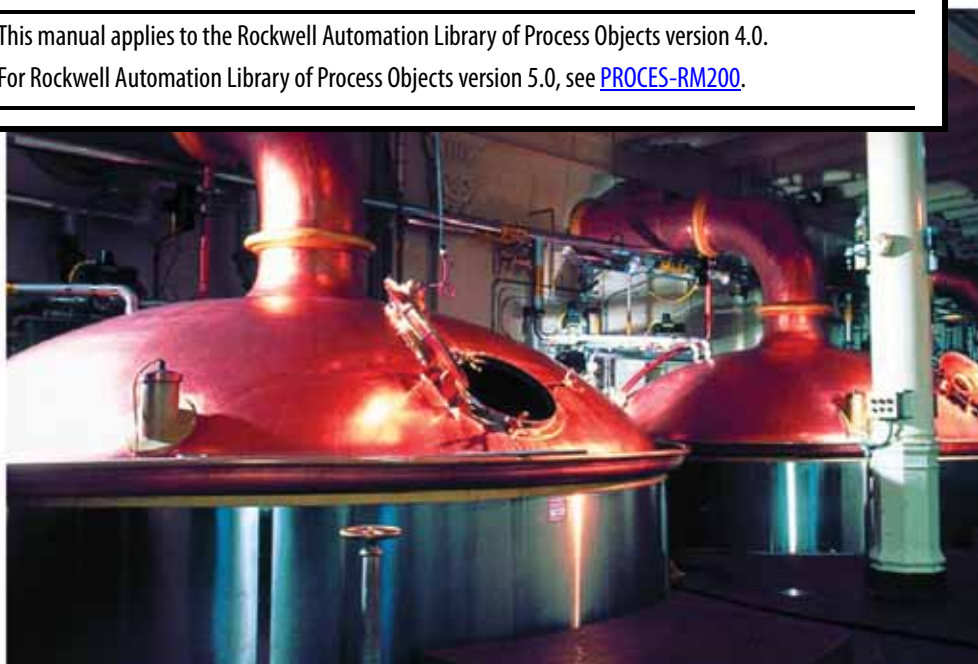
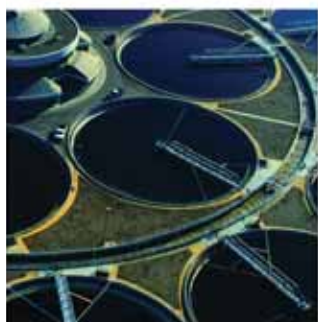


Rockwell Automation Library of Process Objects: Configuration and Usage

Version 4.0 or Later

IMPORTANT This manual applies to the Rockwell Automation Library of Process Objects version 4.0.
For Rockwell Automation Library of Process Objects version 5.0, see [PROCES-RM200](#).



Important User Information

Read this document and the documents listed in the additional resources section about installation, configuration, and operation of this equipment before you install, configure, operate, or maintain this product. Users are required to familiarize themselves with installation and wiring instructions in addition to requirements of all applicable codes, laws, and standards.

Activities including installation, adjustments, putting into service, use, assembly, disassembly, and maintenance are required to be carried out by suitably trained personnel in accordance with applicable code of practice.

If this equipment is used in a manner not specified by the manufacturer, the protection provided by the equipment may be impaired.

In no event will Rockwell Automation, Inc. be responsible or liable for indirect or consequential damages resulting from the use or application of this equipment.

The examples and diagrams in this manual are included solely for illustrative purposes. Because of the many variables and requirements associated with any particular installation, Rockwell Automation, Inc. cannot assume responsibility or liability for actual use based on the examples and diagrams.

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

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

Throughout this manual, when necessary, we use 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.



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, and recognize the consequence.

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

Labels may also be on or inside the equipment to provide specific precautions.



SHOCK HAZARD: Labels may be on or inside the equipment, for example, a drive or motor, to alert people that dangerous voltage may be present.



BURN HAZARD: Labels may be on or inside the equipment, for example, a drive or motor, to alert people that surfaces may reach dangerous temperatures.



ARC FLASH HAZARD: Labels may be on or inside the equipment, for example, a motor control center, to alert people to potential Arc Flash. Arc Flash will cause severe injury or death. Wear proper Personal Protective Equipment (PPE). Follow ALL Regulatory requirements for safe work practices and for Personal Protective Equipment (PPE).

Preface

New and Updated Information.....	7
Additional Resources	8

Chapter 1

How to Install the Library

Download the Library	9
Using HMI and Controller Templates.....	10
Import Logic into a Project	10
Import Visualization Files.....	13
Import Images.....	13
Import Global Object Files.....	15
Import HMI Faceplates.....	16
Import the Macros	17
Import HMI Tags	18
Configure Your HMI Application for Language Switching	23
Process Library Language Switching.....	23
Configure Language Switching	23
Configure FactoryTalk View SE Clients.....	24
Configure FactoryTalk View ME Runtime Device	24

Chapter 2

Configure an HMI Template and Security

Considerations.....	25
Download Template Files.....	26
Observe the Template Structure	27
Explore Template Features.....	28
Display and Alarm Hierarchy.....	29
Template Headers.....	32
Navigation Utilities.....	32
Alarm Annunciation.....	33
System Status	35
Macros	35
Customize an HMI Template.....	37
Modify the Template	37
Configure HMI Security.....	69
Configure Object Area.....	69
Configure Security Area.....	69
Create User Groups	76
Assign Security Codes to User Groups	78
Import Area-based HMI Tags	82
Create Area User Groups	85
Create FactoryTalk Users	90
Security Code Differences for FactoryTalk ME Groups.....	95
Create an OWS Client File	96

Common Configuration Considerations	Chapter 3
	Library Programming Considerations..... 99
	Ladder Diagram Considerations 101
	Prescan 101
	Command Source Configuration..... 103
	Alarm Considerations 103
	Alarm with FactoryTalk Alarms and Events Server 111
	Alarm with FactoryTalk View ME Software 118
	Global Object Configuration..... 125
	/X and /Y Positioning 125
	Maintain Library Releases..... 127
	Customize the Library..... 128
Use the Process Library	Chapter 4
	Create a Logic Instance 130
	Create an HMI Instance 133
	Device Configuration 139
	Modify Navigation and Area Tags 140
Online Configuration Tool	Appendix A
	Before You Begin..... 143
	Configure Parameters By Using a Spreadsheet 144
PlantPAx Alarm Builder Tool	Appendix B
	Before You Begin..... 147
	Build AE Alarms 148
	1. Select Controller 149
	2. Associate HMI..... 156
	3. Define HMI Alarm Server 162
	4. Organize Tags and HMI Displays (Optional)..... 163
	5. Build AE XML Import File 169
	6. Import XML File to AE Alarm Database..... 173
	Build ME Alarms 178
	1. Select Controller 178
	2. Associate HMI..... 178
	3. Define HMI Alarm Server 179
	4. Organize Tags (Optional)..... 179
	5. Build ME Alarms XML Import File 179
	6. Import XML File to ME Alarm Database 182

	Appendix C	
Color Change Utility	Install Tool File	186
	Use the Utility with Library Objects	187
	Modifying the Color Palette	190
	Use the Utility with Other FactoryTalk View Software Files	191
	Appendix D	
View Security Permission Tags	How Do I Access an Attachment?	193
	Index	
		195

Notes:

This manual describes how to build and use library components that comprise the Rockwell Automation® Library of Process Objects.

Procedures include how to customize HMI templates based on your system requirements. Step-by-step instructions guide you through how to apply and maintain HMI security depending on the responsibilities of plant personnel.

The appendix features associated productivity tools, such as the PlantPAx Configuration Tool for creating HMI alarms.

We recommend that you use this manual along with two additional references:

- [PROCES-RM013](#) — Describes the logic per Library object
- [PROCES-RM014](#) — Describes the display elements per Library object

**New and Updated
Information**

This table contains the changes that are made to this revision.

Topic	Page
Build HMI template and configure security	25
Process Library security tag descriptions	193

IMPORTANT	Process Library Security Information tags are attached to this PDF as a Microsoft Excel file. To access the Excel spreadsheet, click the paper clip icon in the left pane of the PDF. Procedures are documented on page 193 .
------------------	---

Additional Resources

These documents contain additional information concerning related products from Rockwell Automation.

Figure 1 - Additional Resources for Related Library Information

Resource	Description
Logix 5000™ Controllers Add-On Instructions Programming Manual, publication 1756-PM010	Provides information for how to define, configure, and program Add-On Instructions.
FactoryTalk® View SE Edition User Guide , publication VIEWSE-UM006	Provides details on how to use this software package to develop and run HMI applications that can involve multiple users and servers, which are distributed over a network.
FactoryTalk View Machine Edition User Guide, publication VIEWME-UM004	Provides details on how to use this software package for creating an automation application.
FactoryTalk Alarms and Events System Configuration Guide, publication FTAE-RM001	Provides details on how to install, configure, and use FactoryTalk Alarms and Events services as part of a FactoryTalk-enabled automation system.
PlantPAx® Distributed Control System Selection Guide, publication PROCES-SG001	Provides information to assist with equipment procurement for your PlantPAx system.
PlantPAx Distributed Control System Reference Manual, publication PROCES-RM001	Provides characterized recommendations for implementing your PlantPAx system.
PlantPAx Distributed Control System Infrastructure Configuration, publication PROCES-UM001	Provides screen facsimiles and step-by-step procedures to configure infrastructure components for your system requirements.
PlantPAx Distributed Control System Application Configuration, publication PROCES-UM003	Provides the steps necessary to start development of your PlantPAx Distributed Control System.
Rockwell Automation Library of Process Objects: Logic Instructions, publication PROCES-RM013	Provides descriptions of the Add-On Instructions in the Library and their parameters and local tags.
Rockwell Automation Library of Process Objects: HMI Instructions, publication PROCES-RM014	Provides descriptions of the HMI visualization files for all Add-On Instructions.

You can view or download publications at <http://www.rockwellautomation.com/literature/>. To order paper copies of technical documentation, contact your local Allen-Bradley® distributor or Rockwell Automation sales representative.

How to Install the Library

This chapter describes procedures for installing the HMI and controller elements that comprise the Rockwell Automation® Library of Process Objects. There are two methods:

- Use predefined application templates
- Import individual library objects

The following table lists the topics in this chapter.

Topic	Page
Download the Library	9
Using HMI and Controller Templates	10
Import Logic into a Project	10
Import Visualization Files	13
Configure Your HMI Application for Language Switching	23

Download the Library

For the latest compatible software information and to download the Rockwell Automation Library, see the Product Compatibility and Download Center at <http://www.rockwellautomation.com/rockwellautomation/support/pcdc.page>.

Using HMI and Controller Templates

Both controller and HMI templates are provided with the Rockwell Automation Library of Process Objects. These templates provide a starting point for building your PlantPAx® applications.

- The HMI templates are offered in various screen resolutions to align with the most commonly used monitors. The templates have a framework inclusive of a FactoryTalk® Alarms and Events alarm banner.

See [Chapter 2](#) for procedures on customizing an HMI template and configuring HMI security.

- The controller templates are provided in supported Logix firmware revisions. The templates provide a recommended task structure in preparation to import the Process Library Add-On Instructions and build the control strategies.

TIP The template 'all' controller applications include all the Add-On Instructions. The 'basic' application includes the most commonly used Add-On Instructions.

Import Logic into a Project

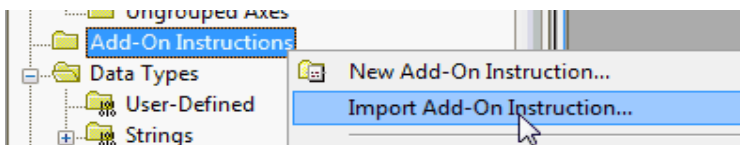
An Add-On Instruction is defined once in each controller project, and can be instantiated multiple times in your application code. To use pre-engineered logic, import each Add-On Instruction into a controller project.

Complete these steps.

1. In the Studio 5000 Logix Designer® application, open a new or existing project.

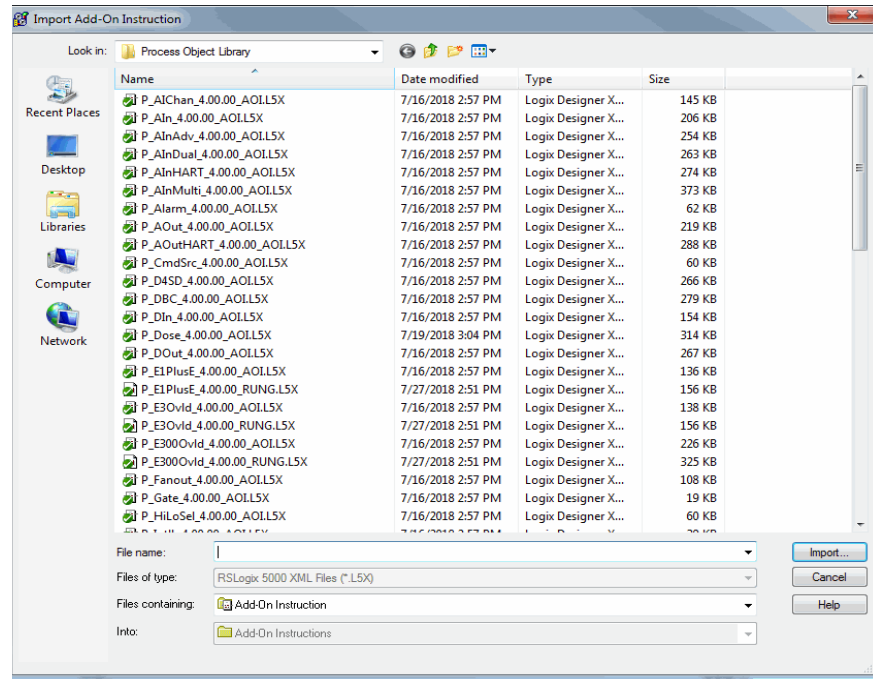
IMPORTANT Add-On Instruction definitions can be imported, but not updated, online.

2. Right-click the Add-On Instructions folder in the Controller Organizer and choose Import Add-On Instruction.



The Import Add-On Instruction dialog box appears.

3. Select the Add-On Instruction and click Import.

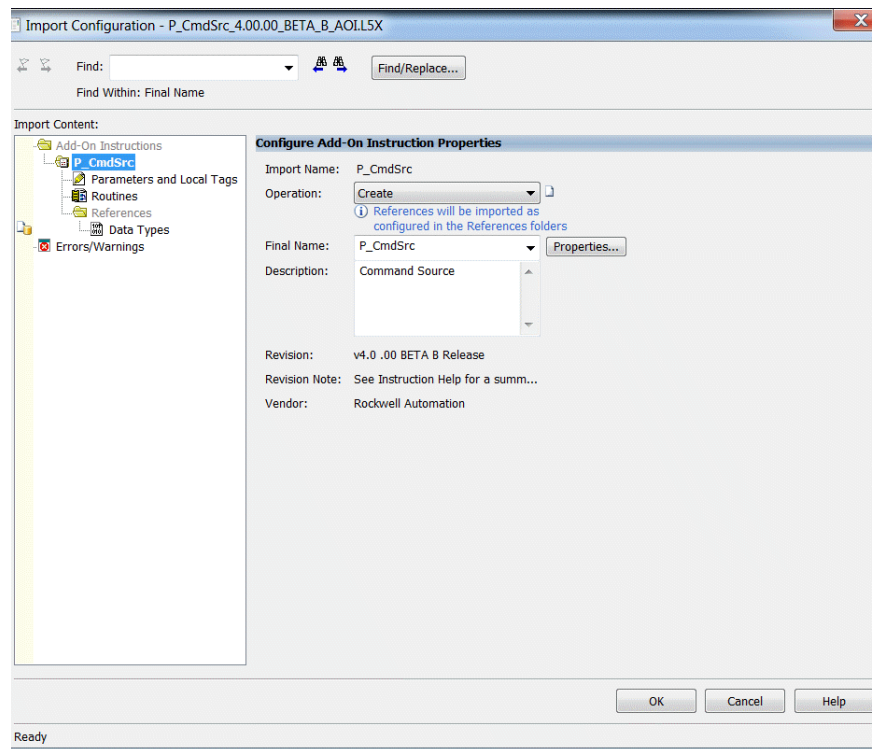


TIP The P_CmdSrc, P_Alarm, and P_Gate Add-On Instructions are used within many of the other instructions. We recommend that you import these three instructions first.

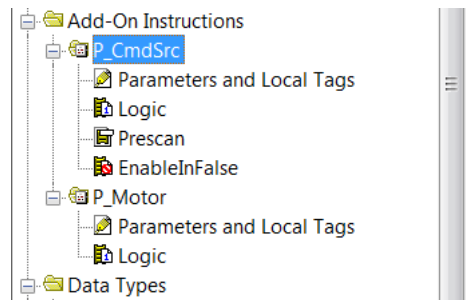
Some Add-On Instructions are provided in RUNG import files.

TIP If a RUNG import file is provided, import the rung into a ladder diagram routine to get all required additional tags, data types, and message configurations.

- On the Import Configuration dialog box, click OK.



- Once the import is complete, the Add-On Instructions are visible in the Controller Organizer.



IMPORTANT See [page 70](#) for procedures on how to change the size of the STRING-Area string data type after importing the **first** Add-On Instruction.

Import Visualization Files

Each Add-On Instruction has associated visualization files that provide a common user interface. You must import these files in the following order:

- Images (.png files)
- Global objects (.ggfx file type)
- HMI faceplates (.gfx file type)
- Tags (.csv file type)
- Macros (FactoryTalk View SE software only) (.mcr file type)

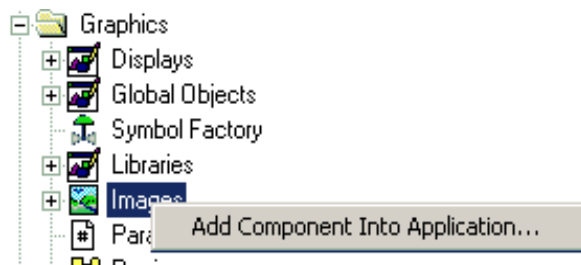
A global object is an HMI display element that is created once and referenced multiple times on multiple displays in an application. When changes are made to the original (base) object, the instantiated copies (reference objects) are automatically updated. Use of global objects, with tag structures in the ControlLogix® system, aid in consistency and save time engineering.

The import procedures in this section are to be followed in the sequence as documented to add the visualization files to your project.

Import Images

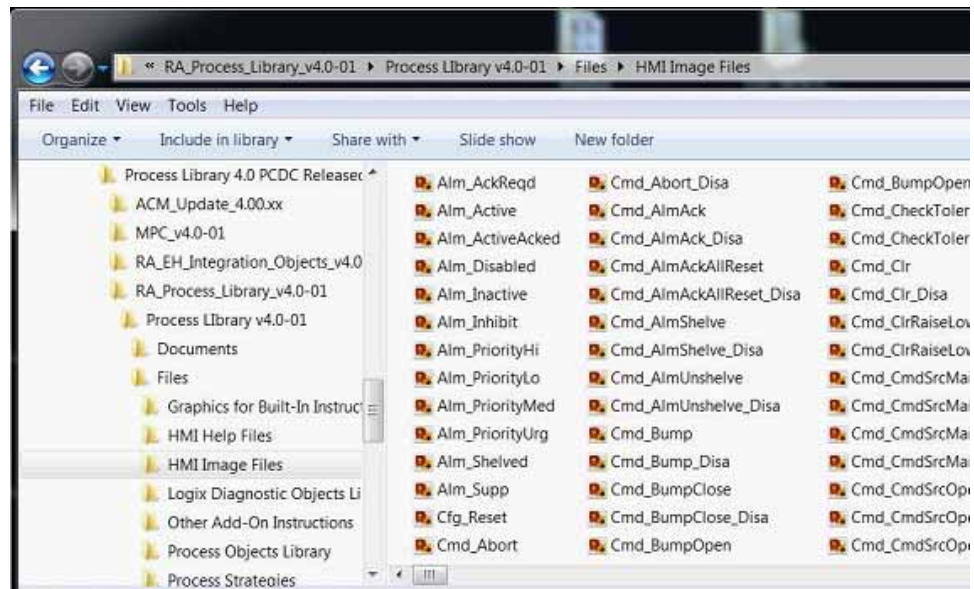
Complete these steps to use the common icons for the global objects and faceplates for all Process objects.

1. In your FactoryTalk View SE or ME software (depending on which one you are using), click the '+' to open the Graphics folder.



2. Right-click Images and choose Add Component Into Application.
The Add Component Into Project dialog box appears.
3. Browse to your downloaded Rockwell Automation library files.
4. Click the graphics folder.

5. Click the Images folder.



IMPORTANT Change the path to the image folder and the file type to PNG. PNG files provide more control with color transparency.

6. Click the pull-down menu (as circled) and select a file type.
For example, PNG Images (*.png)
7. To highlight all .png files, press Ctrl-A.
8. Click Open to import the images.

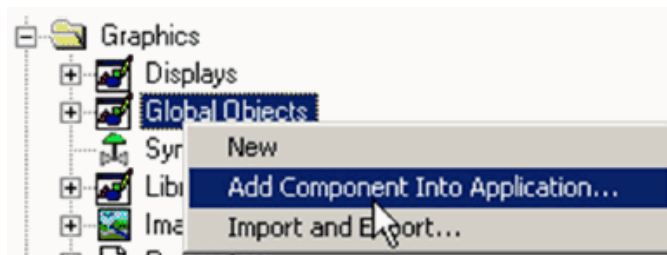
Import Global Object Files

Global objects serve two purposes:

- Faceplate objects files contain common elements that are used to build faceplate displays.
- Graphics Library files contain device symbols that you can use to build your application displays. Click the symbol to open the corresponding faceplate display.

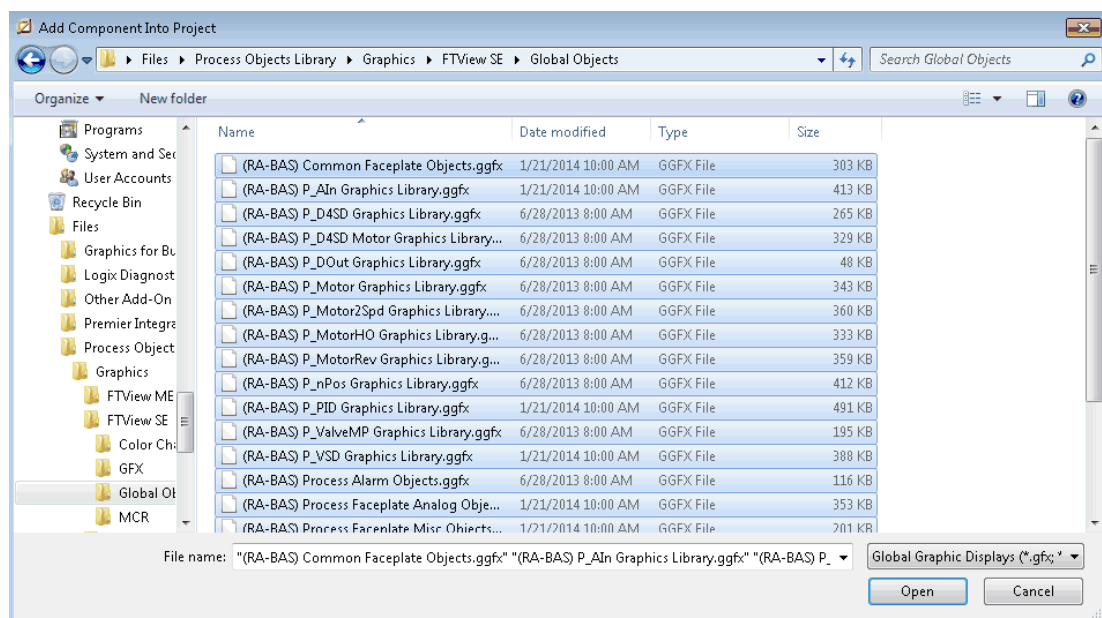
Complete these steps to import global objects.

1. Right-click Global Objects and choose Add Component Into Application.



The Add Component Into Project dialog box appears.

2. Browse to your downloaded Rockwell Automation Library files.
3. Click the Graphics folder.
4. Click the FactoryTalk View SE or FactoryTalk View ME folder depending on your application.
5. Click the Global Objects folder.

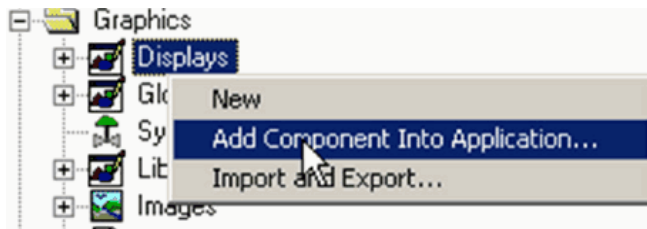


6. To highlight all global object (.ggfx) files, press Ctrl-A.
7. Click Open to import the objects.

Import HMI Faceplates

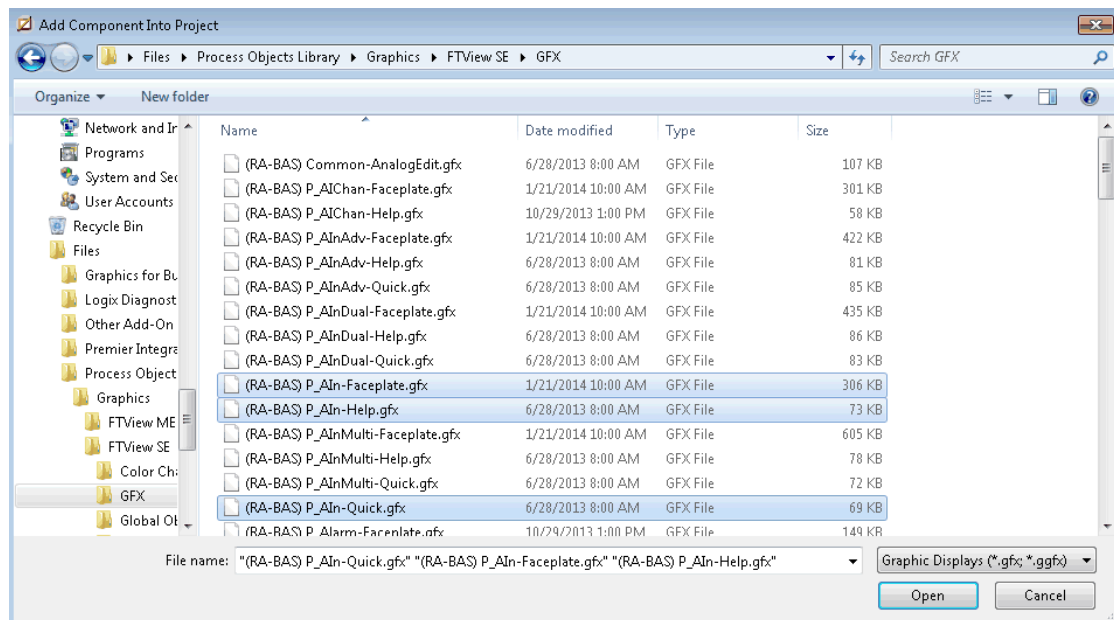
Faceplates provide operators, maintenance workers, engineers, and others with visual components to enable interaction with instrument data. Complete these steps to import faceplates.

1. Right-click Displays and choose Add Component Into Application.



The Add Component Into Project dialog box appears.

2. Browse to your downloaded Rockwell Automation library files.
3. Click the graphics folder.
4. Click the FactoryTalk View SE or FactoryTalk View ME folders depending on your application.
5. Click the GFX folder.



6. Click only the displays that you need; do not import all of them.

IMPORTANT To select multiple display files after the initial file selection, press and hold the Ctrl key while selecting additional files.

The highlighted example with P_AIn shows that each Add-On Instruction requires a Faceplate.gfx, Help.gfx, and Quick.gfx. Most motors, valves, and other devices need displays for Command Source Configuration and Help and Alarm Configuration and Help.

See the Rockwell Automation Library of Process Objects: HMI Instructions, publication [PROCES-RM014](#) for a list of required .gfx files.

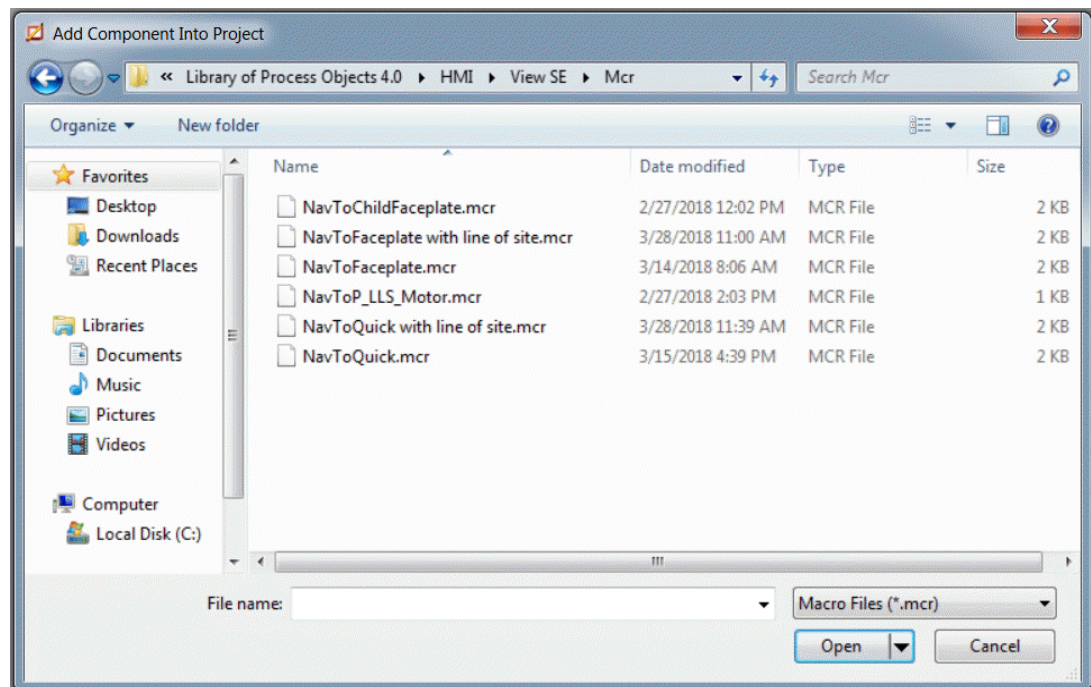
7. For FactoryTalk View SE applications, also import the Common Analog Edit display.

Import the Macros

These instructions are for FactoryTalk View SE projects only. Macros must be imported to support faceplate-to-faceplate navigation by tag name.

Complete these steps to import macros.

1. Right-click Macro and select Add Component Into Application.

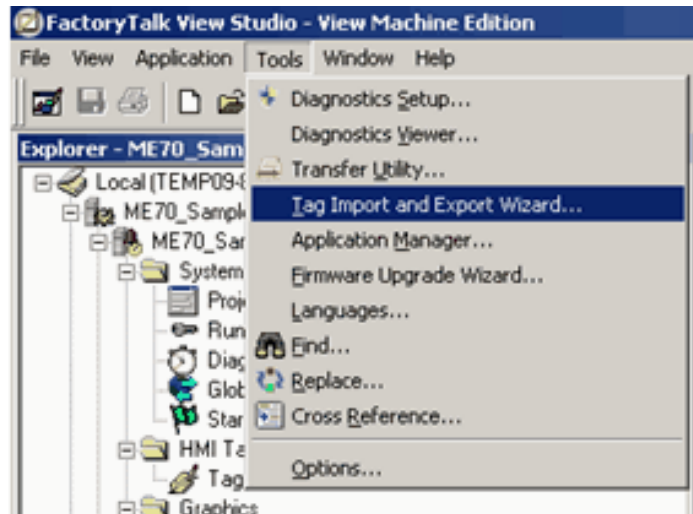


2. Press Ctrl-A to select all.
3. Click Open.

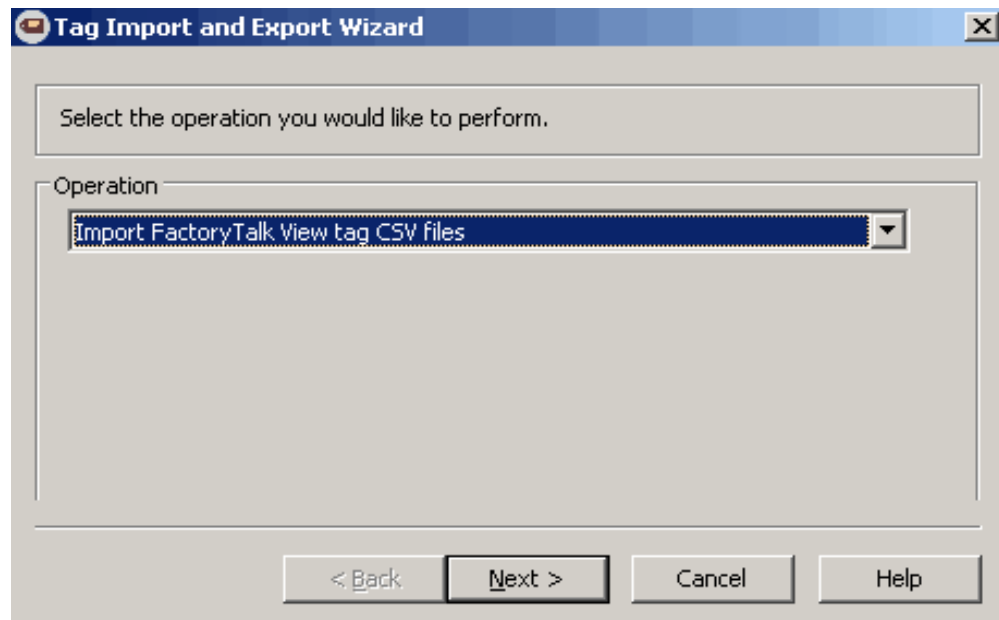
Import HMI Tags

Complete these steps to import HMI tags.

