
In this example the current time is 5:15:44, so the selected time would be 5:05:44



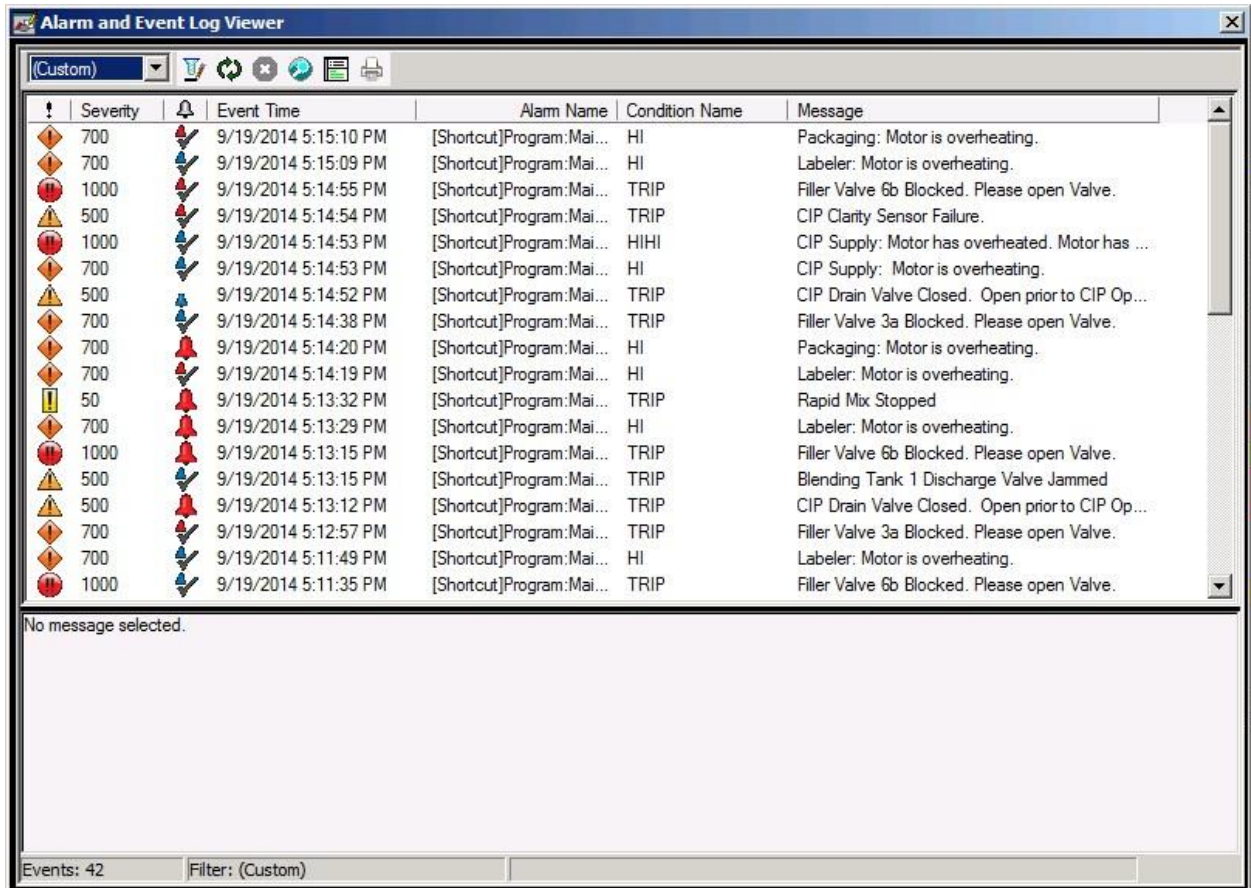
8. Click **OK** to accept the first condition.




9. Click **Apply Filter**



10. Notice the Filter is now set to **Custom** indicating the filter has been created during runtime.

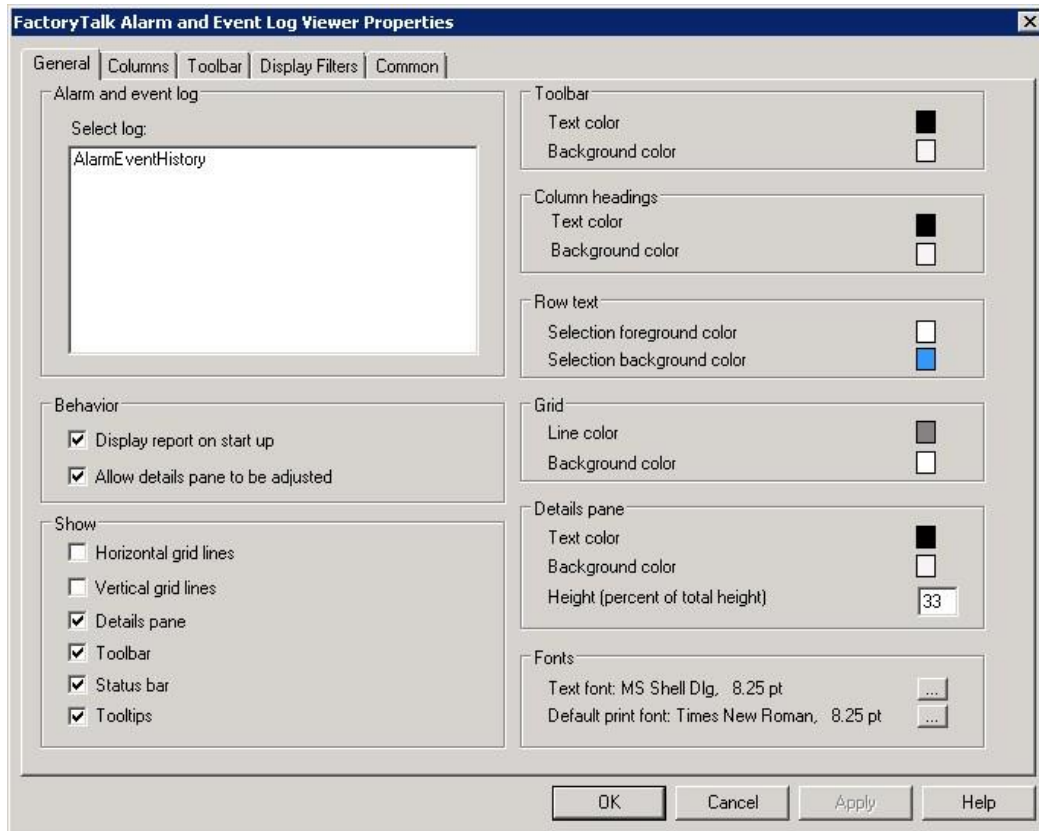


11. Scroll through the alarms to verify you only see events past the filter time.
12. Use the **X**  to close the Alarm and Event Log Viewer display when finished.

### Design features: Alarm and Event LogViewer

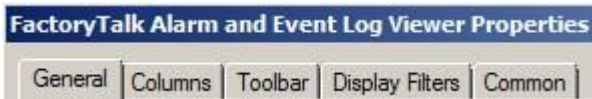
1. Switch back to FactoryTalk View Studio. From the Explorer, double-click on the **alarmlogviewer** display to open it.
2. Double-click on the **Alarm and Event Log Viewer Design View** object in the display and the Alarm and Event Log Viewer Properties will open.

The **Select log**: list displays the alarm database names retrieved from the System > Connections > Databases folder in the FactoryTalk Directory. The first name in the list is selected when this page is presented unless an alarm database was previously selected by the designer. The selected alarm database is queried for historical data requested at run time. Only one alarm database can be selected from the list.



Note that there is only one selection here, **AlarmEventHistory** which we configured as the Database definition in the RSLinx Enterprise Server Alarms and Events History and the FactoryTalk Tag based Alarm and Event History.

3. Click on each tab to look at all the properties.



A few key points are:

**General**

The log must be selected

Text fonts can be configured

**Columns**

Hide and show columns

Resize columns

Reorder columns

**Toolbar**

Hide and show buttons including the new

Reorder buttons

Display Filters

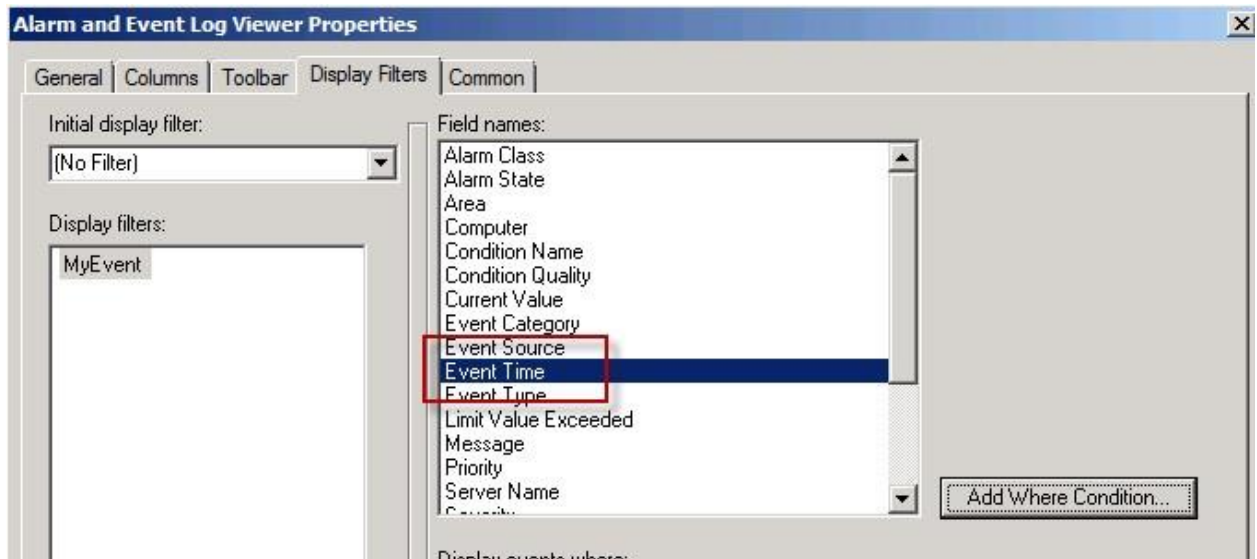
This Log Viewer object can have pre-configured filters.




Let's take a look at where the new features **Edit Filter** and **Event Time** filter are configured during design time.

4. Click on the **Toolbar** tab. As we saw at runtime, the **Edit Filter** button has been added in this release.



5. Click on the **Display Filters** tab. In this example you can see that the **Event Time** filter type can be used to create a 'permanent' filter that can be selected at runtime.



6. Feel free to configure the filter or make configuration changes to the **Alarm and Event Log Viewer** properties. Skip to Step 9 if there are no changes made.
7. On the **Graphics** toolbar, click **Test Display** button . Alternately, you can save your changes and use the  button on the Alarm and Event Summary display of the Client to open the display again to see your changes.
8. On the **Graphics** toolbar, stop testing the display by clicking the **Edit Display** button .
9. Close the **alarmlogviewer** display in FactoryTalk View Studio when you are done. . Click **No** if you are prompted to save any changes.

**Congratulations!!** You have just:

- Used the Alarm Status Explorer to Suppress, Filter and Shelve alarms.
- Examined the Alarm and Event Log Viewer to view historical data
- Examined two new Alarm and Event functions Created a custom Event Time filter at runtime to view historical data based on event time.



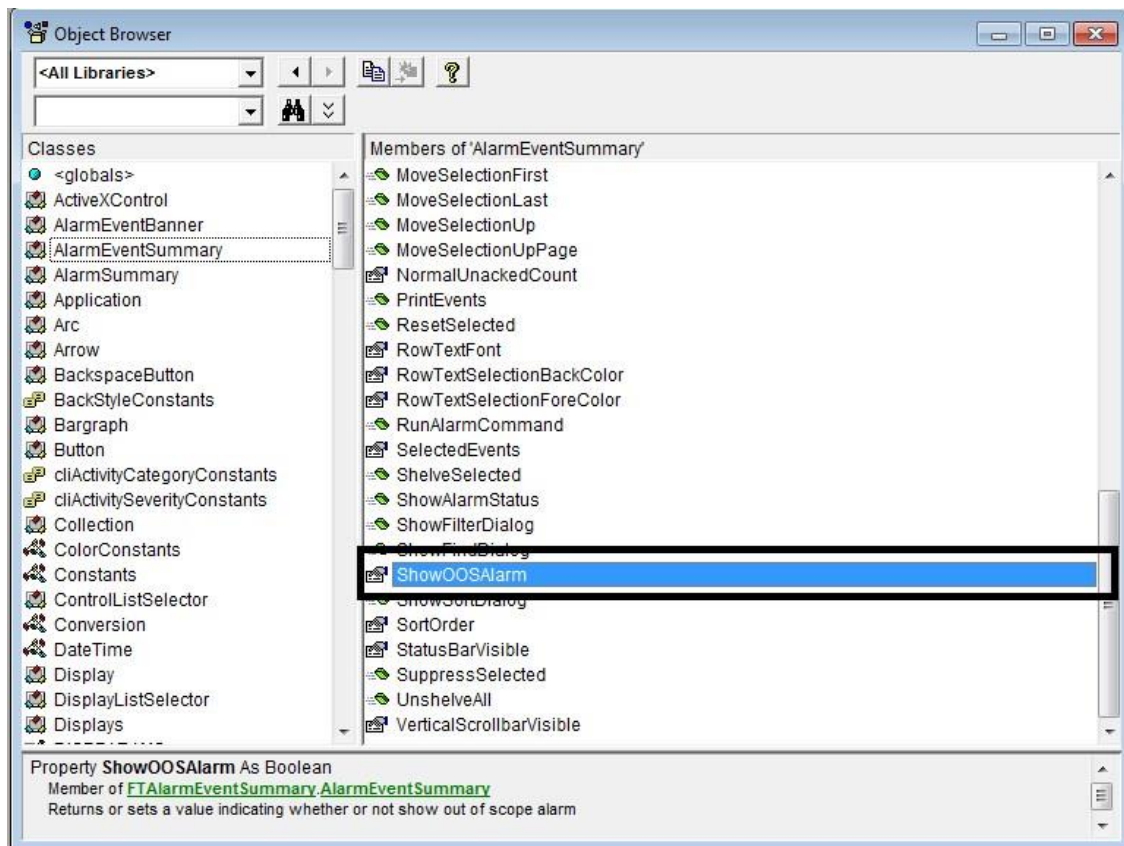
## Out of Scope Alarms



Prior occurrences of an alarm that have gone from Alarm and then back In Alarm cannot be acknowledged. These occurrences are referred to as out of scope and are displayed in the alarm list with a different display and displayed with light grey text. When the most recent occurrence of an alarm is acknowledged, prior occurrences are removed from the event list.

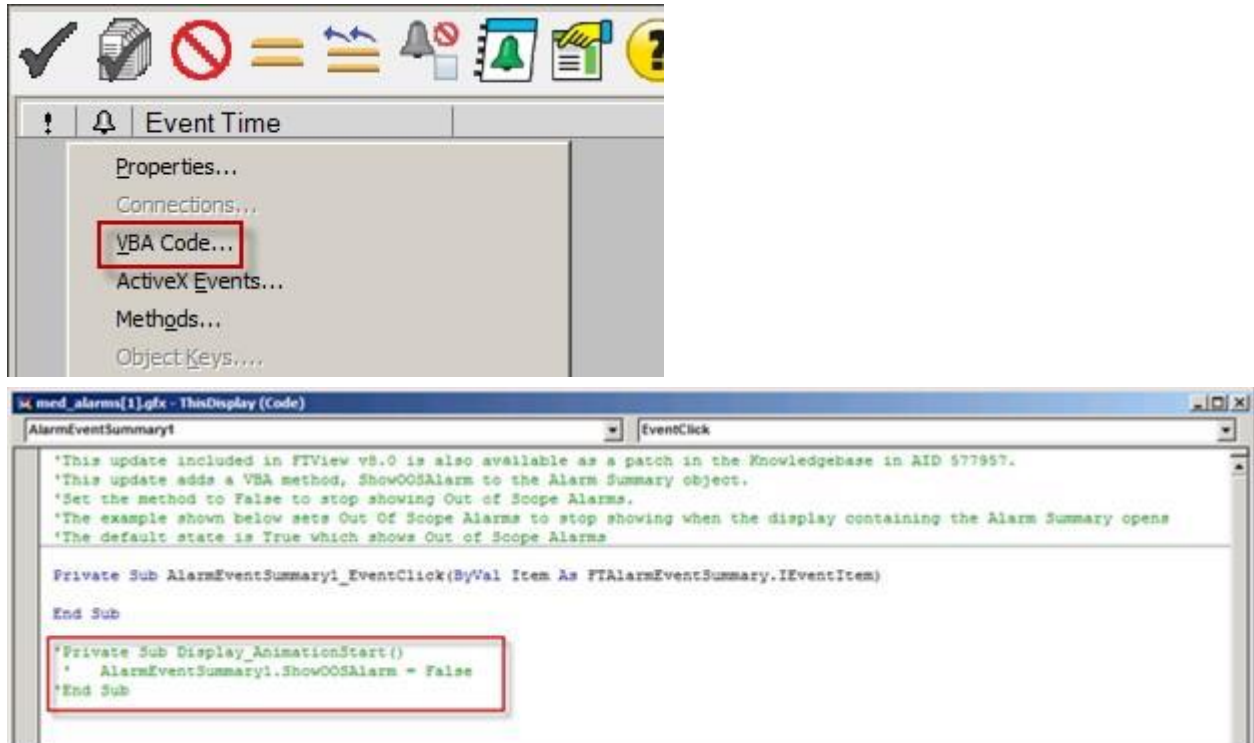
Event Time	Alarm Name	Message
5/1/2014 1:16:33 PM	[Shortcut]Program Main Labeler_Motor	Labeler_Motor is overheating
5/1/2014 1:16:33 PM	[Shortcut]Program Main Packaging_Motor	Packaging_Motor is overheating
5/1/2014 1:16:39 PM	[Shortcut]Program Main Alarm_RapidMixStopped	Rapid Mix Stopped
5/1/2014 1:16:59 PM	[Shortcut]CIPValve0	Recovered Water Valve Jammed. Open valve.
5/1/2014 1:16:59 PM	[Shortcut]CIPValve1	Water Valve Jammed. Open Valve
5/1/2014 1:16:59 PM	[Shortcut]CIPValve2	Chemicals Valve Jammed. Open valve.
5/1/2014 1:16:59 PM	[Shortcut]CIPValve3	Sanitizer Valve Jammed. Open valve.
5/1/2014 1:17:02 PM	Program Labeler Alarm_Labeler1_Breakdown	Labeler Breakdown
5/1/2014 1:17:02 PM	[Shortcut]Program PackD Alarm_Packer1_Starved	Packer Starved
5/1/2014 1:17:02 PM	[Shortcut]Program FillerD Alarm_Filler1_Blocked	Filler Blocked
5/1/2014 1:17:05 PM	[Shortcut]CIPValve0	Recovered Water Valve Jammed. Open valve.
5/1/2014 1:17:05 PM	[Shortcut]CIPValve1	Water Valve Jammed. Open Valve
5/1/2014 1:17:05 PM	[Shortcut]CIPValve2	Chemicals Valve Jammed. Open valve
5/1/2014 1:17:05 PM	[Shortcut]CIPValve3	Sanitizer Valve Jammed. Open valve.
5/1/2014 1:17:23 PM	Program Main Blending_Discharge_Jam_Tank1	Blending Tank 1 Discharge Valve Jammed

There is an update to the Alarm Summary object that enables you to stop displaying Out of Scope alarms. This update adds a VBA method, **ShowOOSAlarm** to the Alarm Summary object, as shown below.



1. Switch to FactoryTalk View Studio. Expand the **Displays** folder in the Application Explorer window under HMI\_Area > InstantFizz\_HMI > Graphics. Double-click on **med\_alarms** to open the display.
2. Right-click and select **VBA Code** from property menu of the Alarm Summary.





The example above is not executing as it has the comment symbol before the code lines. Feel free to remove the comment symbols (') to enable the routine and set the method to false to see that OutOfScope alarms won't be displayed in the Alarm Summary. Note: You must save the display to enable the changes before using Test display or opening the display in View Client.

### 3. Close VBA window.

The Alarm Summary object was enhanced in FactoryTalk View SE v8.0 to enable you to stop displaying Out of Scope alarms. This update adds a VBA method, **ShowOOSAlarm** to the Alarm Summary object, as shown below.

This update is also available for the previous release, FactoryTalk Alarms and Events 2.60, and is available in the Rockwell Automation Knowledgebase in [AID 557977 FTAE Summary displays Out Of Scope alarms, FactoryTalk Alarms and Events 2.60.00 \(CPR9 SR6\)](#).

**Congratulations!!** You have completed the section on the available FactoryTalk Alarms and Events graphical objects and explored Filtering and Subscription. Let's quickly review:

- **Alarm and Event Summary object:** Use to acknowledge, disable, suppress, filter, and sort alarms and shelve Tagbased alarms during run time
- **Alarm and Event Banner object:** Use to monitor and respond to the most current alarms requiring immediate attention
- **Alarm and Event Log Viewer object:** Use to view and filter historical alarm information stored in Microsoft SQL Server databases

- **Alarm Status Explorer object:** Use to enable or disable alarms and suppress or unsuppress alarms and shelve or unshelve Tag-based alarms.
- Out of Scope Alarms: Use update in AID 557977 with VBA to stop showing Out of Scope alarms

---

## Section 6: Testing displays

### About this section

In this section of the lab you will:

- Test Run Displays in the FactoryTalk View Studio Graphics Editor
- Configure a FactoryTalk View Client File called InstantFizz
- Observe Startup Macro
- Observe Client Keys
- Run the FactoryTalk View Client File
- Verify Client Keys work at Runtime

### Test display

Being able to test your display within View Studio without having to run it in a Client is a very powerful feature of FactoryTalk View.