1. From the Tools pull-down menu, choose Tag Import and Export Wizard.

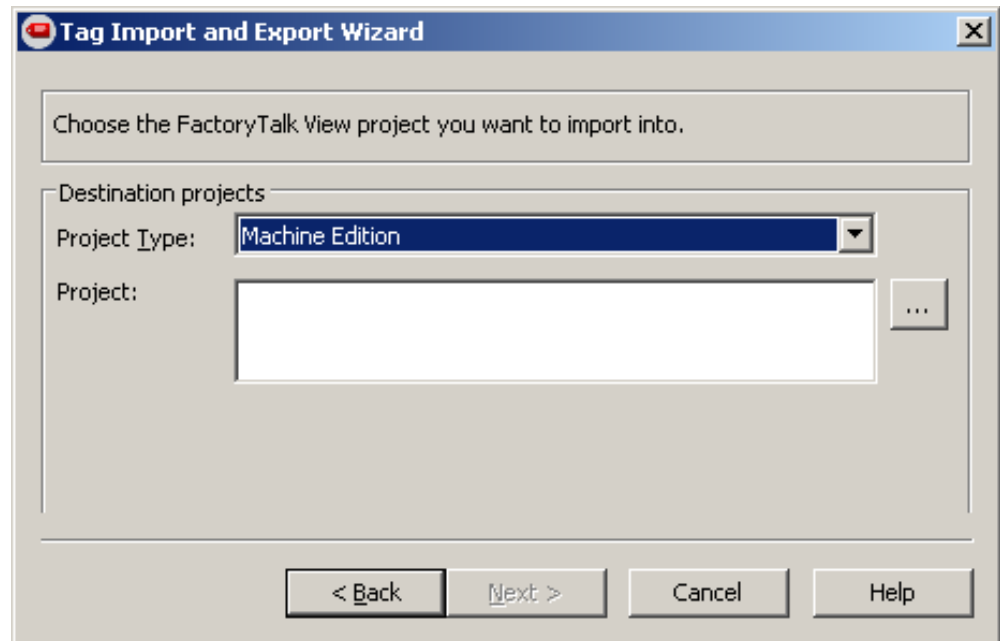


The Tag Import and Export Wizard dialog box appears.



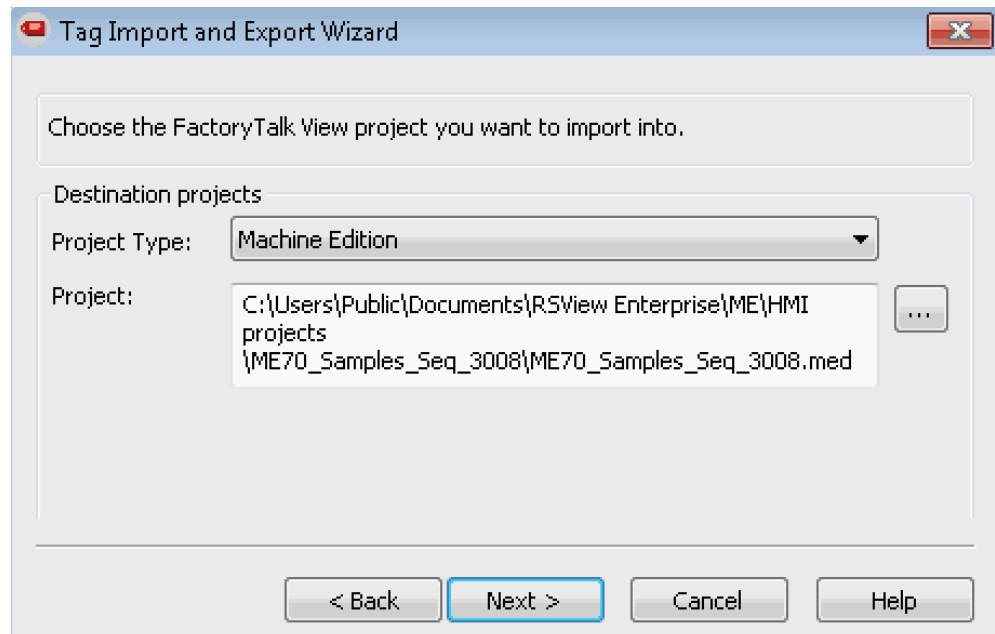
2. From the Operation pull-down menu, choose Import FactoryTalk View tag CSV files and click Next.

The Tag Import and Export Wizard dialog box reappears with a blank Project text box.



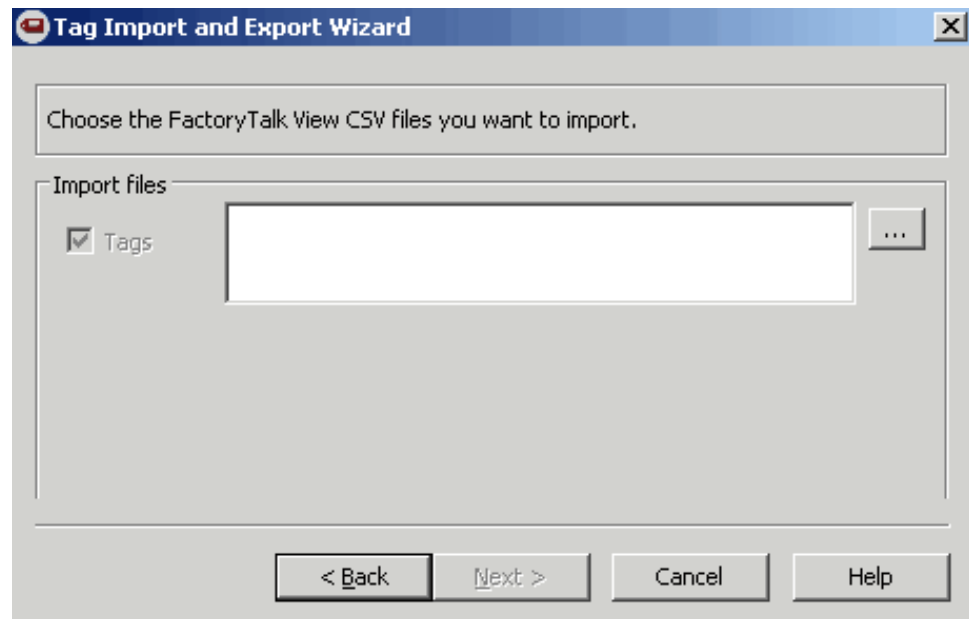
3. From the Project text box, click Browse (...) and select the .med project file that you want the HMI tags imported into and click Open.

The Tag Import and Export Wizard dialog box reappears with the .med file in the Project text box.



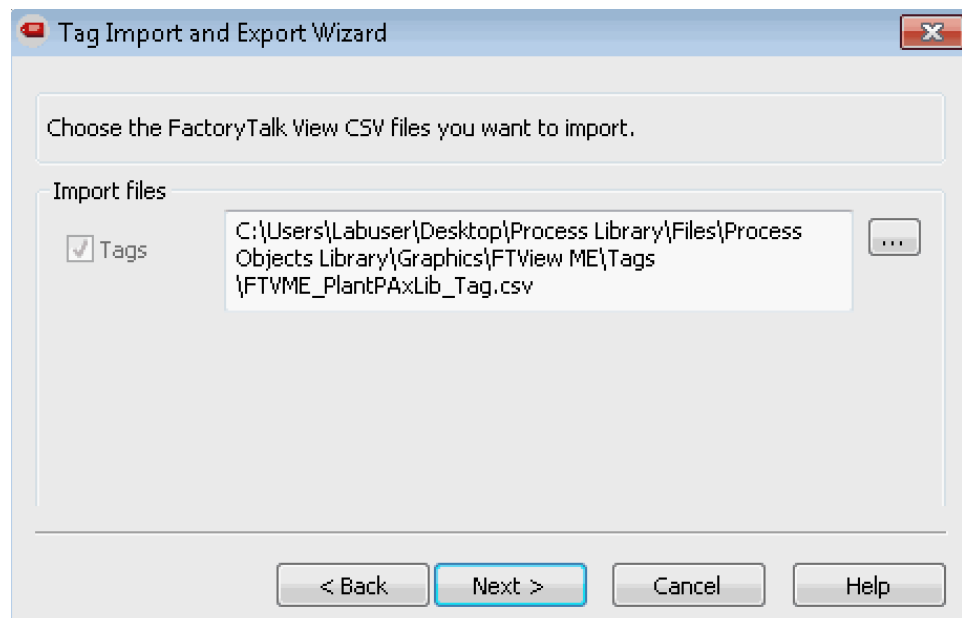
4. Click Next.

The Tag Import and Export Wizard dialog box reappears with a blank text box.



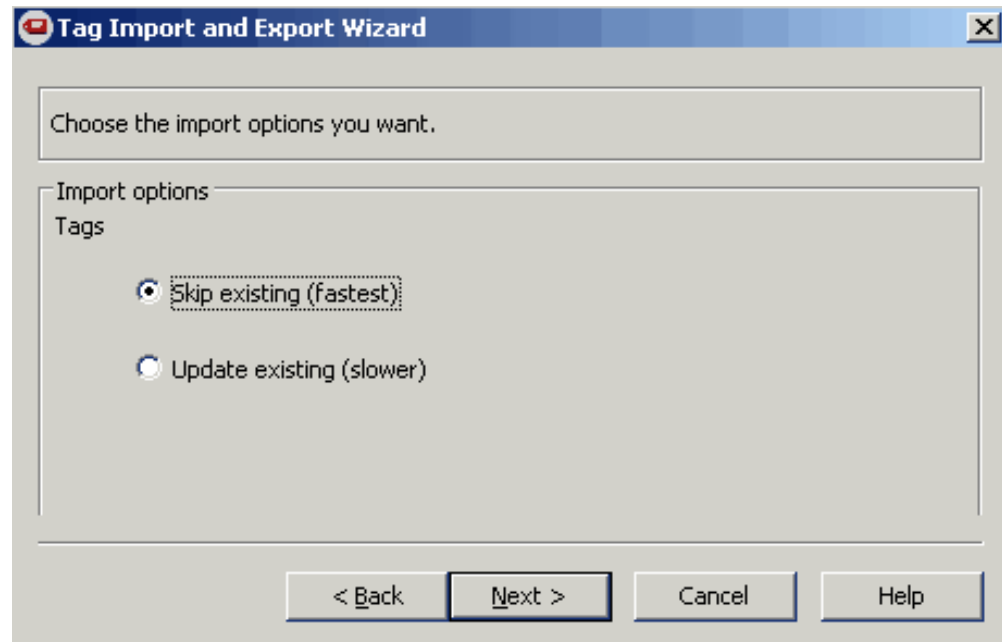
5. From the Import files text box, click Browse (...) and select the .csv file that is contained within the downloaded Library zipped file.
6. Click Open.

The Tag Import and Export Wizard dialog box reappears with the selected .CSV file.



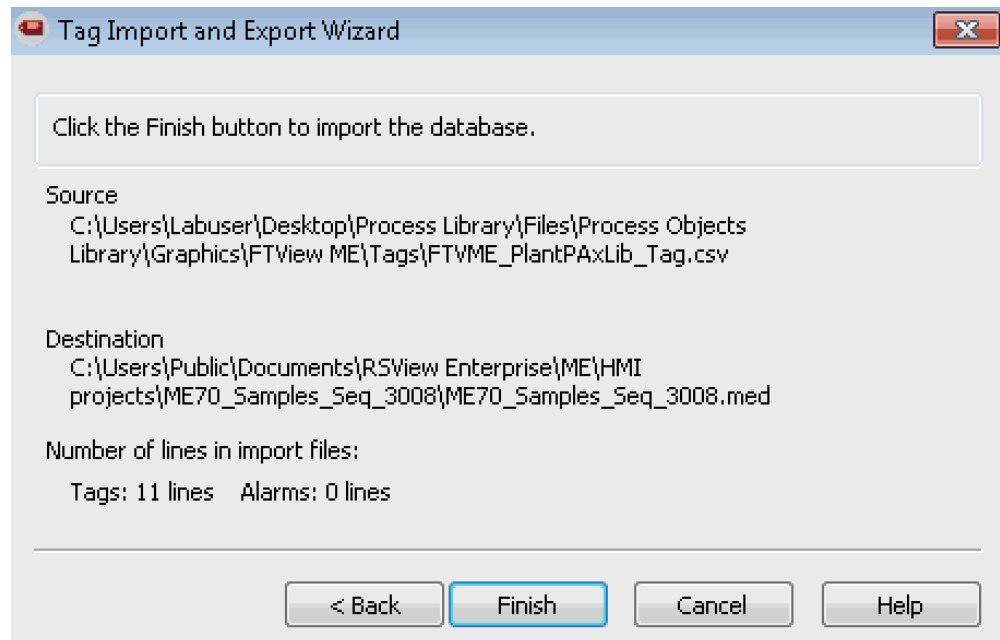
7. Click Next.

The Tag Import and Export Wizard dialog box reappears.



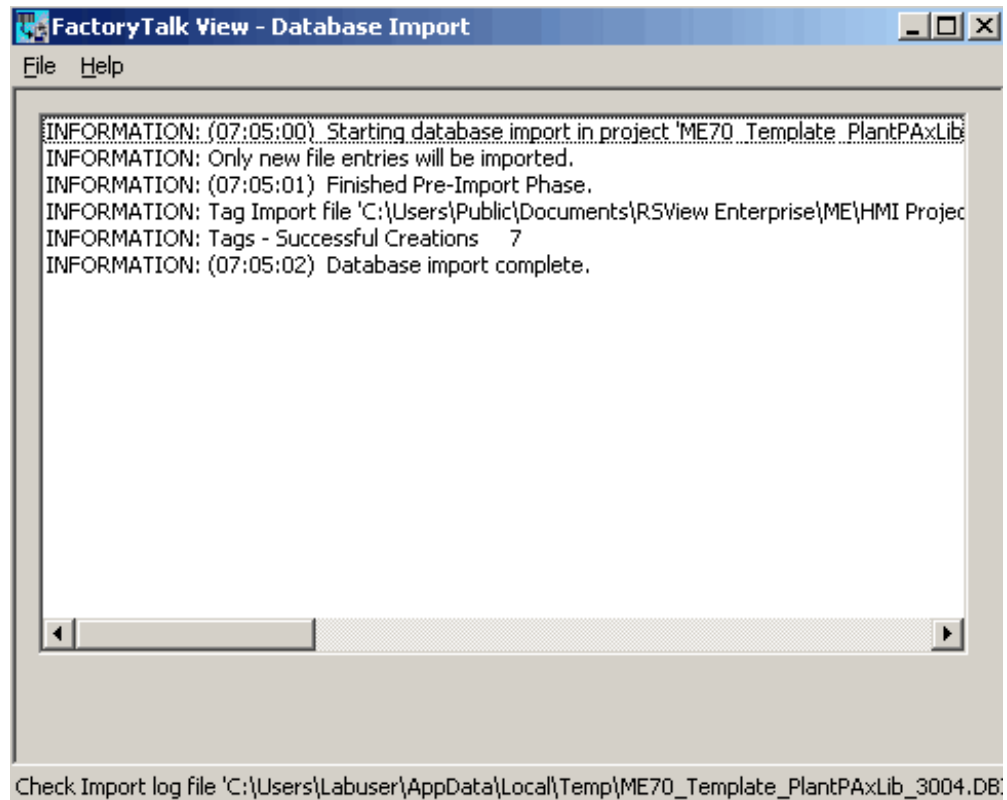
8. Use the Skip existing (fastest) option and click Next.

The Tag Import and Export Wizard dialog box reappears.



9. Click Finish to import the HMI files.

The FactoryTalk View - Database Import dialog box appears with the information that the import is complete.



10. To close the window and complete the import, click the 'X' in the upper right corner of the window.

Configure Your HMI Application for Language Switching

Process Library Language Switching

FactoryTalk View language switching provides the ability to configure multiple languages for an application and switch them dynamically at runtime.

The static strings that are used in the Rockwell Automation Library of Process Objects (Process Library) graphic elements have been designed to use the native FactoryTalk View Language Switching feature. This feature gives operators the ability to switch between languages in one application at runtime.

FactoryTalk View SE/ME Clients can run in any of the languages that an application supports. In a network distributed application, multiple clients can run in different languages simultaneously.

Translated versions of the Process Library static strings (distributed with the Process Library in the /Files/Translations folder) can be imported via the FactoryTalk View Studio Language Configuration dialog box. See [Configure Language Switching on page 23](#) for a summary of the steps necessary to incorporate language switching in your applications.

For more information and details about language switching, see the following FactoryTalk View publications:

- FactoryTalk View Site Edition User Guide, publication [VIEWSE-UM006](#)
- FactoryTalk View Machine Edition User Guide, publication [VIEWME-UM004](#)

Configure Language Switching

Complete these steps to configure language switching for an application:

1. Install the Windows languages that the application is going to use.

IMPORTANT Windows locale formatting determines how the application shows time, date, and floating point values at design time and runtime. When the application language is switched, the Windows locale settings for the new language are used even if that language has not been installed. You do not need to edit the default locale settings.

2. Create, open, or import the application in the language of your choice.
3. Add additional languages to your application.

IMPORTANT Static strings with undefined translations are shown with question mark '?' characters at runtime. Set en-US as your default English application language and choose the option to 'display undefined strings in the default language.'

4. Import Process Library graphic elements.
5. Import Process Library translated text strings for each of the desired languages.
6. To provide operators with a mechanism to switch between the application languages at runtime, use the Process Library **Language Switch** Global Objects in your application.

The **Language Switch** global objects are on the (RA-FRAME) Standard Objects.ggfx display for FactoryTalk View SE and (RA-FRAME-ME) Standard Objects.ggfx display for FactoryTalk View ME software.

Configure FactoryTalk View SE Clients

Complete these steps to configure FactoryTalk View SE clients:

1. On client computers, install the Windows languages that the application supports.
2. In client setup files, specify an initial language for the client to run.

Configure FactoryTalk View ME Runtime Device

Complete these steps to configure FactoryTalk ME runtime device:

1. For applications that are going to run on a personal computer, install the Windows languages that the application is going to use.
2. For applications that are going to run on a PanelView™ Plus or PanelView Plus CE terminal, configure the fonts that the application is going to use.

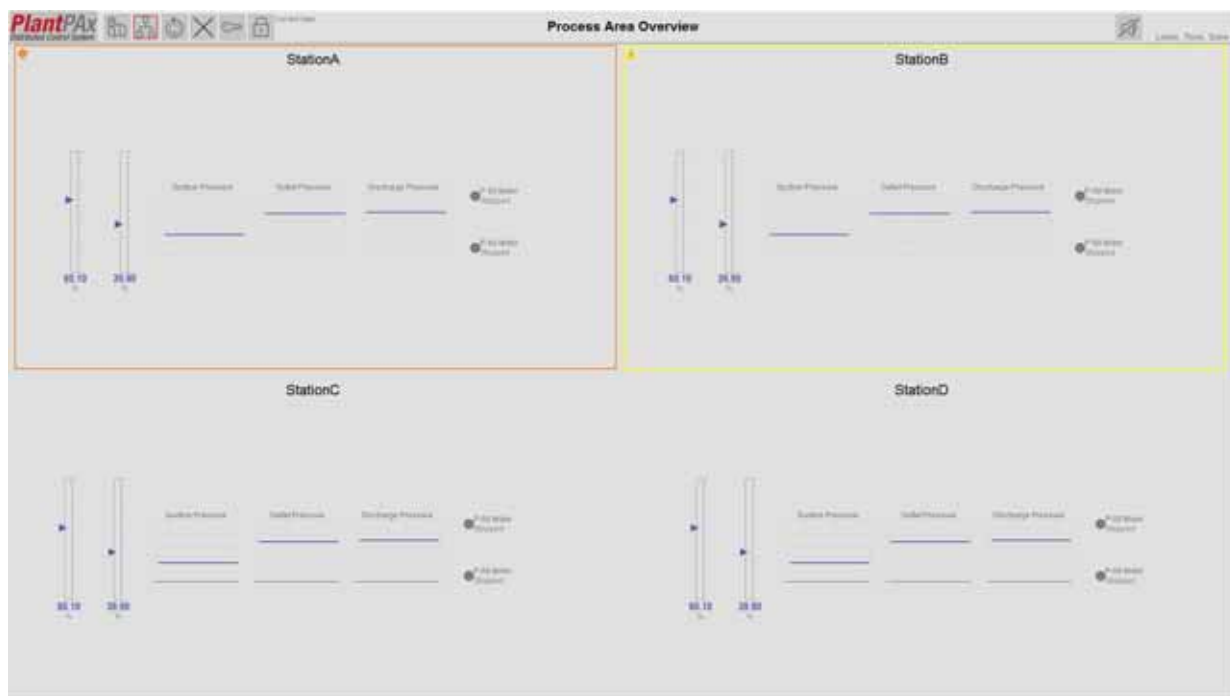
Configure an HMI Template and Security

A flexible, configurable template is available to provide a starting point for creating and developing an HMI application. The template supports Full HD displays and multi-monitors, which align with PlantPax® system best practices.

This chapter has two important subsections:

- Customize a template, starting on [page 37](#)
- Configuring HMI security, [page 69](#)

We encourage you to review the template features before implementing displays and alarms on a single or four-monitor workstation (as shown).



Considerations

Before you complete tasks in this chapter, we suggest that you do the following:

- Determine the number of HMI, data, and alarm and event servers to be used where they are hosted
- Make sure that PASS servers are installed and available
- Review system server configurations as described in the PlantPax Distributed Control System Infrastructure Configuration User Manual, publication [PROCES-UM001](#)

The following table lists the topics in this chapter.

Topic	Page
Download Template Files	26
Explore Template Features	28
Customize an HMI Template	37
Configure HMI Security	69
Create an OWS Client File	96

Download Template Files

The Process Library zip file contains a template folder that can be downloaded from the Product Compatibility and Download Center (PCDC). Click <http://www.rockwellautomation.com/rockwellautomation/support/downloads.page> to access the PCDC.

Double-click the Files folder and choose the Template folder. There are FTView (HMI) Templates and Logix (controller) Templates.

The FTView folder includes these HMI templates:

- **FTVSE_xx⁽¹⁾_x_Template_ProcessLib_x_x_xx.apa** – Provides faceplates, global objects, images, and macros for the current version of the Process Library FactoryTalk® View SE
- **FTVSE_xx⁽¹⁾_x_P2fQuickStart_x_x_xx.apa** – Sample ‘P2fTemplate’ includes framework displays for headers, button bars, alarming, and overview displays for single monitor (1 Mon) and four monitors (4 Mon).

IMPORTANT For most systems, you load the HMI template on the PASS server by using the Engineering workstation (EWS). Your system data is viewed on the Operator workstation (OWS).
Template resolution is 1920 x 1080, Full HD.

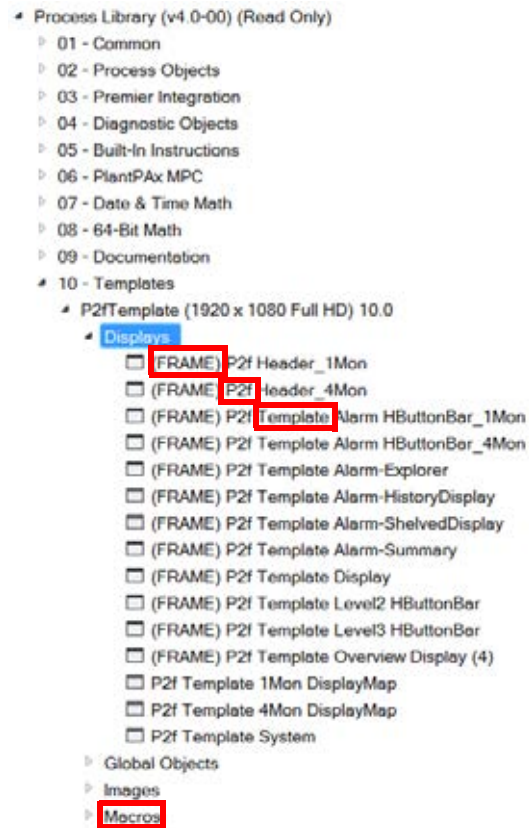
(1) ‘x’ refers to the latest version of Microsoft Windows and software version of FactoryTalk View.

Observe the Template Structure

The example tree diagram shows the HMI template structure.

The displays that are included in the template have the following naming structure:

- **(FRAME)** is a prefix that is used on items that are intended only to be used as is with minimum customization.
- **P2f** is a unique template identifier. This prefix identifies the template that is being applied and the files that are part of the template.
- **Template** is used to identify a template that can be copied multiple times to create application-specific content. If the word 'Template' is omitted, the file is not a template display. The rest of the name is a brief description of the display file. We suggest that these names be similar across templates.
- **Macros** configure the initial display structure during runtime. The Startup macros open the framework files in the orientation intended.



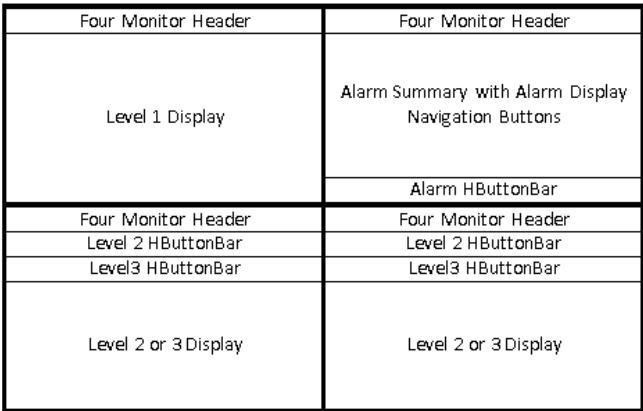
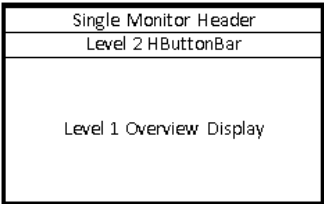
Explore Template Features

This section describes features of the HMI template. To access the template, see [Download Template Files on page 26](#).

We suggest you familiarize yourself with these features to enhance productivity with your template, which includes the following:

- Basic display layout
- Navigation
- Alarm annunciation
- FactoryTalk View Alarm and Event displays

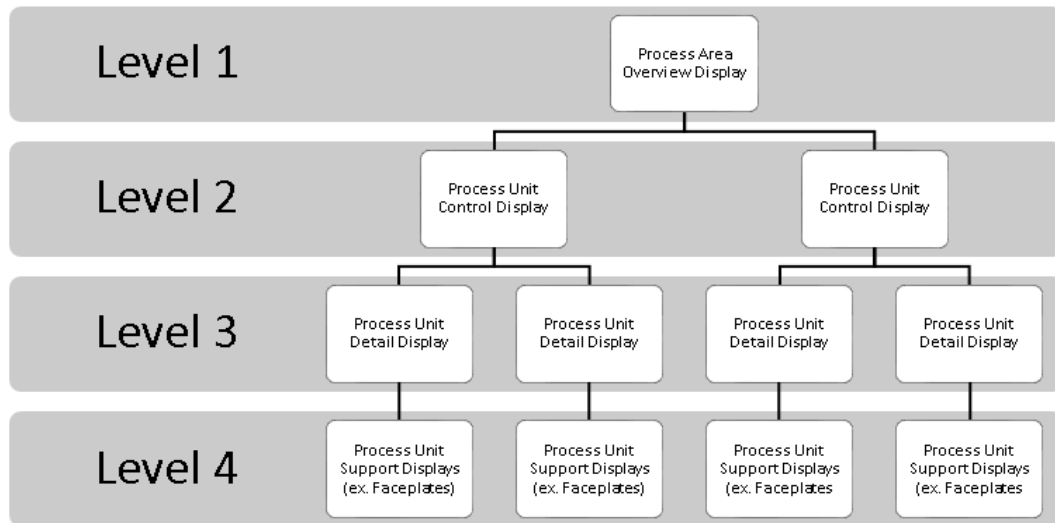
The template supports both single- and four-monitor configurations, as shown in the illustrations.



Display and Alarm Hierarchy

The HMI template requires that displays and alarms be organized in a specific hierarchy. There are four levels that are used in the display and alarm hierarchy, each level providing more detail than the previous level.

Figure 2 - Example of HMI Template Hierarchy



Level 1 displays are high-level overviews. These displays provide an overview that can be assimilated quickly, provides clear indication of current performance, and immediately highlights anything that needs the attention of a viewer.

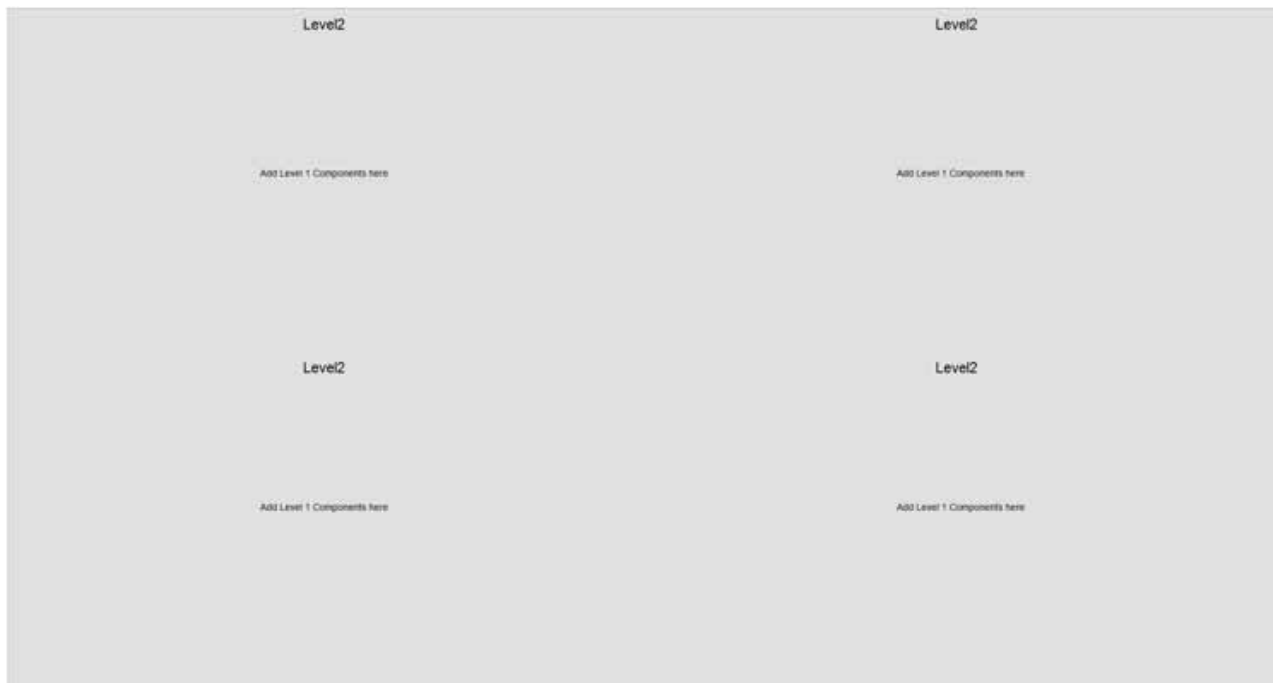
Level 2 displays are the main displays for users to perform their tasks. They contain information and control required to perform most user tasks. We recommend that you create these displays first.

Level 3 displays contain more detail and controls. These displays show details of subunits, individual equipment items, components, and related controls and indications. The displays are used for detailed investigations and interventions, and for troubleshooting or manipulating items not accessible from Level 2 displays.

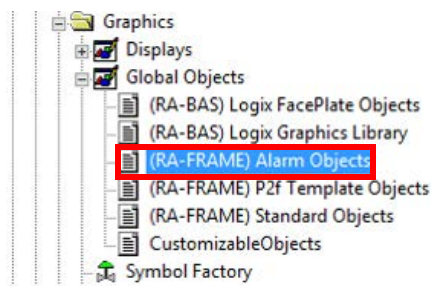
Level 4 displays provide the most detail of subsystems, individual sensors, or components. A faceplate is a type of level 4 display.

Level 1 Overview Display

The Level 1 Overview Display provides clear indication of current performance at a high level, and highlights anything that needs your attention. Control is not to be performed from this display.