Behavior when test running a display might not always be identical to run-time behavior if changes made during development are not saved. To make the behavior as close as possible you should save a display before testing it.

The Microsoft VBA IDE (Visual Basic for Applications Integrated Design Environment) lets you write, edit, test run, and debug code.


Not everything can be done by test running your display. Some FactoryTalk View commands are ignored when run in test display mode. For example, screen navigation commands, using parameter placeholders in a display, and using parameter values in a reference global object will not function in test display mode. To test these features, run the display in a FactoryTalk View SE Client.

Testing a graphic display in FactoryTalk View Studio is not the same as running the display in the FactoryTalk View SE Client. Before you deploy an application, it is recommended that you test it in the FactoryTalk View SE Client, to verify that everything works as intended.

#### FactoryTalk Alarms and Events Objects

The existing (legacy) HMI Tag Alarm Summary object will not be animated when you run a display in test display mode. The FactoryTalk Alarms and Events Objects will work in test display mode.

You can test the objects in a graphic display quickly, by switching to test display mode in the Graphics editor. Let's try this to animate the **Labeling** display.


1. Open the **med\_labeling** display in FactoryTalk View Studio.
2. From the FactoryTalk View Studio toolbar, click on the **Test Display** button  on the Graphics toolbar.

Observe that the **med\_labeling** graphic begins to animate.




3. Click on the **Labeler Control** button . You may need to select **STOP** first and then **START**.

What happened? Notice that the labeler animation works.

4. Click the **Edit Display** button  to get back to edit mode.
5. Close the display. Select **No** if asked to save the display.
6. Open the **security** display and then click on the **Test Display** button to start animating it.
7. Click on the **Log In** button (i.e. the Locked icon).

What happened? Nothing. Look at the **Diagnostics List**. The Login command on the button was issued but the command is ignored in FactoryTalk View Studio. Certain commands cannot be executed in test display mode. You will need to configure and run a client to be able to execute these commands.

 The command 'Login' is ignored when issued from the development environment or an HMI Server.

Now you will need to configure a client file and run the client to finish testing the display.

## Configure a client file

Before you configure the client file, look at a few of the components that were preconfigured for you to use with your client.

## Configure a macro

A **macro** is a list of commands or command symbols stored in a text file. To run a macro you use its name just as you would a command. The commands in the macro will be executed in the order in which they are listed.

A macro can be specified on startup or shutdown of a client or display. It can be called from a command line in FactoryTalk View Studio or from the Factory Talk View Administration Console for system administration.

FactoryTalk View has multi-tasking capabilities that you can take advantage of when you create macros. Generally, the commands in a macro are executed in the order in which they are listed, with one command finishing before the next begins execution.

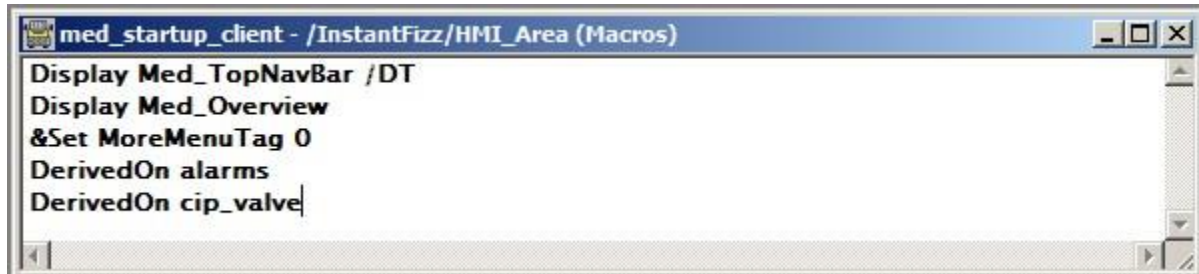
Some commands (such as Print) finish quickly and the next command can start. Others, such as Set, take longer. In the case of Set, it does not finish until the message has been sent to the controller. In cases like that, you can set up the macro so that the next command can be executed before the previous command is finished. Use the ampersand character (&) to do this.

To invoke the command wizard from the macro file, double-click in the macro, or select Edit – Commands... (Ctrl- M).

### Observe the configured macro

These commands will dock displays at the top and bottom of the client window, and set tag values in the controller.

1. From the Explorer, expand the **Macros** folder.
2. Double-click on the **med\_startup\_client** macro. The macro will open.



This macro will open a header display (**med\_TopNavBar**) in a docked area on the top of the client. Then it will open the **Plant Overview** display (**med\_Overview**) that will be undocked. Following this, the macro will set the HMI tag MoreMenu Tag to the value 0 and then start two Derived Tag files.

3. Close the macro. If prompted to save changes, select the **No** button.

### Configure a client key

Sometimes it is useful to have a single key stroke perform a function or multiple functions in your application. For example, when you press F5 in Internet Explorer you will refresh the page. FactoryTalk View SE has similar functionality.

**Client Keys** allow the operator to interact with the system at run time to do things like change displays or set tag values. Client keys are defined for an application. They are enabled whenever the application is running on a FactoryTalk View SE Client.

In addition to Client Keys, there are object and display keys. Object and display keys are defined in the Graphics editor. They are active only when their associated object or display is active. However, object and display keys take precedence over client keys.

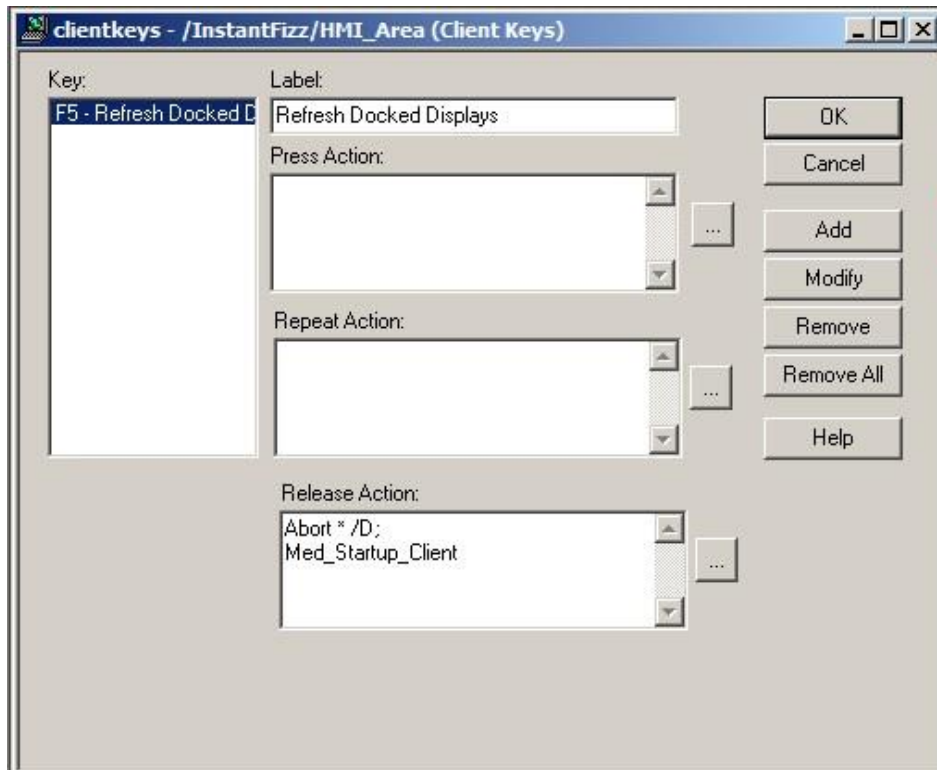
The order of precedence for key animation is: object keys, display keys and client keys. This means, that if a key has object and client key definitions, when the object has focus at run time and the key is pressed, the object key action will be carried out and the client key action will not.

### Observe pre-configured client keys

These commands will perform a refresh of your client. This is useful for testing since you may need to make changes to displays and you don't want to have to close and open the client each time you add or change something on a display.

Let's investigate a pre-configured Client Key file.

1. From the Explorer, open up the **clientkeys** client key file from the Client Keys folder by double-clicking on it.



On the release action of function key F5, all displays including docked displays will be closed. The header display will then be re-opened and docked at the top and the footer display will be docked at the bottom of the client.

2. Close the **ClientKeys**. If prompted to save changes, select the **No** button.

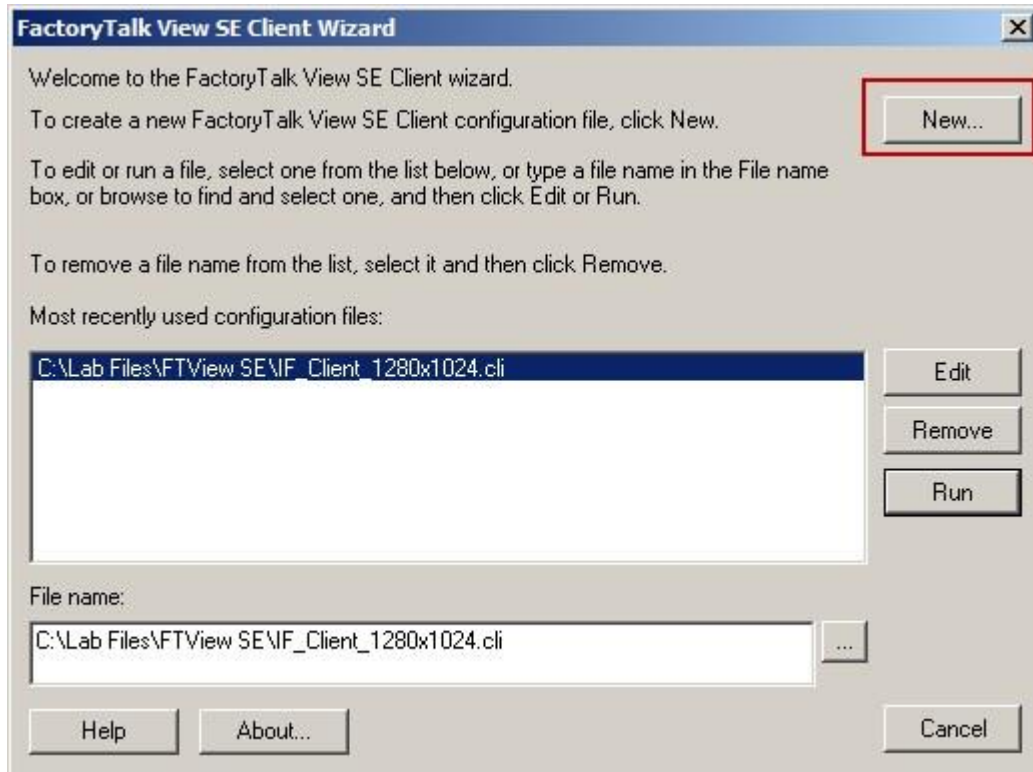
### Configuring a client file

The FactoryTalk View Client can be launched from FactoryTalk View Studio. If a client is already running, close it using the **Exit** button from the navigation display.

1. Configure and launch a new client. Select the **SE Client** button on the tool menu.



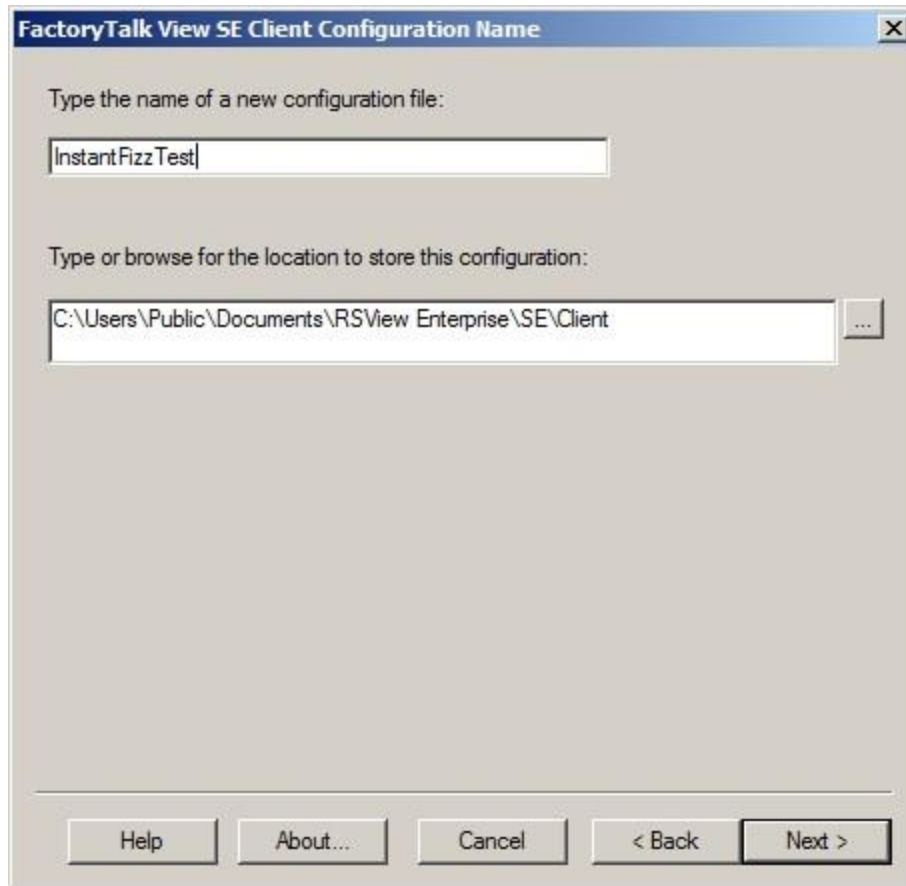
2. When the **Launch FactoryTalk View SE Client** dialog opens, select the **New...** button.



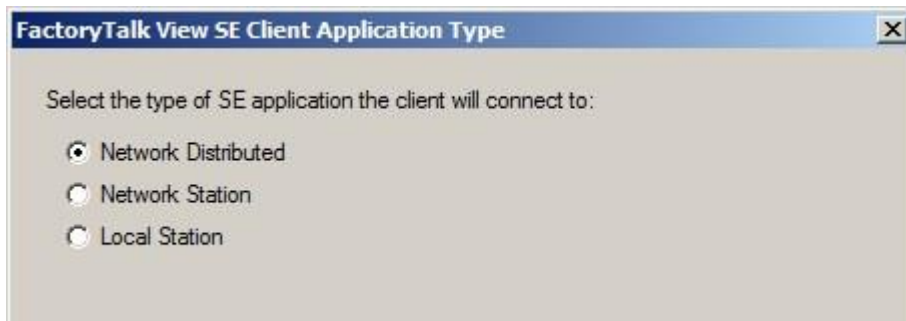
The **FactoryTalk View SE Client Wizard** will open.

3. Type '**InstantFizzTest**' for the name of the configuration file and click the **Next** button. You can leave the location field at default.

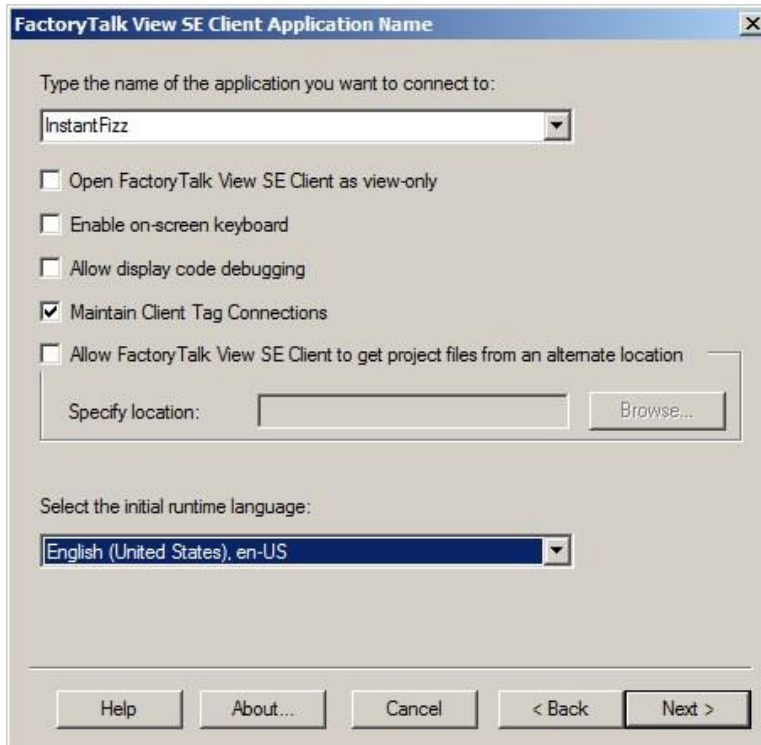




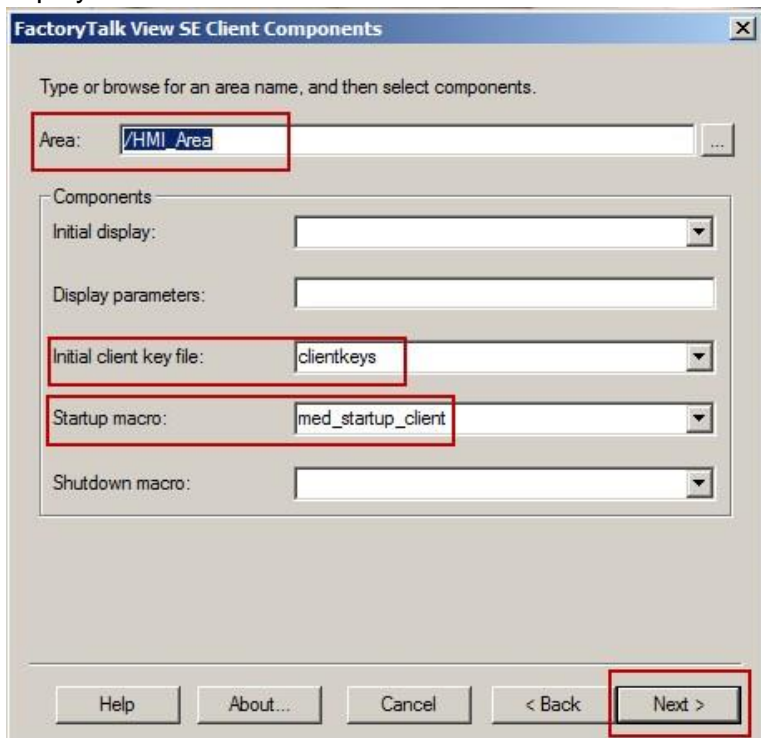
4. Select the **Network Distributed** radio button and click the **Next** button.



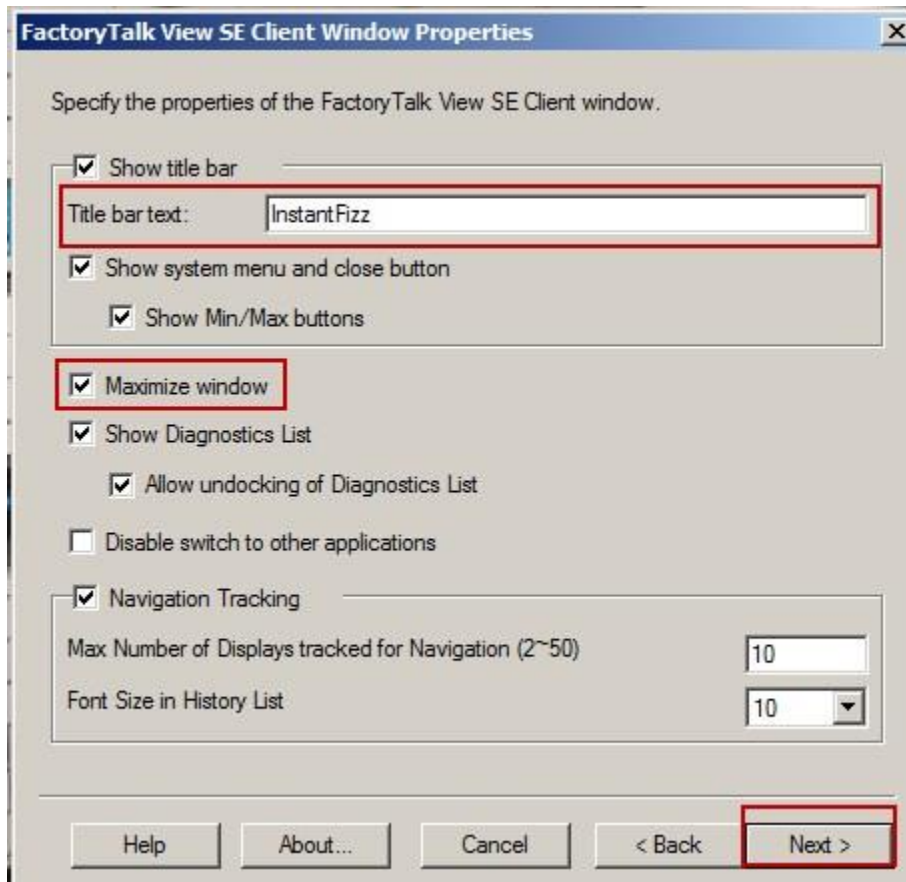
5. Select the **InstantFizz** application and the **English** initial language, leave the defaults, and click the **Next** button.



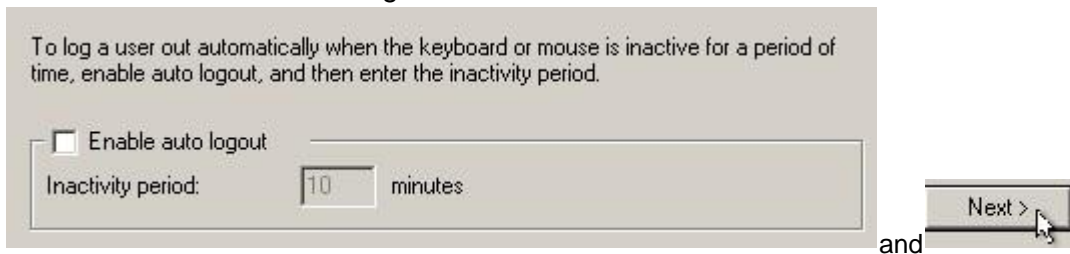
6. Select the **ClientKeys** key file as the **Initial client key file**, select the **Startup\_MediumClient** macro as the **Startup macro**, and click the **Next** button. Note: If you used parameters in the initial display, they would be specified in the Display parameters field. If you use a network application, the area would need to be specified for the initial display.