Global objects have been developed that can be used on the Overview Display to annunciate alarms. The global objects are on the (RA-FRAME) Alarm Objects global object display.



Level 2 and 3 Displays

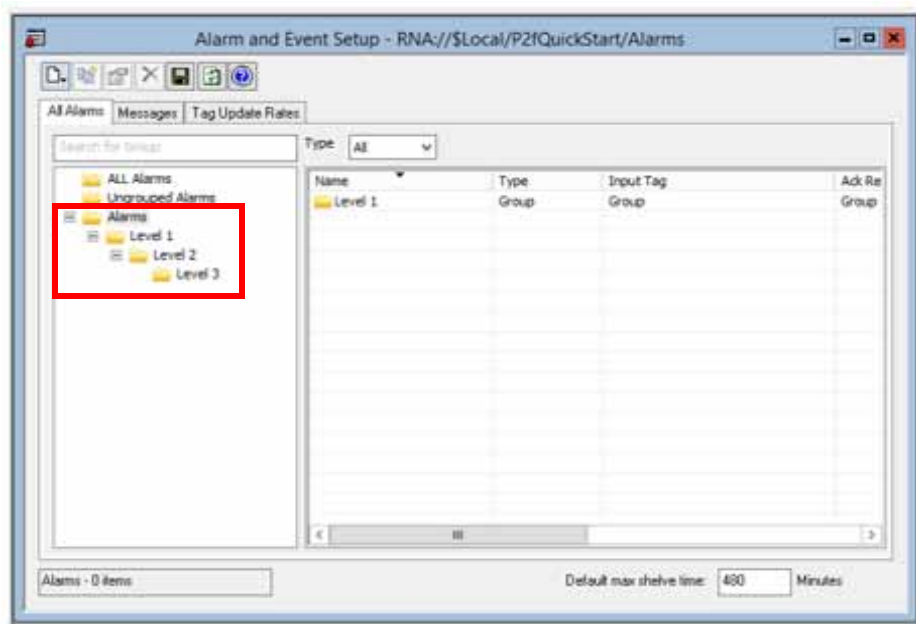
The template display can be used to configure level 2 and 3 displays. Level 2 displays are the main displays to perform tasks. Level 3 displays are used when additional detail or controls are required.

On four-monitor clients, monitors 3 and 4 display the level 2 or 3 displays. Button bars are used to navigate through the level 2 and 3 displays.



Alarm Group Hierarchy

Alarm groups can be configured following the same hierarchy as displays.



Template Headers

Headers contain functionality that provides access to information. There are two headers: one for a single monitor client, the other a four-monitor client.

Single Monitor Header

This header includes navigation display, system status, and alarm access.



Four-monitor Header

The four-monitor header includes the same buttons as the single monitor header, except for the alarm access button. The multi-monitor header also excludes an alarm and event banner. In the four-monitor configuration, the alarm summary is continuously displayed on monitor 2.

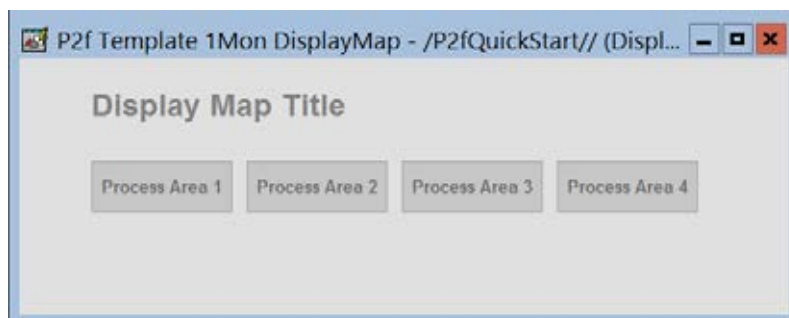


Navigation Utilities

Navigate your plant areas by using the following tools.

Map

The Display Navigation Map lets you switch between multiple process areas. A display map is required for each monitor configuration.



Navigation Bars

Level 2 HButton Bar is used for navigating through level 2 displays. For example, Units or Process Areas.

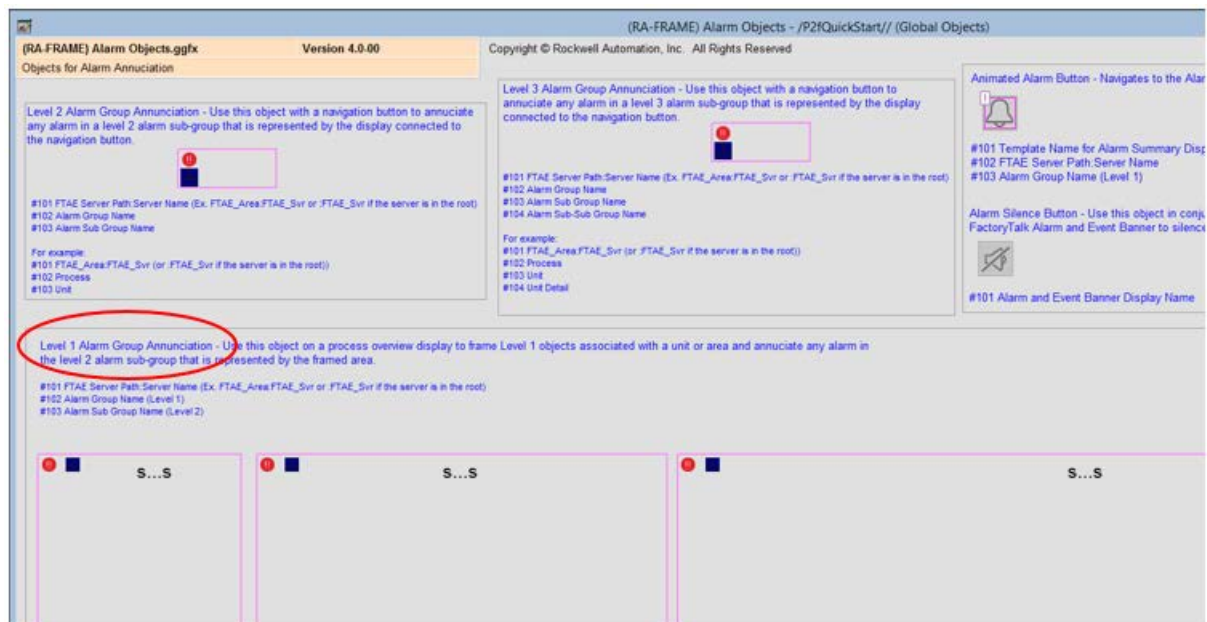


Level 3 HButton Bar is used for navigating through level 3 displays. For example, Unit Detail or Process Area Detail.



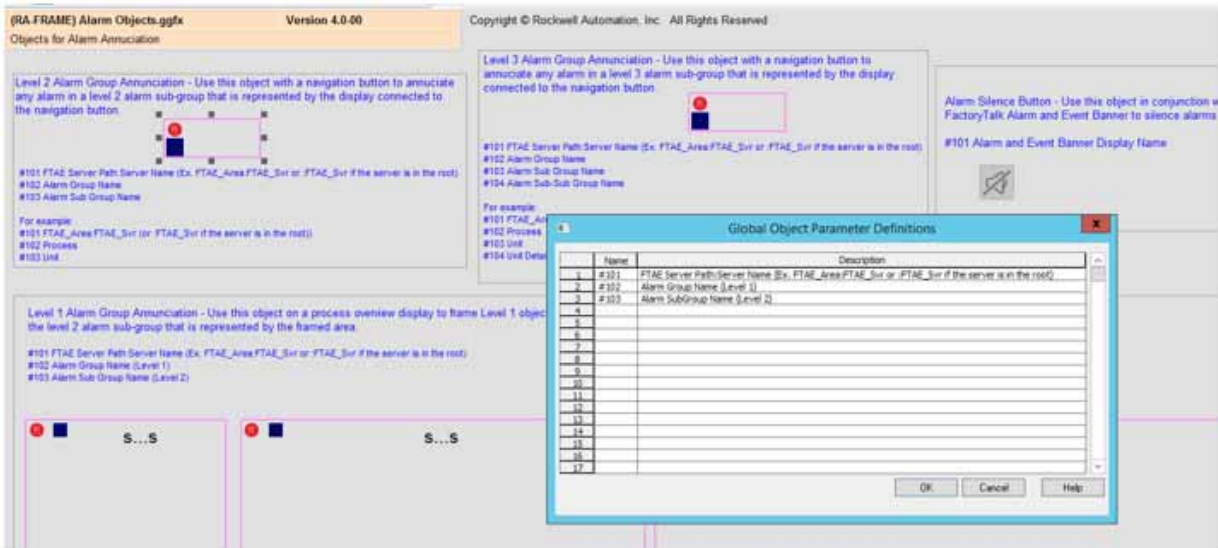
Alarm Annunciation

Level 1 Alarm Group Annunciation objects can be used to 'frame' the display elements that are associated with Level 2 groups. For example, Process Areas or Units. These objects are available in various sizes.



Level 2 and 3 HButtonBars use global objects on top of the navigation buttons to annunciate an alarm that is associated with the Level 2 or 3 alarm group. These global objects are on the (RA-FRAME) Alarm Objects global object display.

The alarm server path and alarm group names must be configured in the global object parameters.

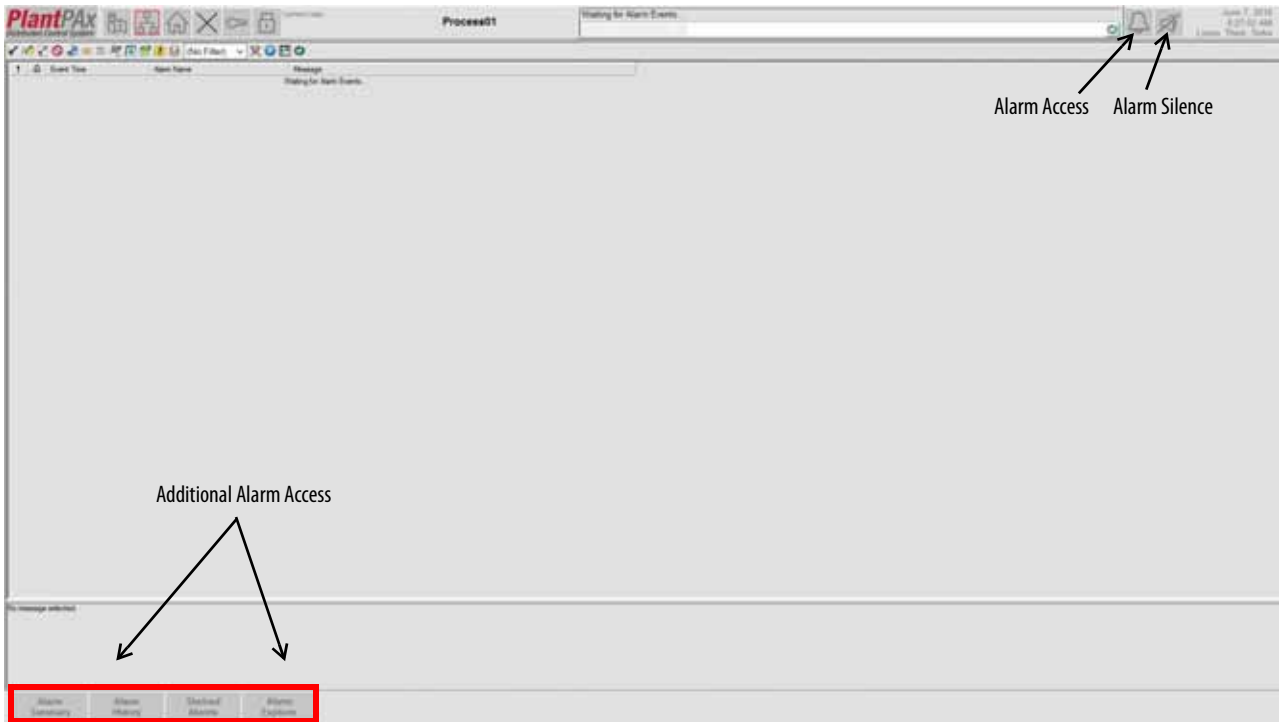


See [page 135](#) for more information on parameter values.

Alarm Tools

Click the bell icon (on the single monitor header) to access active alarms. Tabs appear near the bottom-left of the window to access several alarm screens. Click the speaker icon to silence the alarms.

TIP: On the four-monitor client, the alarm summary and alarm navigation are continuously displayed on monitor 2.



System Status

You can view the health of your system by displaying diagnostics.



Macros

In a FactoryTalk View Site Edition (SE) application, a macro is a series of commands stored in a text file. In a FactoryTalk View Machine Edition (ME) application, a macro is a list of tag assignments stored in a text file.

Table 1 - Process Library and HMI Macros Descriptions

Macro	Description
Process Library (FactoryTalk View SE)	
NavToChildFaceplate.mcr	Macro navigates to the faceplate for the object specified by the given path and tag names. For this macro, the security uses the parent's faceplate area, and that area is passed as the No. 4 parameter.
NavToFaceplate with line of sight.mcr	Macro navigates to the faceplate for the object specified by the given path and tag names.
NavToFaceplate.mcr	Macro navigates to the faceplate for the object specified by the given path and tag names.
NavToP_LLS_Motor.mcr	Macro navigates to the faceplate for the motor specified by the given path and tag names.
NavToQuick with line of sight.mcr	Macro navigates to the Quick display for the object specified by the given path and tag names.
NavToQuick.mcr	Macro navigates to the Quick display for the object specified by the given path and tag names.
Process Library (FactoryTalk View ME)	
FTViewME_ProcessLibrary_Tags_x_x_xx.CSV where x_x_xx = the current software version and service release number	These tags must be imported into the FactoryTalk View ME project to support switching tabs on any Process Object faceplate.
HMI Template	
P2fTemplate 1Mon FixScreen	Provides displays on a single, Full HD monitor.
P2fTemplate 4Mon FixScreen	Provides displays on four, Full HD monitors.

Importing Security-based Macros

For area-based security, you must import the NavToFaceplate and NavToQuick macros into your application. The NavToFaceplate passes parameters to display faceplates. The NavToQuick access the faceplate for a specified object.

Customize an HMI Template

IMPORTANT The 'NavToFaceplate with line of sight' and 'NavToQuick with line of sight' macros add Line of Sight Security to the application. This functionality allows equipment operation within a straight path of computers configured for this type of activity.

After you import the macros into your distributed system, add the desired computer to the specified area to a group. See [page 87](#) for procedures.

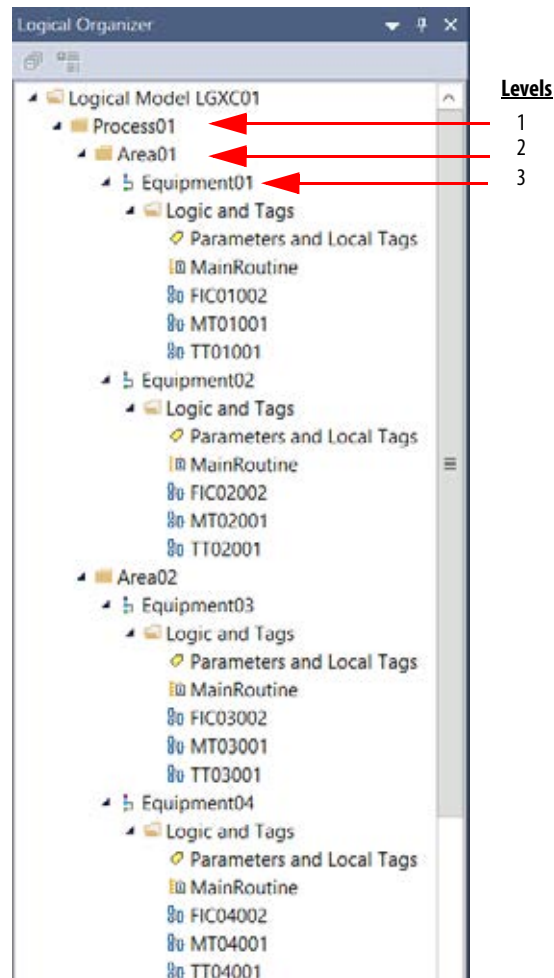
Rename 'NavToFaceplate with line of sight' to 'NavToFaceplate'. The renamed 'NavToFaceplate' (with line of sight) uses the command 'If CurrentComputerHasGroup() Then' to check the location of the login.

Customize an HMI Template

This section details how to customize a template in FactoryTalk View Studio SE software. [Figure 3](#) shows a Studio 5000 Logic Designer® application that shows a level hierarchy. Setting the level hierarchy is critical to alarm setup.

The example shows that the application consists of one process area. Within the process area are two sub areas, which each contains two pieces of equipment.

Figure 3 - Example Application Logical Model

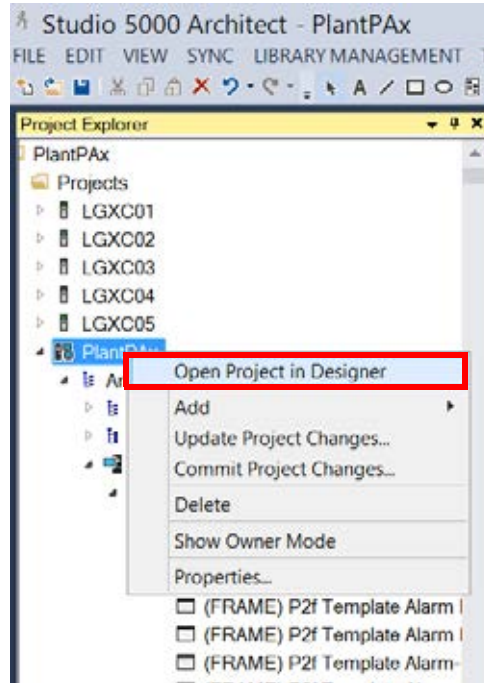


Modify the Template

This section describes how to modify the HMI template by using the Studio 5000 Logix Designer application.

1. Open an Architect project.

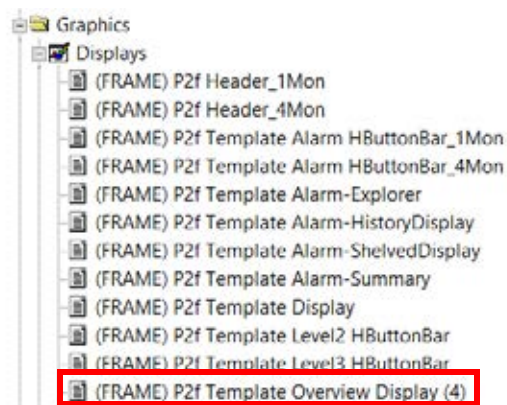
2. In the Project Explorer tree, right-click an application (PlantPAx in the example) and choose Open Project in Designer.



Wait a short time while the FactoryTalk View Studio software opens.

Notice that the project structure and content in FactoryTalk View Studio appears as it does in an Architect project.

3. Right-click '(FRAME) P2f Template Overview Display (4)' and choose Duplicate.



A window appears on the Studio canvas to choose a name.

4. Rename the display 'Process01 Overview Display' and click OK.

The name is for training purposes for this specific application. Your application name is going to be different.

5. Open the Process01 Overview Display and delete the bottom two alarm objects.

6. Right-click the first alarm object and select Global Object Parameter Values.
7. Type the values as shown in the example, and click OK.

	Name	Value	Tag	Description
1	#101	Area/Alarm:PlantPAX_AE	...	FTAE Server Path:Server Name (Ex. FTAE_Area:FTAE_Svr or :FTAE_Svr if the server is in the root)
2	#102	Process01	...	Alarm Group Name (Level 1)
3	#103	Area01	...	Alarm Sub-Group Name (Level 2)

OK Cancel Help

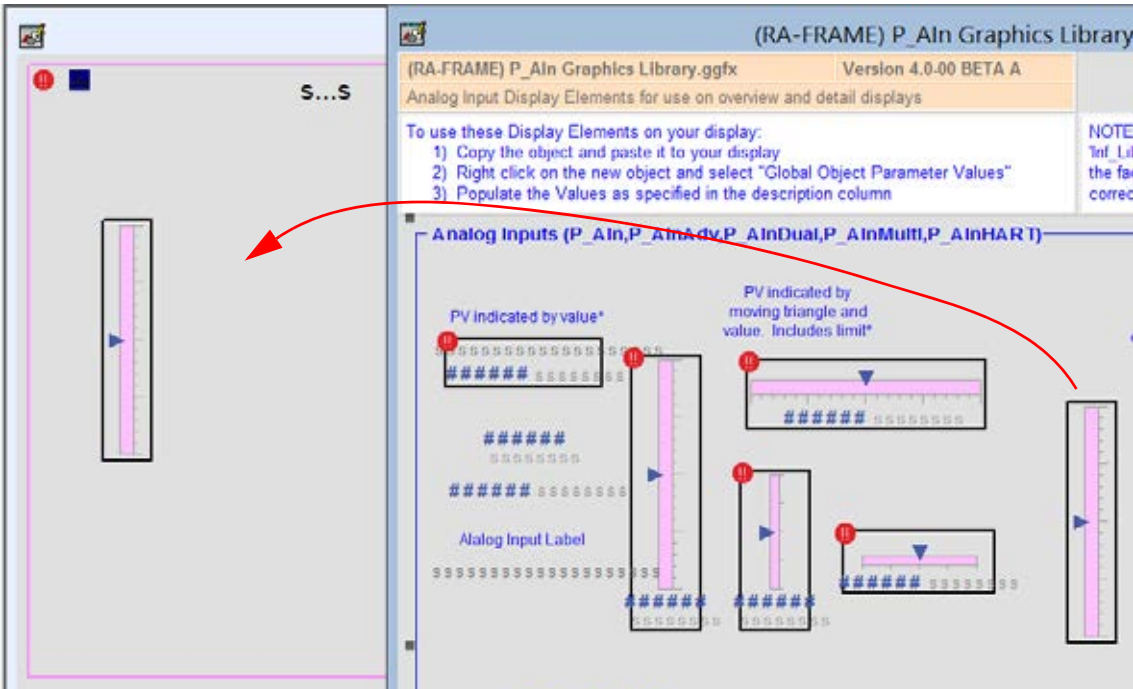
8. Repeat [step 6](#) and [step 7](#) for the second global object on the display.
Set parameter #103 to Area02 for this alarm object.
The alarm objects for Process01 Overview Display are configured for Area01 and Area02.

Adding Key Performance Indicators (KPI)

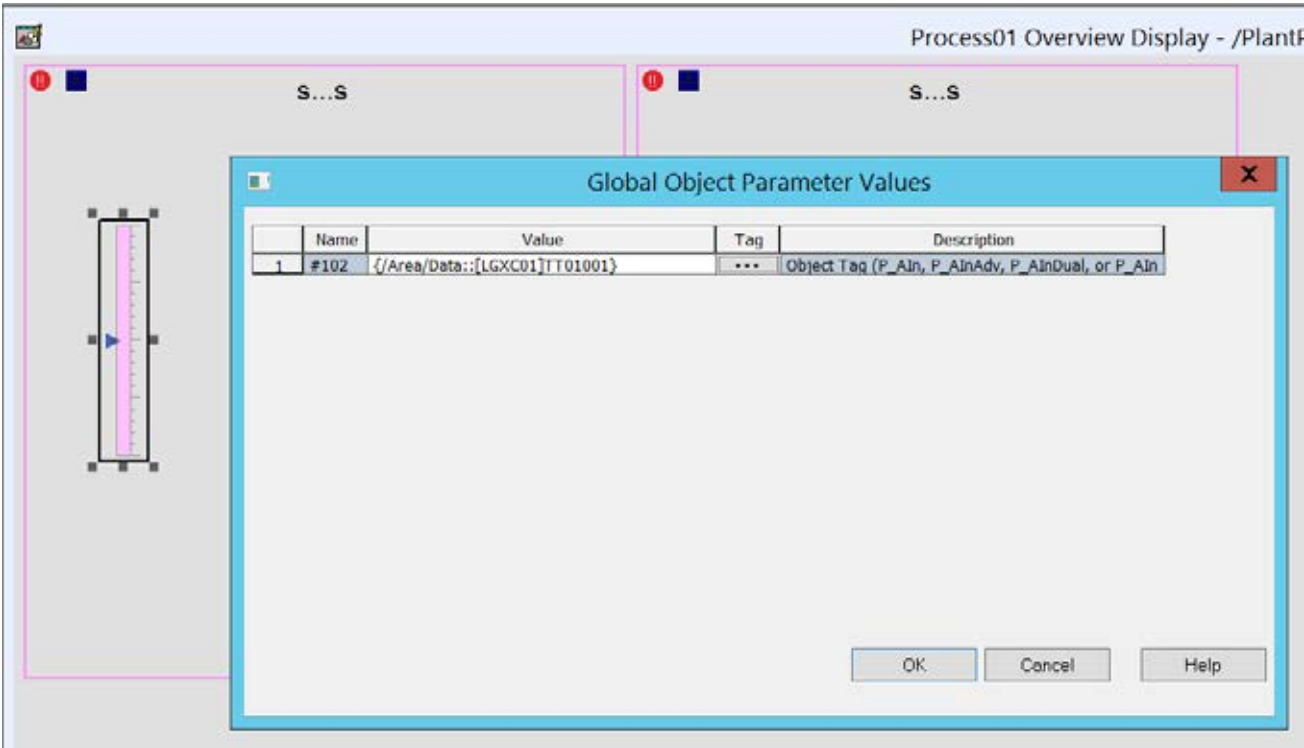
Do these steps to add KPIs to the Process01 Overview Display. In this example, the KPI shows temperature values.

1. In FactoryTalk View Studio, expand Global Objects and double-click '(RA-FRAME) P_AIn Graphics Library'.

A collection of display elements for analog input objects appears.



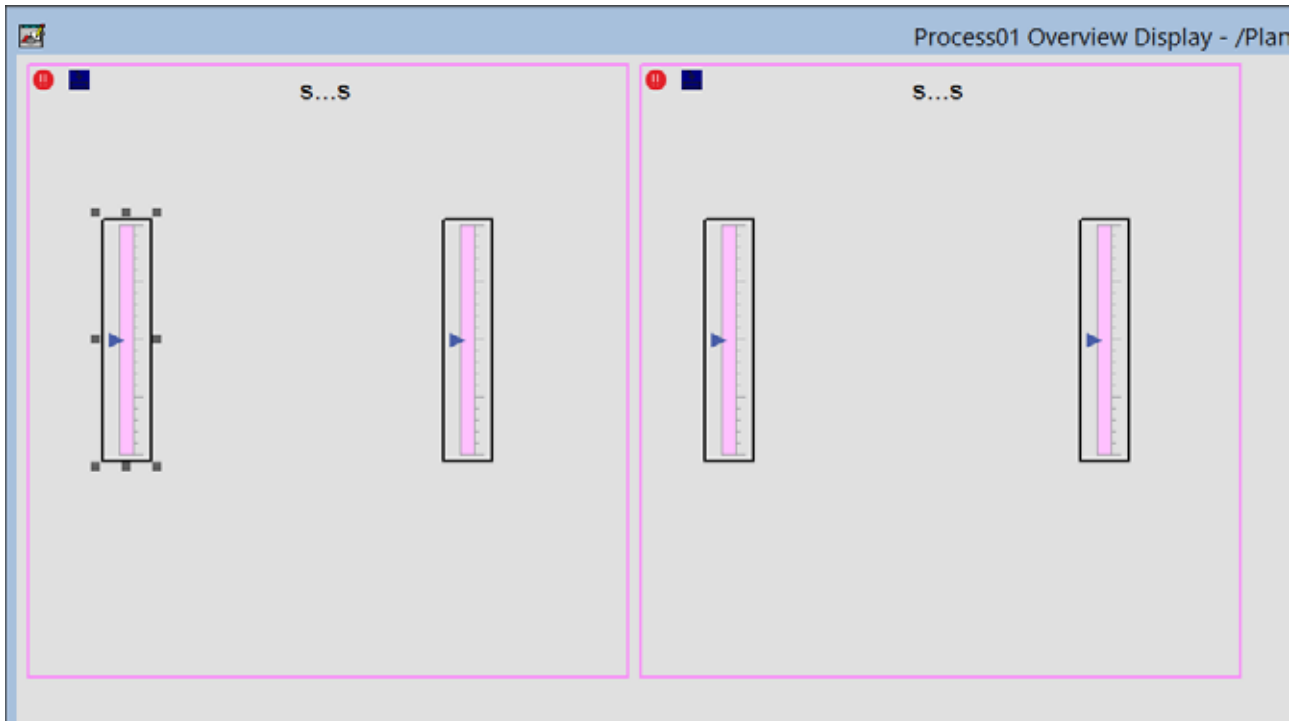
2. Copy the global object and paste into the Process01 Overview Display.
3. Right-click the pasted object and choose Global Object Parameter Values.
4. Type the value as shown in the example.



5. Repeat [step 2](#) through [step 4](#) to add three more KPI objects.

6. Make sure to type the appropriate parameter values into the Global Object Parameter Values table.

The three objects are TT02001, TT03001, TT04001.

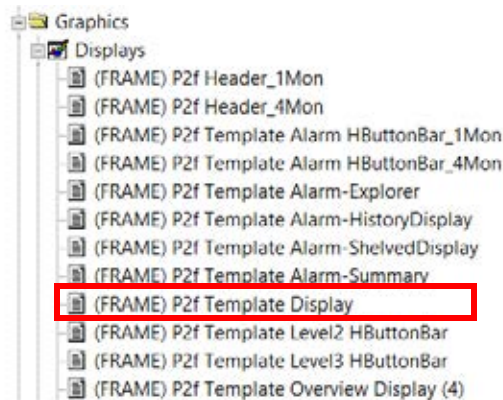


At this stage, the Process01 Overview Display is complete.

Creating Area Overview Displays

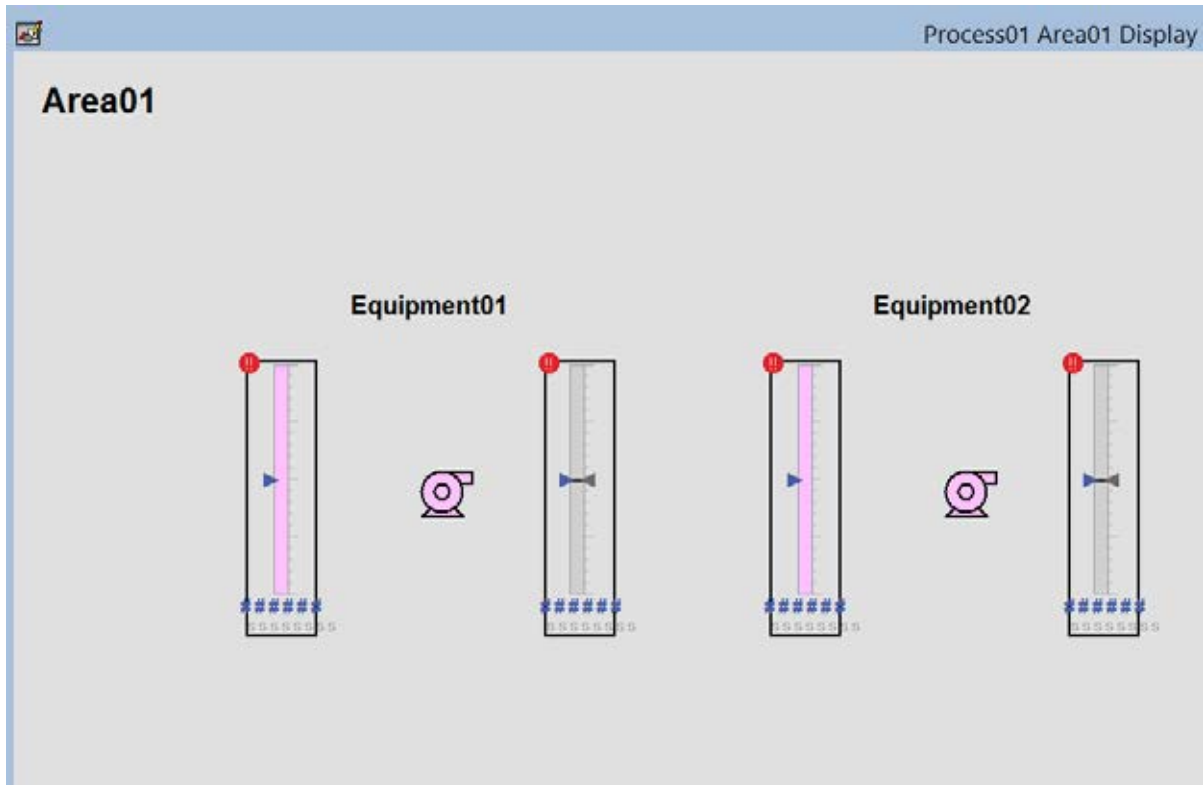
Do these steps to create the Level 2 Area Overview Display. The procedures for creating Area01 **must** be repeated for Area02.

1. Navigate to the HMI template display as shown in the example.
2. Right-click '(FRAME) P2f Template Display' and choose Duplicate.



A window appears on the Studio canvas to choose a name.