7. Type **'InstantFizz'** for the title bar text, check the **Maximize window** option, and click the **Next** button



8. Leave the defaults for auto logout and click the **Next** button.



9. Leave the default to save configuration and open FactoryTalk View SE Client now and click the **Finish** button.



The FactoryTalk View Client will start with the specified configuration.

## Running the FactoryTalk View SE client

The InstantFizz application is now running in a FactoryTalk View SE Client window.

### Test the client keys

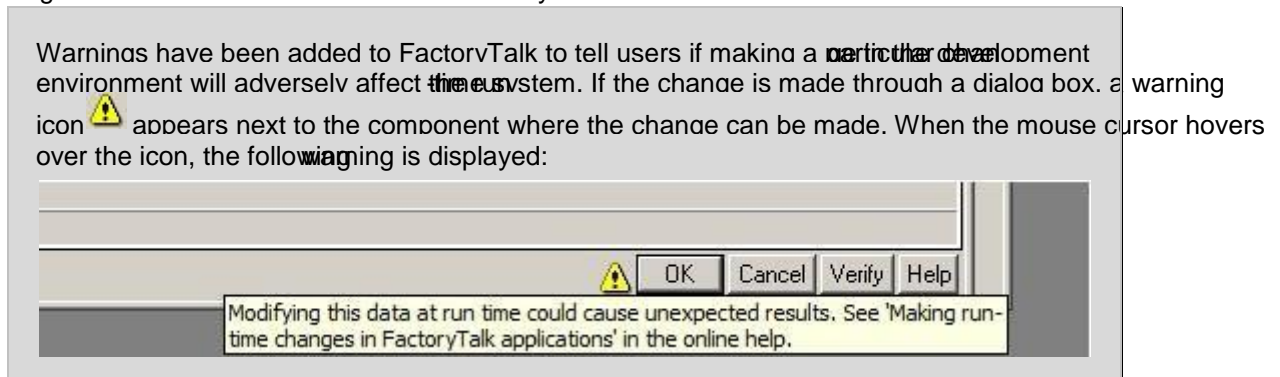
1. Select the **F5** key (using the client keys defined to close all displays and re-display the header and **Plant Overview** display).

Observe that all displays are closed. The header display is re-docked at the top and the **Plant Overview** display is opened below it.

### Verify runtime edits


Changes made to the development environment that can be transferred to the application while it is running are commonly called run-time edits. These fall into four groups:

- Changes that take effect immediately.
- Changes that require a non-disruptive action, such as reopening a graphic display, before they take effect.
- Changes that require a disruptive action, such as restarting a server or a run-time client, before they take effect.
- Changes that cause adverse effects immediately.



For example, you can add a FactoryTalk alarm, add a tag or change a graphic without the need to restart the client.

We will modify one of the displays in FactoryTalk View Studio and observe the update in the client.

1. In the client, click on the **Filling** button  from the navigation display. Look around the display. We will modify the display in FactoryTalk View Studio.
2. Return to **FactoryTalk View Studio**.
3. Open the **med\_fillingcapping** display.

4. Select the **grouped information object** at the bottom and then select the **delete** key.



5. Save the **med\_fillingcapping** display .
6. Close the **med\_fillingcapping** display .
7. Go back to client.
8. Select the **F5** key (using the client keys defined to close all displays and re-display the header and **Plant Overview** display).
9. Observe that all displays are closed. The header display is re-docked. Re-open the **Filling** display. Notice that the **grouped information object** is no longer present.

We have just completed an online change. After modifying a display in FactoryTalk View Studio we only needed to re-open that particular display on the client to see the updates. We did not need to restart the client.

## Section 7: Events and Using the Event Detector Editor

An **Event** is an action that is triggered by the evaluation of the expression associated with it. The action can be any FactoryTalk View SE command or macro. When the expression changes from False on the previous evaluation to True on the current evaluation, the action is triggered. An expression can be a tag name or it can be an equation made up of tag names, mathematical and logical operations, special functions, and If-Then-Else logic.

An Events file consists of a number of events and a maximum update rate for them. At run time, multiple Events files can be active at once. You can run a maximum of 20 Events files at one time, each containing up to 1000 events.

The Event Detector is a graphical editor that is built into FactoryTalk View SE that a user can use to set up the event that has to be triggered and the condition which is defined by an expression that needs to be evaluated for the event to be triggered

In this section, we will learn how to create a new event that will run a Macro, enable this event and then test this by using Test Display.

---

## About this section

In this section of the lab you will:

- Configure an Event Detector file
- Test an Event Detector file

## Create an Event Detector file

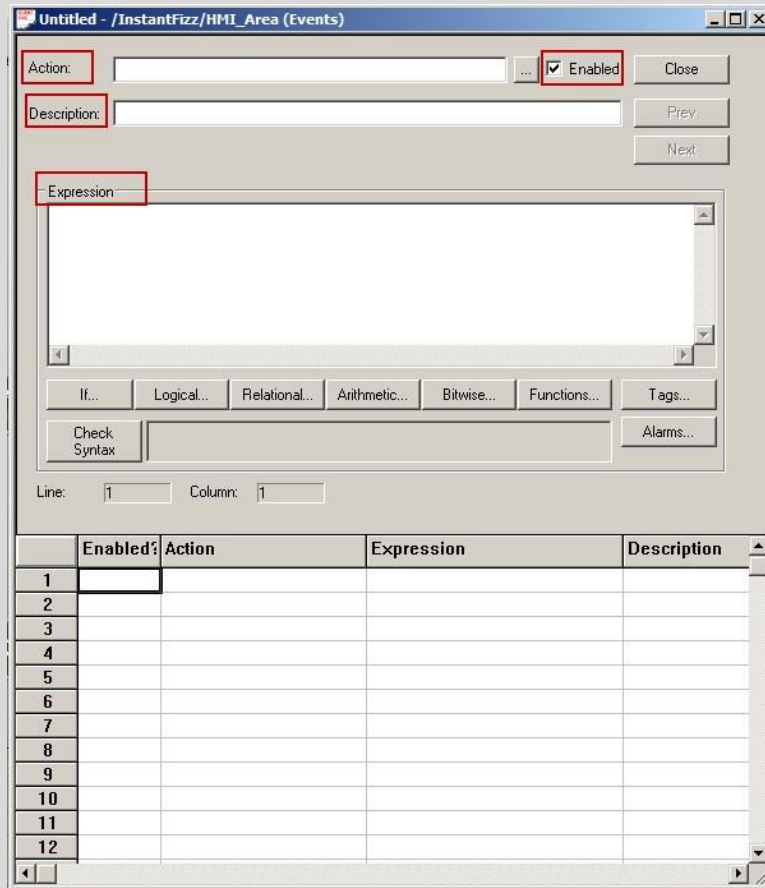
Let us take a look at the Event Detector editor.

1. In FactoryTalk View SE project explorer, right click **Events >New**



The Events Detector editor opens.

The different fields that need to be configured to set up an Event in the Event Detector editor are described in the tip box below with a description:



### Action

The command, macro, or symbol that runs when the expression is True.

### Enabled

For an individual event in an Event file to run when the Event file is running and the event is True, this checkbox must be set to Enabled. The default is Enabled.

### Description

Brief description of what the event does. This description is for your information only and does not appear anywhere else.

### Expression

The expression that will trigger the action when it changes from a False value (0) to a True value (non-zero). An expression is a mathematical or logical equation that returns a value. It can contain tag names, constants and mathematical, relational, logical and/or bitwise operators. A tag name can also stand alone as an expression.

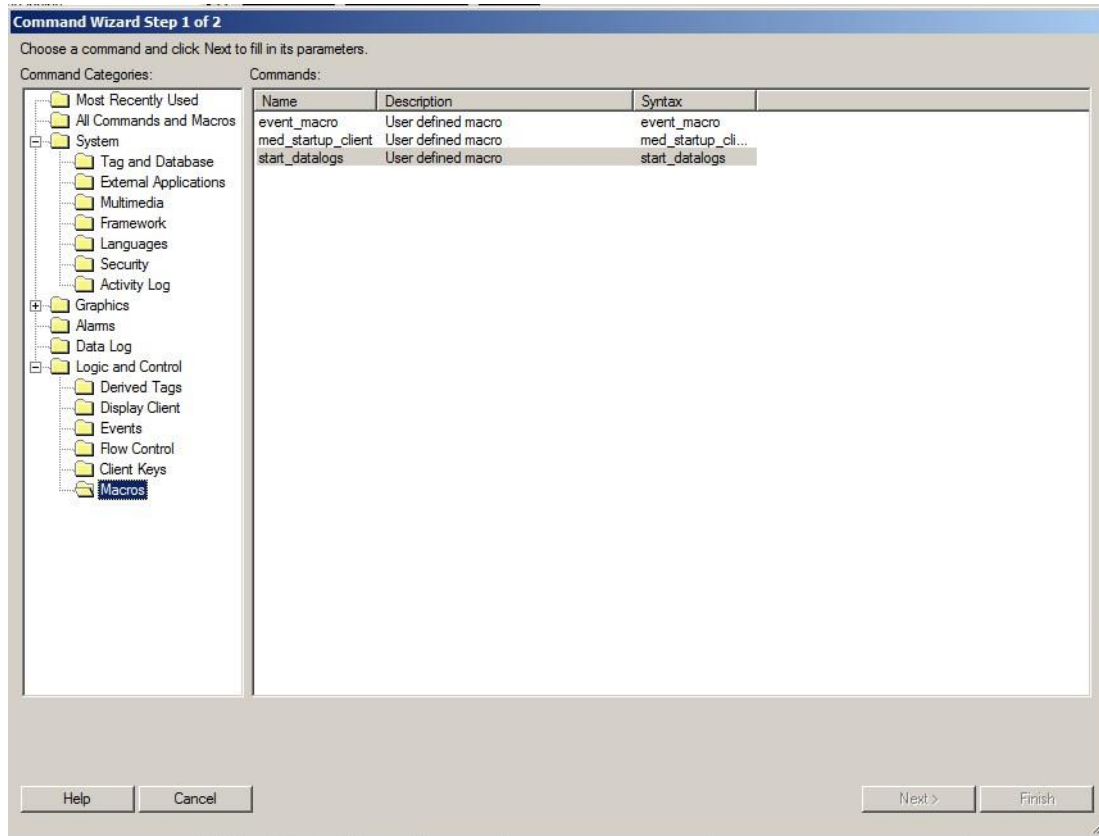
Let us add a new event. The event that we will create is a macro event. When the expression we specify in the Expression Editor evaluates to true, the macro **event\_macro** will be executed.

## Creating a New Event

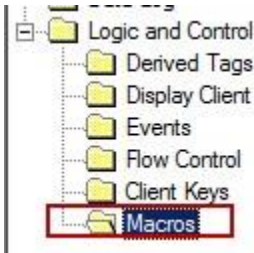
1. Click the Ellipsis button to the right of the Action field



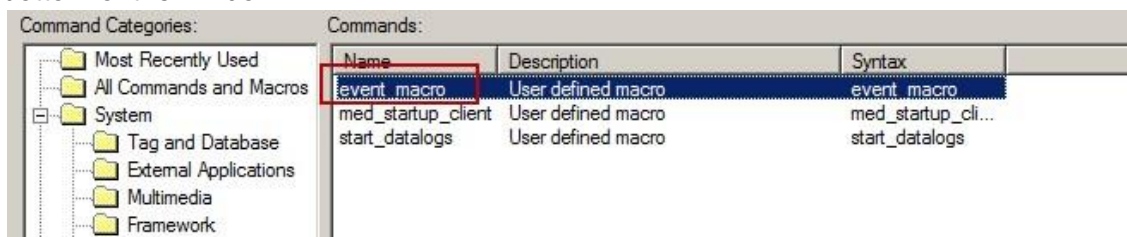
- In the Command Wizard that launches, observe the different command categories and commands that can be set up as an action.



- For our demo, we will choose **Macros** from **Logic and Control > Macros**

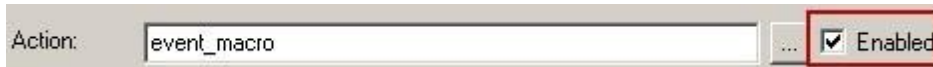


- Select the **event\_macro** from the Commands pane on the right and click on **Finish** at the bottom of the window.

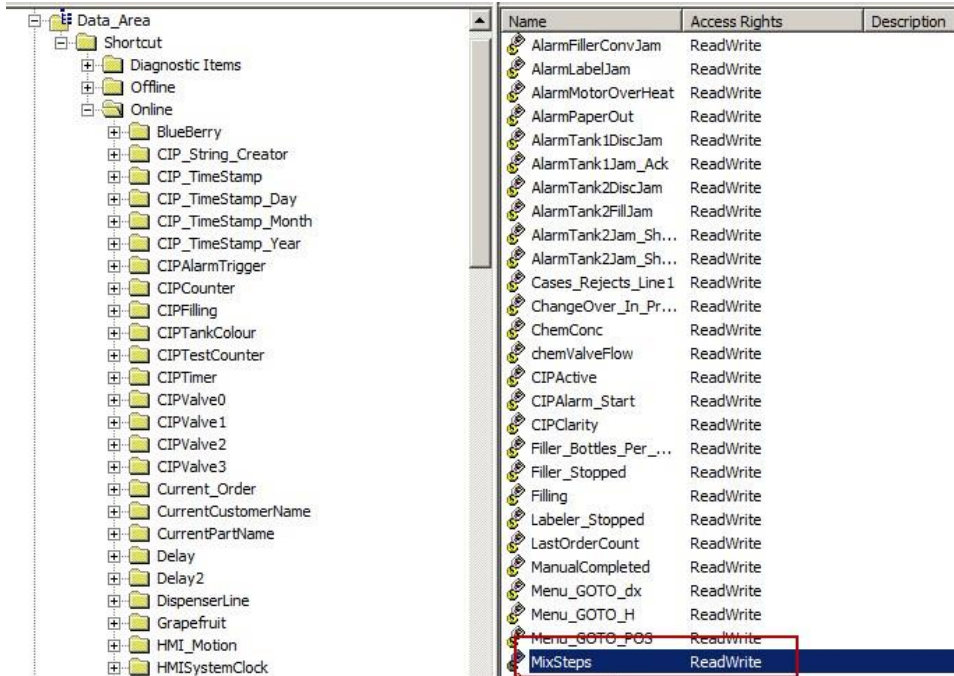


Notice that the **event\_macro** is added to the Action field in the Events Detector editor.

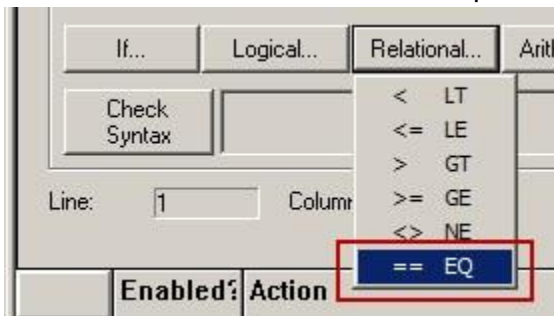
- Optionally, add a brief description in the **Description** field if you would like one.
- Check to make sure that the **Enabled** field is checked.



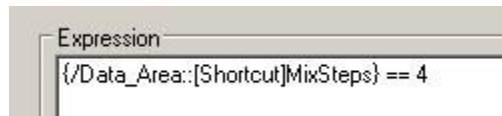
- In the **Expression** editor, click on the **Tags** button, and select the tag: **Mix Steps** (*InstantFizz > Data\_Area > Shortcut > Online > Mix Steps*)



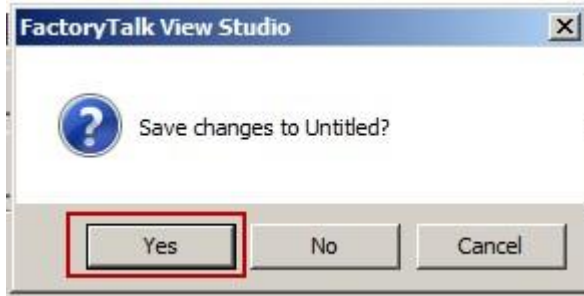
- Click **OK** to close the Tag Browser.
- Click the **Relational** button in the Expression editor and select **== EQ**



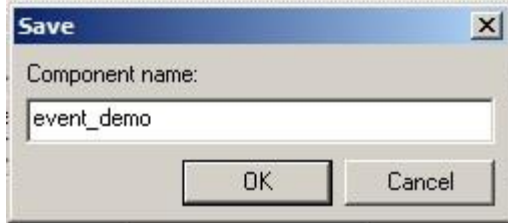
- Type in the value **'4'** once the == is added to the expression. Your expression will look like this :



- Click the **Accept** button in the Events Detector editor.
- Click the **Close** button
- Select **Yes** in the **Save changes to Untitled?** prompt window



14. Type **'event\_demo'** in the **Save** prompt



15. You will see the **event\_demo** event created under the **Events** folder in the Project Explorer



Let us take a look at what the **event\_macro** looks like. This macro was already created for you. Recall this is the macro that we want to run when our event, **event\_demo** is triggered.

16. Double click the **event\_macro** from the **Macros** folder in the Project Explorer. The event\_macro macro will open.



This macro will set the value from the tag RapidMixTimer.ACC to a local HMI tag RapidMix\WaterAmount. This is a simple scenario where a macro is performing a tag write.

17. Close the event\_macro window without making any changes.

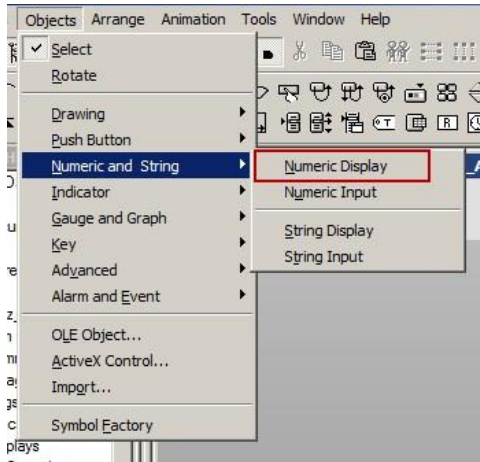
Let us test the event that we have setup.

### Testing Event

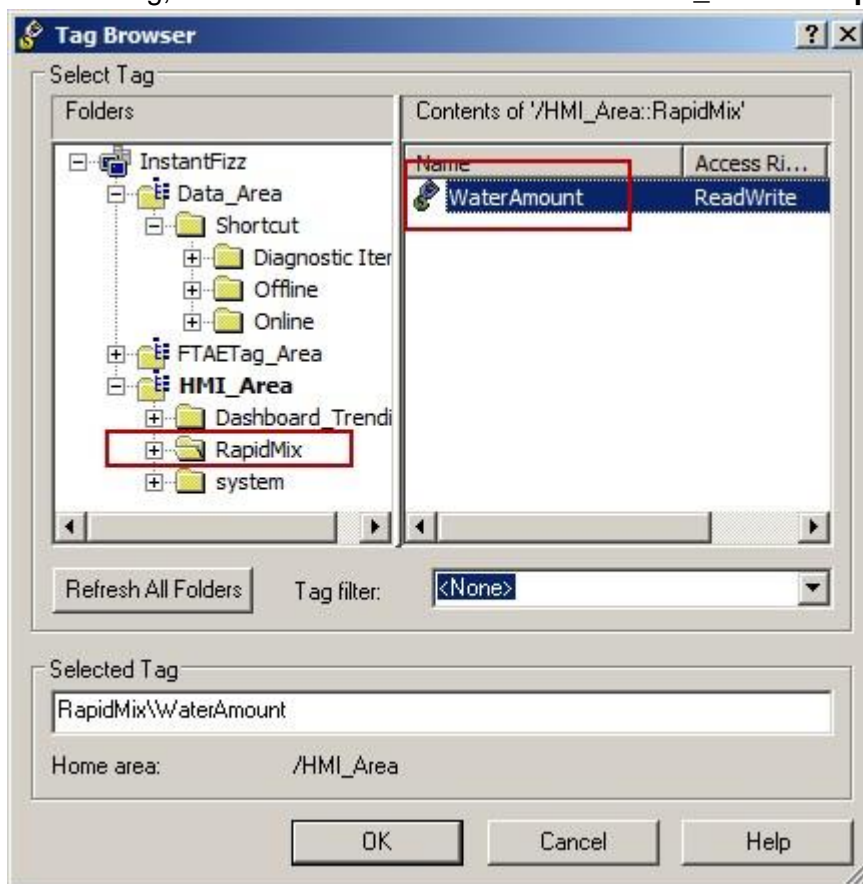
1. Open the display **med\_rapid\_mix** from the **Displays** folder in the Project Explorer.

An event is typically not used to display tag values or navigate displays. However, for the purpose of our demo, we will add a Numeric Display object to the screen so we can observe that the tag value of the WaterAmount tag changes.

2. Select a Numeric Display object by click on **Objects > Numeric and String > Numeric Display**

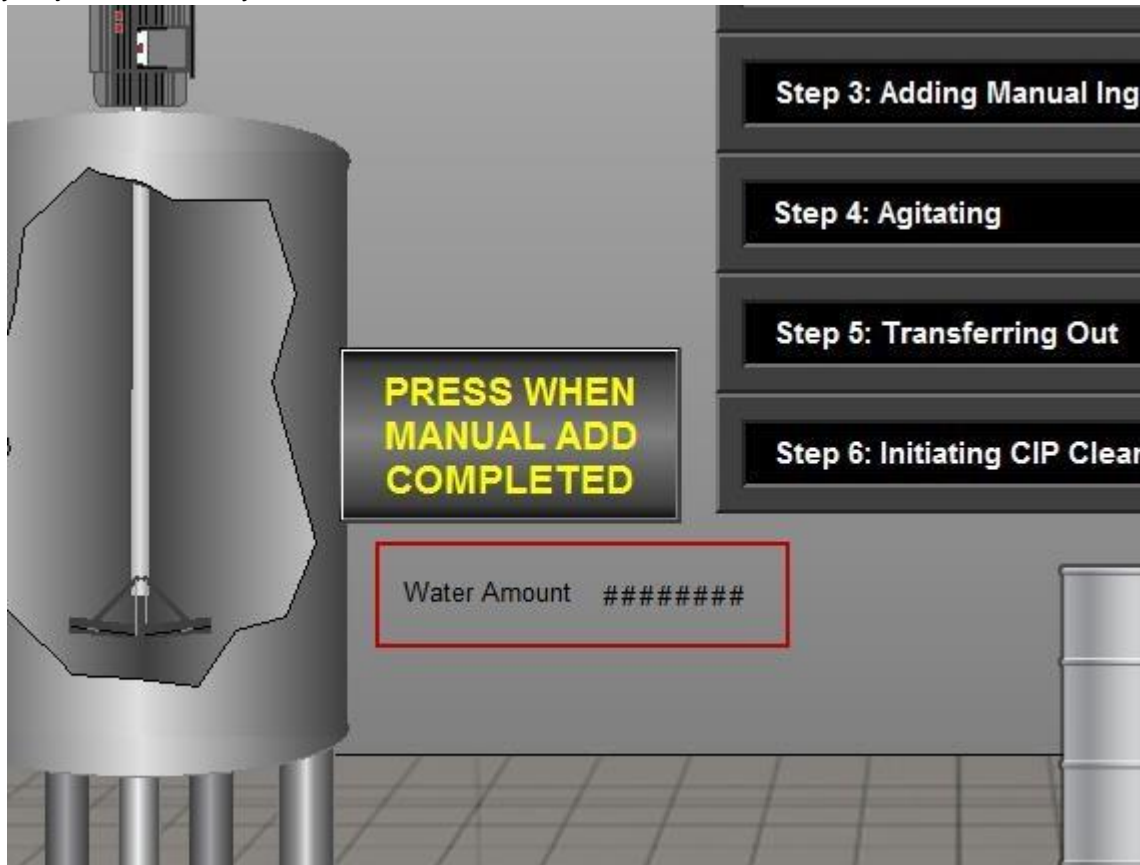


3. Draw the Numeric Display object onto the **med\_rapid\_mix** display.
4. In the Numeric Display Properties window that opens up, click the **Tags** button to launch the **Tag Browser** window.
5. Select the tag, **WaterAmount** from **Instant Fizz > HMI\_Area > RapidMix**



6. Click **OK** to close the Tag Browser.

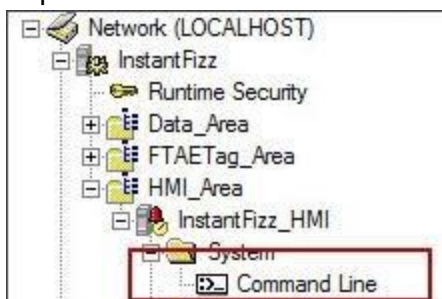
7. Click **OK** in the Numeric Display Properties window to close.
8. Also, add a text object from **Objects > Drawing > Text**. In the **Text Properties** window that pops up, type '**Water Amount**' as the Text. You can modify other properties in this window. When you are done, click the **OK** to close the Text Properties window. The two objects that you just added may look like this:



9. **Save** and close the med\_rapid\_mix display.

We will now issue a command line command from within FactoryTalk View Studio to initiate the event from the HMI server.

10. Double click **Command Line** from **HMI\_Area > InstantFizz\_HMI > System** in the Project Explorer

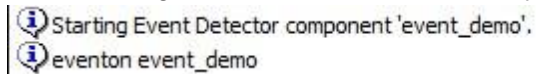


11. In the Command Line window that opens, type '**eventon event demo**' as shown in the screenshot below:





12. Hit the **Enter** key.
13. This will trigger the event event\_demo from the HMI server. You can verify this by looking at the event log at the bottom of the FactoryTalk View Studio window

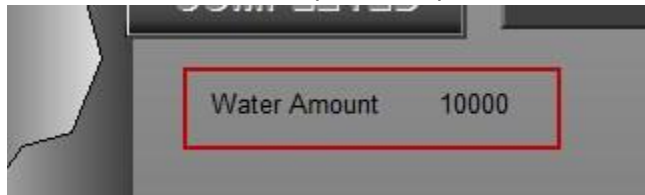


14. Switch to the running client window and open the Rapid Mix display.

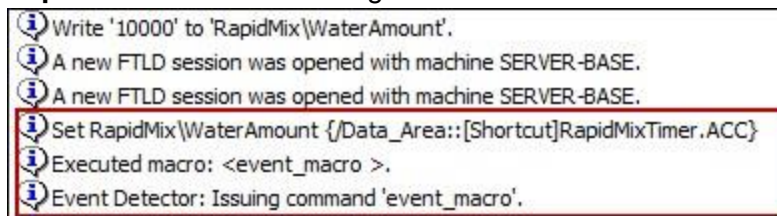


15. On the Rapid Mix display, click on the **Stopped** button to start the rapid mix process.
16. Press the **PRESS WHEN MANUAL ADD COMPLETED** button when it becomes enabled.  
At that point, the **MixSteps** tag will receive a value **4** after **Step 3 Adding Manual Ingredients** is complete.

Notice that as soon as Step 3 is complete, the Numeric Display object will change its value to **10000**.



17. If you look at the event log at the bottom of the FactoryTalk View Studio window, you will observe that the Event Detector first issued a command to launch the macro, **event\_macro**. Following this, the **event\_macro** macro was executed. As a result of this macro, the **RapidMix\WaterAmount** tag was set with the value from **RapidMixTimer.ACC**.



This tells us that the event was triggered by the HMI server. In our scenario, the event that was triggered was to run the macro event\_macro.

Congratulations! We have learned what an Event is and used the Event Detector editor to create a new event that will run a Macro. We also enabled this event in the HMI Server Properties and tested this event by using Test Display.

---

## Section 8: FactoryTalk View Application Documenter

FactoryTalk View ME/SE Application Documenter is a utility that provides detailed information on HMI projects in Factory Talk View SE applications (both Network and Local) and Factory Talk View ME applications. The utility allows a user to view, export and print the content of the HMI project components including the tag cross reference information associated with each component as well as the VBA code (for SE applications) associated with the graphic displays.

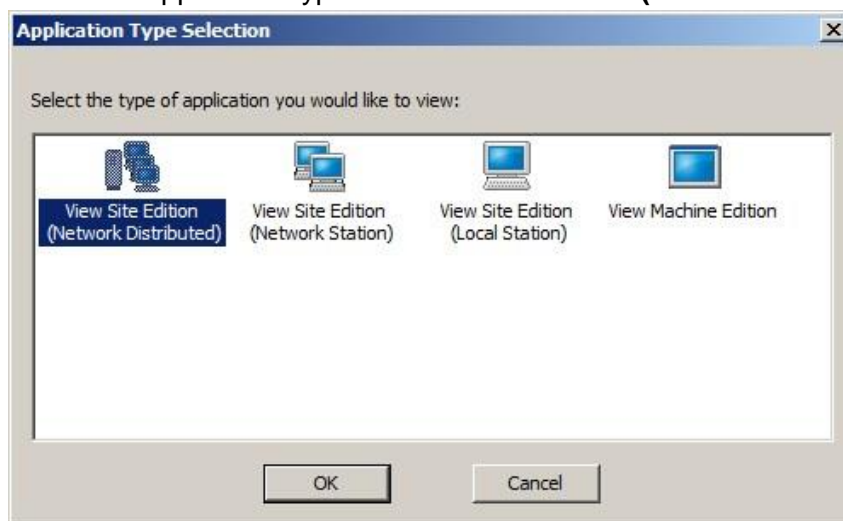
### About this section

In this section of the lab you will:

- Learn how to use the FactoryTalk View Application Documenter

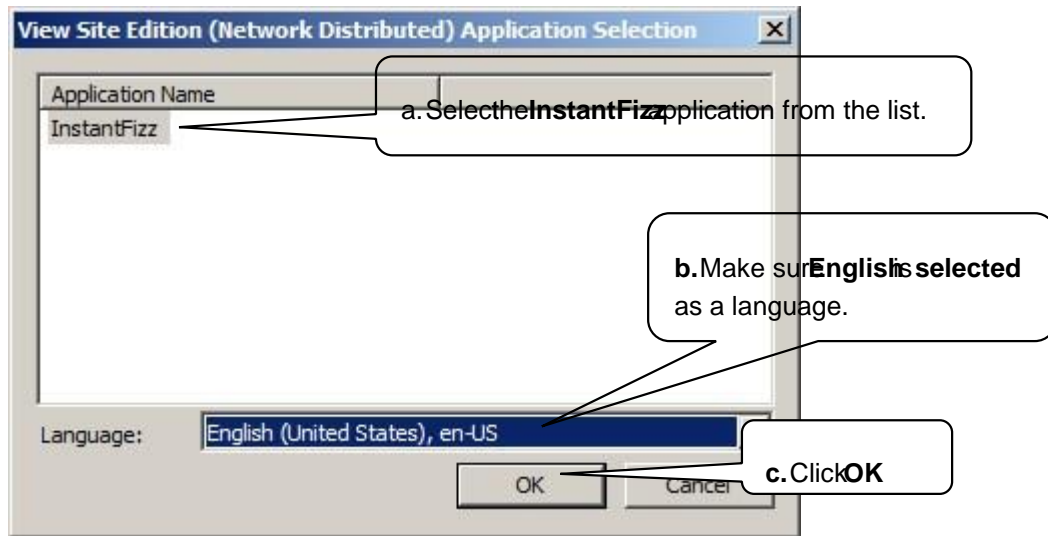
### Starting FactoryTalk View Application Documenter and Opening a Project

1. Start Application Documenter. From the **Start** menu, select **All Programs > Rockwell Software > FactoryTalk View > Tools > Application Documenter**.
2. Select the application type as **View Site Edition (Network Distributed)** then press OK.

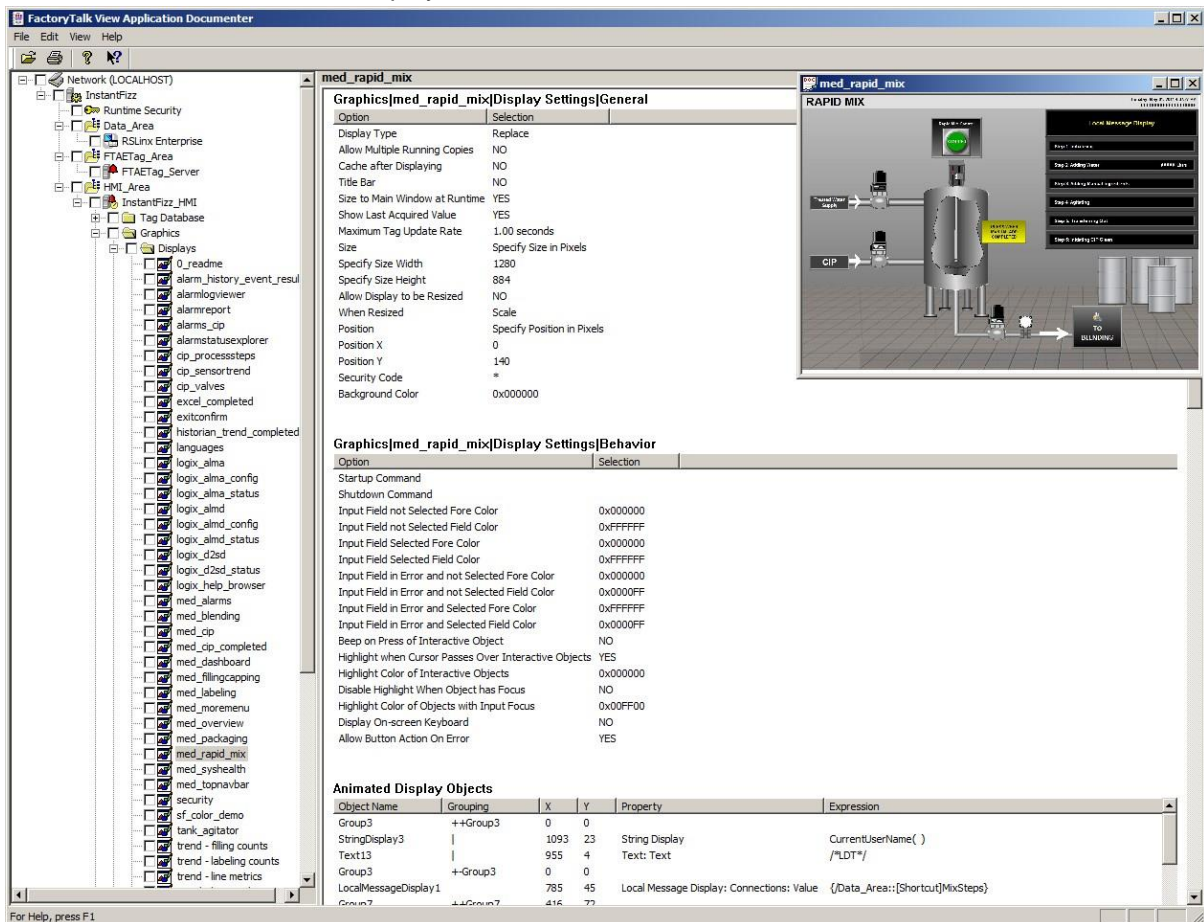


3. Select the InstantFizz application. Follow steps a to c.





- Once the application is opened, the Application Documenter main window will open, presenting the list of project components to choose from in the left pane. Expand **Displays** and click on any display to open it. Details of this display will be shown on the right pane. Note that a screen shot of the display is also shown.



- Application Documenter is capable of generating a report for the selected components and exporting it into an html format. To generate a report select **Export** from the **File** menu. The

html reports can be exported to any location specified. Note that html report will be generated only for the components with the check box checked.

The report contains links to other pages in the report. These other pages are stored in the xml format in the HMI project and/or area subfolders, depending on the components the report has been generated on.



You now have the ability to copy and view the exported files on any computer, regardless of whether that computer has FactoryTalk View installed on it or not.

6. Printing functions are available from the **File** menu. Similar to the Exporting feature, the report will be printed only for the components with the check box checked.

Feel free to play with this useful tool to check on things like tag and parameter file cross referencing.

---

## Section 9: Extending FactoryTalk View SE functionality using VBA

### About this section

In this section of the lab you will:

- Learn what VBA is and how you can use the VBA editor to add functionality in FactoryTalk View SE
- Learn how to enable VBA function to an object on a display
- Do a sample code walk through to understand the different sections of the VBA code

### Introduction to VBA

Microsoft Visual Basic for Applications (VBA) is an integral part of FactoryTalk View Site Edition. With the Microsoft VBA IDE (Visual Basic for Applications Integrated Design Environment) you can write, edit, test run, and debug code to extend some of the features and functionality of FactoryTalk View SE.

When a graphic display opens on an FactoryTalk View SE client computer, the VBA code executes independently on that client. For example, suppose the same graphic display is opened on two different client computers at the same time. When an operator on the first computer clicks a button on the display, the action causes the VBA code to run. The same display, open on another client computer, does not execute the VBA code until an event on that client triggers it. VBA code is event driven. This means that an event needs to trigger the code to run. Code that runs in response to an event is called an event handler.

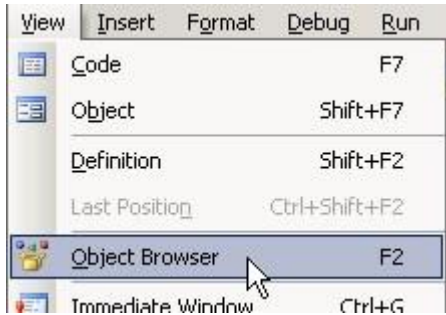
Using VBA code allows a developer to add custom features and functionality such as custom alarm events, validating operator input, creating custom operator forms, manipulating the FactoryTalk View SE Client window and sending custom messages to the Diagnostics log.

The VBA editor provides code editing, navigation keys, and keyboard shortcuts in the code window. It also includes tools that complete keywords as you type (Intellisense), provides a list of options at each stage in building an expression, and verifies syntax after you enter a line of code.

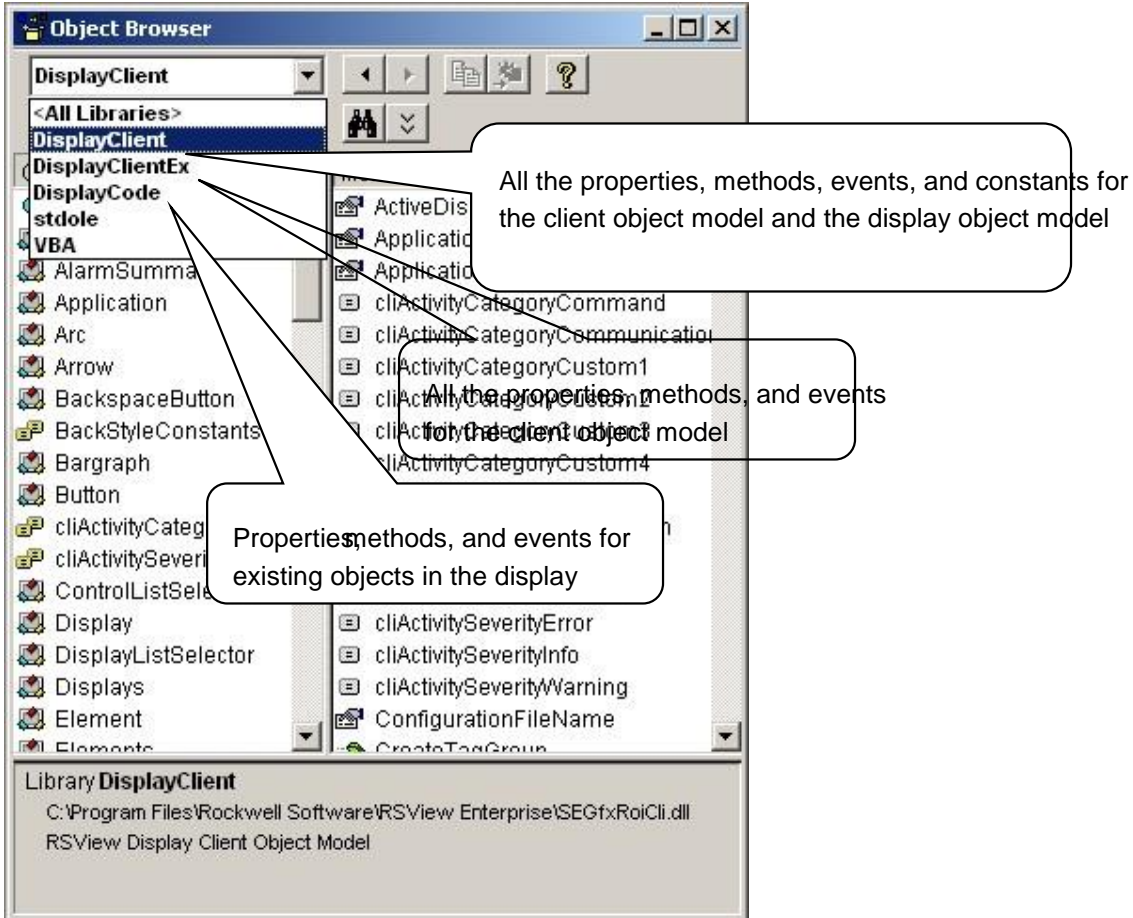
We will learn more about the VBA editor as we proceed!

### View the FactoryTalk View SE object model

1. Return to FactoryTalk View Studio
2. From the **Displays** folder in the Project Explorer, open the display named **med\_VBA**
3. Open the Visual Basic Editor by clicking on **View > Visual Basic Editor**
4. In the Visual Basic Editor, select the **View > Object Browser** menu item or **press** the **F2** key.



- To get help on an object or property, select it and click the help button.



- Close the Object Browser by clicking the X in the corner.
- You can either close the Visual Basic Editor or simply Minimize the window.

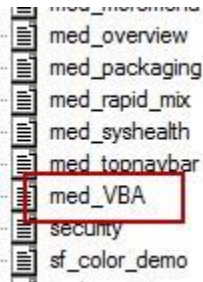
## Understanding VBA Code and Making Edits to the VBA code

A majority of the VBA code has already been written for your convenience. We will do a walkthrough of the code and understand the different sections of the code. You will have an opportunity to edit the code and add a code snippet in. Here is the scenario that we are implementing in this section of the lab.