3. Rename the display 'Process01 Area01 Display' and click OK.
4. Update the display title text to 'Area01'.
5. Expand Global Objects and copy objects from RA-FRAME Displays to represent Equipment01 and Equipment02.
6. For each object that is added to the display, right-click the object and configure the global object parameter values.



7. Repeat [step 1](#) through [step 6](#) for Area02.

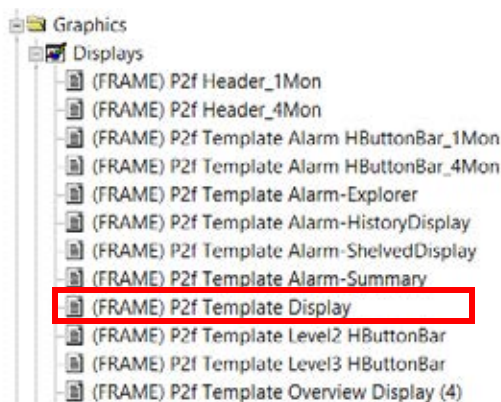
At this stage, you have configured the Level 2 displays.

Creating Equipment Detail Displays

Do these steps to create the Level 3 Equipment Detail Displays. The procedures for creating the first detail display **must** be repeated for all equipment.

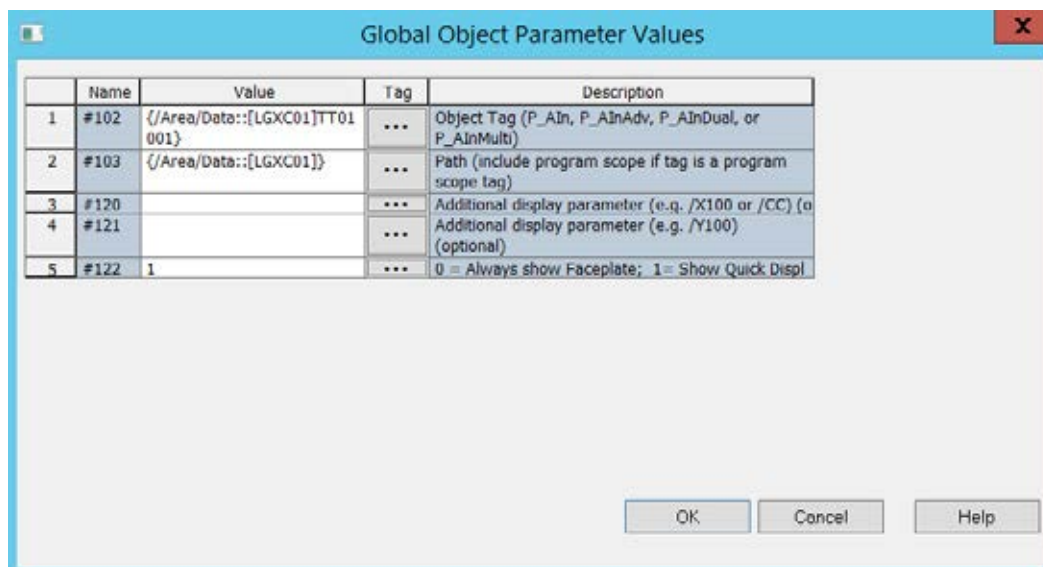
1. Navigate to the HMI template display as shown in the example.

- Right-click '(FRAME) P2f Template Display' and choose Duplicate.

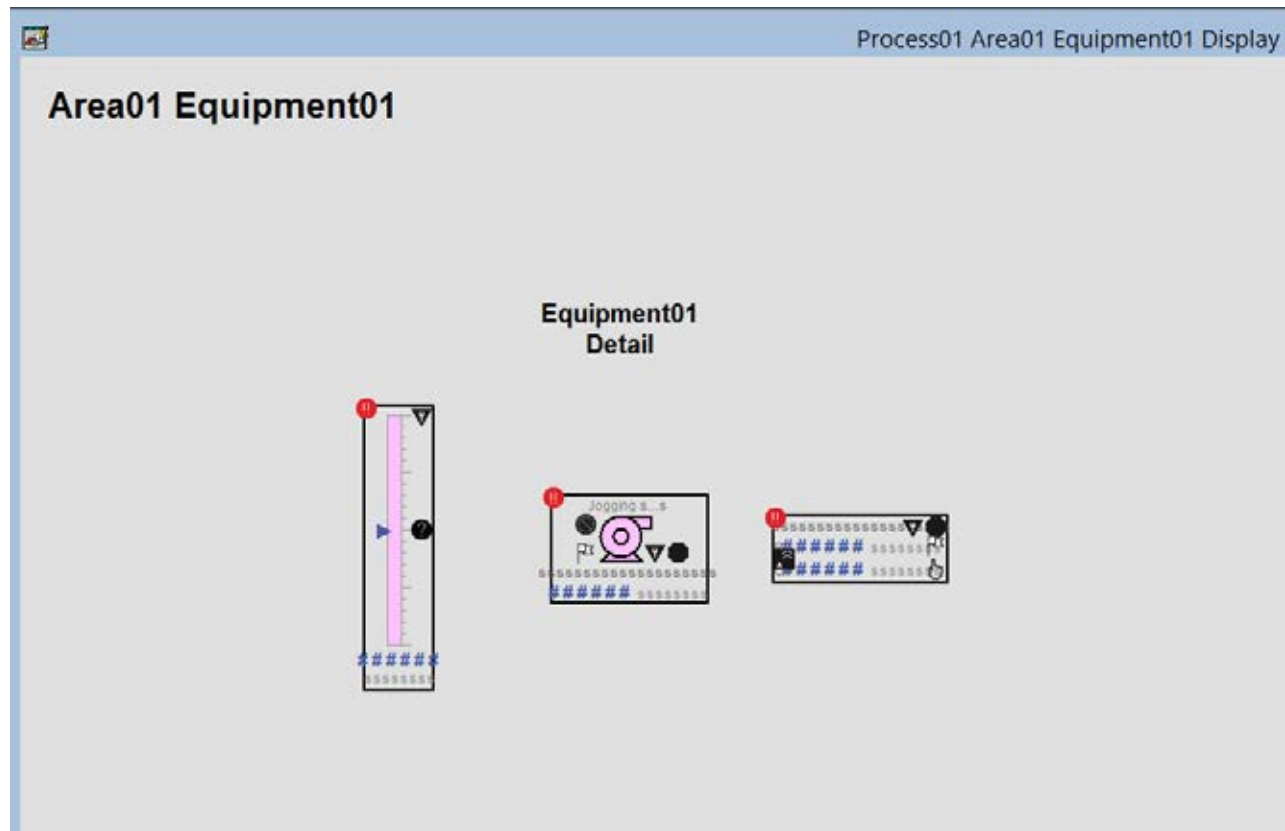


A window appears on the Studio canvas to choose a name.

- Rename the display 'Process01 Area01 Equipment01 Display' and click OK.
- Update the display title text to 'Area01 Equipment01'.
- Expand Global Objects and copy objects from RA-BAS Displays to represent Equipment01.
- For each object that is added to the display, right-click the object and configure the global object parameter values.



At this stage, your equipment detail looks as shown in the example.

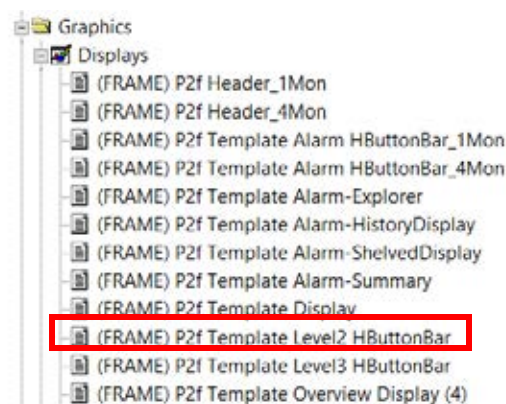


7. Repeat [step 1](#) through [step 6](#) for Equipment02 through Equipment04. Make sure that the titles are correct for the additional equipment.

Creating Level 2 Navigation Displays

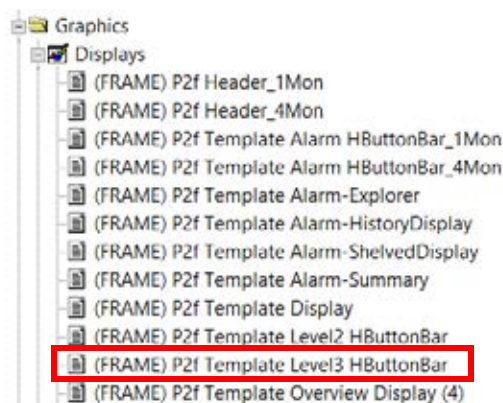
Do these steps to create Level 2 button bars that are used for navigating displays.

1. In FactoryTalk View Studio, right-click '(FRAME) P2f Template Level2 HButtonBar' and choose Duplicate.



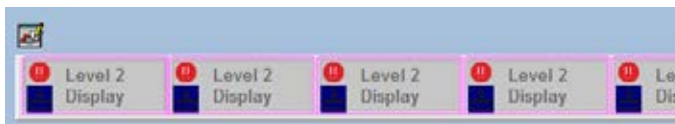
A window appears on the Studio canvas to choose a name.

2. Rename the display 'Process01 HButtonBar' and click OK.
3. Right-click '(FRAME) P2f Template Level3 HButtonBar' and choose Duplicate.

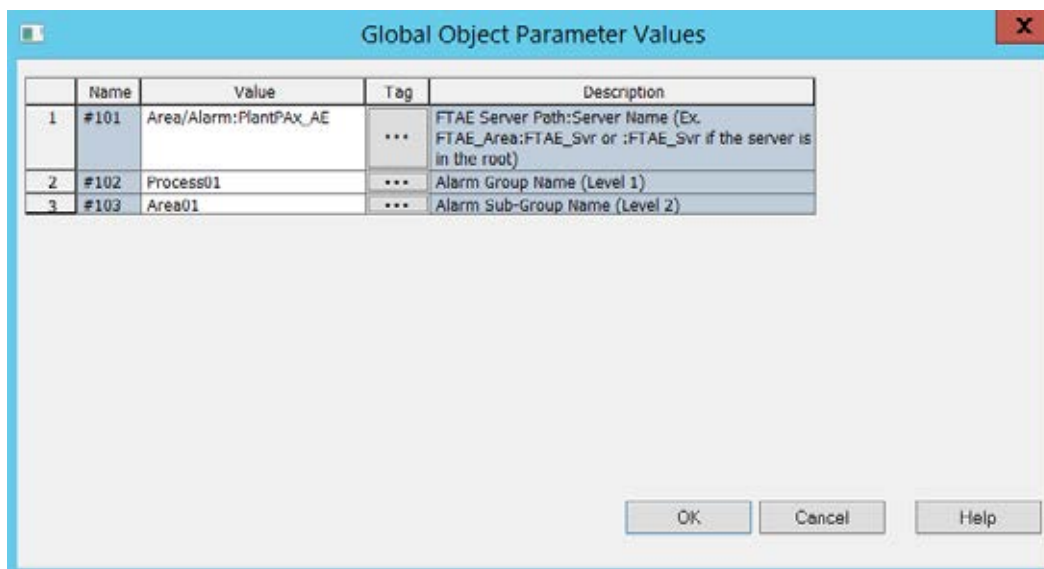


A window appears on the Studio canvas to choose a name.

4. Rename the display 'Process01 Area01 HButtonBar' and click OK.
5. Repeat [step 3](#) and [step 4](#) to create another button bar, but rename 'Process01 Area02 HButtonBar'.
6. Open the Process01 HButtonBar display.



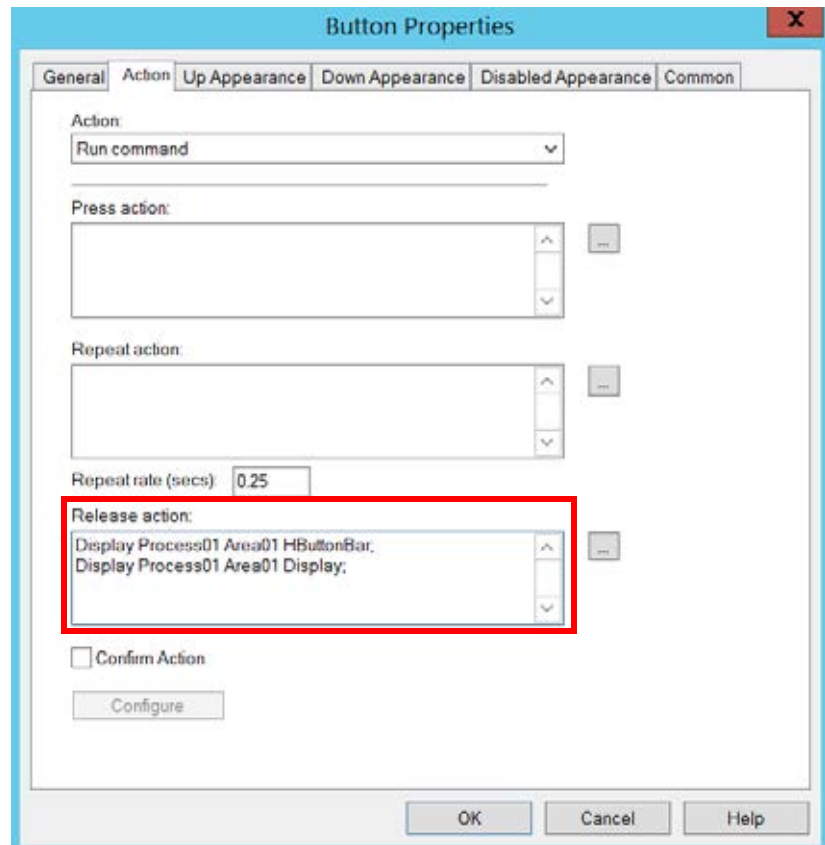
7. Right-click the alarm object that surrounds the first navigation button and choose Global Object Parameter Values.
8. Configure the global object parameter values as shown.



9. Repeat [step 7](#) and [step 8](#) for the alarm objects that surround the second navigation button.

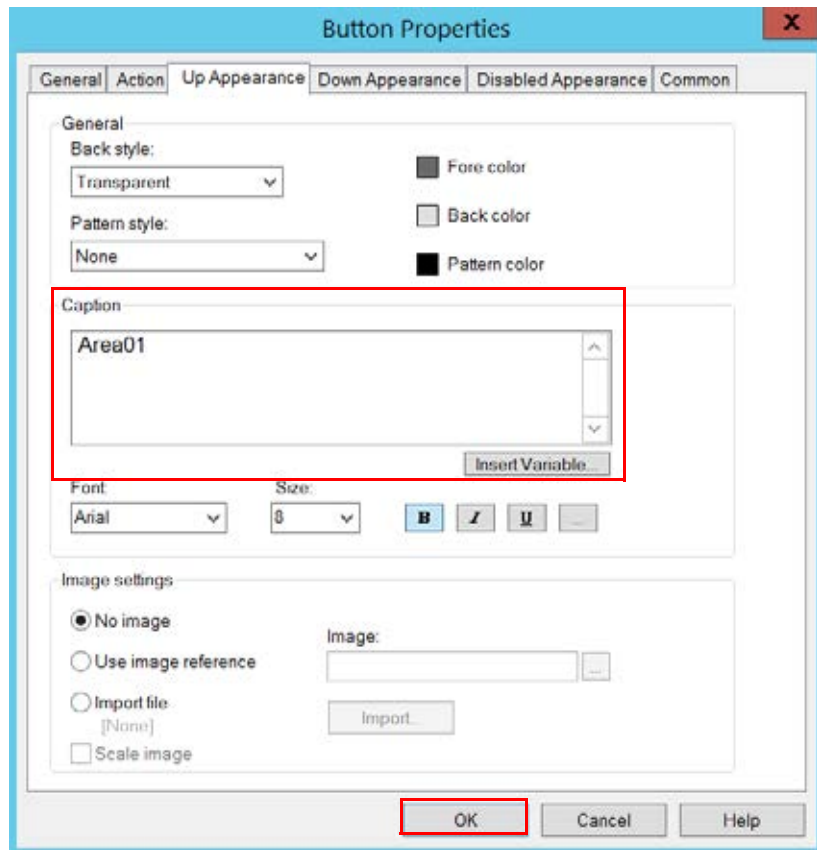
Make sure parameter #103 is Area02.

10. Right-click on the first navigation button and choose Properties.
11. Click the Action tab and update the Release Action based on the example shown.



12. Click the Up Appearance tab.

13. Update the caption as shown in the example.



14. Click OK.
15. For the second navigation button, repeat [step 9](#) through [step 13](#).
Make sure the action and caption are for Area02.
16. Delete all unused navigation buttons from the Process01 HButtonBar display.

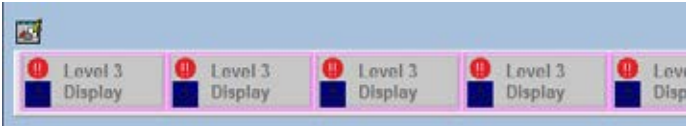


17. Close the Process01 HButtonBar display.

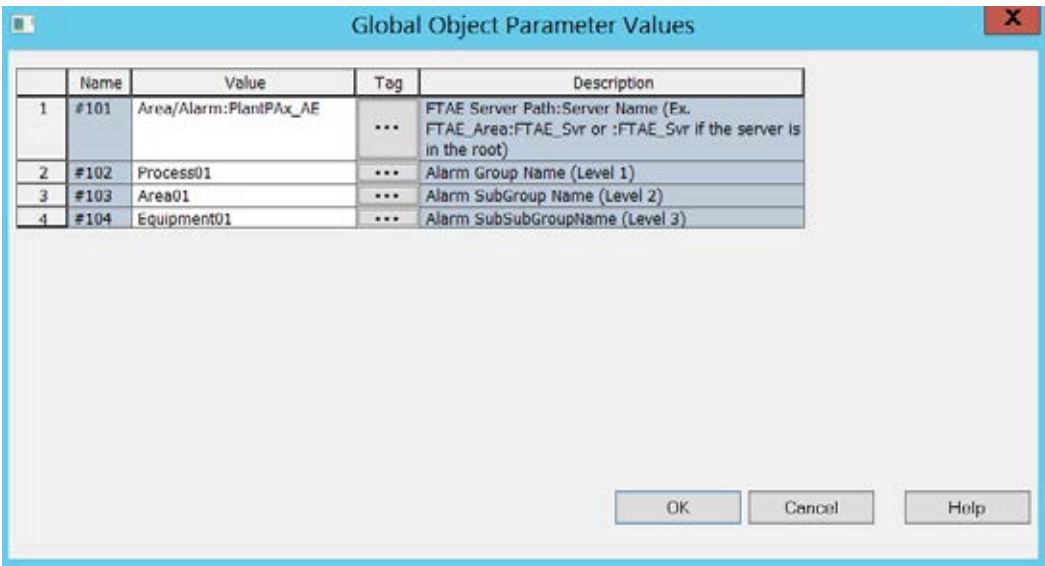
Creating Level 3 Navigation Displays

Do these steps to create Level 3 button bars that are used for navigating displays.

- 1. Open the Process01 Area01 HButtonBar display.

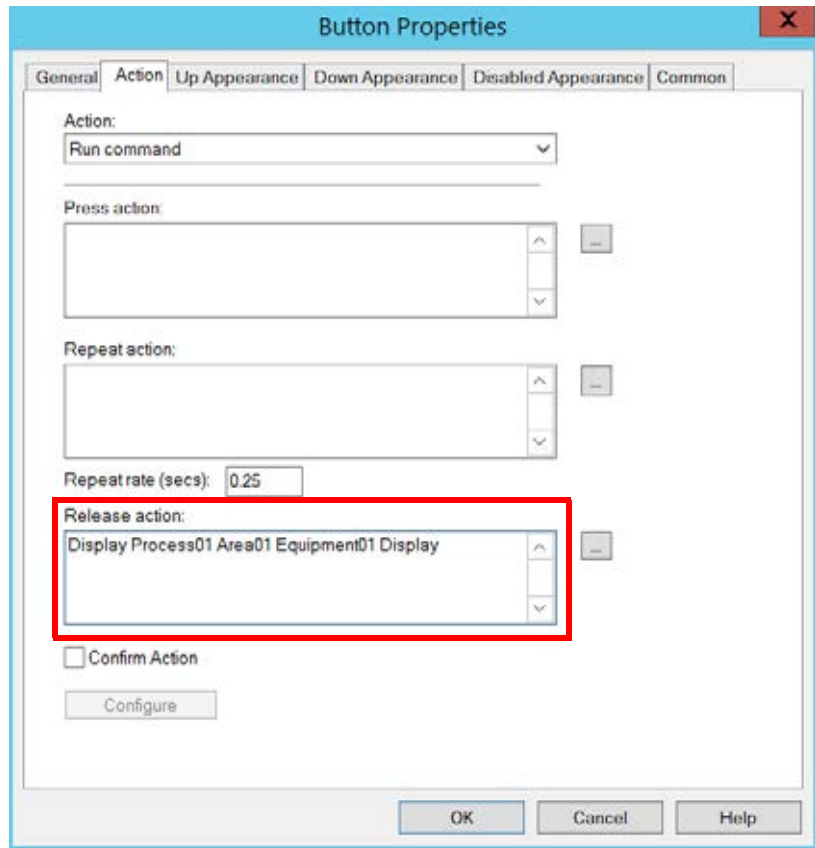


- 2. Right-click the alarm object that surrounds the first navigation button and choose Global Object Parameter Values.
- 3. Configure the global object parameter values as shown.



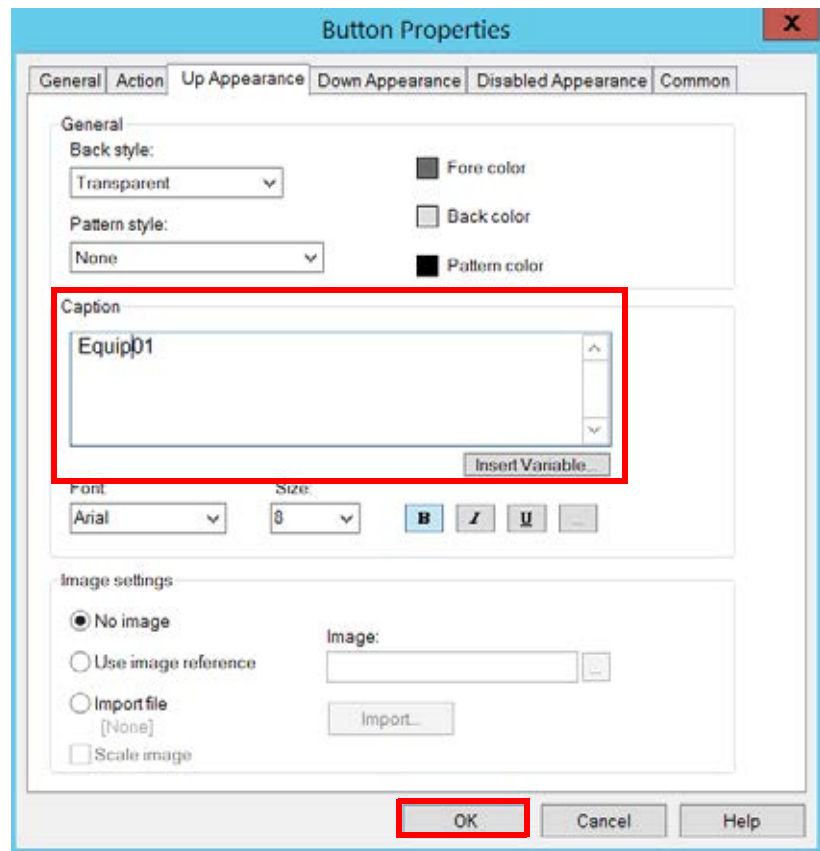
- 4. Repeat [step 2](#) and [step 3](#) for the alarm objects that surround the second navigation button.
Make sure parameter #104 is Equipment02.
- 5. Right-click on the first navigation button and choose Properties.

- Click the Action tab and update the Release Action based on the example shown.

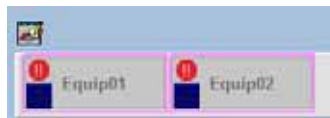


- Click the Up Appearance tab.

8. Update the caption as shown in the example.



9. Click OK.
10. For the second navigation button, repeat [step 5](#) through [step 9](#).
Make sure the action and caption are for Equip02.
11. Delete all unused navigation buttons from the Process01 Area01 HButtonBar display.



12. Close the Process01 Area01 HButtonBar display.
13. Open the Process01 Area02 HButtonBar display.
14. Repeat [step 1](#) through [step 13](#) for this display using Equipment03 and Equipment04.

At this stage, your HMI application contains the following displays.

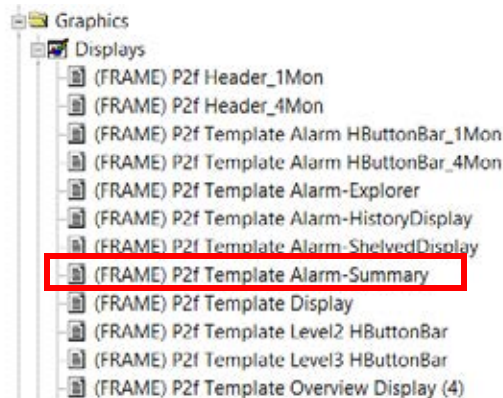


Create Alarm Displays

Do these steps to create the Alarm displays for the HMI application.

IMPORTANT As stated at the beginning of the Customizing an HMI Template section, organize and group your alarms based on the level hierarchy (Levels 1...3) of your system. For additional information, see Chapter 4 in the PlantPAx DCS Application User Manual, publication [PROCES-UM003](#).

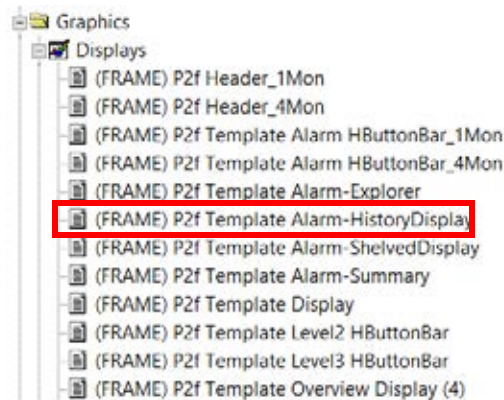
1. In FactoryTalk View Studio, right-click '(FRAME) P2f Template Alarm-Summary' and choose Duplicate.



A window appears on the Studio canvas to choose a name.

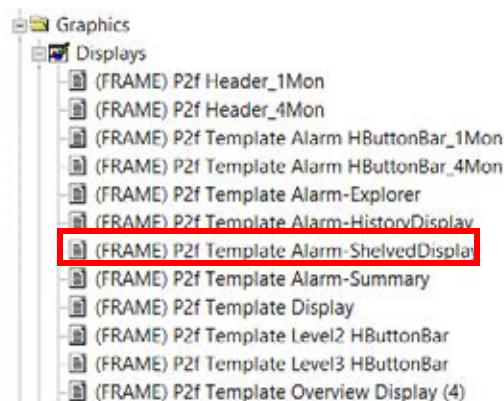
2. Rename the display 'Process01 Alarm-Summary' and click OK.

3. Right-click '(FRAME) P2f Template Alarm-HistoryDisplay' and choose Duplicate.



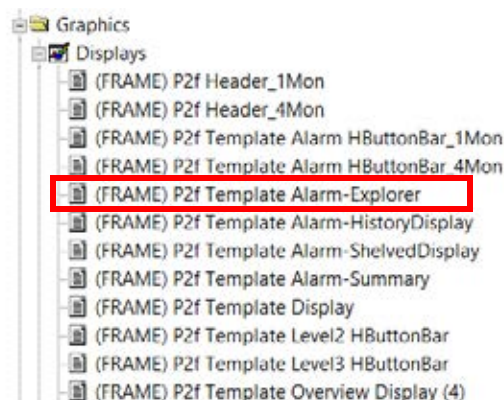
A window appears on the Studio canvas to choose a name.

4. Rename the display 'Process01 Alarm-HistoryDisplay' and click OK.
5. Right-click '(FRAME) P2f Template Alarm-ShelvedDisplay' and choose Duplicate.



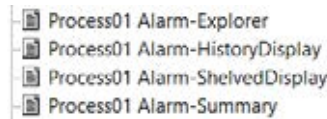
A window appears on the Studio canvas to choose a name.

6. Rename the display 'Process01 Alarm-ShelvedDisplay' and click OK.
7. Right-click '(FRAME) P2f Template Alarm-Explorer' and choose Duplicate.



A window appears on the Studio canvas to choose a name.

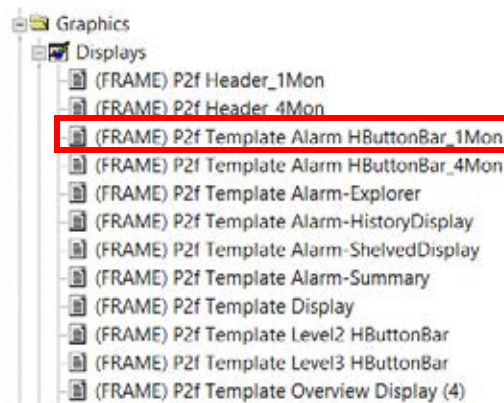
8. Rename the display 'Process01 Alarm-Explorer' and click OK.



9. Open the Process01 Alarm-Summary Display.
10. Close the Process01 Alarm-Summary Display.

IMPORTANT The next steps allow your alarm displays to perform on single and four-monitor operator workstations.

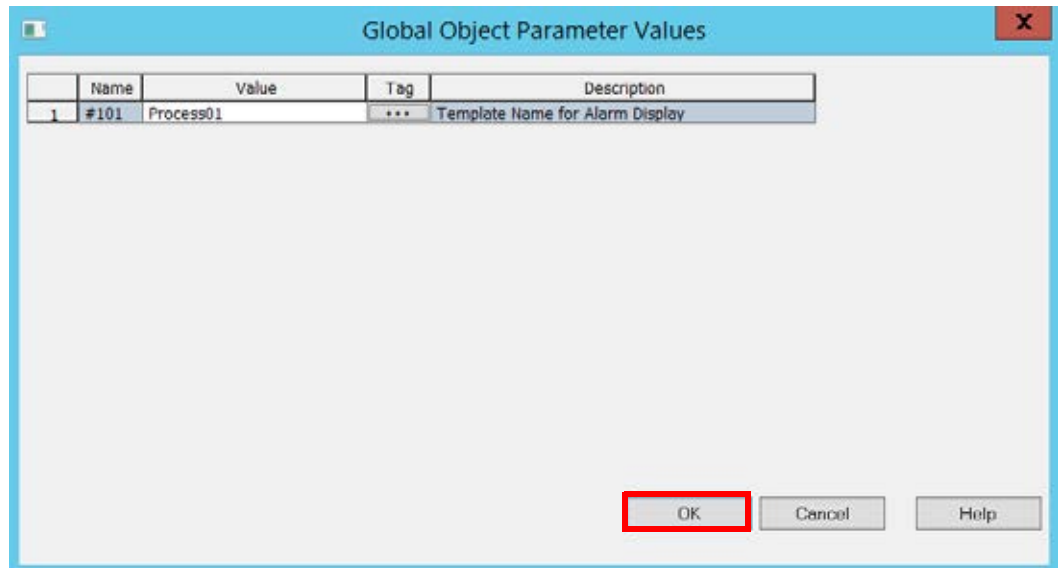
11. Right-click '(FRAME) P2f Template Alarm-HButtonBar_1Mon' and choose Duplicate.



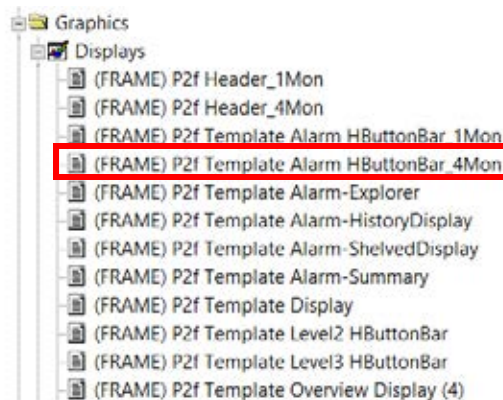
A window appears on the Studio canvas to choose a name.

12. Rename the display 'Process01 Alarm HButtonBar_1Mon' and click OK.
13. Open the Process01 Alarm HButtonBar_1Mon Display.
14. Right-click the button group and choose global object parameter values.