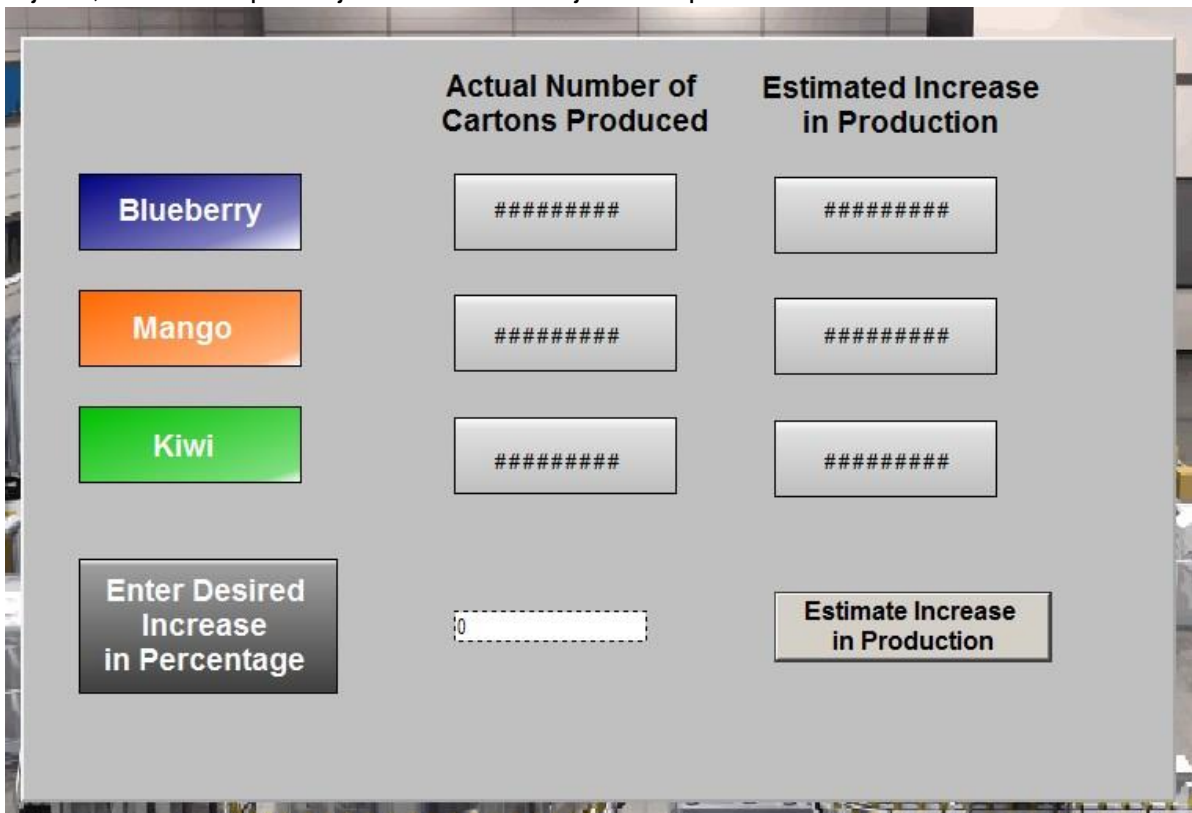
A Plant Supervisor would like to see the number of cartons of Blueberry Juice, Mango Juice and Kiwi juice that were produced. He also wants to type in the percentage of increase in production that he would like to see. When he presses a button on the screen, the Supervisor would like the screen to show him the estimated number of cartons of Blueberry, Mango and Kiwi that could be produced if the percentage of production was increased by the value that he typed in.

First, let us take a look at the display.

1. Double click **med\_VBA** from the **Graphics >Display** folder in the Project Explorer.

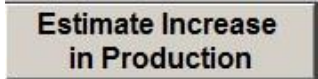


2. The display should open as shown below. Observe what the different Numeric Display objects, Numeric Input Object and Button object is expected to do on the screen.



Recall from the introduction that code that runs in response to an event is called an event handler. In our scenario, we want the **Estimate Increase in Production** button to be pressed in order to display the results. This will make the button an event handler.

When you want an object to interact with VBA you must configure the ExposeToVBA property to either Type Info Extension or VBA Control. Let us see how this was done for the **Estimate Increase in Production** button.

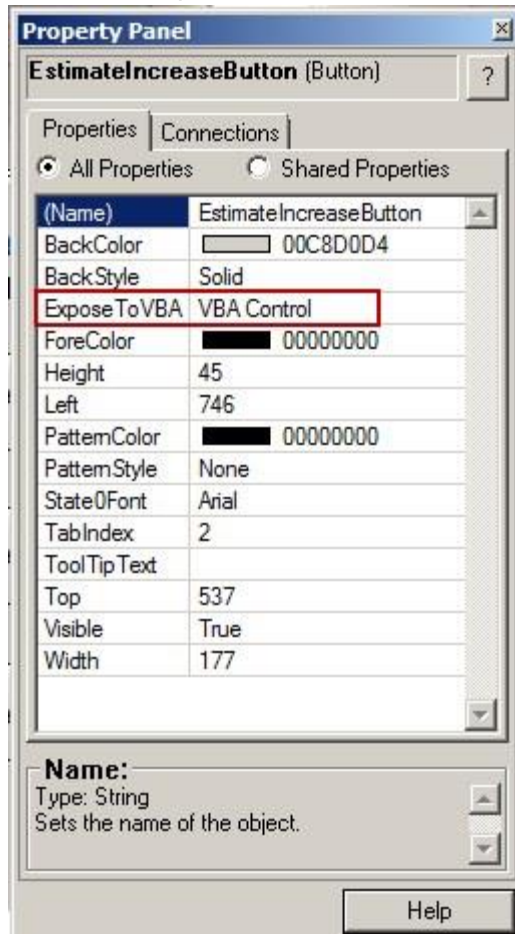


3. Right-click on the **Estimate Increase in Production** button.
4. From the menu that pops up, click on the **Property Panel** option to open the Property Panel window.



5. Click on the **Properties** tab if it is not displayed.

6. From the Properties tab, notice that the **Expose to VBA** property is set to **VBA Control**



**Not Exposed** means that the object cannot interact with VBA code on the display. This is for performance reasons. If there were 10,000 objects on a display and VBA code was used to change a property of one of the objects, the code would have to search through 10,000 objects to find the right one. This is the default setting when an object is created.

**Type Info Extension** means that you can read and write properties of that object but its VBA events cannot be used. Again this is for performance reasons.

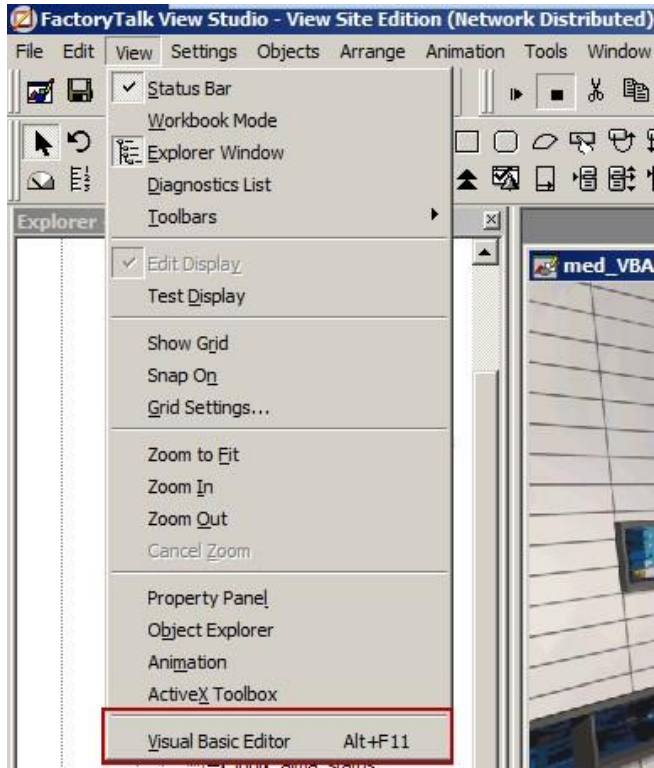
**VBA Control** is for creating an event handler for that object. For example, for the button object you create the click event.

7. Close the Property Panel by clicking on the **X** in corner.

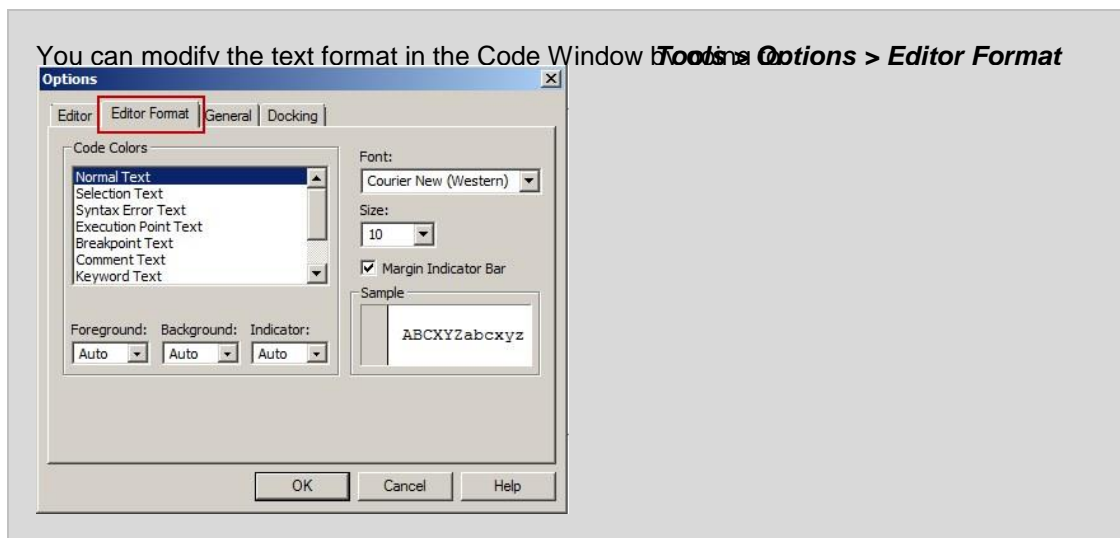
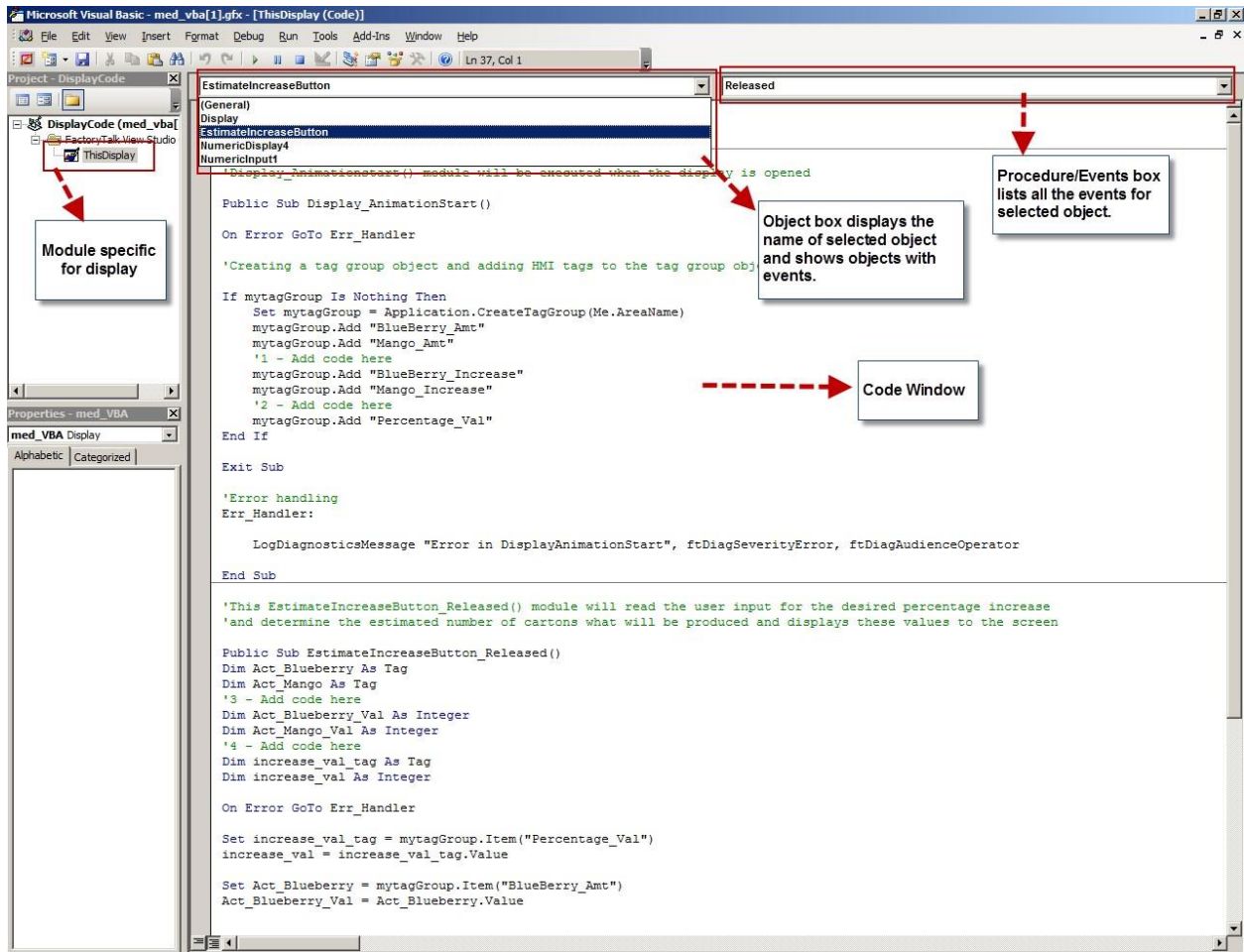
### Launching the VBA Editor

1. Open the VBA editor : **View > Visual Basic Editor**





2. Observe the Microsoft Visual Basic IDE and understand the different sections in the IDE. Double click the **ThisDisplay** under FactoryTalk View Studio Objects on the left if you cannot see the code window. You may have to maximize the VBA IDE window to see the different sections on the screen.



3. The **EstimatedIncreaseButton** object is added to the Object Box because we have added VBA code to the **Estimate Increase in Production** button on the display.

To implement the scenario, we will need 4 important sections in the code:

1. Public Declaration
2. Display\_AnimationStart() subroutine
3. EstimatedIncreaseButton() subroutine
4. Error handling

## Code Walk-through to understand different components of the VBA applications

1. The public declaration is typically the first section of the VBA code.

```
'Public Declaration section  
Option Explicit  
Public mytagGroup As tagGroup
```

The text in green **'Public Declaration section** is a comment in the code module. Adding code comments is a very good coding practice and improves readability and helps anyone reviewing code understand what a line of code is doing. Notice that the `'` (single quote) at the beginning of the line is what tells the compiler that this is a comment line

The **Option Explicit** keyword forces declaration of all variables in a code module. This will therefore 'catch' any variables that are not declared but inadvertently used in the code by a user when testing the code. If you have an undeclared variable that is used in the code without 'Option Explicit', your code will halt with a run time error which is highly undesirable.

The **Public** keyword indicates that a declared variable has 'public' scope or can be accessed by code from anywhere in the code project. In this scenario, we have declared the mytagGroup which is a tag group object as public, because we need multiple functions to be able to access the object.

2. Display\_AnimationStart() subroutine will be executed as soon as the **med\_VBA** display is active or is navigated to.

```

'Display_Animationstart() module will be executed when the display is opened 1

Public Sub Display_AnimationStart() 2
On Error GoTo Err_Handler 3

'Creating a tag group object and adding HMI tags to the tag group object

If tagGroup Is Nothing Then
    Set mytagGroup = Application.CreateTagGroup(Me.AreaName)
    mytagGroup.Add "BlueBerry_Amt"
    mytagGroup.Add "Mango_Amt"
    '1 - Add code here
    mytagGroup.Add "BlueBerry_Increase"
    mytagGroup.Add "Mango_Increase"
    '2 - Add code here
    mytagGroup.Add "Percentage_Val"
End If

Exit Sub 5

'Error handling 6
Err_Handler:

    LogDiagnosticsMessage "Error in DisplayAnimationStart", ftDiagSeverityError, ftDiagAudienceOperator

End_Sub 7

```

**1** Similar to the previous code section, the green comment text informs a reviewer that the module will be executed when the display is navigated to.

**2** The Public Sub Display\_AnimationStart() line indicates that this is a public subroutine

**3, 5 & 6** These lines are important to handle error conditions. If there is a run-time error, we do not want the code to stop executing as this will halt any FactoryTalk View SE runtime processes. We also do not want a message box to pop up indicating an error condition as this message box will then wait for a user intervention. Using error handling code will log any error condition and proceed to the next executable section of the code without halting the process. The LogDiagnosticMessage has multiple parameters that you can choose depending on how you want the error condition to be logged.

**4** Note that Blueberry\_Amt, Mango\_Amt, Blueberry\_Increase, Mango\_Increase, Percentage\_Val are all HMI local tags that have already been created and configured. Browse to the Tag Editor from FactoryTalk View Studio to learn how these tags are set up.



The tag group and tag objects in the Tag Objects Module are used in the VBA code to access tags. This method is preferred since it does not require a display object for each tag being accessed or rely on the screen update rate for tag values. We have already created a Tag group in the first section of code in this section, so we are adding references to the tag group.

Here is an opportunity to create some code. We have filled in the tag references for the Blueberry and Mango tags, but have not added references for Kiwi.

- a) Hit the **Enter** key after the '1-Add code here' line to start a new line. Type **mytagGroup.Add "Kiwi Amt"**
- b) Hit the **Enter** key after the '2-Add code here' line to start a new line. Type **mytagGroup.Add "Kiwi Increase"**

**7** The End Sub keyword marks the end of a subroutine or a function. Control is passed to the next executable line of code at this point.

3. EstimatedIncreaseButton() subroutine will be executed when the button is pressed from the display

**Estimate Increase  
in Production**

```

'This EstimateIncreaseButton_Released() module will read the user input for the desired percentage increase
'and determine the estimated number of cartons what will be produced and displays these values to the screen

Public Sub EstimateIncreaseButton_Released()
Dim Act_Blueberry As Tag
Dim Act_Mango As Tag
'3 - Add code here
Dim Act_Blueberry_Val As Integer
Dim Act_Mango_Val As Integer
'4 - Add code here
Dim increase_val_tag As Tag
Dim increase_val As Integer

On Error GoTo Err_Handler

Set increase_val_tag = mytagGroup.Item("Percentage_Val")
increase_val = increase_val_tag.Value

Set Act_Blueberry = mytagGroup.Item("BlueBerry_Amt")
Act_Blueberry_Val = Act_Blueberry.Value

Set Act_Mango = mytagGroup.Item("Mango_Amt")
Act_Mango_Val = Act_Mango.Value

'5 - Add code here
'6 - Add code here

mytagGroup.Item("BlueBerry_Increase").Value = (Act_Blueberry_Val * increase_val) / 100 + Act_Blueberry_Val
mytagGroup.Item("Mango_Increase").Value = (Act_Mango_Val * increase_val) / 100 + Act_Mango_Val
'7 - Add code here

Exit Sub

Err_Handler:

    LogDiagnosticsMessage "Error in Determining Estimated Increase", ftDiagSeverityError, ftDiagAudienceOperator

End Sub

```

1 This line of code is the start of the EstimateIncreaseButton subroutine. This subroutine will collect the user input, determine the estimated increase in cartons produced and display the results to the screen

2 This is a local declaration for tags and other variables that we will be using later in this module. The declaration has been added for Blueberry and Mango. Let us add the tag declaration and integer value declaration for Kiwi.

- a) Hit the **Enter** key after the '3-Add code here line to start a newline. Type **Dim Act Kiwi as Tag**
- b) Hit the **Enter** key after the '4-Add code here line to start a newline. Type **Dim Act Kiwi Val as Integer**

3 Set increase\_val\_tag = mytagGroup.Item("Percentage\_Val") line assigns the tag group object item Percentage\_Val to the increase\_val\_tag variable. This is where the tag value is read into VBA

- c) **increase\_val = increase\_val\_tag.Value** assigns the actual numerical value of the **increase\_val\_tag.Value** to the integer variable **increase\_val**
- d) Observe how the same value assignments have been done for Blueberry and Mango.

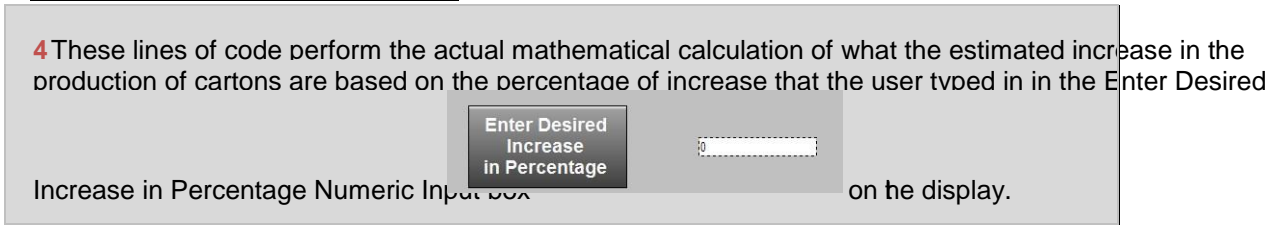
We will now add the assignments for Kiwi.

- e) Hit the **Enter** key after the '5-Add code here line to start a newline. Type the following line

**Set Act Kiwi = mytagGroup.Item("Kiwi amt")**

- f) Hit the Enter key after the '6-Add code here line to start a newline. Type the following line

**Act Kiwi Val = Act Kiwi.Value**



We will now add the line of code that calculates the increase in production for Kiwi.


- g) Hit the Enter key after the '7-Add code here line to start a new line. Type the following line **mytagGroup.Item("Kiwi Increase").Value = (Act Kiwi Val \* increase val) / 100 + Act Kiwi Val**

- h) Finally, save the changes you have made in the VBA IDE by selecting **File > Save**

Great! We have now completed the code review and walk through and added the missing lines of code to make our display a completely functional display that will calculate the Estimated increase in cartons production. Let us test this display and observe the results.

### Testing the display with VBA Code

Return to the FactoryTalk View Studio environment. You will see the **med\_VBA** display. If this is not already open, double click the display to open it from the Project Explorer.

1. Click the **Test Display** button 
2. Watch the display update with the Actual Number of Cartons Produced for Blueberry, Mango and Kiwi
3. In the **Numeric Input** object to the right of the **Enter Desired Increase in Percentage** text, type in a value such as **20**.



---

4. Hit the **Enter** key.

5. Press the **Estimate Increase in Production** button.

Notice how the Estimated Increase in Production values are populated in the Blueberry, Mango and Kiwi Numeric Display objects on the right side.

6. Close the **med\_VBA** screen

Congratulations! You have learned how to write VBA code to solve a simple scenario that involves reading values from tags, accepting user input and writing results to tags that will be displayed on screen.

## Section 10: FactoryTalk ViewPoint

### Before you begin

FactoryTalk ViewPoint is an add-on to FactoryTalk View Site Edition (SE) and FactoryTalk View Machine Edition (ME) running on PanelView Plus that provides for a fully scalable, fully animated view of existing applications from a Web browser.

To make information about your plant or process available on demand, from a Web browser in your office, home, or hotel, all you have to do is select FactoryTalk View graphic displays you want to make ready for the Web, and then publish the displays to a FactoryTalk ViewPoint Server (also called the Web server).

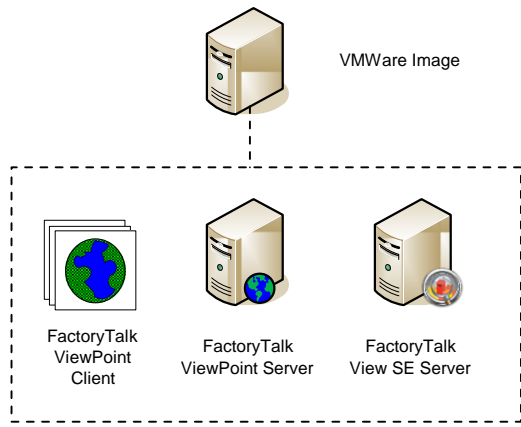
You don't need to install any Rockwell Software products on the browser computer: all you need to connect to a published FactoryTalk ViewPoint Web application is the name (or IP address) of the computer or PanelView Plus hosting the FactoryTalk ViewPoint Server that stores the application.

Then, you can enter a simple address directly into the Web browser, or click a link to the Web address from some other application. For example, you could receive a link to the FactoryTalk ViewPoint Server in an e-mail message, a Microsoft Office PowerPoint presentation, or an Excel spreadsheet.

When a web browser connects to a published web application, the initial display (or displays, if specified in a startup macro) selected for the application opens first. To navigate to other displays, use buttons and touch animation in the initial display, or use the web browser's navigation tools.

As of FactoryTalk ViewPoint 2.0 release, graphic objects in the displays are fully animated and you can use the objects to write to tags or to start and stop HMI components. For example, a numeric input object with a tag connection will display the tag's current value and will also permit downloads.

In the lab you'll be using InstantFizz application. A typical Network system could have separate machines for the FactoryTalk View SE Server, the FactoryTalk ViewPoint Server, and the FactoryTalk ViewPoint client. For this lab we will be running all three roles on the same server.



### About this section

This section provides you with an opportunity to explore the basics of FactoryTalk ViewPoint SE. FactoryTalk ViewPoint SE is used to develop and run web applications for FactoryTalk View SE network and local projects.

The FactoryTalk ViewPoint client is a comprehensive web client that can connect to FactoryTalk ViewPoint server.

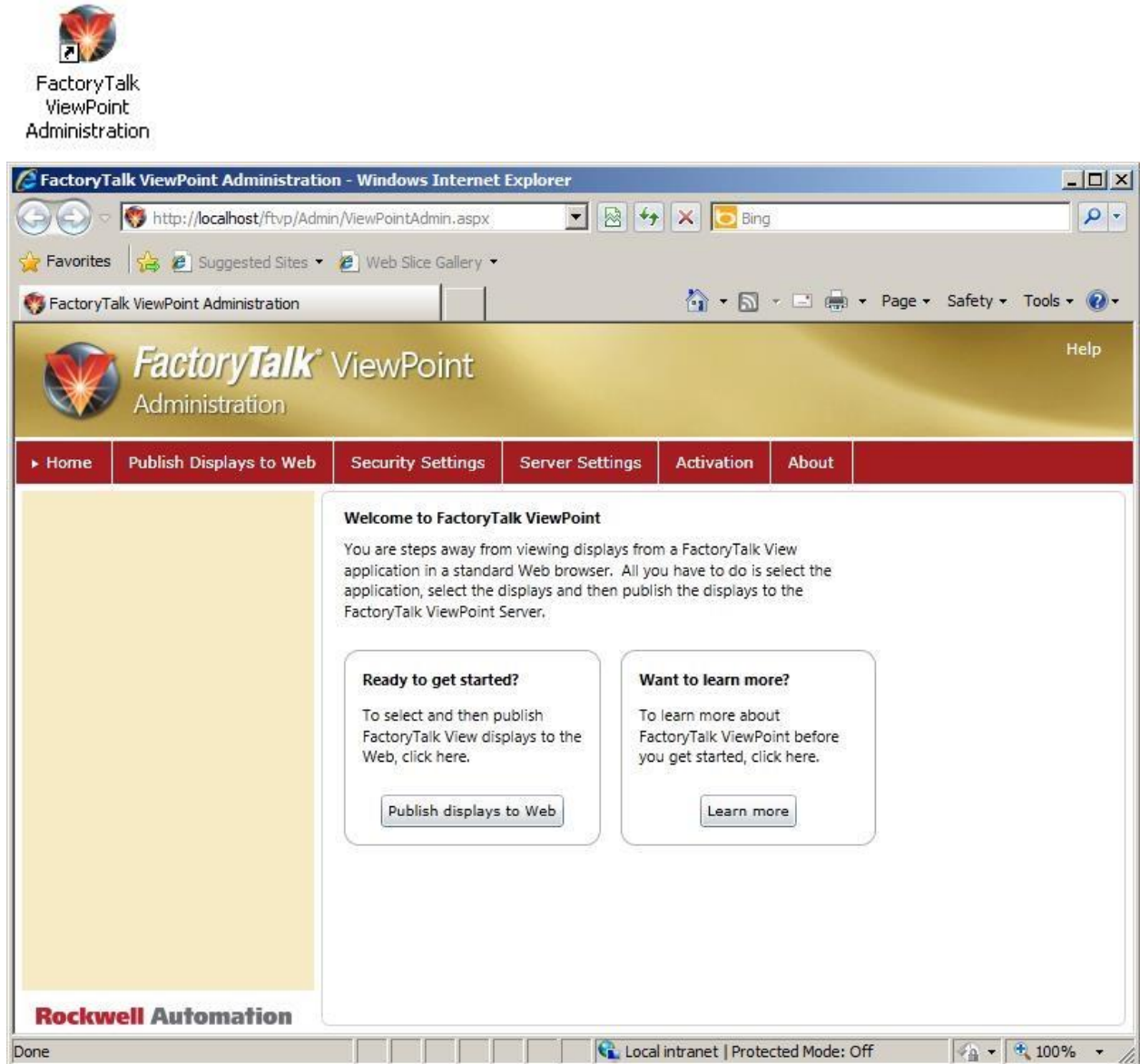
As you complete the exercises in this session, you will gain an understanding of the functionality and capability of FactoryTalk ViewPoint by:

- Connecting to the FactoryTalk ViewPoint Administration Site
- Configuring FactoryTalk ViewPoint Administration Security
- Exploring FactoryTalk ViewPoint Server Settings
- Activating FactoryTalk ViewPoint
- Publishing a FactoryTalk View SE application
- Running an application in a browser
- Configuring FactoryTalk ViewPoint Application Security
- Utilizing Tag Write Functionality

## Connecting to the FactoryTalk ViewPoint Administration Site

FactoryTalk ViewPoint's configuration is web enabled. This allows configuration of the FactoryTalk ViewPoint Server from any machine with network access to the ViewPoint Server. The FactoryTalk ViewPoint Administration is served to the connecting client computer. No additional software packages beyond an internet browser need to be installed.

1. Double-click the **FactoryTalk ViewPoint Administration** shortcut on the desktop.



The FactoryTalk ViewPoint Administration pages allow you to publish displays to the web, configure security, set up activation, view server settings and access the on-line help.

## Configuring FactoryTalk ViewPoint Administration Security

FactoryTalk ViewPoint Security has been redesigned in version 2.0 to accommodate tag writing capabilities that were introduced in this product version. It still allows you to secure access to the FactoryTalk ViewPoint Administration site and the published web application but with added flexibility and

granularity. Security is now role based where a 'role' refers to a FactoryTalk Security group, as you will see in this section.

1. Select **Security Settings** on the red navigation bar in the FactoryTalk ViewPoint Administration Site window.



You will notice two security tabs:



**Administration** - used for securing access to FactoryTalk ViewPoint Administration console and **Secured Displays** – used for configuring run-time security.

In this section we will focus on the Administration portion of FactoryTalk ViewPoint security. Run-time security will be covered later in the lab.

In the User Groups area you can see two default groups that have been automatically created by FactoryTalk View SE:

**Administrators**, **Authenticated Users** and **Windows Administrators**. In addition you can see two user groups called **Operators** and **VP\_Admin** that has been pre-configured for this lab.



2. In the next steps we will create new user groups for Factory Talk ViewPoint specific roles. Minimize FactoryTalk ViewPoint Administration console and open FactoryTalk Administration Console: select **Start > All Programs > Rockwell Software > FactoryTalk Administration Console**.



3. Select the **Network** directory and then **OK**.

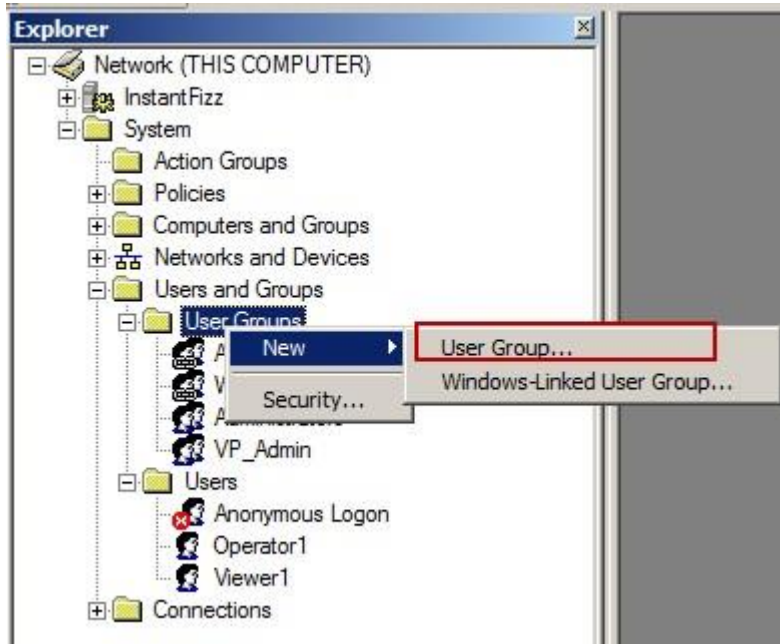


4. Expand **Users and Groups** folder and **User Groups** and **Users** subfolders.

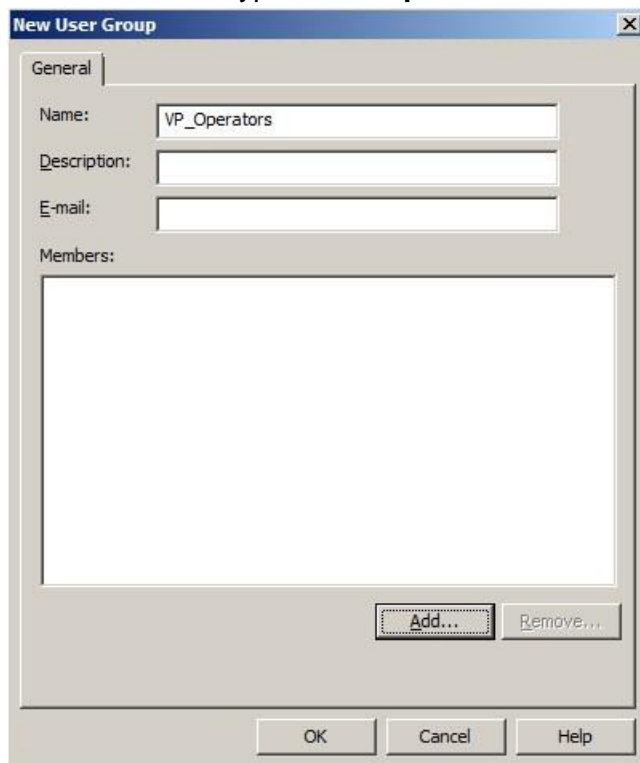


Under User Groups you will notice a previously mentioned pre-configured group called **VP\_Admin**. Configured members of this group are Windows Administrators which include the **labuser** user.

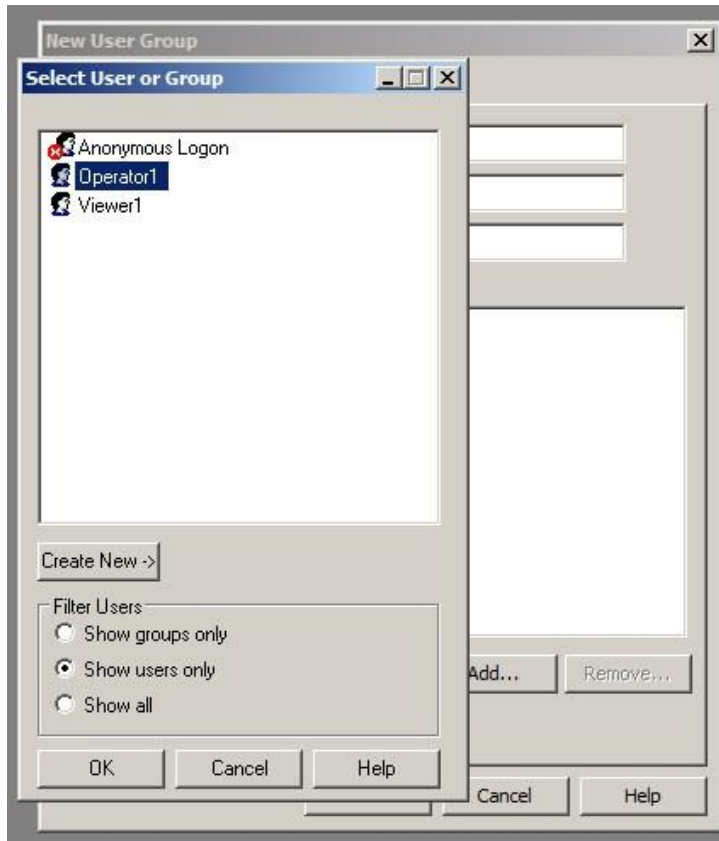
Also, note the two pre-configured users: **Operator1** and **Viewer1**. Currently, these users are not members of any user group. In order to use these users in FactoryTalk ViewPoint application, we will create two new ViewPoint specific groups 5. Right-click on **User Groups** folder and select **New > User Group...**



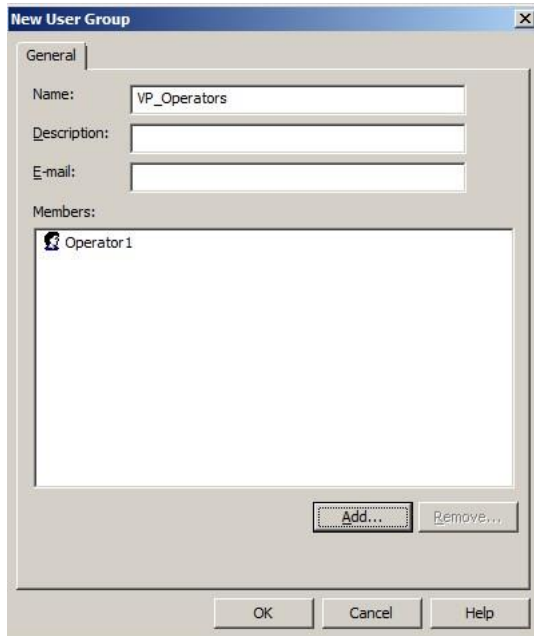
6. In the **Name** field type in **VP\_Operators** and then click **Add...**



7. Select **Show users only**, click on **Operator1** and select **OK**.

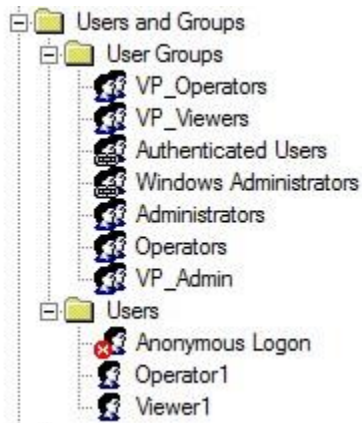



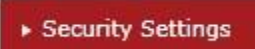
8. Select **OK** again to finish creating **VP\_Operators** group.



9. Repeat steps 5 - 8 to create **VP\_Viewers** group and add **Viewer1** user as its member. Upon completion, your User Groups tree should look like the screen capture below:



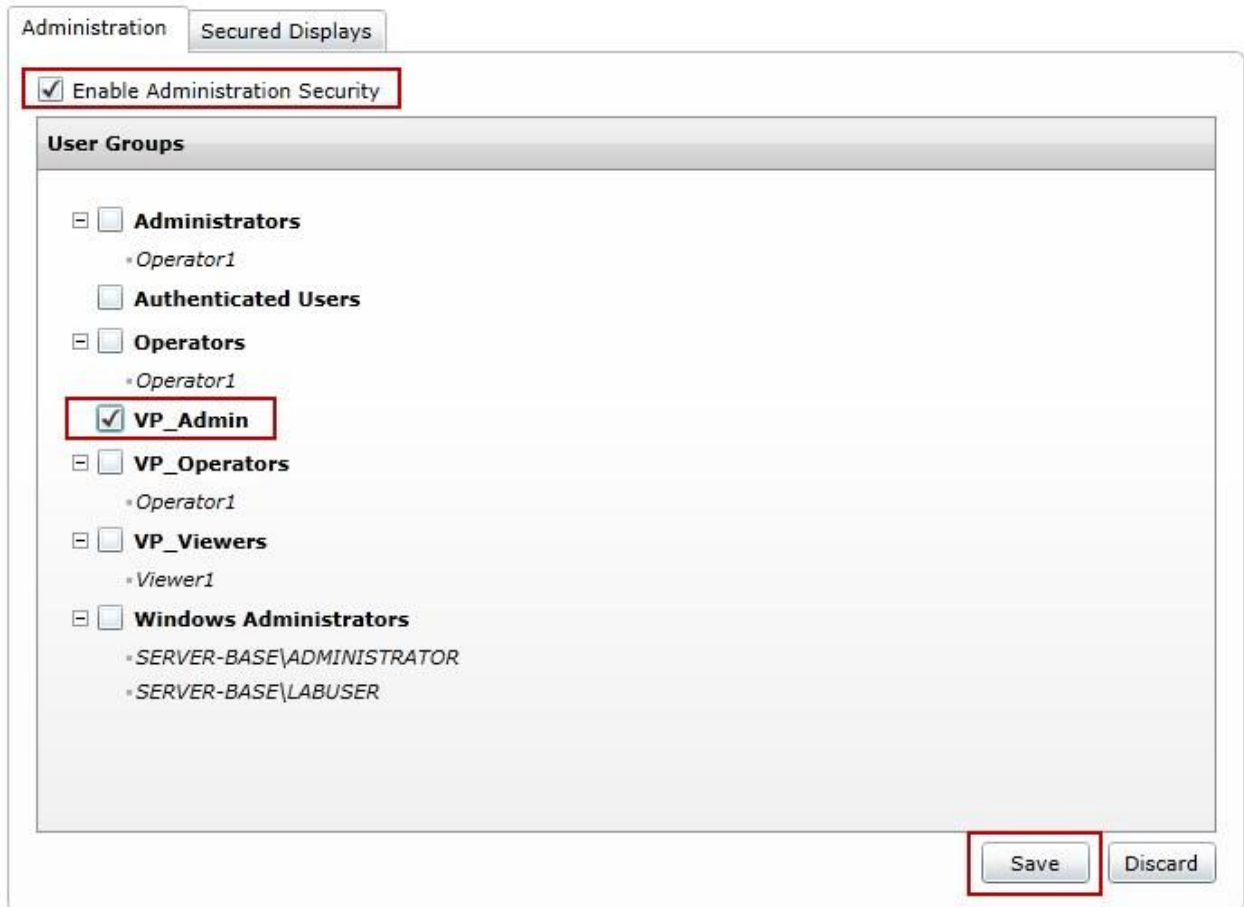


10. Close **FactoryTalk Administration Console** and click on  icon on the Task bar to restore the FactoryTalk ViewPoint Administration Console if it is minimized.
11. Press **F5** to refresh the browser and select **Security Settings**  again (as you did in step 1). The two new user groups that you just created, VP\_Operators and VP\_Viewers, now show up in the User Groups list.
12. Expand all user groups to see the users that belong to each of the groups. Your FactoryTalk ViewPoint User Groups tree should look like the screen capture below:

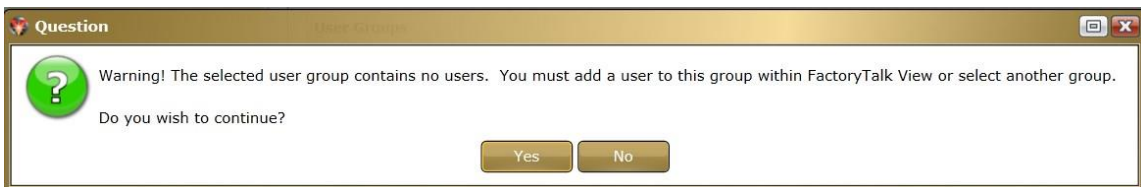


Next, we want to setup the security so that only the users in the VP\_Admin group can have access to FactoryTalk ViewPoint Administration Console. Remember that VP\_Admin has been pre-configured to include Windows Administrators group as its member. This means that all the users belonging to Windows Administrators are also members of the VP\_Admin user group.