15. Type the value shown in the example and click OK.



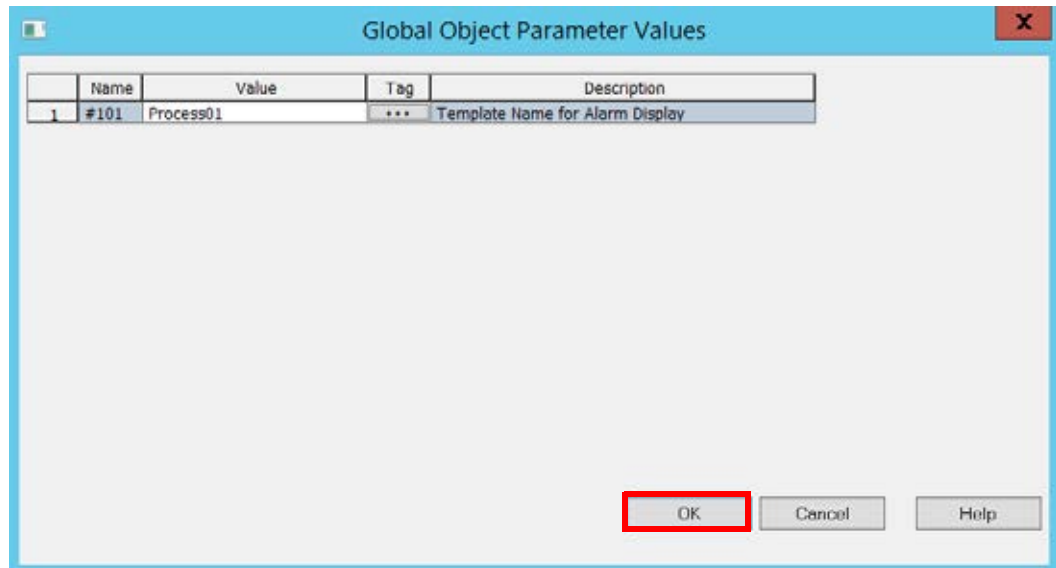
16. Close the Process01 Alarm HButtonBar_1Mon Display.
17. Right-click '(FRAME) P2f Template Alarm-HButtonBar_4Mon' and choose Duplicate.



A window appears on the Studio canvas to choose a name.

18. Rename the display 'Process01 Alarm HButtonBar_4Mon' and click OK.
19. Open the Process01 Alarm HButtonBar_4Mon Display.
20. Right-click the button group and choose global object parameter values.

21. Type the value shown in the example and click OK.

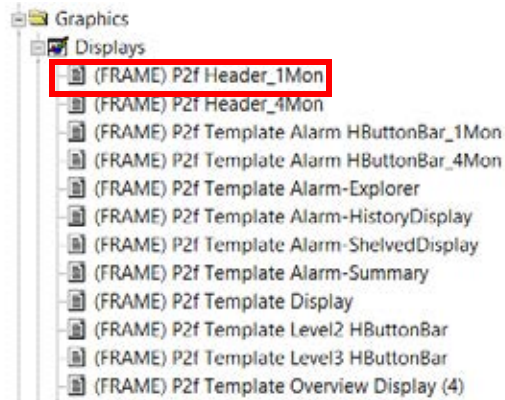


22. Close the Process01 Alarm HButtonBar_4Mon Display.

Create a Single-monitor Header Display

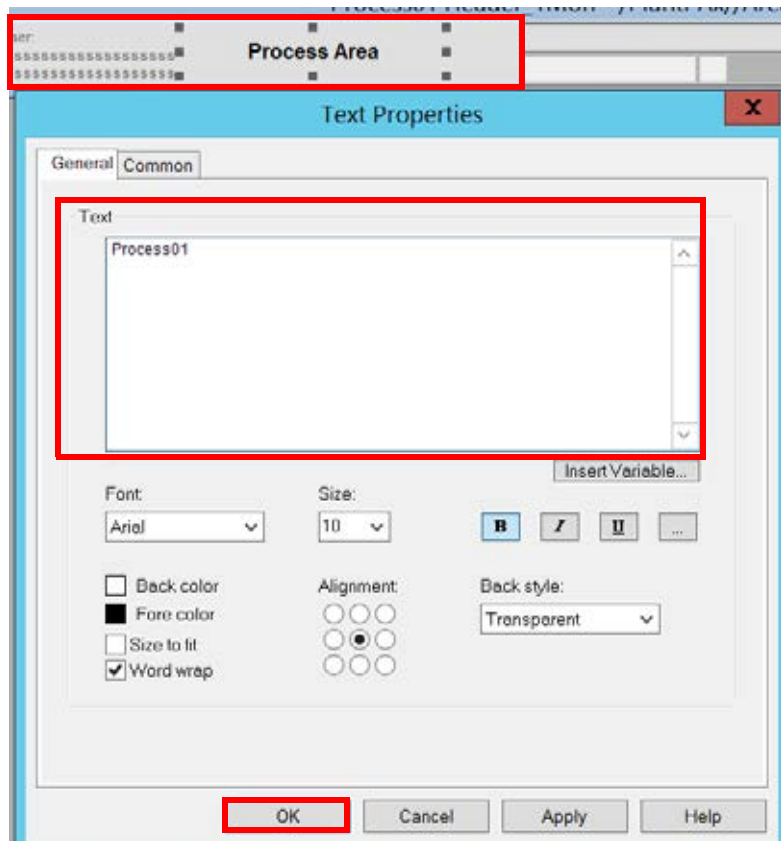
Do these steps to create a banner at top of a display for a single monitor display.

1. In FactoryTalk View Studio, right-click '(FRAME) P2f Header_1Mon' and choose Duplicate.

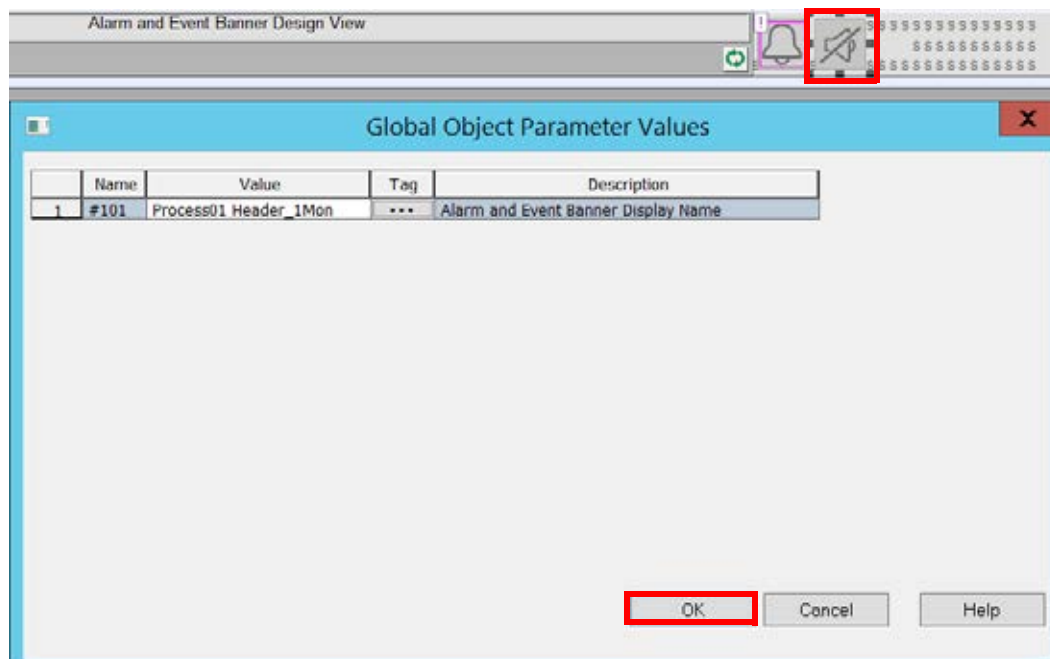


A window appears on the Studio canvas to choose a name.

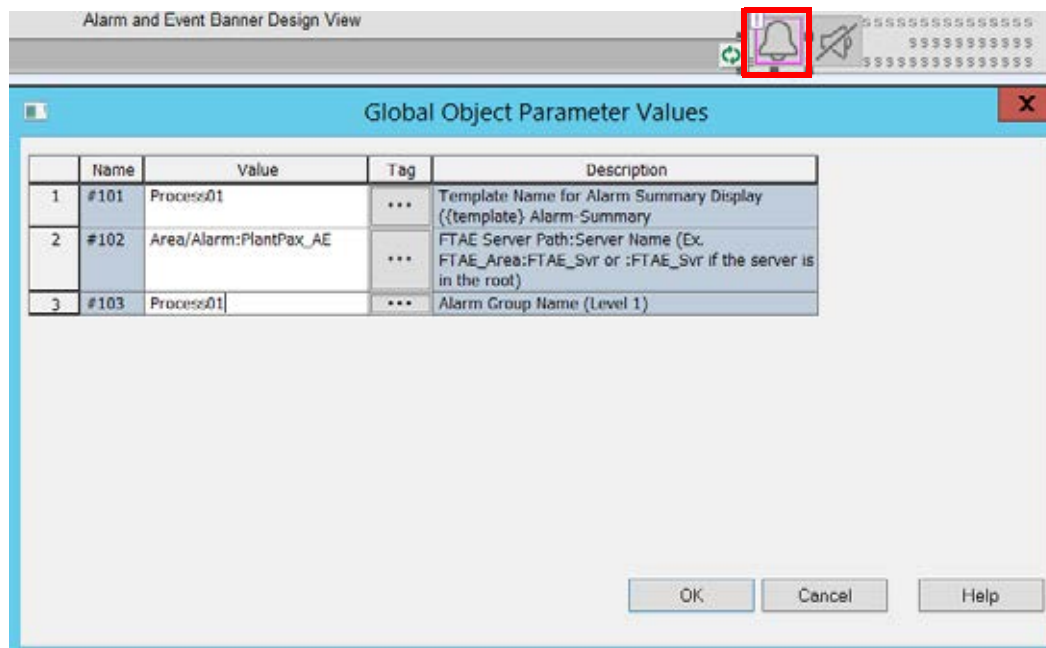
2. Rename the display 'Process01 Header_1Mon' and click OK.
3. Open the Process01 Header_1Mon Display.
4. Double-click the header title text and update as shown in the example.



5. Click OK.
6. Right-click the Horn Silence button and choose global object parameter values.
7. Type the values as shown in the example, and click OK.

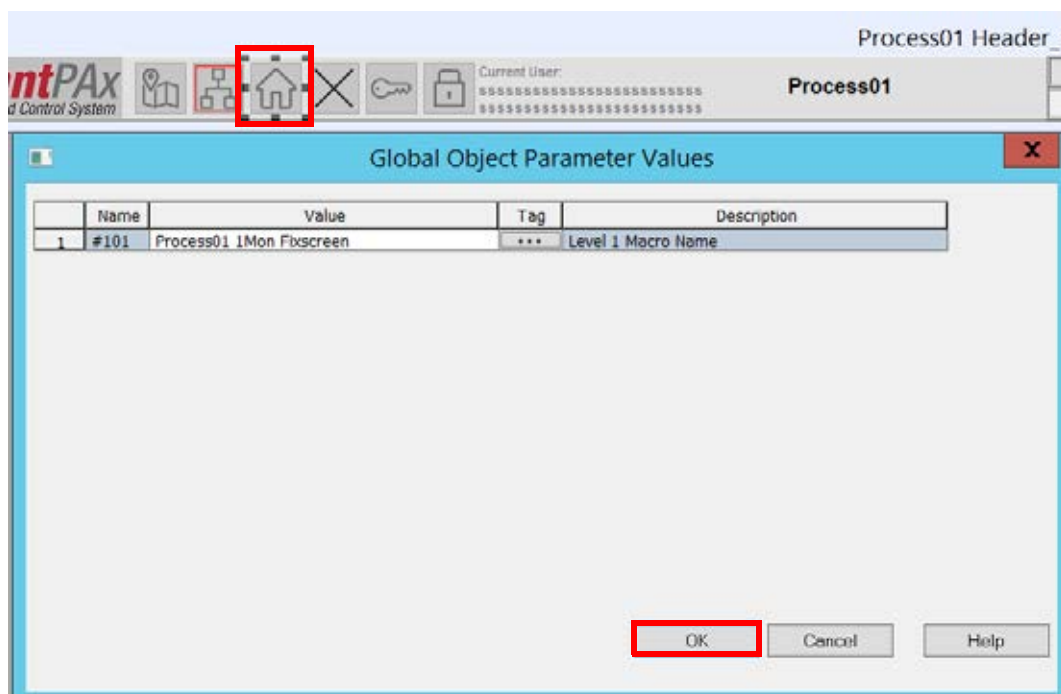


8. Right-click on the Alarm Navigation button and choose global object parameter values.
9. Type the values as shown in the example and click OK.

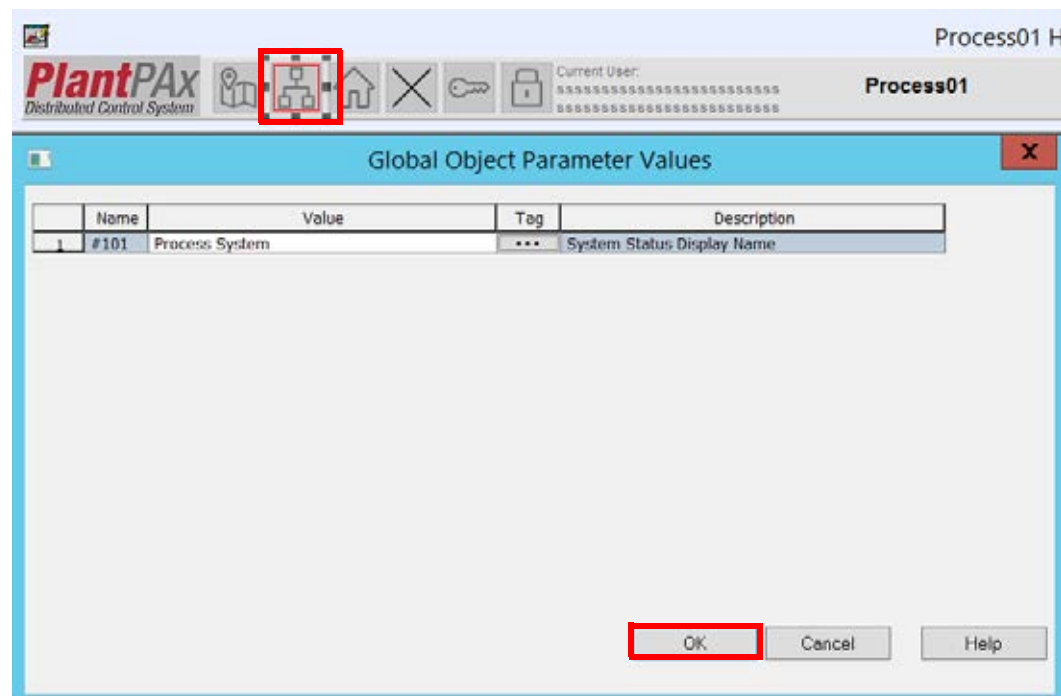


10. The single monitor header display includes an alarm banner.

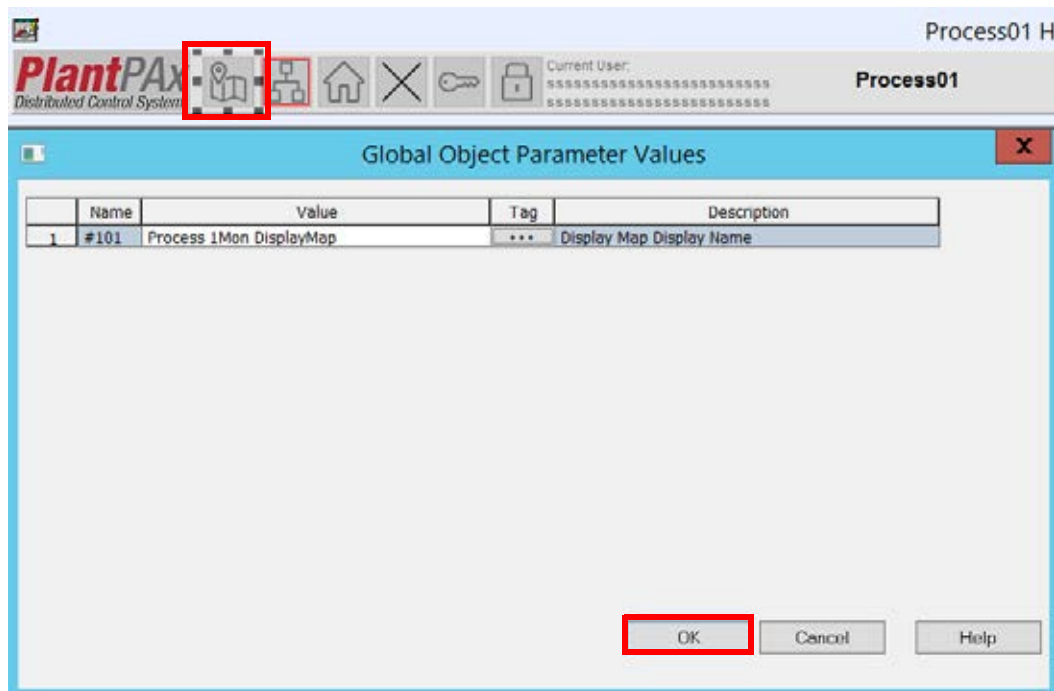
11. Right-click the Home button and choose global object parameter values.
12. Type the values as shown in the example and click OK.



13. Right-click the System button and choose global object parameter values.
14. Type the values as shown in the example and click OK.



15. Right-click the Display Map button and choose global object parameter values.
16. Type the values as shown in the example and click OK.

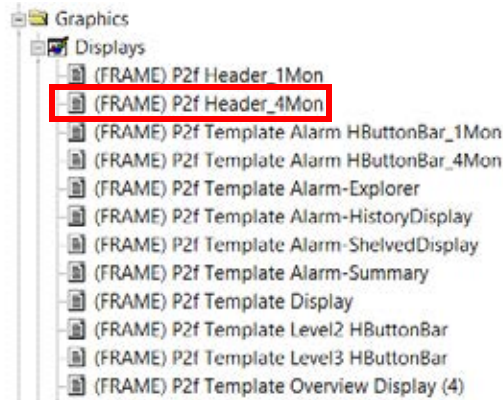


17. Close the Process01 Header_1Mon Display.

Create a Four-monitor Header Display

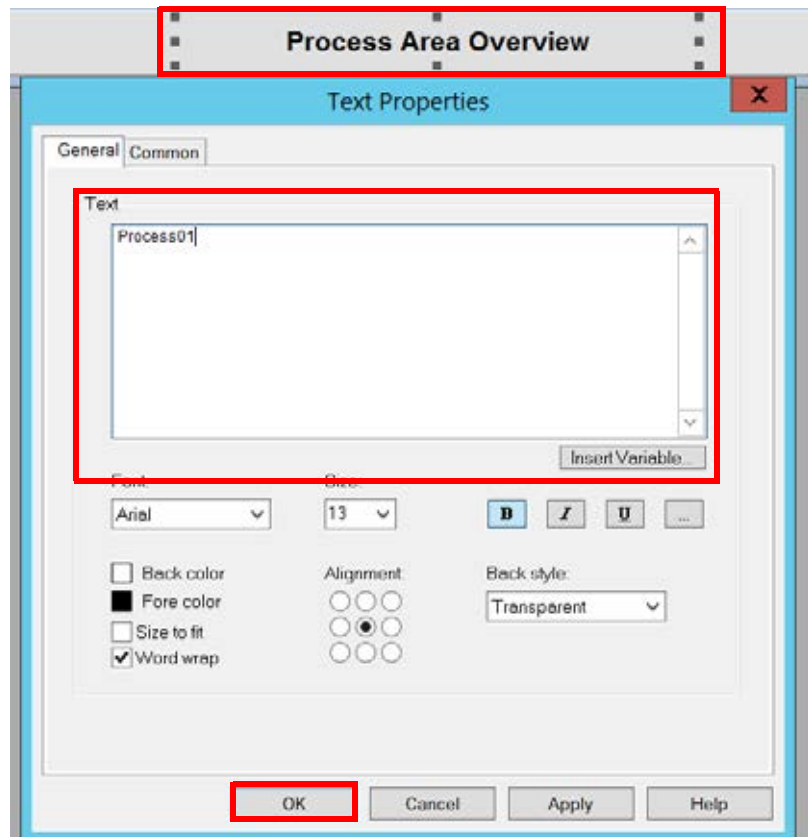
Do these steps to create a banner at top of a display for a four-monitor display.

1. In FactoryTalk View Studio, right-click '(FRAME) P2f Header_4Mon' and choose Duplicate.



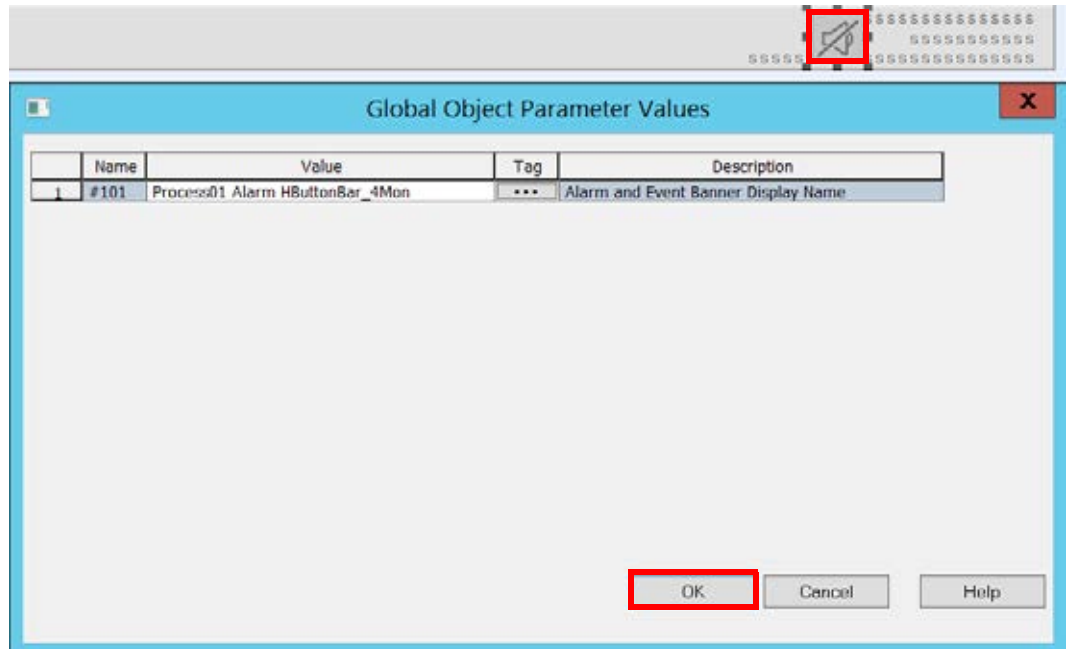
A window appears on the Studio canvas to choose a name.

2. Rename the display 'Process01 Header_4Mon' and click OK.
3. Open the Process01 Header_4Mon Display.
4. Double-click the header title text and update as shown in the example.

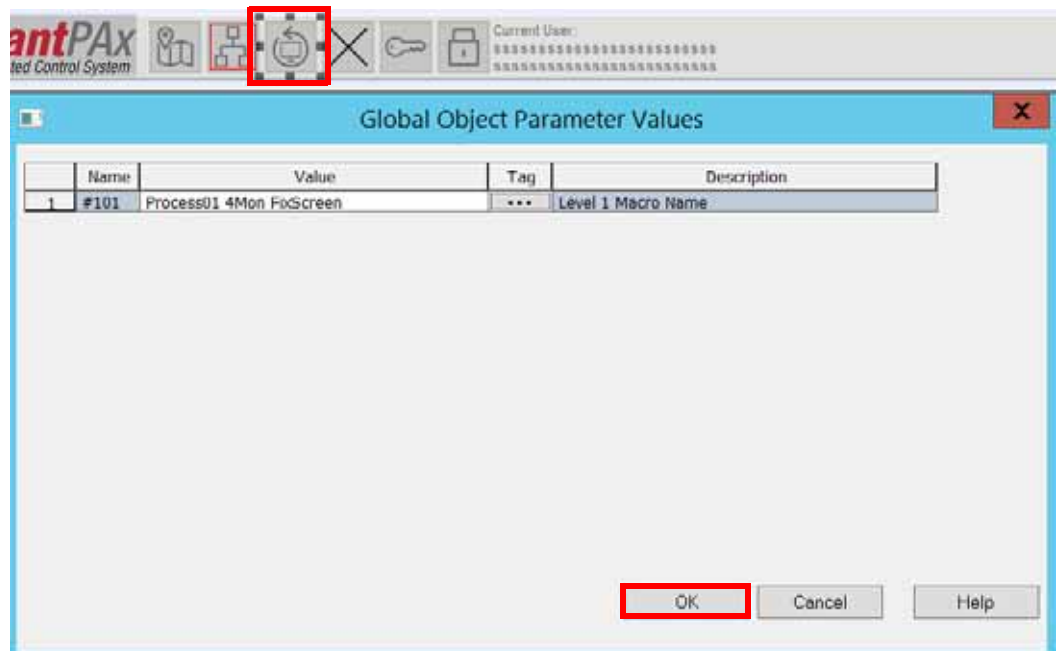


5. Click OK.

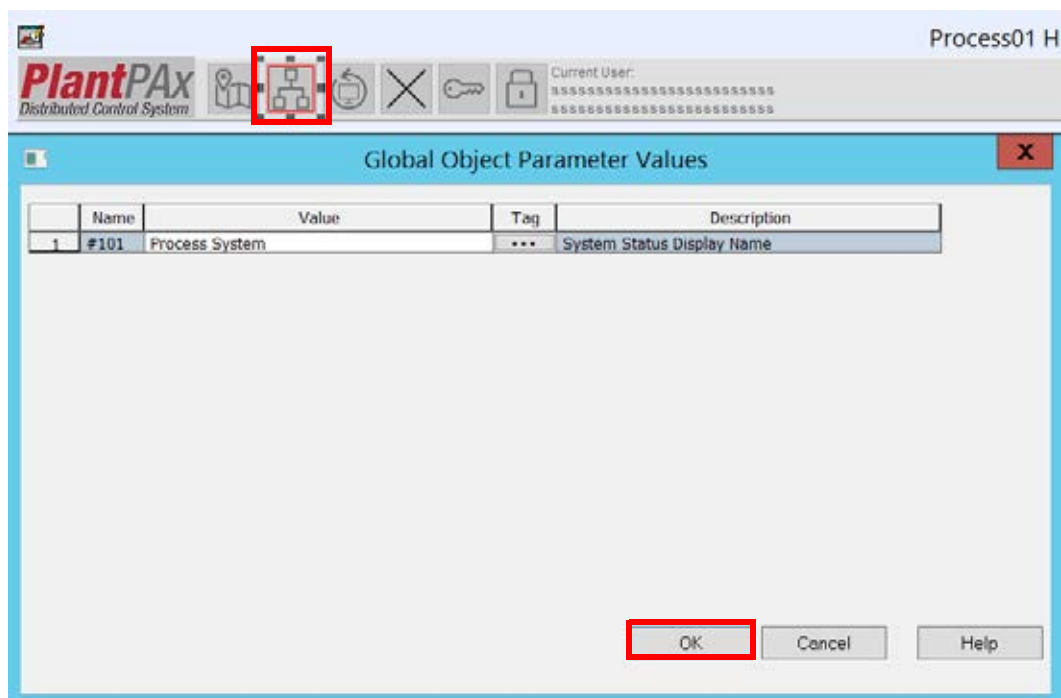
6. Right-click the Horn Silence button and choose global object parameter values.
7. Type the values as shown in the example, and click OK.



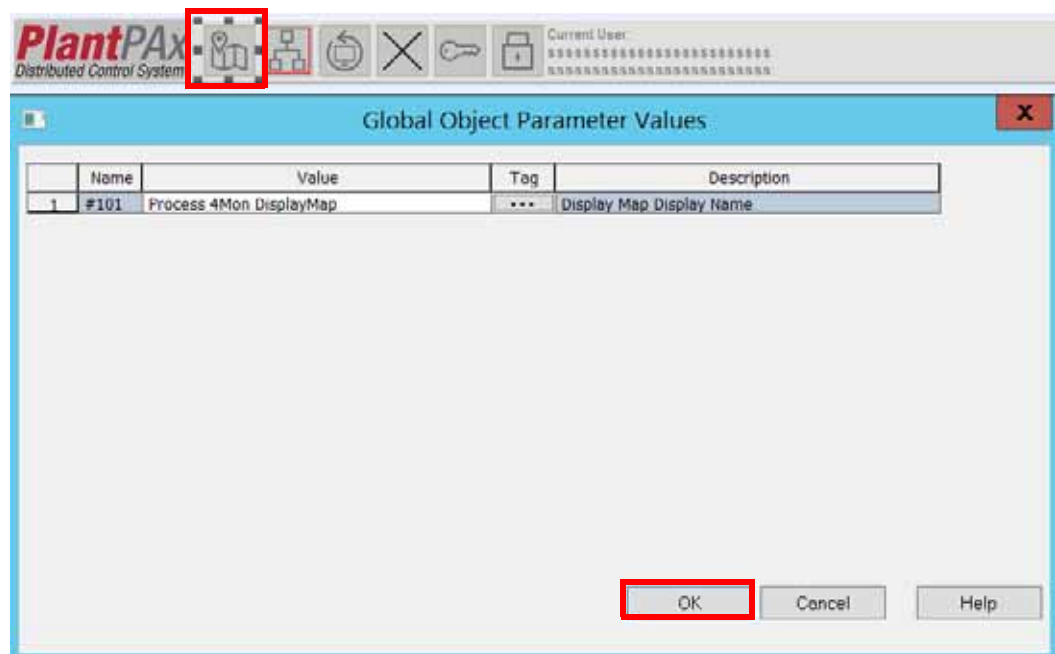
8. Right-click the Fix Screen button and choose global object parameter values.
9. Type the values as shown in the example and click OK.



10. Right-click the System button and choose global object parameter values.
11. Type the values as shown in the example and click OK.



12. Right-click the Display Map button and choose global object parameter values.
13. Type the values as shown in the example and click OK.

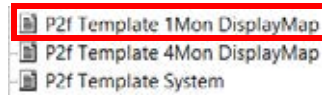


14. Close the Process01 Header_4Mon Display.

Create the Display Map Single Monitor Display

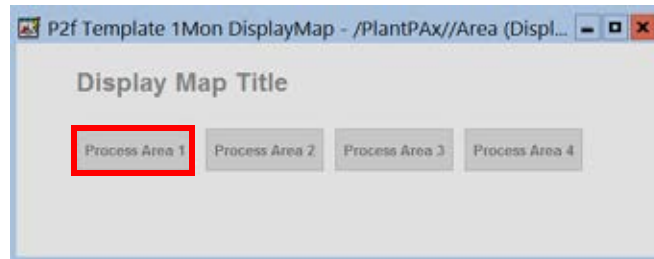
Do these steps to configure high-level navigation that lets operators view different process areas within the same HMI application. This instruction is for single monitors.

1. In FactoryTalk View Studio, right-click 'P2f Template 1Mon Display Map' and choose Duplicate.

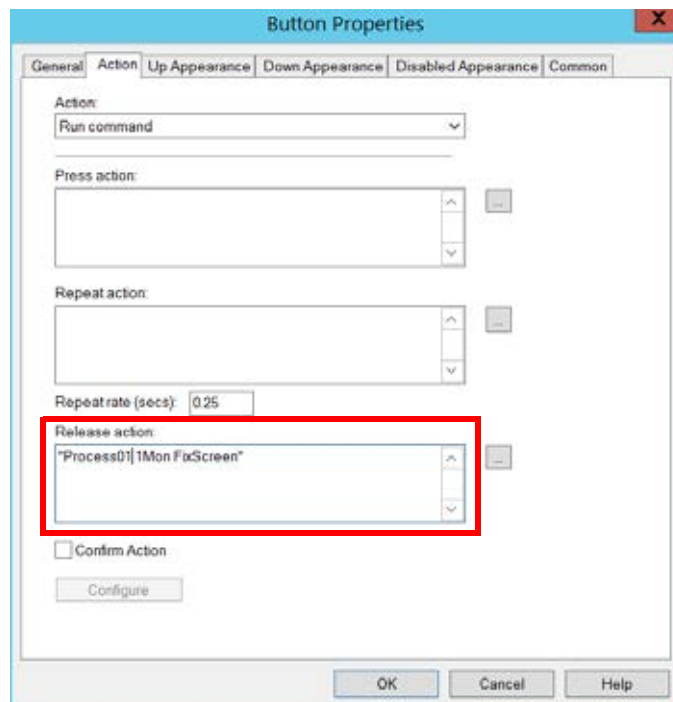


A window appears on the Studio canvas to choose a name.