13. Select the check box next to **Enable Administration Security**, then select the check box next to **VP\_Admin** and save the changes by clicking on the **Save** button.



14. The message window will show up warning you that the selected user group contains no users.



It is true that VP\_Admin group has no explicitly added users. However, since the Windows Administrators group is a member of VP\_Admin group, all the Windows Administrators users have also been added indirectly to VP\_Admin group.

Click **Yes** to continue.

15. Press **F5** to refresh the browser.

16. You will notice that security is now enabled and you are required to log in to access the Administration Site. Log in using the username 'Operator1' and password 'Operator1'.

▶ Home About

▶ Log on  
▶ Administration

User name Operator1

Password ●●●●●●●●

Directory Network

Remember me on this computer

Log on

Although the entered credentials for Operator1 are valid, the logon attempt will fail because the user Operator1 is not a member of the VP\_Admin group.

▶ Home About

▶ Log on  
▶ Administration

User name

Password

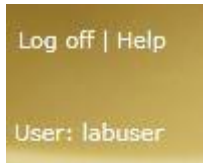
Directory Network

Remember me on this computer

Log on

User 'Operator1' is not authorized to access the ViewPoint Administration. Please log on with appropriate credentials.

17. We had only given access to the Administration site to VP\_Admin users which are also Windows administrators.  
Since the labuser is the member of this group, log on using the username 'labuser' and the password 'rockwell'. Do not select **Remember me on this computer** checkbox. The logon attempt will be successful this time.
18. In the top-right corner of the browser window you can see which user is currently logged on and you are also given the option to log off.



19. Click on **Security Settings** tab again.

20. Clear the check box next to **Enable Administration Security**

Enable Administration Security

and save the changes by clicking on the **Save**  button. This will allow all users full access to FactoryTalk Administration web page so that security logins are no longer required for the rest of this lab.

21. Press F5 to refresh the browser.

## FactoryTalk ViewPoint Server Settings

Additional information about the FactoryTalk ViewPoint Server can be found on the Server Settings tab.

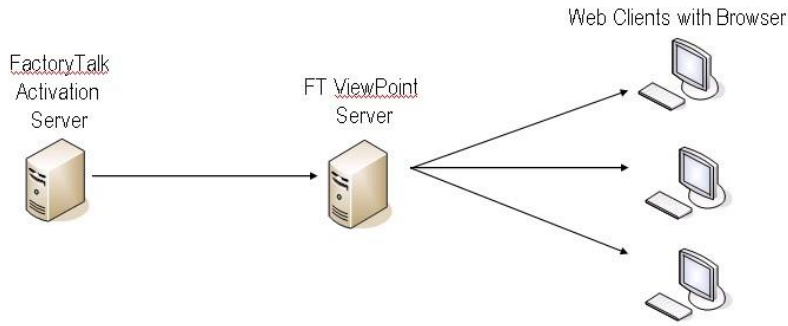
A screenshot of a web application interface. At the top, there is a horizontal navigation bar with several tabs: "Home", "Publish Displays to Web", "Security Settings", "Server Settings" (which is highlighted with a red border and a small arrow icon), "Activation", and "About". Below the navigation bar, the main content area is divided into two sections. On the left, there is a vertical sidebar with a yellow background and a red arrow icon pointing to the text "Server Settings". On the right, there is a table of configuration settings. The table has two columns: the left column contains labels for various settings, and the right column contains their corresponding values. The settings listed are: FactoryTalk Directory (SERVER-BASE), Application Type (Site Edition (Network Distributed)), Application Name (InstantFizz), Web server name (localhost), Web client port (80), and Base URL (http://localhost/FTVP).

Most important is the name of the server, the port it's using and the URL that is used to connect to that server.

## Activating FactoryTalk ViewPoint

Licensing is not required for FactoryTalk ViewPoint server functionality; however, the FactoryTalk ViewPoint clients do require a client access license (CAL) to connect to a FactoryTalk ViewPoint server. Client access licenses (CALs) are concurrent FactoryTalk activations and are provided to the FactoryTalk ViewPoint clients by the FactoryTalk ViewPoint server. When a client consumes a CAL, it allows three browser or tabbed instances of the published web application to be open at a time.

The FactoryTalk ViewPoint server is configured to consume CALs from the FactoryTalk Activation server. In the case where multiple FactoryTalk ViewPoint servers are being used, licenses can be shared among the FactoryTalk ViewPoint servers by configuring each server to consume a selected number of CALs.



1. Select the **Activation** tab.

By default, a ViewPoint server will automatically consume 3 licenses after install but this can also be configured by entering a new number of licenses, ranging from 1 to the number of available licenses, and selecting 'Apply'. As you can see, there is a total of 25 licenses for this lab. We will leave the number of assigned licenses set to 3.

## Publishing a FactoryTalk View SE application

FactoryTalk ViewPoint Administration also allows you to publish web applications to the FactoryTalk ViewPoint server.

FactoryTalk ViewPoint supports both Site Edition Network Distributed and Station as well as Site Edition Local applications.

Let's go ahead and publish an application.

1. Select the **Publish displays to Web** tab.

Home **► Publish Displays to Web** Security Settings Server Settings Activation About

Follow these 3 steps to make a FactoryTalk View application Web-ready:

- 1 Select an application**
- 2 Select displays
- 3 Publish displays

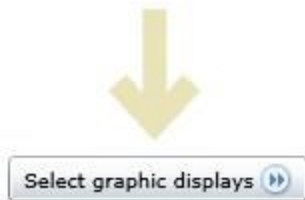
**FactoryTalk Directory:** SERVER-BASE  
**Application Type:** Site Edition (Network Distributed)  
**Application Name:** InstantFizz

Select a FactoryTalk View application type:  
 Site Edition (Network Distributed) ▼

Select the application:  
 InstantFizz ▼

Select **Site Edition (Network Distributed)** if it is not selected and make sure that the application **InstantFizz** is selected.

2. Click the **Select graphic displays** button in the bottom right corner of the browser.



FactoryTalk ViewPoint will only maintain one FactoryTalk ViewPoint application configuration at a time.

Selecting a different FactoryTalk View application, or selecting a different set of displays to publish, will replace the current Web application configuration and the published content.

3. A list of displays within the ViewPoint application will be displayed. Expand the area **HMI\_Area**, if it's not already, by clicking on the + next to **HMI\_Area** to view the entire list.
4. The **med\_dashboard** display is not currently published so let's add it to the list by selecting the checkbox beside the display name.

Web Enable	Initial Display	Name
<input checked="" type="checkbox"/>	<input type="radio"/>	logix_d2sd
<input checked="" type="checkbox"/>	<input type="radio"/>	logix_d2sd_status
<input checked="" type="checkbox"/>	<input type="radio"/>	logix_help_browser
<input checked="" type="checkbox"/>	<input type="radio"/>	med_alarms
<input checked="" type="checkbox"/>	<input type="radio"/>	med_blending
<input checked="" type="checkbox"/>	<input type="radio"/>	med_cip
<input type="checkbox"/>	<input type="radio"/>	med_dashboard

All displays are initially selected, but you can choose which displays you would like to publish. By default the initial display—the first display that opens in the Web browser—is the first display listed in the Administration Site.

To remove a display from a published application, uncheck the display prior to publishing.

5. You can change the initial display by selecting an alternative display in the **Initial Display** column or you can select a startup macro for opening displays on startup. If you specify a startup macro, the Initial Display setting is ignored.

Since the ViewPoint application uses a docked display, select the startup macro named **med\_startup\_client** if it is not already selected. This macro displays the navigation bar at the top of the screen as a docked display and then the **med\_overview** screen by running the following FactoryTalk View SE commands:

```
Display Med_TopNavBar /DT
```

```
Display Med_Overview
```

```
&Set MoreMenuTag 0
```

Using docked displays and a startup macro is the best approach for configuring FT View applications with multiple display windows that also need to run in FactoryTalkViewPoint.

<input checked="" type="checkbox"/>	<input type="radio"/>	cip_processsteps
<input checked="" type="checkbox"/>	<input type="radio"/>	cip_sensortrend

Re-publish all selected displays [What is this?](#)

Select a startup macro:

6. Select the **Publish Displays** button. Observe progress as the Site analyzes the selected displays and prepares these displays to be hosted in the browser before publishing.

Depending on the size of the displays, the publishing process may take several minutes when performed for the first time. Subsequent publish actions are optimized to analyze and prepare any displays that have been changed or selected to be published. In the interest of time, a majority of the displays have already been published and are currently selected in the Web Enable column. Republished will be only the displays that have been modified in the previous sections of the lab.

You also have the option of re-publishing all selected displays, whether or not they have been modified. The Re-publish all selected displays option deletes the entire existing published content before the new content is created based on selected displays. This selection is useful after upgrading to a new version of ViewPoint to ensure that the published application contains the latest features.




When the publishing process is complete, you will be provided with a link to the publishing report which will describe any errors or warnings that may have occurred during publication and a link to your initial FactoryTalk ViewPoint display. Any objects which are not supported in FactoryTalk ViewPoint will be listed as a warning.

7. Click the **View publishing report** link to open a separate browser window containing the publishing report. When you are finished reviewing the report, close the browser window containing the report.

Done	<a href="#">/HMI_Area/cip_processsteps</a>
Done	<a href="#">/HMI_Area/cip_sensortrend</a>
Done	<a href="#">/HMI Area/cip_valves</a>

Status: Finished publishing. Warnings are recorded in the publishing report.

[View publishing report](#)



**Application Name:** InstantFizz  
**ViewPoint Server Name:** SERVER-BASE  
**Time of Conversion:** Wednesday, April 23, 2014 11:24 AM

**System messages**

Application Name	Status	Errors	Warnings
InstantFizz	Converted		Yes

**Area: /HMI\_Area**

37 displays  
 Converted: 37  
 Not converted: 0

Display Name	Status	Errors	Warnings
0_readme	Converted		
alarm_history_event_results	Converted		Yes
alarmlogviewer	Converted		Yes
alarms_cip	Converted		
alarmstatusexplorer	Converted		Yes
cip_processsteps	Converted		
cip_sensortrend	Converted		
cip_valves	Converted		Yes
exitconfirm	Converted		Yes
languages	Converted		
logix_alma	Converted		Yes
logix_alma_config	Converted		
logix_alma_status	Converted		
logix_almd	Converted		
logix_almd_config	Converted		
logix_almd_status	Converted		
logix_d2sd	Converted		
logix_d2sd_status	Converted		
logix_help_browser	Converted		Yes
med_alarms	Converted		Yes
med_blending	Converted		Yes
med_cip	Converted		Yes
med_fillingcapping	Converted		Yes
med_labeling	Converted		Yes
med_moremenu	Converted		
med_overview	Converted		Yes
med_packaging	Converted		Yes
med_rapid_mix	Converted		Yes
med_syshealth	Converted		Yes
med_topnavbar	Converted		Yes
security	Converted		Yes
tank_agitator	Converted		Yes
trend - filling counts	Converted		
trend - labeling counts	Converted		

8. After exploring the publishing report, close the browser window that hosts the report.

## Running an application in a browser

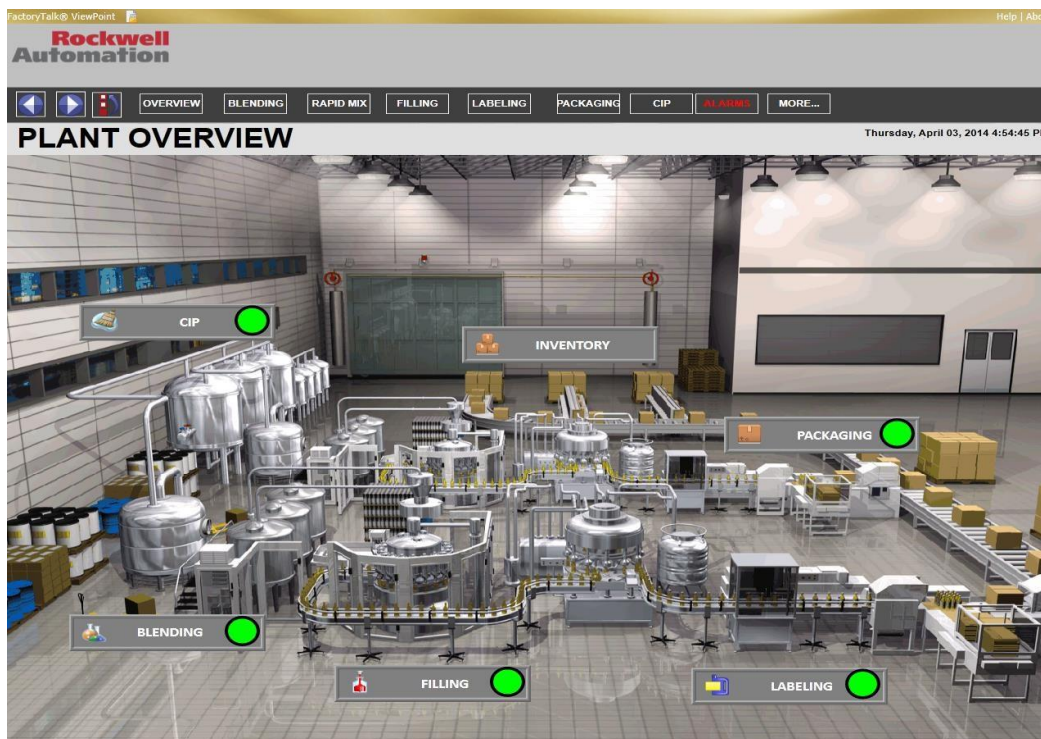
1. After the publishing process finishes, the **Open Web Application** button becomes available. Click on this button.



Open Web application

This opens Internet Explorer and executes the selected startup macro.

2. Maximize the browser. The FactoryTalk ViewPoint screens automatically scale to fit the size of the browser window.



We had specified the **med\_startup\_client** macro during publishing so that when a FactoryTalk ViewPoint client gets connected to the Server, the macro would execute the Display commands providing the navigation bar at the top of the screen as a docked display and the Plant Overview (**med\_overview**) as the main screen. FactoryTalk ViewPoint provides a thin client solution for FactoryTalk View. There is no need to install and maintain any Rockwell Software on the client machine. This lowers a total cost of ownership, minimizes downtime and improves security. FactoryTalk ViewPoint also provides convenient access to FactoryTalk View applications from anywhere (central office, home, internet) which extends the reach of visualization to remote, casual and mobile users such as plant managers, central maintenance engineers, OEMs and System Integrators. Think of all of the remote users in your facility or your customer's facility that could benefit from improved access to plant floor visualization information as you run the InstantFizz application in FactoryTalk ViewPoint.

Starting a FactoryTalk ViewPoint client from within FactoryTalk ViewPoint Administration browser is just one of a number of different ways to start it. You could also use browser features to do this.

3. **Close** FactoryTalk ViewPoint client browse window. You should now be showing FactoryTalk ViewPoint Administration browser. If not, restore this browser instance.
4. Open a new browser tab by clicking on the New Page icon as shown below in the screenshot:



5. In the New Tab address bar type in <http://server-base/FTVP>, the default FactoryTalk ViewPoint page for published content.

You can now conveniently switch between FactoryTalk ViewPoint client and FactoryTalk ViewPoint Administration within the same browser instance. Keep both tabs open as this will be handy for the steps in the next section of the lab.

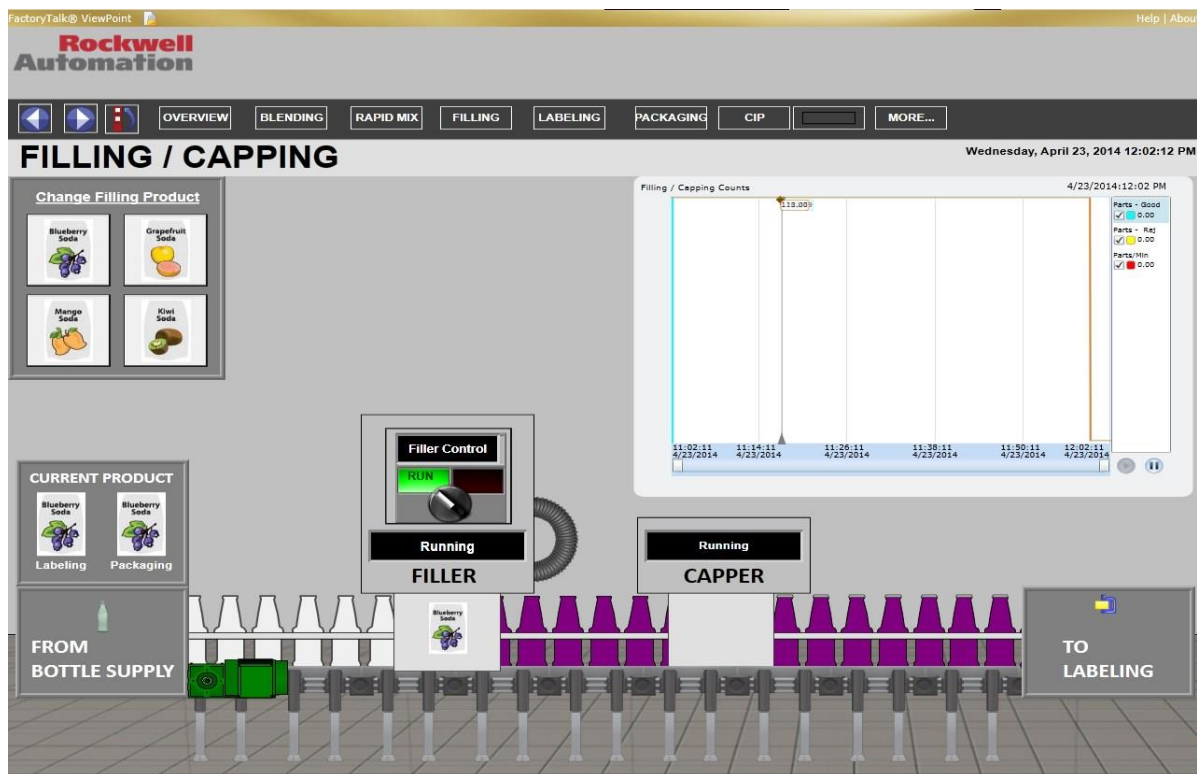
FactoryTalk ViewPoint provides a rich, interactive browser user experience. All of the navigation built into the FactoryTalk View SE application also works in the browser.

- Using the navigation bar at the top, click on the filling button to navigate from the **Overview** display to the **Filling** display. Observe how the navigation functionality works in FactoryTalk ViewPoint.



Filling button

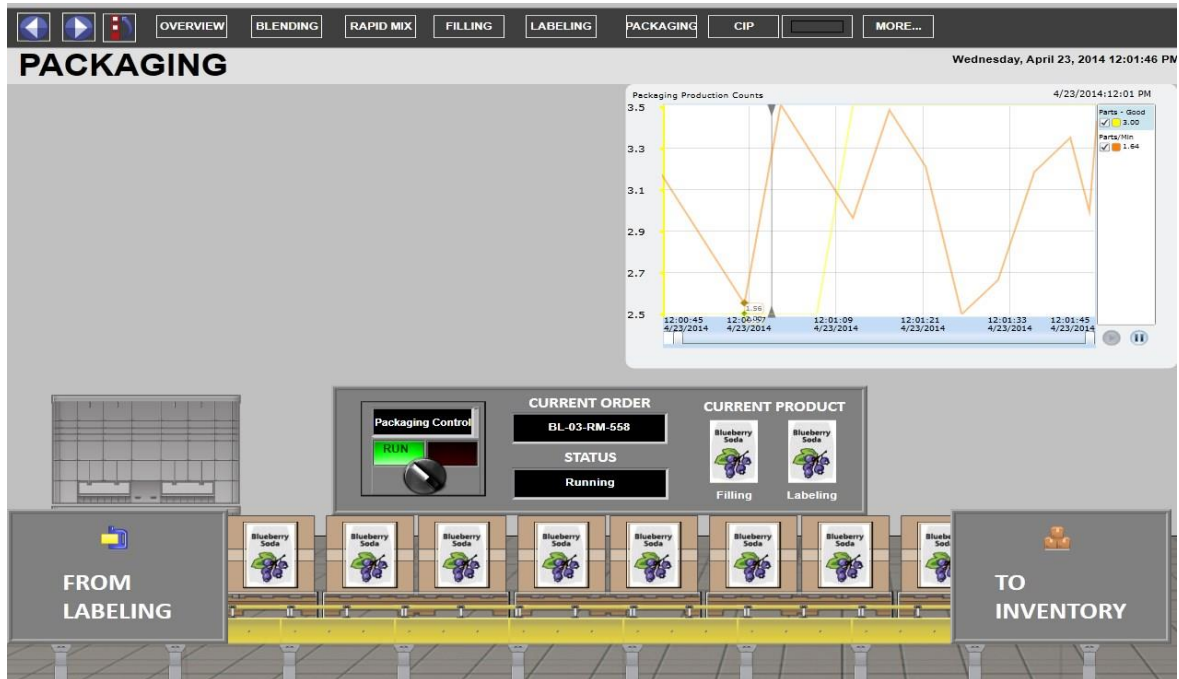
The **Filling** display should now be shown in FactoryTalk ViewPoint. Observe that the animation used to simulate the bottles filling and moving along the bottling line is fully functional in FactoryTalk ViewPoint. FactoryTalk ViewPoint supports full animation of FactoryTalk View SE displays without the need to refresh your browser.



Maximize the browser window if it is not already at full size.