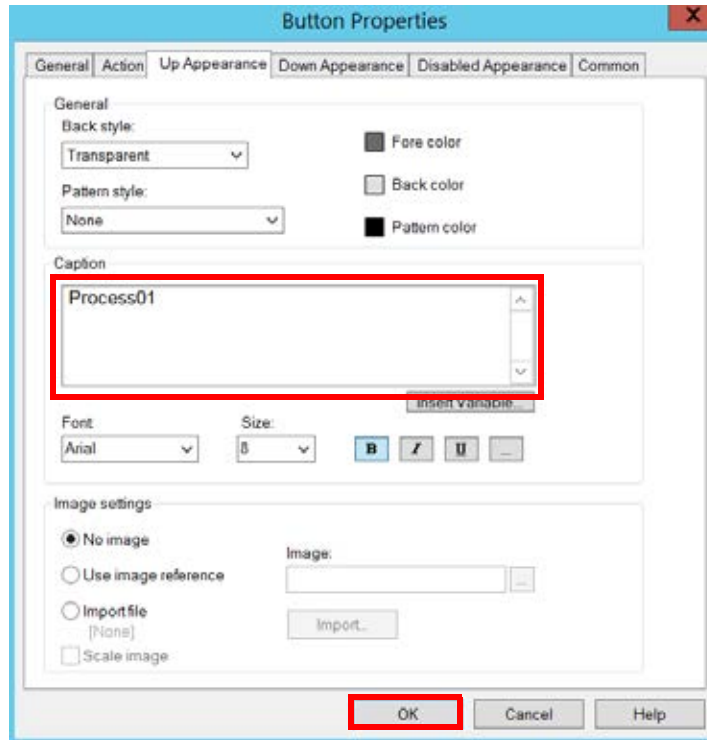
2. Rename the display 'Process 1Mon DisplayMap' and click OK.
3. Open the Process 1Mon DisplayMap Display.
4. Right-click Process Area 1 button and choose Properties.



5. Select the Action tab and update the release action as shown in the example.



- Click the Up Appearance tab and update the caption as shown in the example.

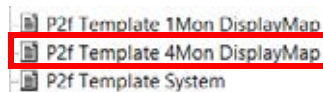


- Click OK.
- Close the Process 1Mon DisplayMap Display.
This display configuration is application-specific.

Create the Display Map Four-monitor Display

Do these steps to configure high-level navigation that lets operators view different process areas within the same HMI application. This instruction is for a four-monitor operator workstation.

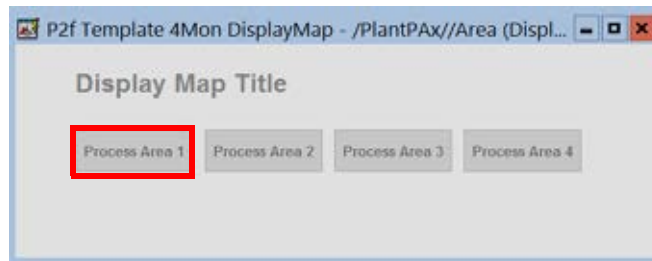
- In FactoryTalk View Studio, right-click 'P2f Template 4Mon Display Map' and choose Duplicate.



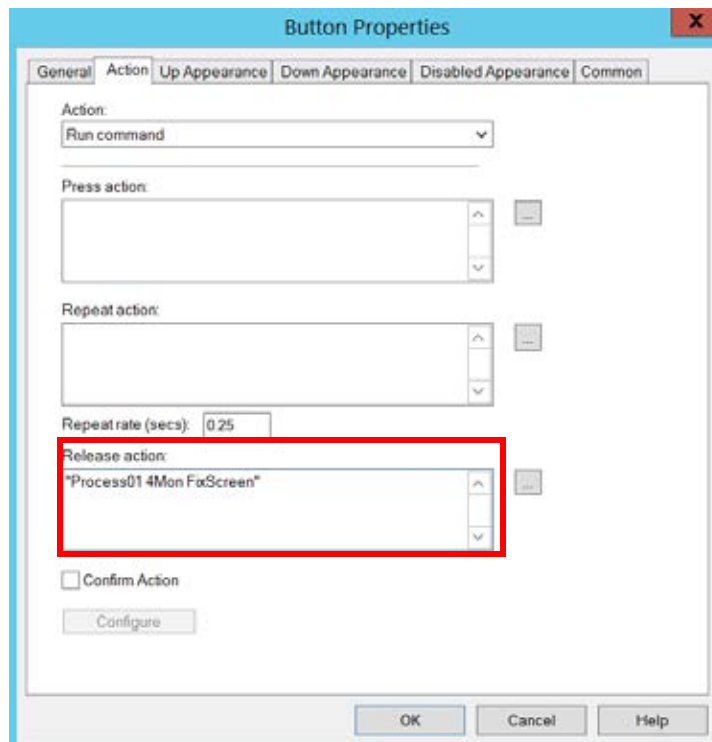
A window appears on the Studio canvas to choose a name.

- Rename the display 'Process 4Mon DisplayMap' and click OK.
- Open the Process 4Mon DisplayMap Display.

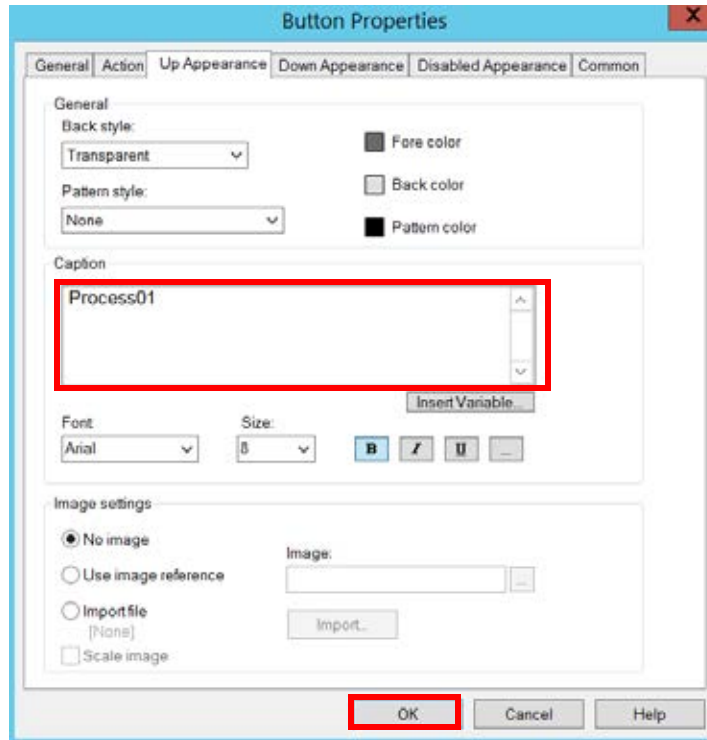
4. Right-click Process Area 1 button and choose Properties.



5. Select the Action tab and update the release action as shown in the example.



- Click the Up Appearance tab and update the caption as shown in the example.

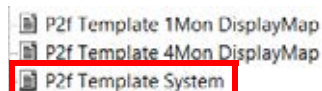


- Click OK.
- Close the Process 4Mon DisplayMap Display.
This display configuration is application-specific.

Create the System Display

Do these steps to configure a display for system diagnostics.

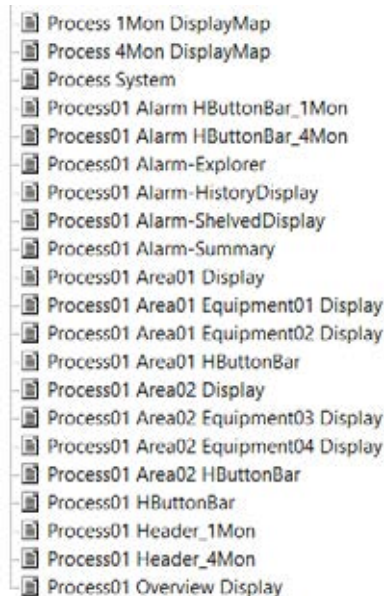
- In FactoryTalk View Studio, right-click 'P2f Template System' and choose Duplicate.



A window appears on the Studio canvas to choose a name.

- Rename the display 'Process System' and click OK.

At this stage, the displays for this example HMI application are created and resemble the example.

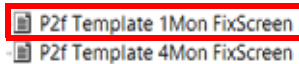


For basic system diagnostics, see Chapter 9 in the PlantPAx DCS Application User Manual, publication [PROCES-UM003](#).

Rename HMI Template Macros

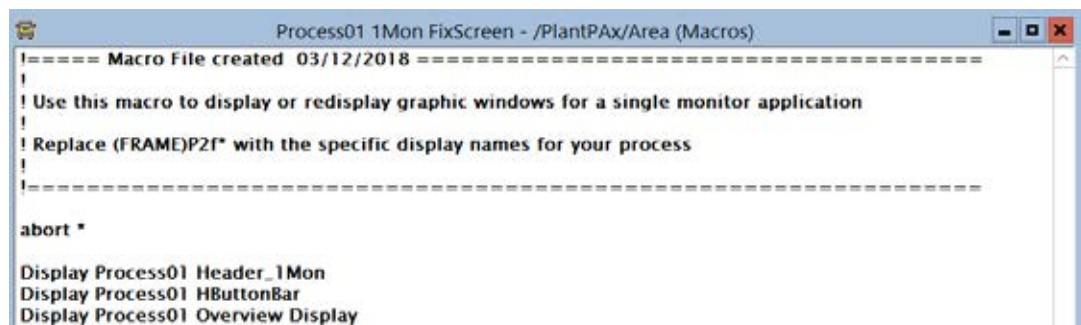
Do these steps to rename HMI macros, which are used for the client startup configuration.

1. From the Project Explorer in FactoryTalk View Studio, expand Macros.
2. Right-click 'P2f Template 1Mon FixScreen' and choose Duplicate.

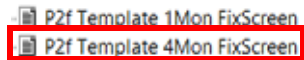


A window appears on the Studio canvas to choose a name.

3. Rename the display 'Process01 1Mon FixScreen' and click OK.
4. Open the Process01 1Mon FixScreen macro.
5. Update the macro as shown in the example.

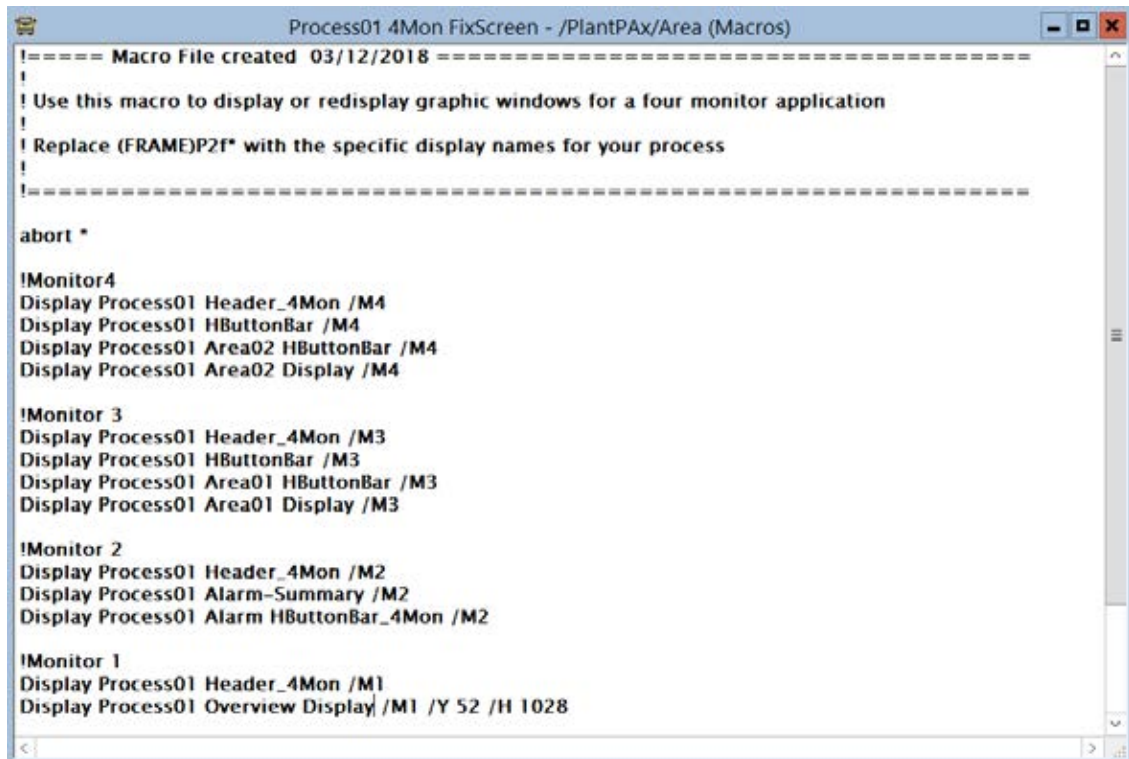


6. Close the Process01 1Mon FixScreen macro.
7. Right-click 'P2f Template 4Mon FixScreen' and choose Duplicate.



A window appears on the Studio canvas to choose a name.

8. Rename the display 'Process01 4Mon FixScreen' and click OK.
9. Open the Process01 4Mon FixScreen macro.
10. Update the macro as shown in the example.



11. Close the Process01 4Mon FixScreen macro.
- Your customization of the HMI template is complete.

Configure HMI Security

This section describes how to configure security to assign HMI privileges to your staff members. Without security privilege, personnel cannot access faceplates for specific areas of the plant.

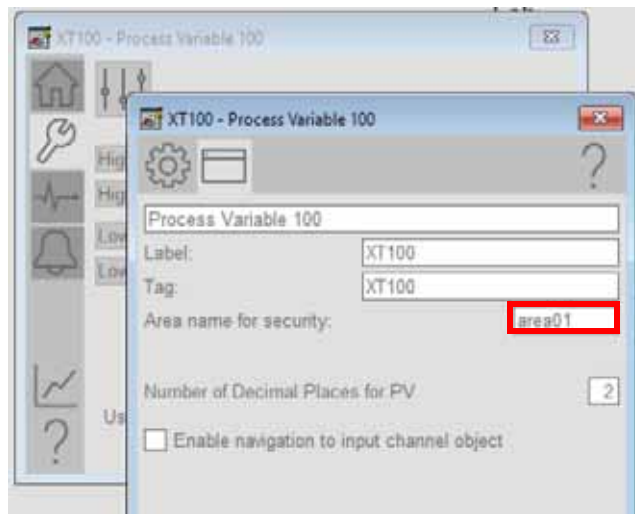
IMPORTANT Before performing the following procedures, you must have assigned user groups with roles to specific plant areas. See Chapter 6 in the Infrastructure User Manual, publication [PROCES-UM001](#).

Configure Object Area

Add-On Instructions for the objects include a parameter for a configurable object area (tag parameter Cfg_Area), which can be used to assign the object to an area of the facility. Only users with the privileges for the assigned area can operate or configure the device. For example, an engineer in Area 1 cannot modify pump attributes in Area 2, unless assigned security for Area 2.

Configure Security Area

Each Library faceplate has a configurable 'Area Name for Security' value, which is used to assign the object instance to an area of the facility.



To make an area assignment, you must do the following:

- Change the area string size (optional)
- Configure the Cfg_Area tag parameter
- Configure the area by using a PlantPAx tool

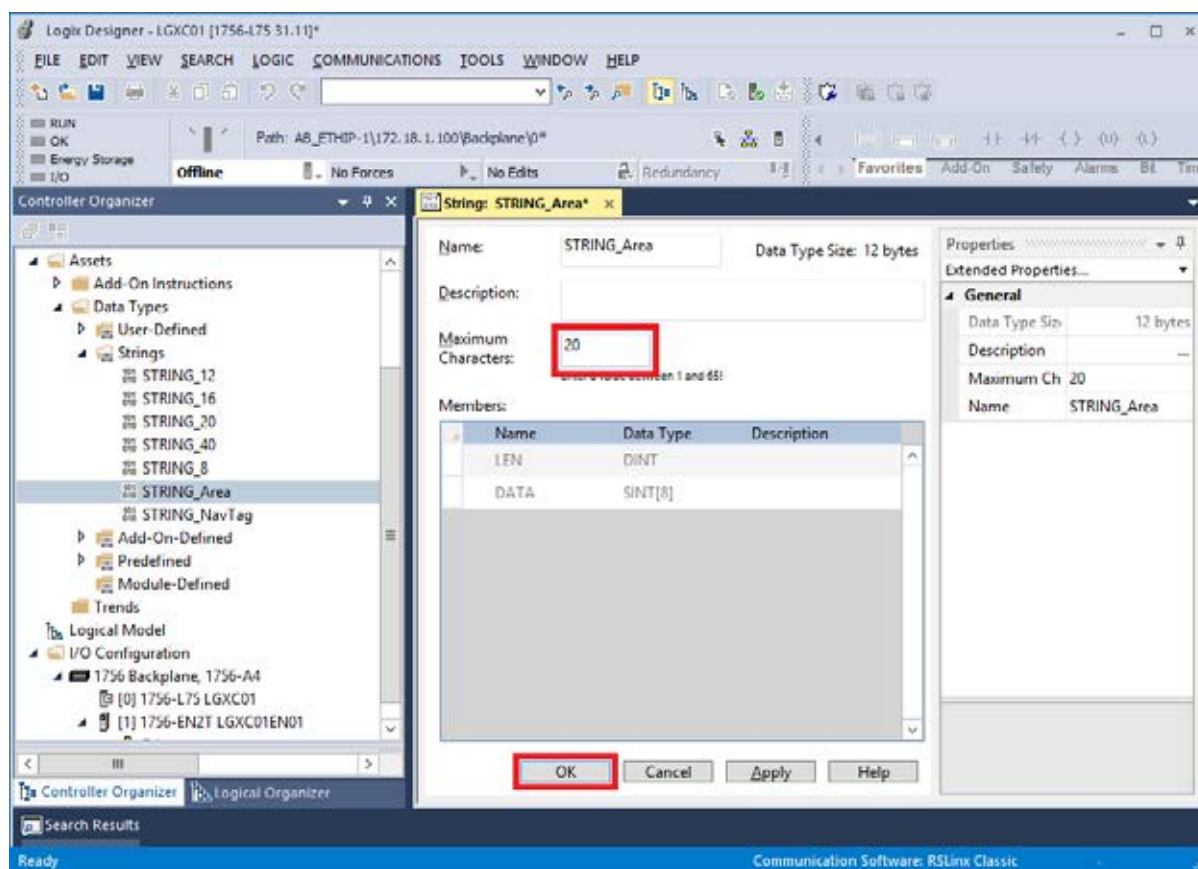
Each task is explained in the following pages.

Changing Area String Size

You can change the default area string size to accommodate your system naming convention. For example, we name our domain 'System.PlantPax.local' where the NetBIOS™ is SYSTEM. We need to increase the string size to contain the number of characters in the area name, such as SYSTEM\Area1.

IMPORTANT The string area name depends on whether you are using a domain. In the example, 'System' is a child domain of parent domain PlantPax. If you are not using a domain, your HMI could be named Area 1. For details, see Chapters 3 and 6 in the PlantPax Distributed Control System Infrastructure Configuration User Manual, publication [PROCES-UM001](#).

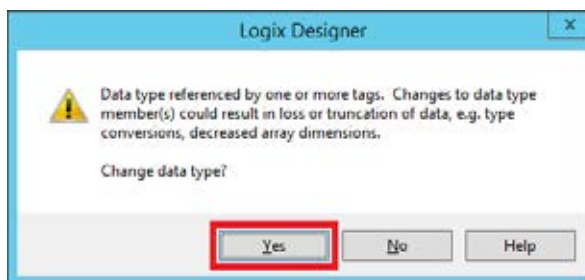
1. In the Logix Designer Controller Organizer, expand Assets and click Data Types>Strings>String Area.



2. Type a value for the maximum characters.
The example shows 20 characters.
3. Click OK.

IMPORTANT Changing the definition of a string type clears the values of all strings of that type. Decide on string sizes early in the project.

A message window appears.



4. Click Yes.

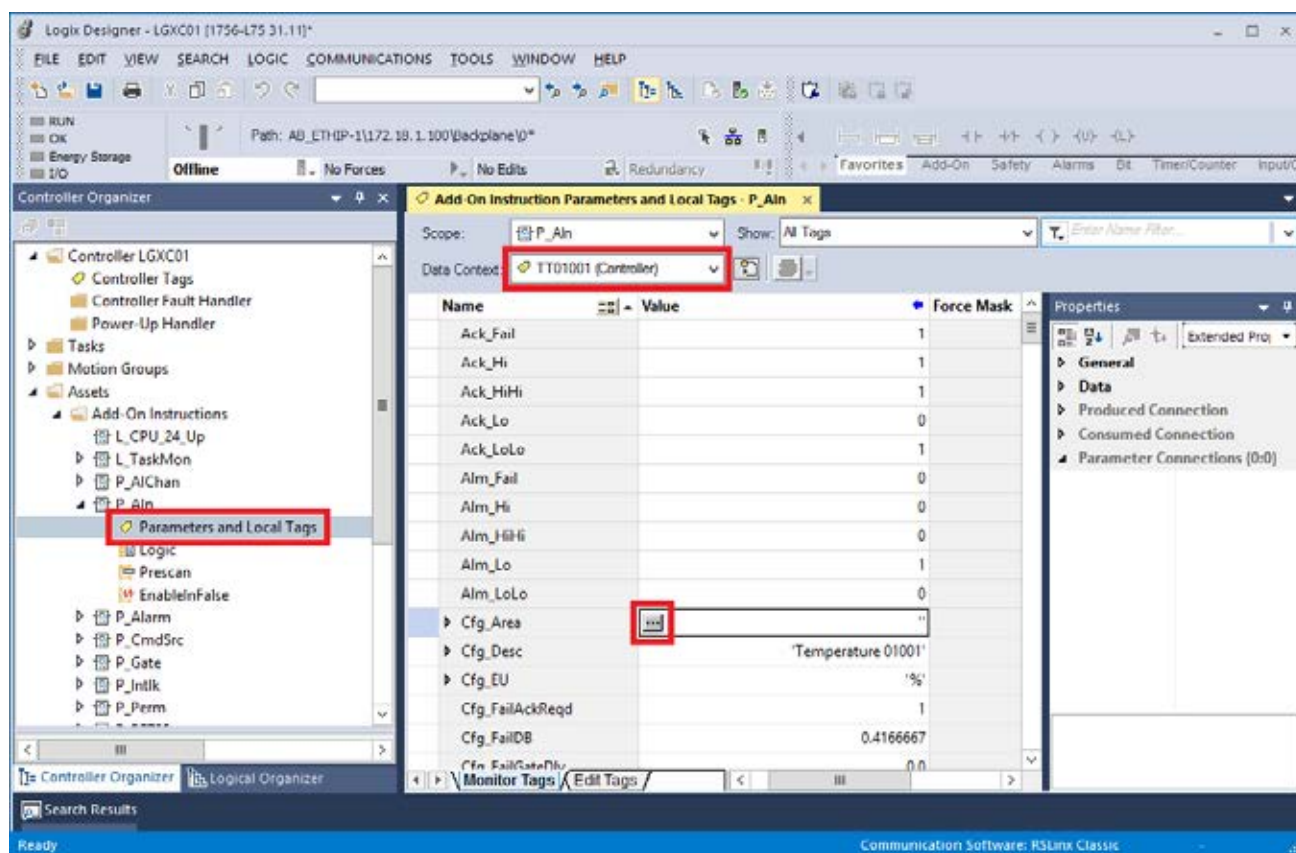
Configuring Area Manually

This section describes how to manually configure an object tag parameter (Cfg_Area) by using Logix Designer to modify tags in the controller application file (ACD file).

1. In the Controller Organizer, expand Assets and click Add-On Instructions>P_AIn.

Our example is P_AIn, but the procedure applies for any Add-On Instruction that requires area string modification.

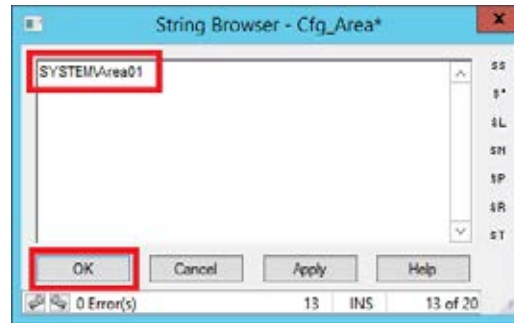
2. Click Parameters and Local Tags to open the Add-On Instruction.



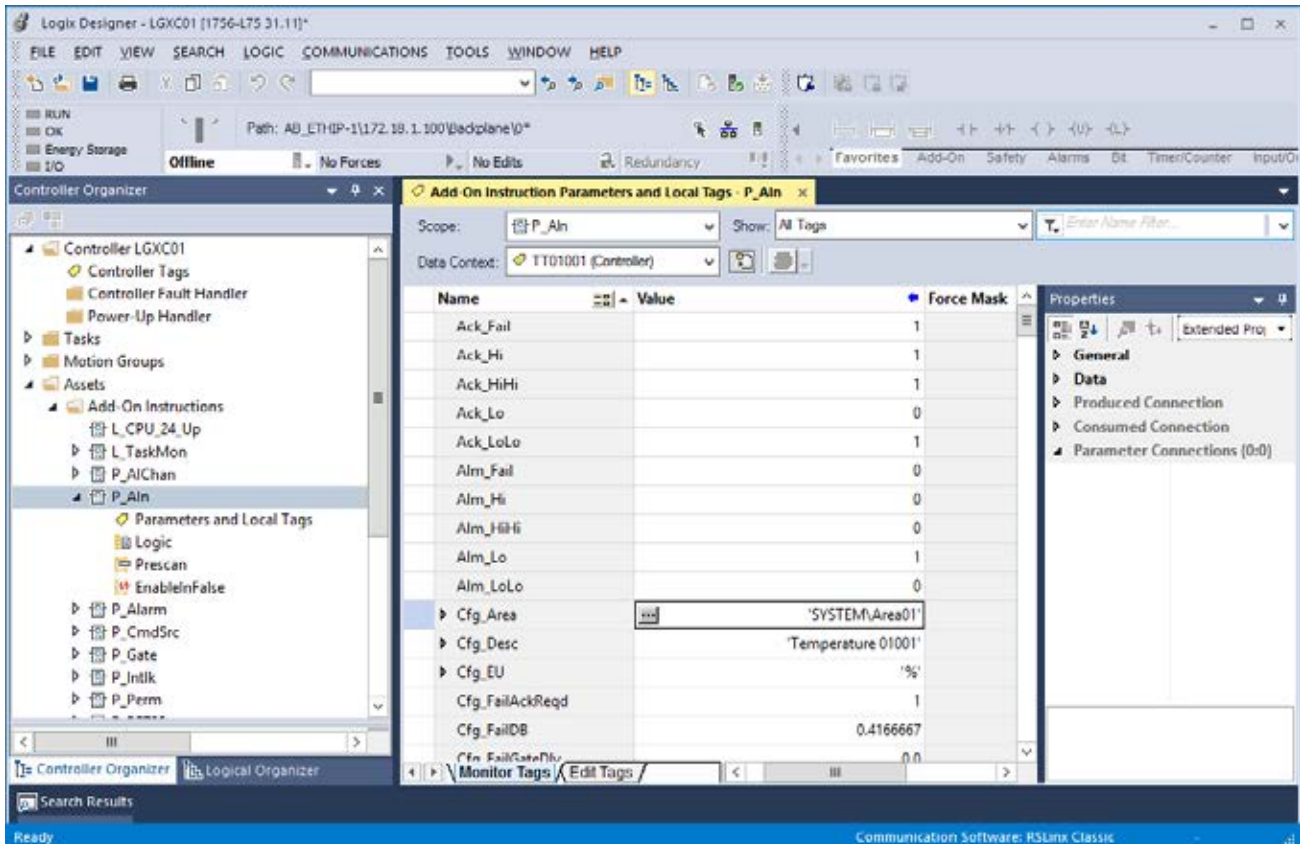
3. From the Data Context pull-down, choose an object.

The object attributes appear.

4. Click the value column for Cfg_Area to access a Browse ('ellipse' ...) button.
5. Click Browse to open the String Browser window.
6. Type your area name and click OK.



The area name appears in the Cfg_Area Value field.



7. Repeat [step 2](#) through [step 6](#) for all objects in your system that require the area string to be modified.

For larger systems, a configuration tool is provided to modify area strings. See [Using the PlantPAx Tool to Change Area on page 73](#).

Using the PlantPAx Tool to Change Area

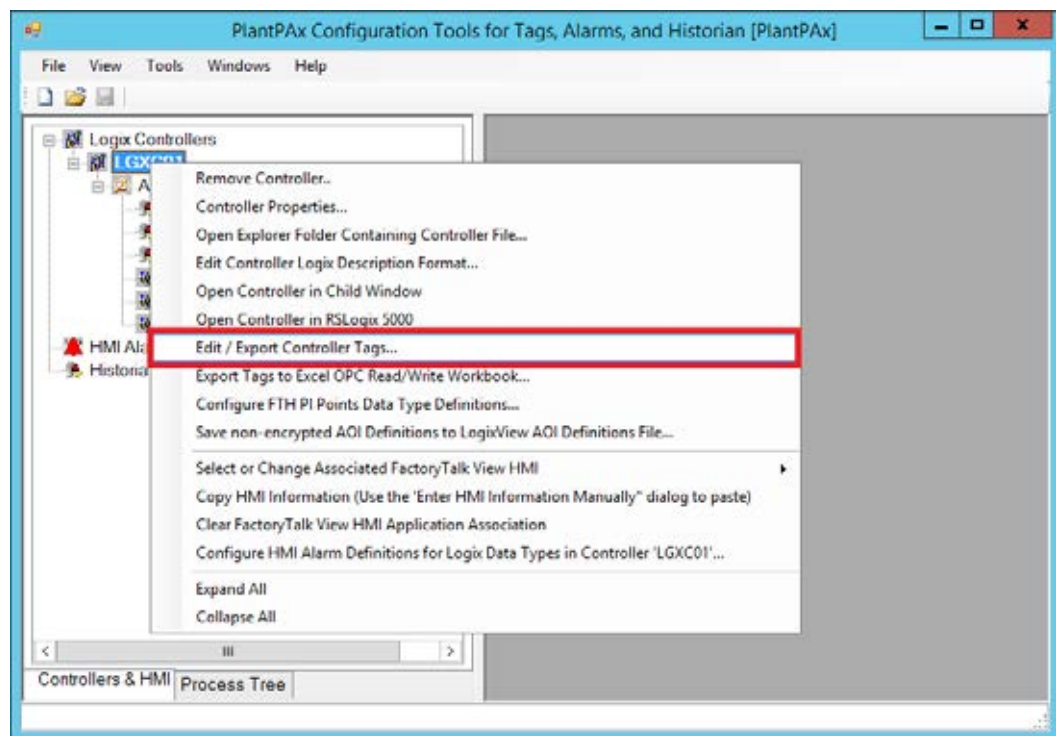
The Rockwell Automation Library of Process Objects contains the 'PlantPAx Configuration Tools for Tags, Alarms, and Historian' software. The PlantPAx Configuration Tool can be used to configure area strings for large systems with multiple objects. This software tool is part of the Rockwell Automation Library of Process Objects download at

<http://www.rockwellautomation.com/rockwellautomation/support/downloads.page>.

1. Download and install the tool on the EWS.
2. Open the tool.

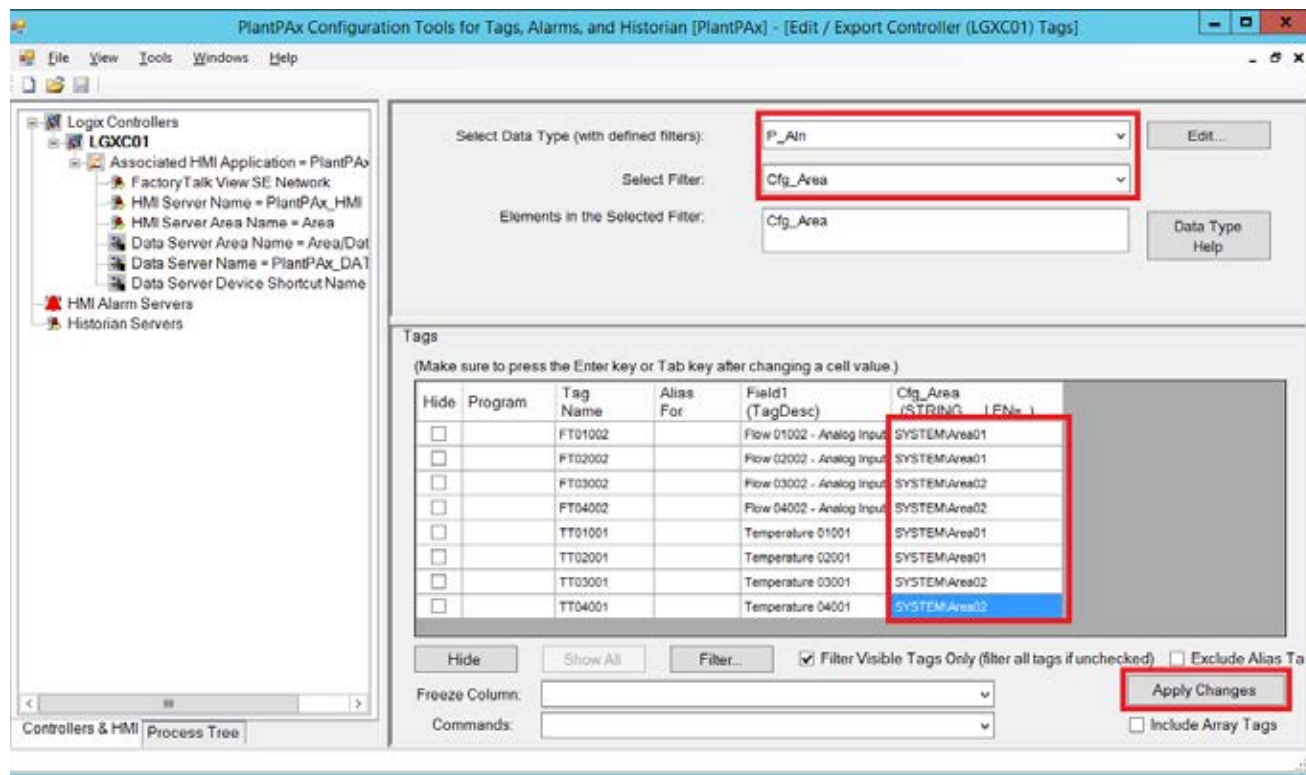
Follow the documentation supplied with the tool to import the controller file into the tool.