7. Navigate from the **Filling** display to the **Packaging** display (click on the  button).

The **Packaging** display should now be shown in FactoryTalk ViewPoint. Once again, observe that the packaging line objects, such as numeric displays and color animation used to show line status, are updating dynamically. FactoryTalk ViewPoint is fully interactive and does not use static 'screen captures' to display information in the browser.



If the Internet Explorer window is maximized, you can double-click on the top blue Internet Explorer title bar to restore the window to the previous size. Then, by clicking on the bottom right hand corner of the internet explorer window while holding down the left mouse key and dragging, resize the browser window again.

Notice the screen resizes at runtime. This allows you to develop a single display and run it on a wide variety of client devices with different screen sizes and resolutions (including wireless, mobile devices).

8. Navigate from the **Packaging** display to the **CIP** (Clean in Place) display (click on the  button).

The **Clean in Place** display should now be shown in FactoryTalk ViewPoint.

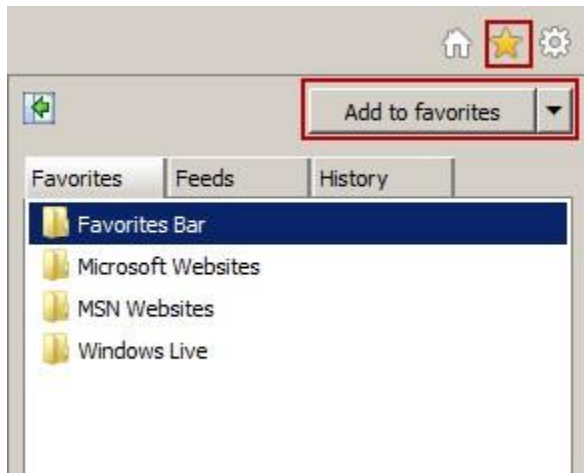
9. In the Internet Explorer click on **Favorites** icon and then on **Add to Favorites...** button. You can do this by

clicking on the Favorites button  in your browser or by pressing the '**ALT**' key to bring up the

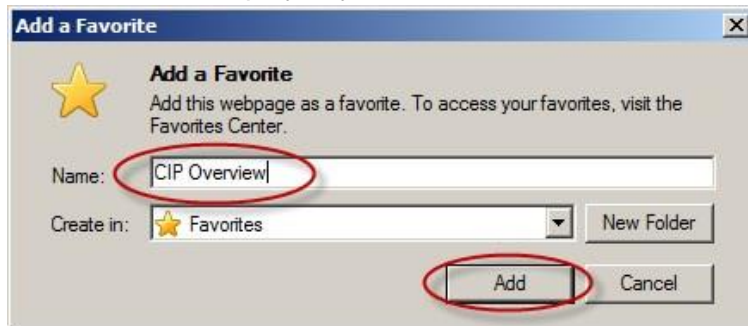




from which Favorites can be selected

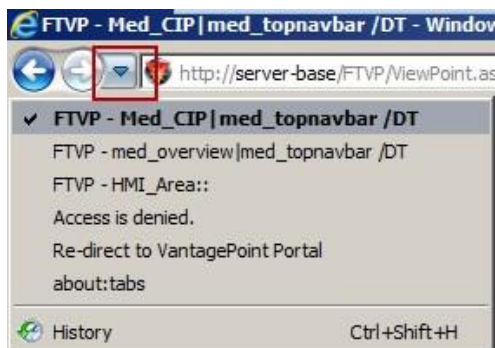


10. In the **Name** field, type in a friendly name for the CIP display (i.e. CIP Overview) and Click **Add** to add the display to your Favorites list.



You can now navigate to CIP Overview display using Favorites feature of the Internet Explorer.

11. In the Internet Explorer click on Recent Pages button and click on the link **FTVP – HMI\_Area::** (which is a short name for the initial application page).



Instead of using FactoryTalk ViewPoint navigation buttons, you navigated back to the Plant Overview display using Internet Explorer navigation feature.

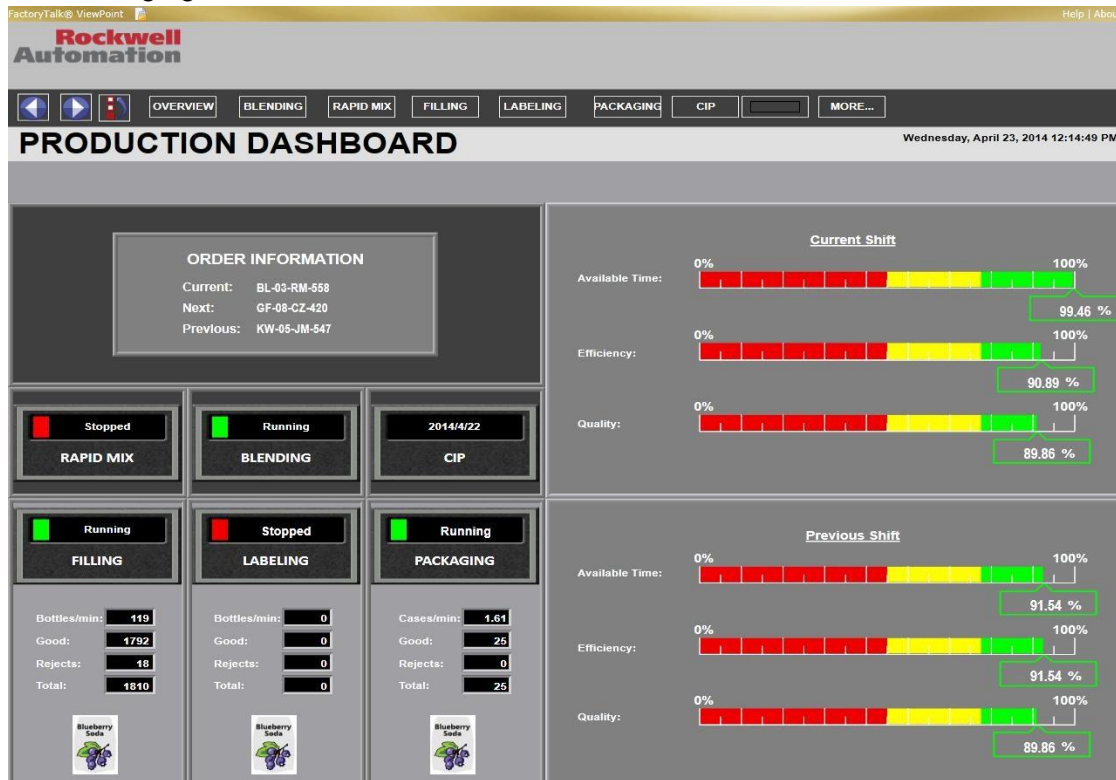
12. From the Plant Overview display navigate to the **Manager Dashboard** display by clicking on the **More**



button and then clicking on the **Dashboard** button



13. The Production Dashboard display provides a nice example of a display specifically designed for a remote, casual ViewPoint user such as a Plant Manager. This display provides an overview of line status and production data. From here you can navigate to other displays or display trends associated with Filling, Labeling and Packaging data.



14. Try putting Internet Explorer browser in full screen mode by hitting the **F11** function key. FactoryTalk ViewPoint application will occupy the entire screen. Press **F11** again to return to a normal view mode.
15. Continue to browse the application with FactoryTalk ViewPoint by navigating to the **Rapid Mix, Blending** and other displays and observe the behavior in FactoryTalk ViewPoint.

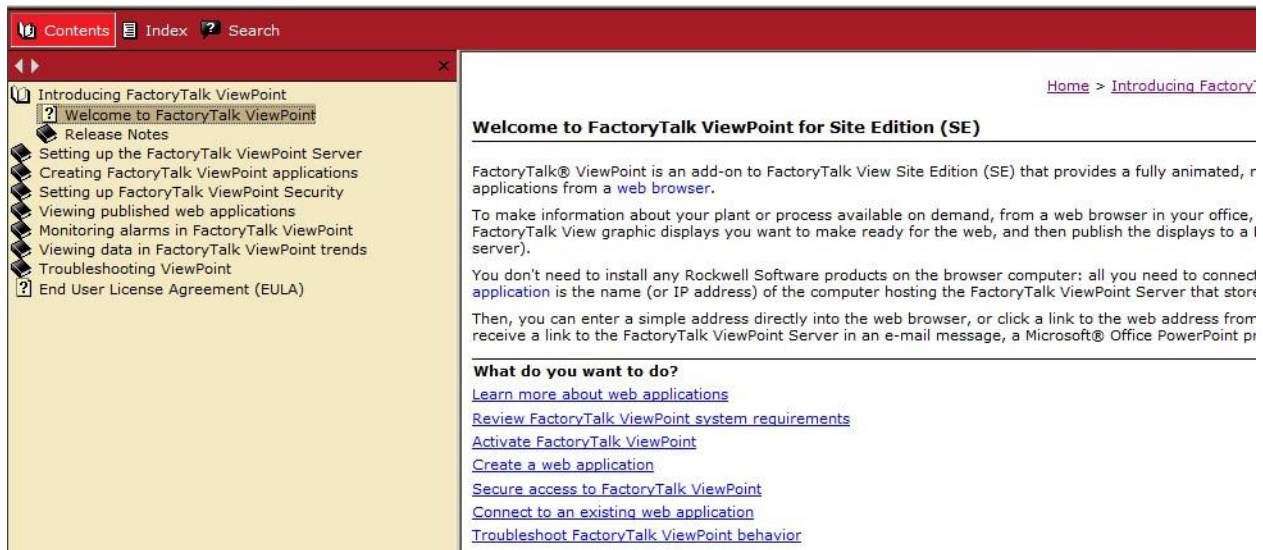


FactoryTalk ViewPoint has an informative help file that is easily accessible from any FactoryTalk ViewPoint client or FactoryTalk Administration site.

16. Open the product Help file by clicking Help in the upper right hand corner.



17. The help file covers important topics like creating and viewing applications, security, alarms, tag write functionality and trends. It also contains the Release Notes which outlines System Requirements, unsupported FactoryTalk View features, and installing any prerequisite software.



Navigate through these topics and any others you choose to view what information is available.

18. Click on **Troubleshooting ViewPoint** from the menu on the left side.



This special section in the Help file allows users to learn about any issues that they may encounter and provides assistance if troubleshooting is required. The user would find the heading that describes what they were doing at the time and clicks on the link that best characterizes the behavior that was observed.

19. Close the Help window by clicking on the 'X' in the upper right hand corner.


Congratulations! You have successfully published the FactoryTalk View SE ViewPoint application and can now run it using your web browser. Note that FactoryTalk ViewPoint automatically converted and published the displays to the FactoryTalk ViewPoint Server; you did not need to make any modifications to the FactoryTalk View SE application to accomplish this.

## Configuring FactoryTalk ViewPoint Application Security

As mentioned in the previous section, there are two portions of security – Administration and Secured Displays. In the next steps we will configure and explore effects of application security at the application and display level.

### Global Application Security

1. Navigate to the **CIP** display again (click on the  button).

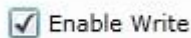
2. The **CIP – Repeat Cycle**  button sets a tag in the controller that initializes the CIP process. **Click** on this button.

You will notice that the button does not respond to click action. This is because tag write operations are disabled in FactoryTalk ViewPoint Security by default.

- In Internet Explorer, switch to the tab hosting **FactoryTalk ViewPoint Administration** console. (if you no longer have this tab open, start a new tab and type in <http://server-base/FTVP/admin> on the New Tab address bar to start a new instance of the console)
- Click on **Security Settings**, and select the **Secured Displays** tab.

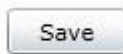


- Application security is currently disabled. Select the **Enable Write** checkbox



that is located in the middle of the screen

and click on the Save button



to save the changes.

- Switch back to the browser tab running the FactoryTalk ViewPoint client and press **F5** to refresh the browser.

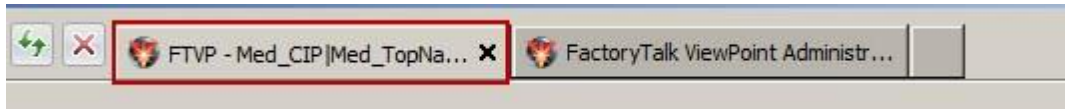
- Click on **CIP – Repeat Cycle**  button again.

This time the click action will take place and you will see CIP process steps being executed. In the Status Area at the bottom of the screen you will see the message:



This means that the Set command associated with the button was executed successfully.

- Close FactoryTalk ViewPoint client browser tab by clicking on the **X** button.



- Back in the FactoryTalk ViewPoint Administration browser instance clear **Enable Write** checkbox

Enable Write

Save

and click on the Save button to save the changes.

### Display Application Security

Next, we will configure Application security so that only **VP\_Operators** group members can execute CIP related operations.

- Select Enable Application Security checkbox.

Enable Application Security

The open table lists all the User groups and their associated Write or View security permissions. At this point, we have not enabled any application security.

Security permissions for: <b>InstantFizz</b>			
User Group	Write	View	
Administrators	<input type="checkbox"/>	<input type="checkbox"/>	
Authenticated Users	<input type="checkbox"/>	<input type="checkbox"/>	
Operators	<input type="checkbox"/>	<input type="checkbox"/>	
VP_Admin	<input type="checkbox"/>	<input type="checkbox"/>	
VP_Operators	<input type="checkbox"/>	<input type="checkbox"/>	
VP_Viewers	<input type="checkbox"/>	<input type="checkbox"/>	
Windows Administrators	<input type="checkbox"/>	<input type="checkbox"/>	



As noted above the table, the Security permissions apply to the entire **InstantFizz** application because the root of the InstantFizz application is selected in the left hand pane. The security configured at this level will be automatically applied to all displays within the application.

- Double-click** on each of the checkboxes in the **View** column so that the permissions look like the following:

Security permissions for: **HMI\_Area**

User Group	Write	View
Administrators	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Authenticated Users	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Operators	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
VP_Admin	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
VP_Operators	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
VP_Viewers	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
Windows Administrators	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

- In the left hand pane expand **HMI\_Area** by clicking on **+** next to it.
- Click on **cip\_processteps** display to highlight it.

The screenshot shows a web interface with two tabs: 'Administration' and 'Secured Displays'. The 'Secured Displays' tab is active. A checkbox labeled 'Enable Application Security' is checked. Below this is a section titled 'Select an Application, Area or Display to edit security settings'. On the left, a tree view shows 'HMI\_Area' expanded, with 'cip\_processteps' selected and highlighted in blue. On the right, a table titled 'Security permissions for: cip\_processteps' shows the following permissions:

User Group	Write	View
Administrators	<input type="checkbox"/>	<input type="checkbox"/>
Authenticated Users	<input type="checkbox"/>	<input type="checkbox"/>
Operators	<input type="checkbox"/>	<input type="checkbox"/>
VP_Admin	<input type="checkbox"/>	<input type="checkbox"/>
VP_Operators	<input type="checkbox"/>	<input type="checkbox"/>
VP_Viewers	<input type="checkbox"/>	<input type="checkbox"/>
Windows Administrators	<input type="checkbox"/>	<input type="checkbox"/>

- Select the **Show Effective Permissions** checkbox in the upper right corner of the right hand pane. You may need to resize your browser window to view the entire description. The security permission table should look like the screenshot below:





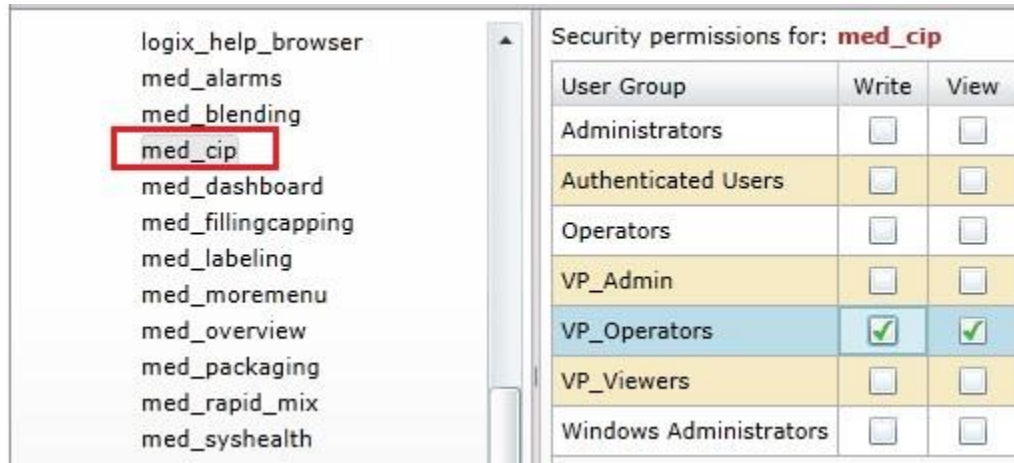
The table indicates that all user groups have permission to view the cip\_processteps display but none of them have tag write permission. As per the Legend, note how both Allow and Deny permissions are Inherited. This means that the permissions are inherited from the level above. In our case the permissions were inherited from the application level (refer back step 1).

- Next, you will assign **Write** permission to **VP\_Operators**. Clear the **Show Effective Permissions** checkbox.
- In the **Write** column, click twice on the checkbox in **VP\_Operators** row to define the Allow permissions.

Security permissions for: **cip\_processteps**

User Group	Write	View
Administrators	<input type="checkbox"/>	<input type="checkbox"/>
Authenticated Users	<input type="checkbox"/>	<input type="checkbox"/>
Operators	<input type="checkbox"/>	<input type="checkbox"/>
VP_Admin	<input type="checkbox"/>	<input type="checkbox"/>
VP_Operators	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>
VP_Viewers	<input type="checkbox"/>	<input type="checkbox"/>
Windows Administrators	<input type="checkbox"/>	<input type="checkbox"/>

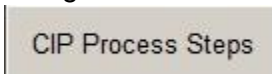
- Click on **med\_cip** to highlight it and configure the security in the same manner as for **cip\_processteps** in the previous step.



- Click **Save** again to save the changes
- Start FactoryTalk ViewPoint client by right-clicking on the Internet Explorer icon in the Task Bar and then select **Internet Explorer** from the context menu. Point the browser to <http://server-base/FTVP>.



- Log in using the username **labuser** and password **rockwell**.
- Navigate to the **CIP** display and click on the button **CIP Process Steps**



**NOTE:** The

'CIP Process Steps' button will not be there if you haven't created it earlier in the lab in the section called 'Commands' on page 70. Please do that section now if you have not done it yet.

- Try to change the value in any of the four numeric input fields on a pop-up display. Notice how you cannot do that because the tag write permission for **CIP - Process Steps** display has only been granted to the members of **VP\_Operators** group.
- Log off by clicking **Log off | t** in the upper right corner.
- Log back in using the username '**Operator1**' and password '**Operator1**'.



16. Navigate to the **CIP** display and from there to **CIP Process Steps** again.
17. In the **Step1: Adding Water** numeric input field type in **6000** and press **Enter**



You've successfully changed the value. This is also confirmed in the Status Area at the bottom of the screen:



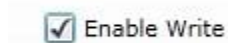
18. **Minimize** FactoryTalk ViewPoint client and **switch** to the FactoryTalk ViewPoint Administration browser instance by **clicking** on the Internet Explorer icon in the task bar and choosing the proper instance.



19. **Clear** Enable Application Security checkbox

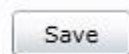


and then select Enable Write checkbox again



Note that we are disabling display level security and enabling application level security.

20. Click on the **Save** button to save the changes.



This will allow all users full read/write control of the entire application so that security logins are no longer required for the rest of this lab.

21. Similarly as you did in step 18, switch back to the FactoryTalk ViewPoint client browser by **clicking** on the Internet Explorer icon in the task bar and choosing the proper instance.
22. Press **F5** to refresh it so that the new security settings take effect.



Note that FactoryTalk ViewPoint does not use FactoryTalk View Security Codes (A to P permissions) and will ignore any security codes used in the published web application. For example, expressions that use the `CurrentUserHasCode()` function are not evaluated in FactoryTalk ViewPoint.

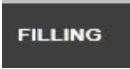
## Tag Write Functionality


FactoryTalk ViewPoint 2.0 has been enhanced to include the ability to write to tags. You have already experienced some of the tag write functionality in the previous section when you were asked to start the CIP process and enter a new value in the CIP process steps. In this section you will learn about this feature in more detail.

In addition to already supported Abort and Display navigation commands, FactoryTalk ViewPoint SE supports the following tag write commands:

- = (Equal)
- Ramp
- Set
- Toggle

In the following steps you will use various control objects that support tag write functionality.

1. On the navigation bar click on the **Filling** button 
2. In the Change Filling Product area click on any of the four buttons to change the current product. Behind each of these buttons there is an object with Touch animation that executes Set command. As soon as the associated tag is set, you will notice that current product has changed. Again, successful command execution is accompanied with the message:

 The call to Execute was successful

3. Click on Filler Control Selector Switch to stop and re-start the production line. The status area message will indicate yet another supported tag write command:

 Button2: Value was written successfully.

4. On the navigation bar click on the **Alarms** button 
5. In the **Simulate Motor Temperature alarm** area there are three buttons that set HIHI, HI and LO Limit digital alarm. **Click** on any of them. You will notice a new alarm messages in the alarm summary.
6. Click the Turn Off button  to turn off the alarm. The alarm message will disappear.

Feel free to explore further – navigate to other displays, click on control buttons and/or enter new numeric input values where possible. Observe the results of your actions and the messages logged in the Status area.

Congratulations! You have learned how to connect to the FactoryTalk ViewPoint Administration Site, configure FactoryTalk ViewPoint Administration Security, explore FactoryTalk ViewPoint Server Settings and publish a FactoryTalk View SE application. We also reviewed how to running an application in a browser, configuring FactoryTalk ViewPoint Application Security and manage Tag write functionality in FactoryTalk ViewPoint.

**Power, Control and Information Solutions Headquarters**

Publication XXXX###ENP - Month Year, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4620  
Supersedes Publication XXXX###ENP - Month Year, Park, De Kleetlaan 12a, 1831 Diegem, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640  
Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846