3. Right-click a Controller and choose Edit/Export Controller Tags.

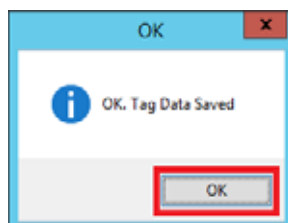


4. From the Data Type pull-down, select a data type.
5. From the Filter pull-down, select Cfg_Area.

Tags that associated with the selected data type appear.

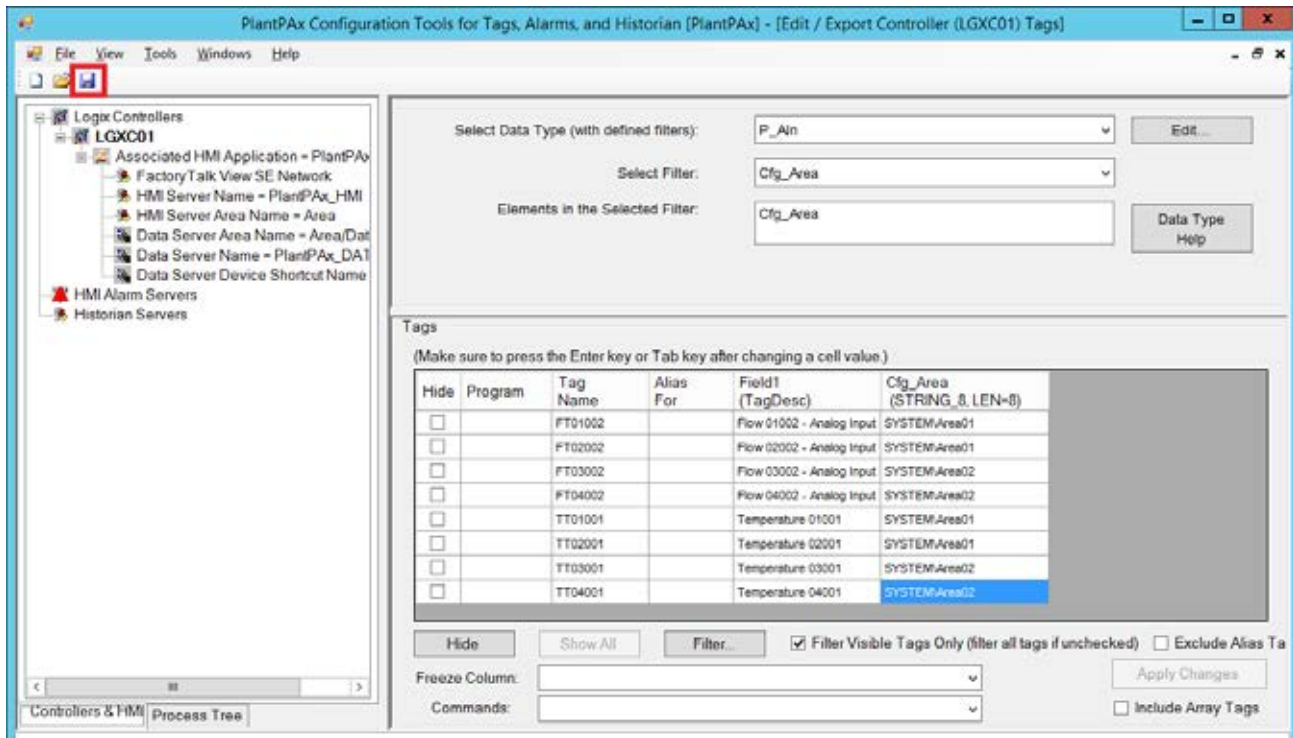


6. For each tag, type the area name for the object.
 7. Click Apply Changes.
- A message window appears.
8. Click OK.

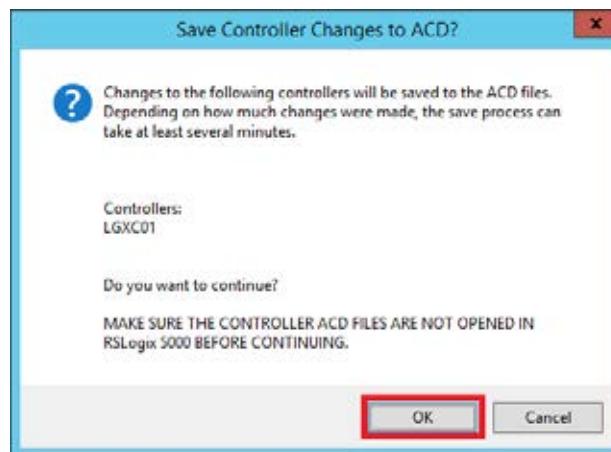


9. Repeat [step 4](#) through [step 8](#) for each data type in your system.

10. Click Save.

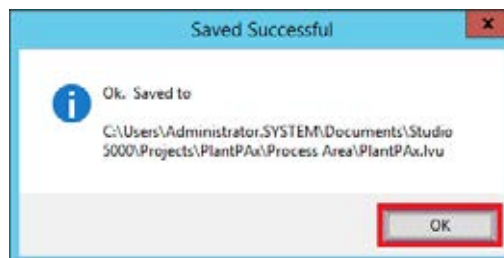


A message window appears.



11. Click OK.

Be patient while the tool updates the controller .acd file.

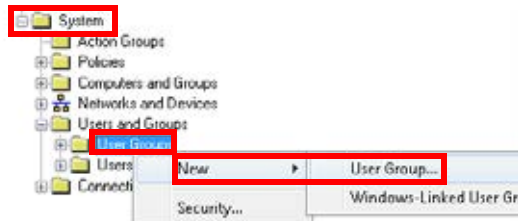


12. Click OK.

Create User Groups

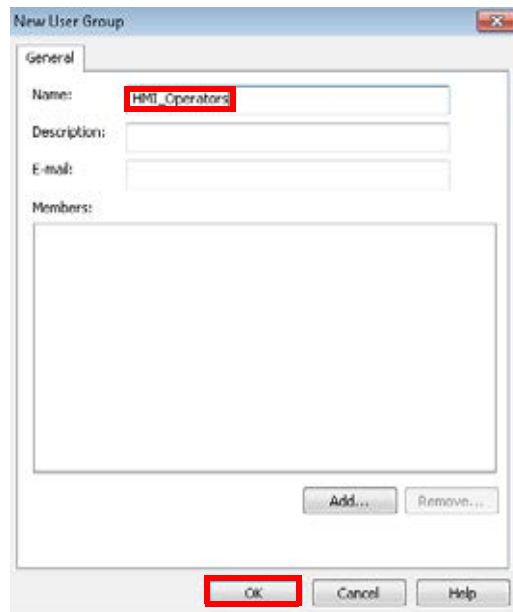
This section describes how to create user groups.

1. From the FactoryTalk View Studio, click '+' to expand System and click Users and Groups.
2. Right-click User Groups and choose New>User Group.

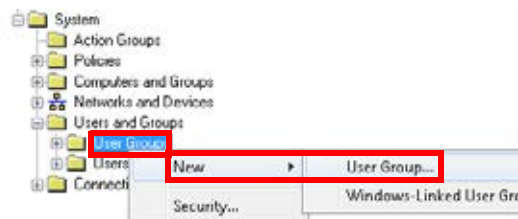


3. Name the user group HMI_Operators and click OK.

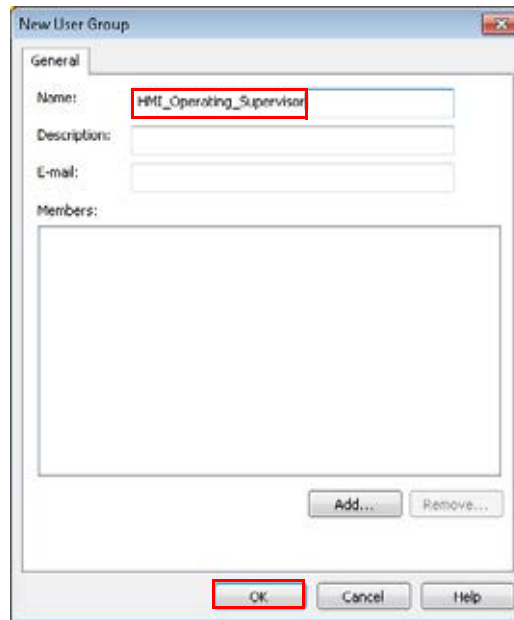
Later we assign the security code A (only) to this user group. See [page 79](#) for procedures.



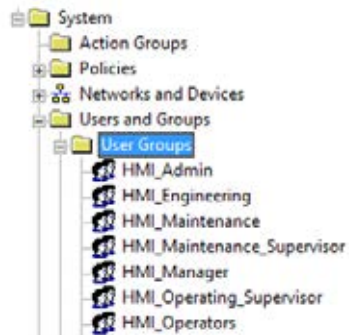
4. Right-click User Groups (again) and choose New>User Group.



5. Name the user group HMI_Operating_Supervisor and click OK.



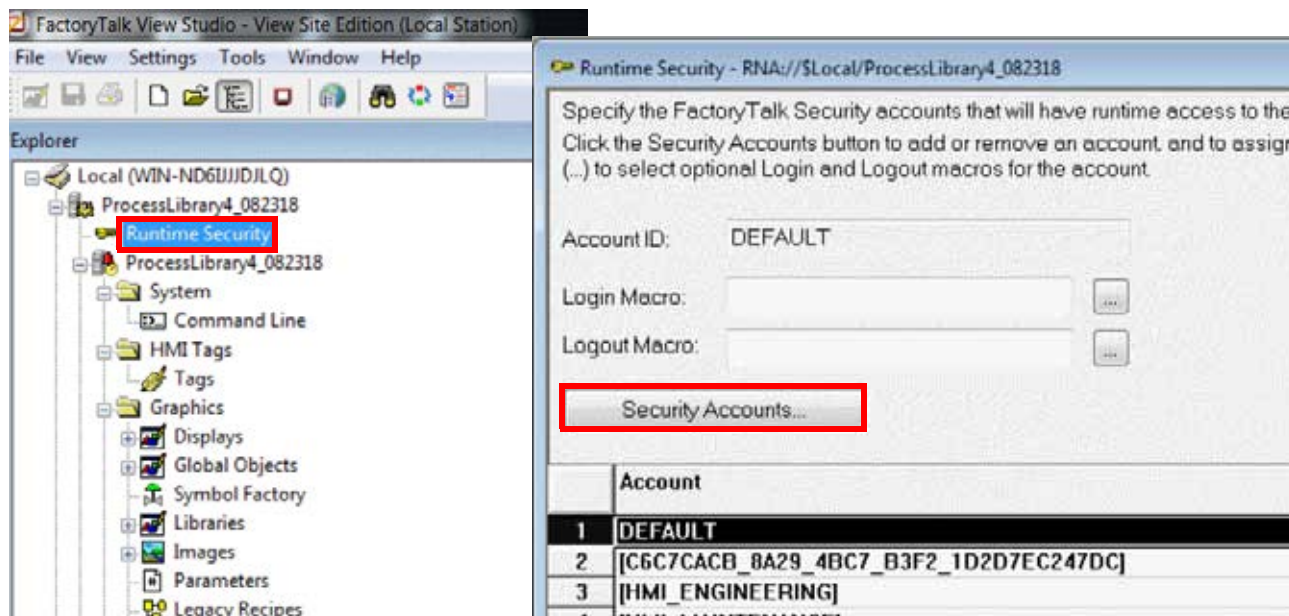
6. Repeat [step 2](#) and [step 5](#) to add the following five user groups:
- HMI_Maintenance
 - HMI_Maintenance Supervisor
 - HMI_Engineering
 - HMI_Manager
 - HMI_Admin
7. Verify that the user groups shown are added.



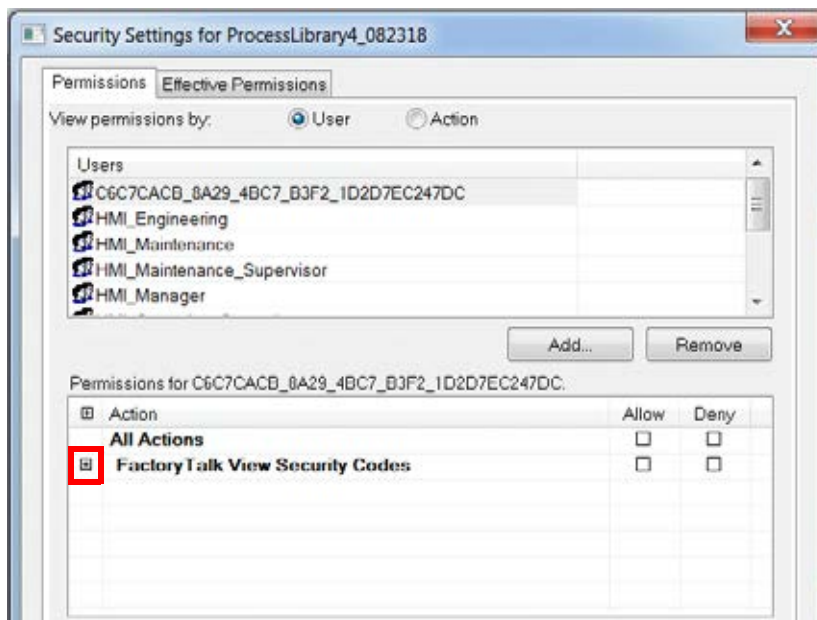
Assign Security Codes to User Groups

This section describes how to assign security codes to user groups. Runtime security is organized based on the role of a user. Each role is assigned a security letter (A ...G).

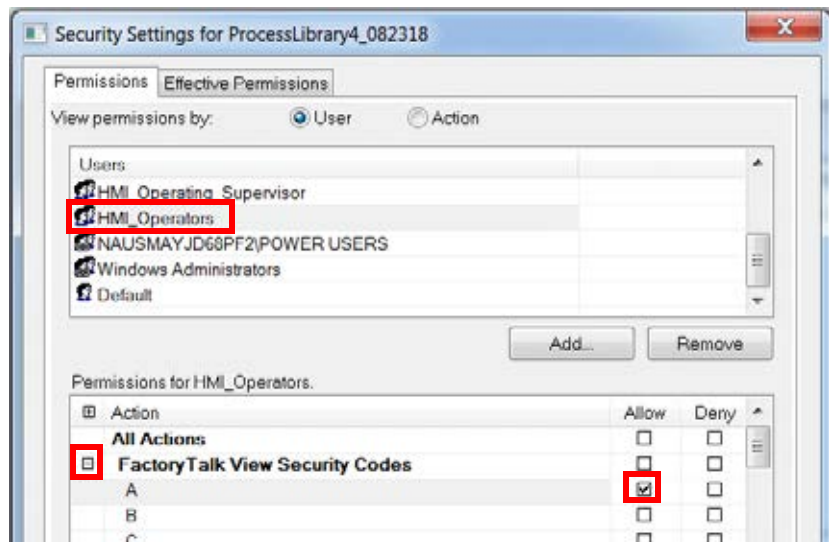
1. From the FactoryTalk View Studio, double-click Runtime Security and choose Security Accounts.



2. To expand the FactoryTalk View Security Codes, click the '+':



3. Scroll and select the User 'HMI_Operators'.
4. Expand the FactoryTalk View Security Codes, then click the Allow check box for security code A.



5. Click OK.
6. Repeat [step 1](#) though [step 5](#) to assign the security codes to the following six user groups:
 - HMI_Operating Supervisor (code B only)
 - HMI_Maintenance (code C only)
 - HMI_Maintenance Supervisor (code D only)
 - HMI_Engineering (code E only)
 - HMI_Manager (code F only)
 - HMI_Admin (code G only)

List of Security Codes

FactoryTalk View software security codes help protect information that is contained within the HMI faceplates. Operators, maintenance personnel, and engineers must have security permissions to modify their respective faceplate tabs.

[Table 2](#) references the security codes that are recommended for each tag.

Table 2 - Security Codes

FTView Tag	Description	Security Codes
Security\AlarmAck	Acknowledge/Reset Alarms	ABCDEF
Security\AlarmConfig	Alarm Configuration	E
Security\AlarmDisable	Disable Alarms	BCDEF
Security\AlarmShelve	Shelve Alarms	ABCDEF
Security\BypassFeedback	Can Bypass Feedback	CDE
Security\BypassInterlocks	Bypass Permissives and Interlocks	BCDEF

Table 2 - Security Codes

FTView Tag	Description	Security Codes
Security\CmdSrcMaint	Acquire/Release Equipment Maintenance Command Source	CDE
Security\CmdSrcOperProg	Acquire/Lock and Release Equipment Operator Command Source	BCDEF
Security\CmdSrcOutOfService	Can put device in/out of service	CDE
Security\ConfigSecurity	Change Security for the device	E
Security\DeviceConfigBehavior	Change the setup of the device (Advanced)	E
Security\DeviceConfigDiagnostics	Configure device diagnostics	CDE
Security\DeviceConfigFailTimers	Modify Alarm Delay Times	DE
Security\DeviceConfigHMI	Change the configuration of the HMI interface for the device	E
Security\DeviceConfigLimits	Configure device limits	DE
Security\DeviceConfigThresholds	Modify Limits and Deadbands	BCDEF
Security\DeviceConfigTimers	Configure device timers	DE
Security\DeviceConfigTuning	Change Tuning; Inflight; and Preacts	DE
Security\EnableSimulation	Put Device in Simulation	E
Security\EnterOperSettings	Enter Setpoints and Control Variables	ABCDEF
Security\OperateEquipment	Command Equipment in Operator Command Source	ABCDEF
Security\OverrideInputs	Override Inputs	CDE
Security\OverrideOutputs	Override Outputs	CDE
Security\ProcedureAdvancedExceptions	Exception Processing (Step Change; Parameter Change; Acquire; Reorder; Activate)	BCDEF
Security\ProcedureChangeParameters	Override Downloaded Phase Parameters	BCDEF
Security\ProcedureChangeSetpoints	Override Downloaded Setpoints	BCDEF
Security\ProcedureControl	Select; run; hold; and restart Procedures; Sequences; and Batches	ABCDEF
Security\ProcedureEquipmentControl	Manual Supervisory EP/EM Control	BCDEF
Security\ProcedureExceptions	Exception Processing (Resume; manual; Auto; Semi-Auto; Pause; Disconnect; Release)	ABCDEF
Security\ProcedureForceSequence	Force Steps/States	BCDEF
Security\ProcedureManualControl	Manual Procedure; Sequence; and Batch Processing (Stop; Abort; Reset)	ABCDEF
Security\ResetAccumulators	Reset Run Time Accumulators	CDE
Security\RespondToPrompts	Respond to Prompts	ABCDEF
Security>ShowFaceplate	Navigate to full faceplate	CDE

Security Code Differences from Version 3.5 to Version 4.0 (or Later)

In the previous versions of the library (version 3.5 and earlier), the faceplates used all the security codes 'A...P'. The drawback with using security codes A...P in the faceplates was that there were no available spare codes for customer use to customize their security model. Also, the security codes had to be assigned per user.

Starting with the Rockwell Automation Library of Process Objects, version 4.0 (or later), release, each object (and faceplate) has a configurable 'Area Name for Security' value. The configurable value can be used to assign the object (or faceplate) to a specific process area of the facility. This functionality frees up most of the 'H...P' codes for customer assignable security and makes assigning users to groups much easier.

With version 4.0 (or later), the user groups are assigned the 'roles' (the appropriate security codes) by using new Security tags. Users are assigned to groups. The new security tags can be imported into the version 4.0 (or later) application from a library supplied import file.

The Rockwell Automation Library of Process Objects, version 3.5, used the following user groups. Each group used various security codes (A...P):

- HMI_ Operators (various codes)
- HMI_ Operating Supervisor (various codes)
- HMI_ Maintenance (various codes)
- HMI_ Maintenance Supervisor (various codes)
- HMI_ Engineering (various codes)
- HMI_ Manager (various codes)
- HMI_ Admin (various codes)

For the version 4.0 (or later) Area-based Security feature (runtime security), the Rockwell Automation Library of Process Objects, version 4.0 (or later), recommends the same seven User Groups per HMI. However, each group is assigned only a **single security code**:

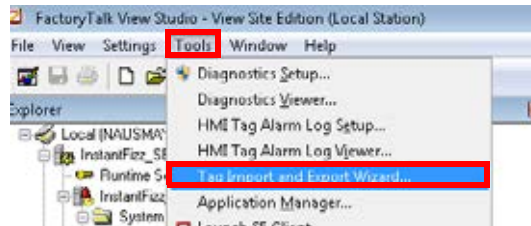
- HMI_ Operators (code A)
- HMI_ Operating Supervisor (code B)
- HMI_ Maintenance (code C)
- HMI_ Maintenance Supervisor (code D)
- HMI_ Engineering (code E)
- HMI_ Manager (code F)

IMPORTANT It is a requirement to have at least one area configured within your projects with basic and advanced user assignments. If you do not add {Area}_Basic and {Area}_Advance groups you have view only access to the HMI faceplates. The area name must match the .Cfg_Area tag string value in the Add-On Instruction instance for security to function correctly.

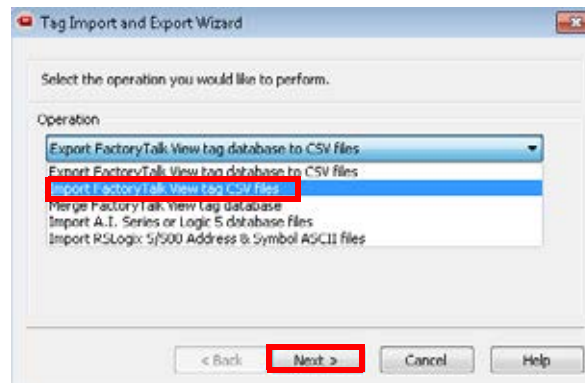
Import Area-based HMI Tags

Complete these steps to import the HMI tags (.CSV file) that is provided with the Process Library download.

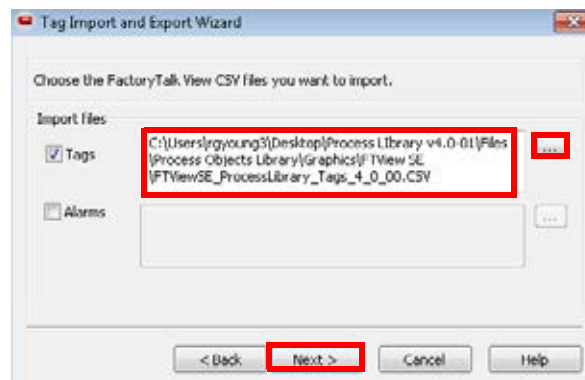
1. From FactoryTalk View Studio main menu, click Tools and choose Tag Import and Export Wizard.



2. From the Tag Import and Export Wizard, select Import FactoryTalk View tag CSV files from the pull-down, and click Next.

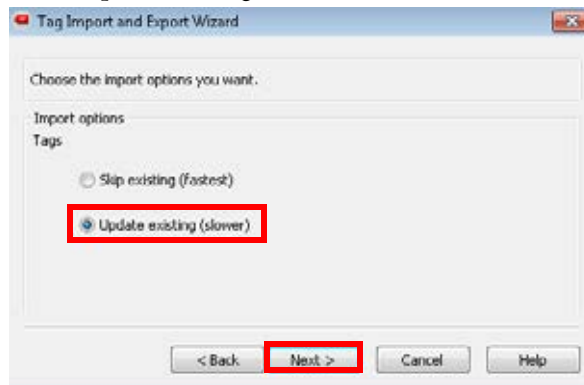


3. Choose the Project Type and then click the ellipsis ('...') to navigate to the folder that contains the HMI project.
4. Click Next.
5. Click the ellipsis ('...') and navigate to the folder with the .CSV file with the Process Library download.

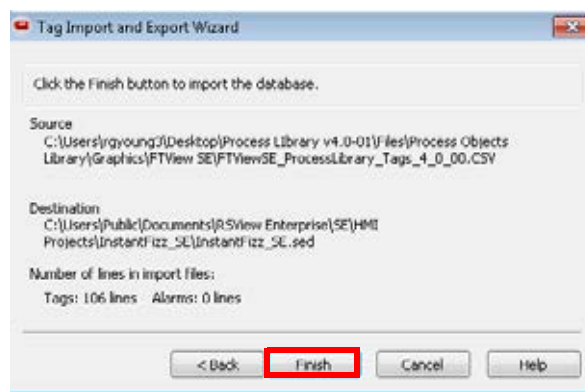


6. Click Next.

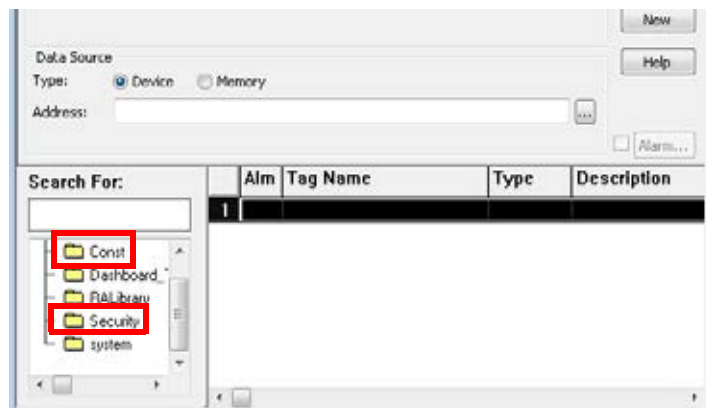
7. Select Update existing (slower) and click Next.



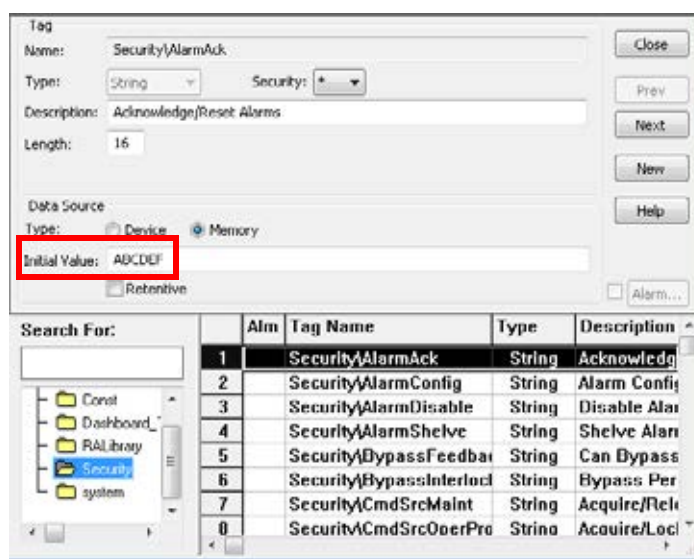
8. Click Finish.



9. After importing the tags, double-click Tags (from the HMI tree) and verify that two HMI folders exist: Const and Security.



TIP Using the A...P codes assigned to the HMI user groups, you can determine which groups have permission for each security task. Add or remove that group's security code in the Initial Value field of the corresponding HMI tag. By importing the tags, all (library download) recommended assignments have been configured.



Before changing initial values to customize your security on HMI faceplates, when the HMI server is started, it loads the initial value of the memory tag into memory. But when changes are made to the initial value in FactoryTalk View Studio, those changes only go to the hard disk drive. To get the newly edited initial value into memory, the HMI server needs to be restarted. Do the following:

- Use the SE Service Manager utility, which stops and restarts the HMI server services. This action forces the HMI server to load the new initial value into memory.
- Create a display (or use an existing display), and draw a numeric/string input and tie it to the tag in question. Enter the new value and press Enter to force the value to update.
- Restart the computer.

For more information, see the Knowledgebase Answer ID 54147 at <https://rockwellautomation.custhelp.com/app>.

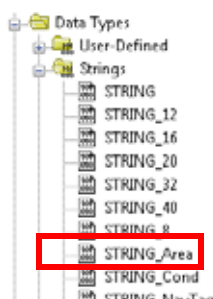
For each object instance in your Logix code, you assign an area using the tag member, Cfg_Area, as shown. Two user groups must be created in the FactoryTalk directory for each area. One named '{cfg_Area}_Advanced' and one named '{cfg_Area}_Basic'. These groups are used to grant or deny permissions on the faceplates for these objects.

The Rockwell Automation Library of Process Objects, version 4.0, added one new controller tag per Object (Cfg_Area):

Data Context: MyP_Motor (Controller) [1]								
Name	Usage	Value	Force Mask	Style	Data Type	Description	Constant	
+ Cfg_Area	Local	*Area01*	{...}		STRING_Area	Process Area for security	☐	

Data Context: MyP_PF755 (Controller) [1]								
Name	Usage	Value	Force Mask	Style	Data Type	Description	Constant	
+ Cfg_Area	Local	*Area02*	{...}		STRING_Area	Process Area for security	☐	

When configuring security for your application, if your area name exceeds eight characters, you must increase the string size of STRING_Area found under Data Types>Strings.



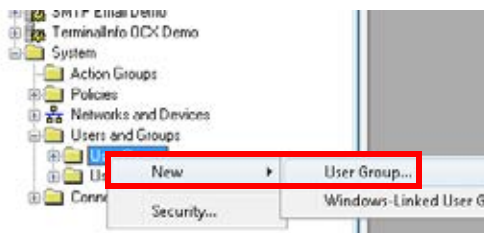
A screenshot of the configuration dialog for 'STRING_Area'. The 'Name' field contains 'STRING_Area'. The 'Description' field is empty. The 'Maximum Characters' field is set to 8. Below this is a 'Members' table with two rows: 'LEN' with Data Type 'DINT' and 'DATA' with Data Type 'SINT[8]'. The 'Data Type Size' is noted as 12 byte(s).

Name	Data Type	Style	Description	External Access
LEN	DINT	Decimal		Read/Write
DATA	SINT[8]	ASCII		Read/Write

Create Area User Groups

Complete these steps to create area user groups for multiple security areas of a production facility.

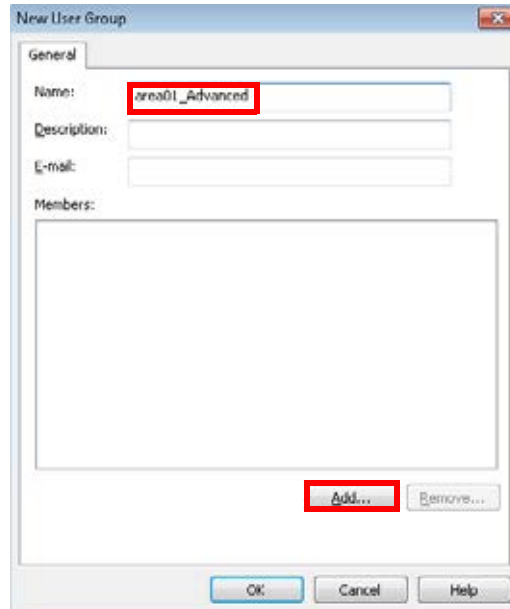
1. From the FactoryTalk Administration Console, click '+' to expand System and then click '+' to expand Users and Groups.



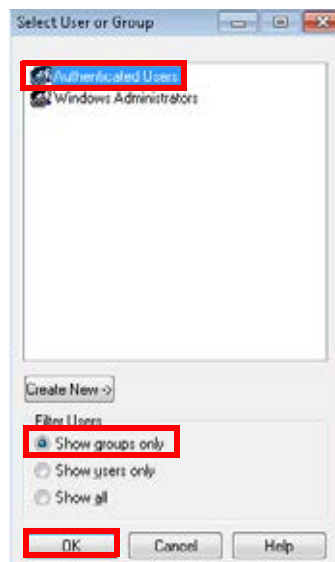
2. Right-click User Groups and choose New>User Group.

On the New User Group dialog box, you must add two groups: 'area01_Advanced,' 'area01_Basic.' These groups define which Area01 Users have basic functions on the faceplate or advanced functions (engineering, maintenance).

3. To add groups, type the name (example, area01_Advanced) and click Add.

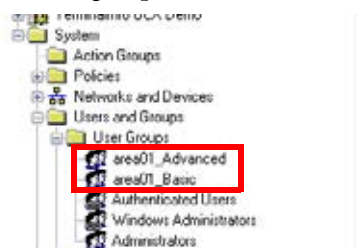


4. Select Authenticated Users and use the default 'Show groups only' and click OK.

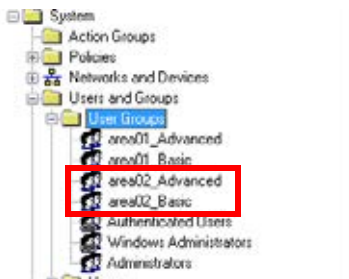


5. Click OK again.

Your two groups for area01 look like the example.



6. Repeat [step 3](#) and [step 4](#) to add two groups for Area02 ('area02_Advanced', 'area02_Basic').

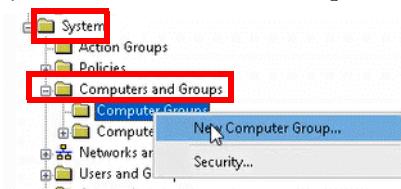


Line of Sight Functionality

The procedures for this functionality, which requires a distributed system, is outlined on [page 36](#). This section describes how to add a desired computer to a group ('Computer Group') and to an area of the plant.

IMPORTANT The files 'NavToFaceplate with line of sight' and 'NavToQuick with line of sight' must be renamed to replace 'NavToFaceplate' and 'NavToQuick'.

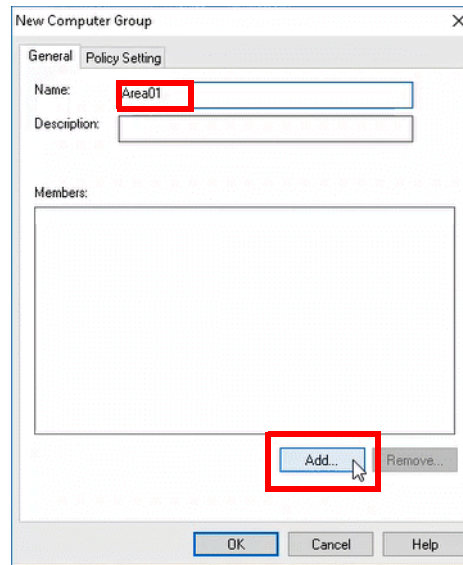
1. From the FactoryTalk View Distributed application, click '+' to expand System and then click '+' to expand Computers and Groups.



2. Right-click Computer Groups and choose New Computer Group.

3. Type the area name and click Add.

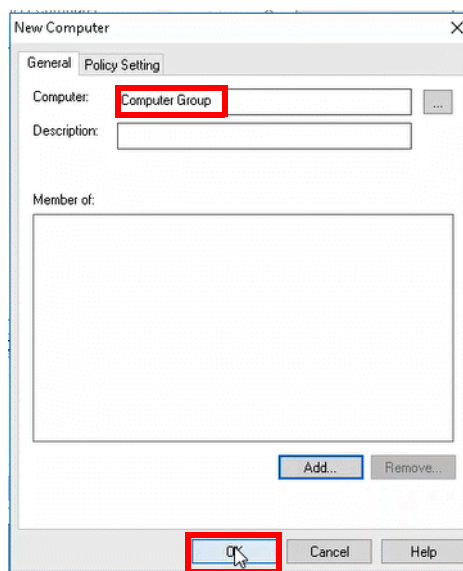
The area name is the same name as the area name configured in the controller. The computer area name does not have the 'Basic' or 'Advanced' suffix.



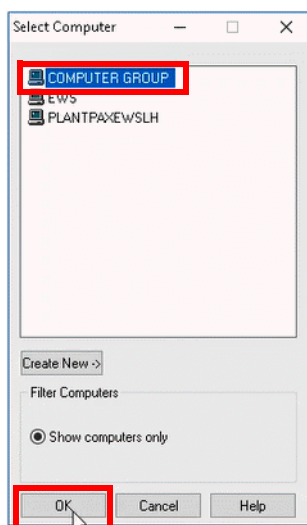
4. On the Select Computer window, click Create New and choose Computer.



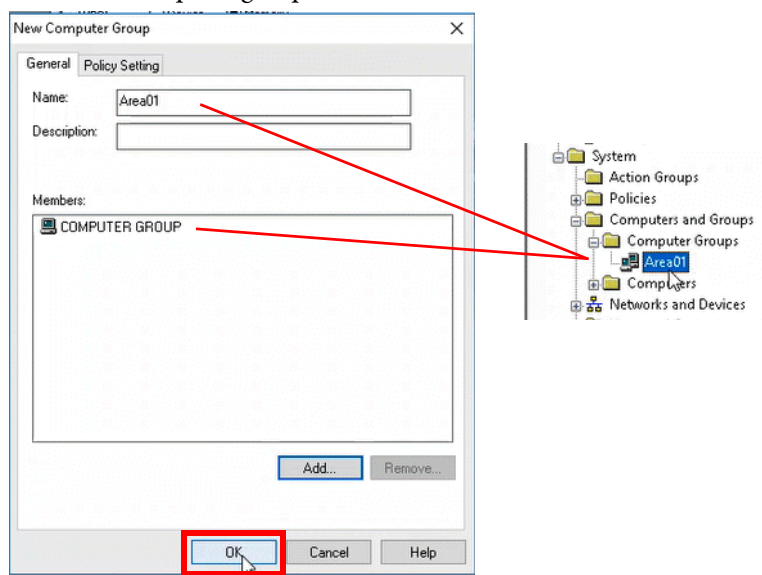
5. Type a desired computer name and click OK.



6. Select the name of the desired computer and click OK.



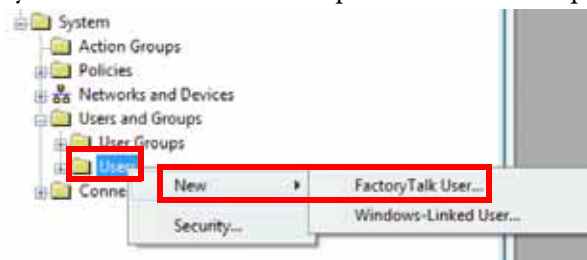
7. To add the computer group to the area, click OK.



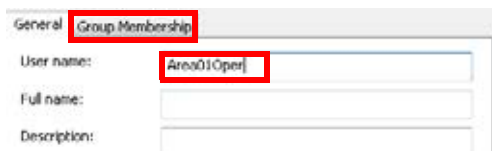
Create FactoryTalk Users

Complete these steps to assign a user to one or more HMI security groups and to any Area groups that they can have access.

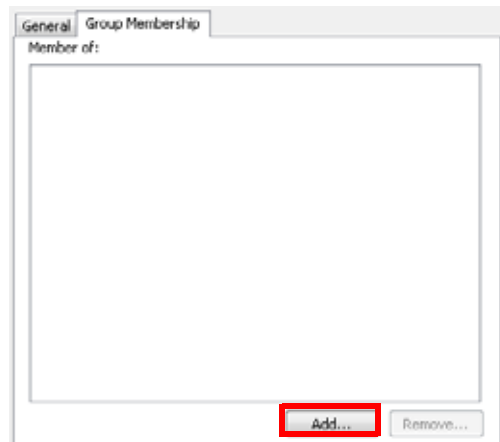
1. From the FactoryTalk Administration Console, click '+' to expand System and then click '+' to expand Users and Groups.



2. Right-click Users and choose New>FactoryTalk User.
3. Type Area01Oper in the User Name text box and click the Group Membership tab.

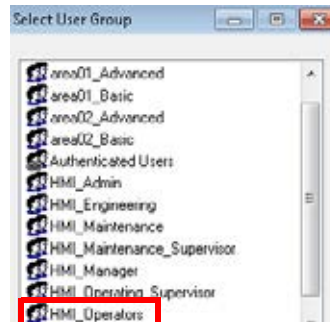


4. Click Add.

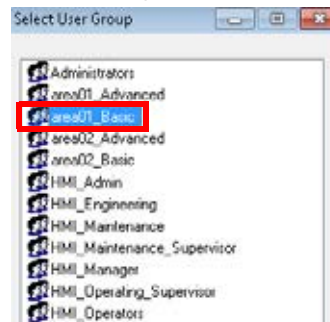


With multiple process areas defined (example Area01, Area02), each user must be assigned to the HMI_{group} (example HMI_Operator) and the 'area' group. Operators can be assigned to the HMI_Operator group and also to the Area01_Basic group. This limits operator access to the faceplate operator controls only.

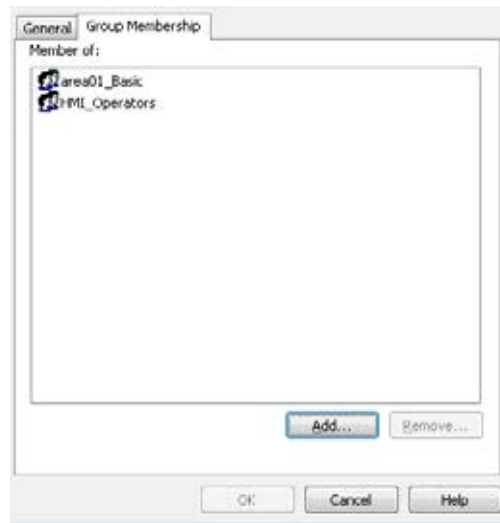
5. For our example, select the group that is named HMI_Operators and click OK.



6. Click Add again, and select the group that is names 'area01_Basic'.



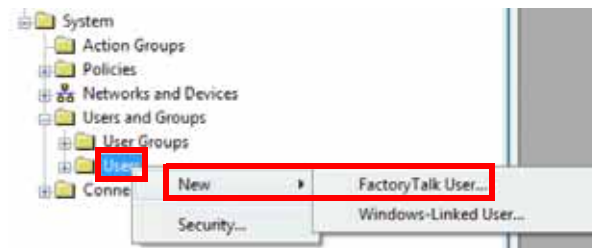
7. Click OK.



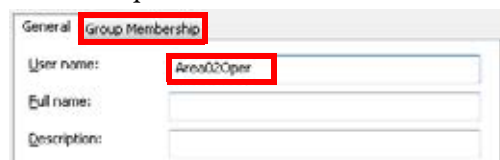
8. Click OK again.

We repeat the same steps to add an operator for Area02.

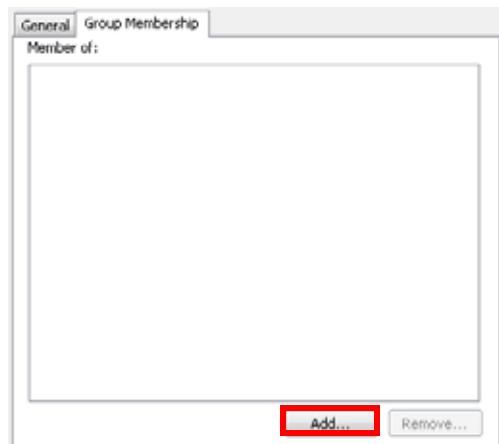
1. From the FactoryTalk Administration Console, click '+' to expand System and then click '+' to expand Users and Groups.



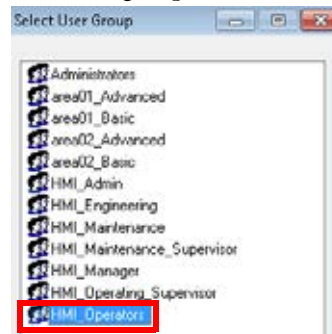
2. Right- click Users and choose New>FactoryTalk User.
3. Type Area02Oper in the User Name text box and click the Group Membership tab.



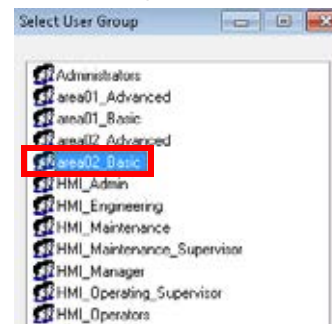
4. Click Add.



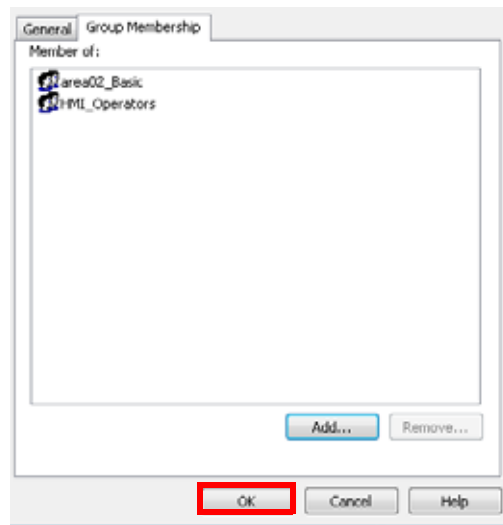
5. Select the group that is named HMI_Operators and click OK.



6. Click Add again, and select the group that is names 'area02_Basic'.

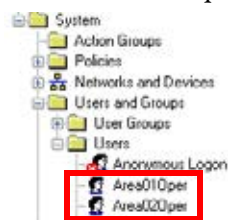


7. Click OK.



8. Click OK again.

You now have an operator that is assigned to Area01 and Area02.



Follow these exact steps to assign **each** HMI_{group} and the 'area' group.

For example, HMI_Engineer and Area01_Advanced. Advanced provides engineers access to the advanced engineering features on the faceplates. If an engineer is to be able to do operator tasks, an engineer needs to be a member of both groups: 'Basic' and 'Advanced'.

Security Code Differences for FactoryTalk View ME Groups

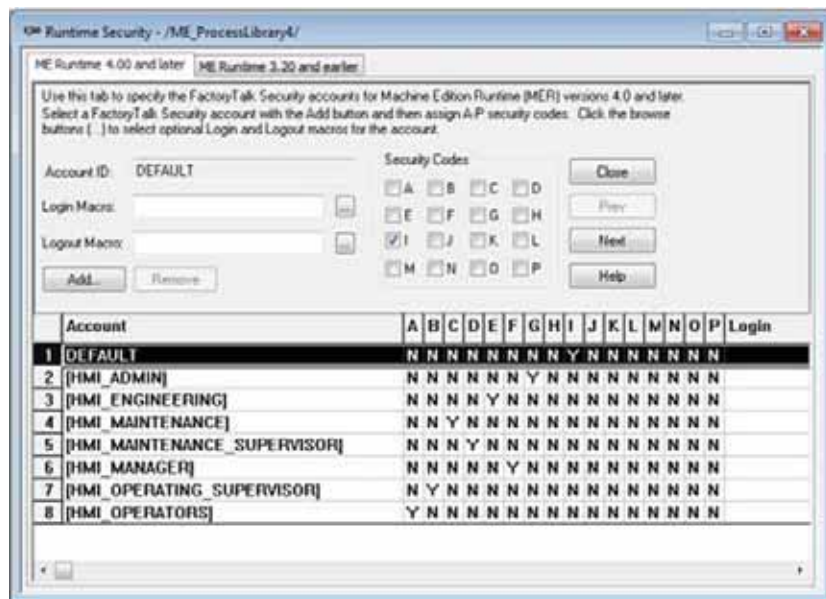
Area-based security is typically not needed on a PanelView™ terminal. Security is located on an OEM skid in the area and doesn't access other areas. In previous Process Library releases, the engineering functions were enabled from the A...P codes, and available to the 'default' user. You could power up the skid, including the PanelView terminal, and be ready to operate the skid without any login credentials.

With the release of Version 4.0 (or later) for the Rockwell Automation Library of Process Objects, at least one area is required to be configured. Since there is no ability in FactoryTalk View Machine Edition to automatically log in a user, the customer has to know a valid UID and password to operate their unit. Thus, you must log into the PanelView system at least one time when a skid has been powered on.

Refer to Knowledgebase Answer ID 617466 at <https://www.rockwellautomation.custhelp.com>.

FactoryTalk View Machine Edition, version 8.0, changes how security codes are applied across multiple groups. Security codes are now ANDed across groups instead of being ORed.

Thus, you cannot assign a user to multiple groups that has A...P codes configured. As stated in this document, we do not specify that the {Area}_Basic and {Area}_Advanced have A...P codes. These groups cannot be included in the 'Runtime Security' list. By doing it this way, one does not have the problem of assigning a user to multiple {Area} groups. The illustration shows an example of 'Runtime Security'.



Create an OWS Client File

Use an Engineering Workstation or Operator Workstation with these procedures

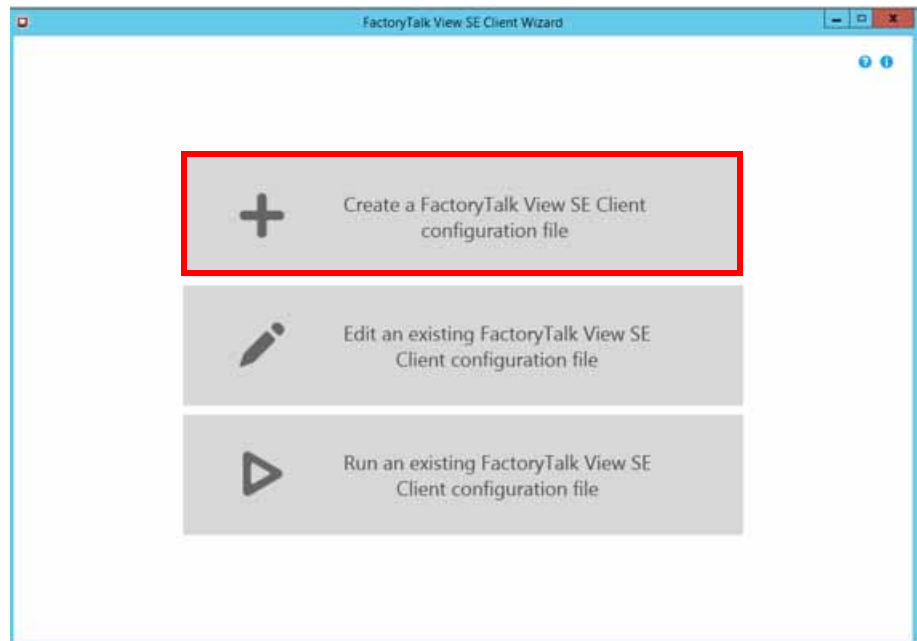


In this section, you create a FactoryTalk View SE client file. Client files are used to launch a FactoryTalk View software client from the Operator Workstation (OWS). Once created, this client file can be copied to each OWS.

IMPORTANT Our procedure shows how to create the client file on an OWS. This procedure also can be used on an EWS.

Complete the following steps.

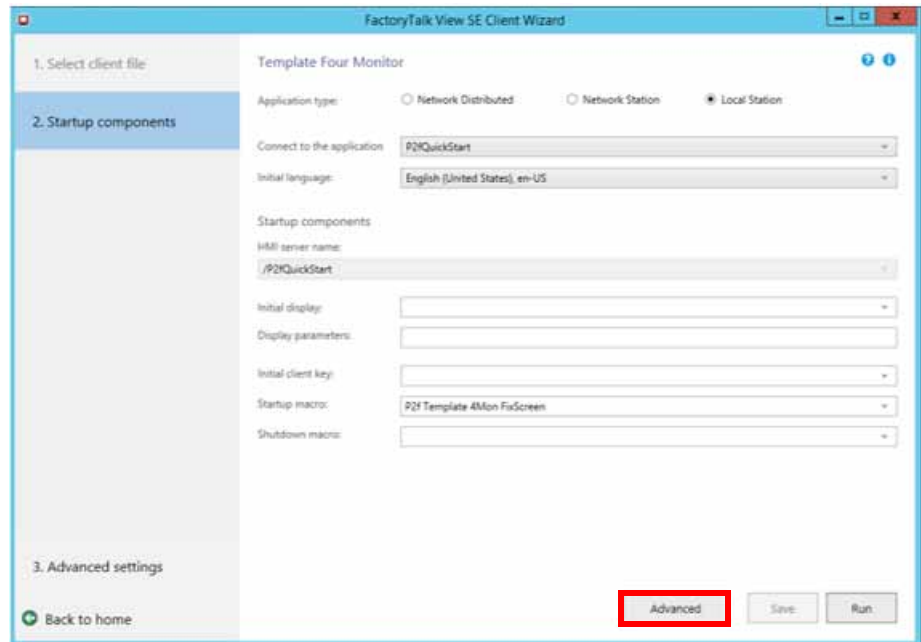
1. Click the Windows Start button and locate the FactoryTalk View Site Edition Client application. Click the application to launch the client wizard.
2. Click the application icon  to open the client wizard.
3. To create a client file, click Create a FactoryTalk View SE Client configuration file.



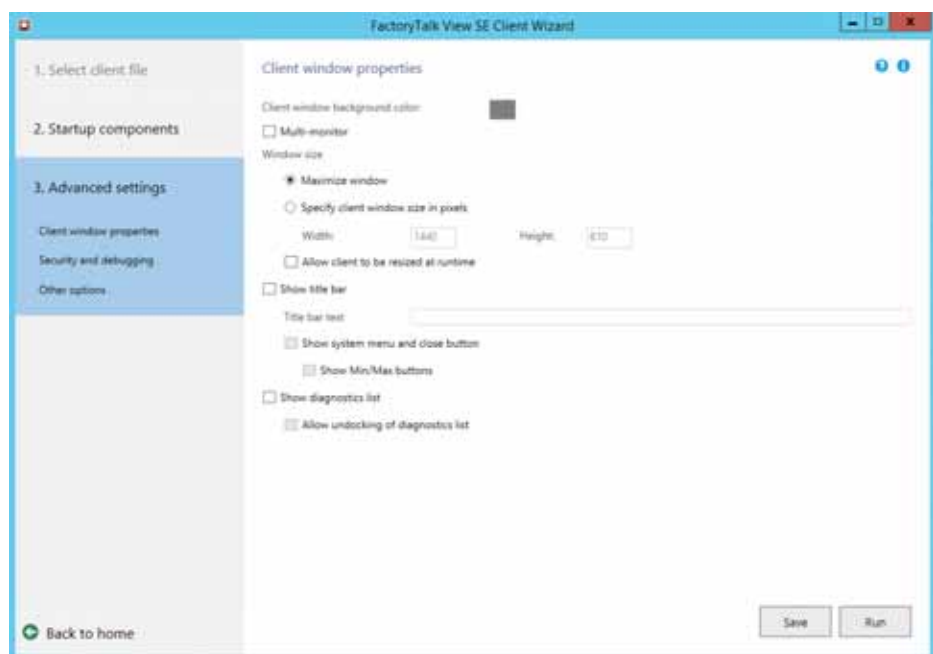
4. Type a client file name and location.
5. Click Continue.



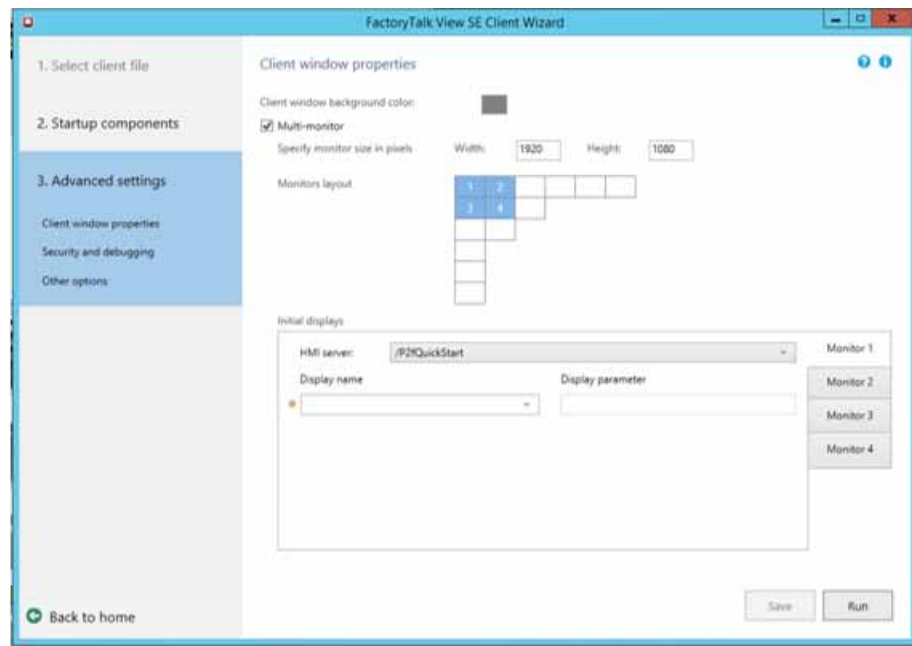
6. In the startup components section, do the following:
 - a. Select an application type
 - b. Select an application name from the pull-down menu
 - c. Select a language from the pull-down menu
 - d. Select an HMI server name from the pull-down menu
 - e. From the Startup macro pull-down menu:
 - If a single monitor display, select the single monitor macro
 - If a 4Mon display, select the 4 monitor macro



7. Click Advanced.
8. Make the selections as shown.



9. Click Save **only** if you are using a single monitor.
10. If you select the multi-monitor option, specify the monitor size and layout.



IMPORTANT You can configure the starting displays for each monitor by using the configuration wizard rather than a startup macro.

11. Click Save.

Common Configuration Considerations

This chapter includes programming considerations that are common for Process Objects.

The following table describes the topics in this chapter.

Topic	Page
Library Programming Considerations	99
Command Source Configuration	103
Alarm Considerations	103
Alarm with FactoryTalk Alarms and Events Server	111
Alarm with FactoryTalk View ME Software	118
Global Object Configuration	125
Maintain Library Releases	127
Customize the Library	128

Library Programming Considerations

Multiple programming languages are available in Logix controllers. Based on the type of application that you are creating, the Add-On Instruction logic can be used with Ladder Diagram, Function Block Diagram, and Structured Text languages.

Review the following programming language examples for the P_Motor instruction. Use whatever language is 'best' for your application.

Figure 4 - Ladder Diagram

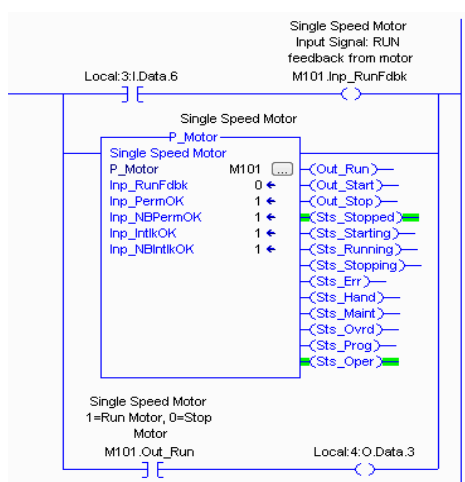


Figure 5 - Function Block Diagram

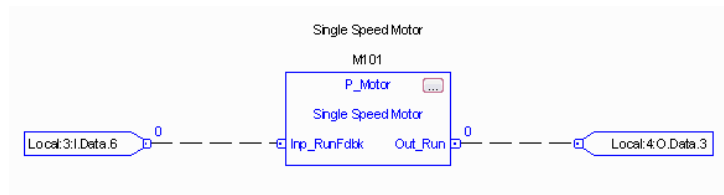


Figure 6 - Structured Text

```

M101.Inp_RunFdbk := Local:3:I.Data.6;
P_Motor(M101);
Local:4:O.Data.3 := M101.Out_Run;

```

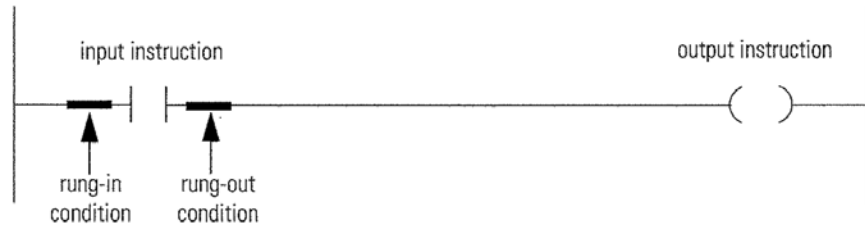
Ladder Diagram logic executes simple Boolean logic, timers, and counters the fastest. Function Block Diagrams and Structured Text can give you an advantage of the more advanced process and drives instructions available in those languages.

There are fundamental differences on how the different languages execute and are compiled. Nevertheless, Add-On Instruction execution time is the same regardless of language in which the instruction is used.

Ladder Diagram Considerations

Although multiple programming languages are available to be used with the Library Add-On Instructions, Ladder Diagram has differences in behavior to consider. The controller evaluates Ladder Diagram instructions that are based on the rung condition that precedes the instruction (rung-in condition).

Based on the rung-in condition and the instruction, the controller sets the rung condition following the instruction (rung-out condition), which affects any subsequent instruction.



If the rung-in condition to an input instruction is true, the controller evaluates the instruction and sets the rung-out condition based on the results of the instruction. If the instruction evaluates to true, the rung-out condition is true; if the instruction evaluates false, the rung-out condition is false.

IMPORTANT The rung-in condition is reflected in the EnableIn parameter and determines how the system performs each Process Add-On Instruction. If the EnableIn signal is **true**, the system performs the main logic routine of the instruction. Conversely, if the EnableIn signal is **false**, the system performs the EnableInFalse routine of the instruction.

The main logic routine for the instruction sets/clears the EnableOut parameter, which then determines the rung-out condition. The EnableInFalse routine cannot set the EnableOut parameter. If the rung-in condition is **false**, then the EnableOut parameter and the rung-out condition are also **false**.

Prescan

During the transition into Run mode, the controller performs a Prescan before the first logic scan. Prescan is a special scan of all routines in the controller. The controller scans all main routines and subroutines during Prescan, but ignores jumps that could skip the execution of instructions. The controller executes all FOR loops and subroutine calls. If a subroutine is called more than once, it is executed each time that it is called. The controller uses Prescan of built-in instructions to reset non-retentive data values.

During Prescan, input values are not current and outputs are not written. The following conditions generate Prescan:

- Toggle from Program to Run mode
- Automatically enter Run mode from a power-up condition

Prescan does not occur for a program when the following occurs:

- Program becomes scheduled while the controller is running
- Program is unscheduled when the controller enters Run mode

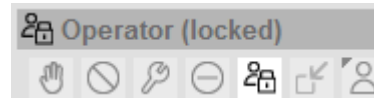
IMPORTANT The Prescan performs the logic routine for the Process Add-On Instruction as all **false** and then performs its Prescan routine as **true**.

TIP When hard-coding configuration bits in Ladder Diagram instances of Add-On Instructions, we recommend using OTL (output latch) instructions. The OTL instruction writes configuration bits to 1 and the OTU (output unlatch) instruction writes configuration bits to 0. Avoid using OTE (output energize) instructions for writing to Add-On Instruction bits. Because OTE is a non-retentive instruction, when the controller logic prescan occurs (on Powerup or controller Program to Run transition), the bit referenced in the OTE is cleared to zero.

Example: You want to force the configuration of a PID loop to enable cascade loop mode, and you want the loop to power up in cascade. You write code that sets the Cfg_HasCasc bit of the P_PIDE instance. The prescan logic of the Add-On Instruction examines this bit. If the loop does not have cascade loop mode configured, the request to power up in cascade is ignored. Suppose that you use an OTE instruction to hard-code Cfg_HasCasc to 1. The prescan of OTE, a non-retentive instruction, sets Cfg_HasCasc to 0 during prescan. The P_PIDE Add-On Instruction prescan sees Cfg_HasCasc as 0 and powers up in whatever loop mode it was in previously, not the desired cascade loop mode. By using an OTL instruction to write a 1 to hard-code this configuration, the result is the desired action: Powering up in cascade loop mode.

Command Source Configuration

Command Source indicators provide a visual reference to the current owner of the process device. The sources available are listed on the Operator tab, as shown in the example.



For descriptions of the symbols, see the Rockwell Automation Library of Process Objects: Display Elements, publication [PROCES-RM014](#).

Alarm Considerations

Effective alarm management is an important function of a process control system. This section describes how to use FactoryTalk® View Alarms and Events software to create alarms for library objects to help safeguard personnel and plant assets.

IMPORTANT Version 4.00 of the Rockwell Automation Library of Process Objects requires FactoryTalk View software version 10.00.01 or later.

The Library uses instances of a dedicated Add-On Instruction, P_Alarm, for alarm handling. For example, the P_Motor instruction uses four P_Alarm instances for the following four alarms:

- Fail to Start (used in the examples of this section)
- Fail to Stop
- Interlock Trip
- I/O Fault

The methods that are described in this section can be used to configure any alarm for objects in the Library of Process Objects. Recommended procedures for connecting the Add-On Instruction instance alarms to a HMI are presented in the following subsections:

- [Alarm with FactoryTalk Alarms and Events Server on page 111](#)
- [Alarm with FactoryTalk View ME Software on page 118](#)

The PlantPAx® Alarm Builder tool (see [Appendix B](#)) helps to streamline the digital alarm tag definition process and facilitates bulk configuration.

[Table 3](#) summarizes the P_Alarm alarm types that are used with the Library, and indicates which objects use the alarm.

Table 3 - P_Alarm Types by Library Objects

Alarm Type	Alarm Description	Library Objects
ActuatorFault	Raised if the Inp_ActuatorFault input is true. This alarm is provided for use by valves that generate a fault contact, such as actuator motor overload trip. P_ValveM0 only: If the actuator fault is configured as a shed fault, the Stop output to the valve is triggered and a reset is required to command the valve open or closed.	• P_ValveC • P_ValveM0
AnyReject	At least one input signal has been rejected because of any of the following: • It is outside the configured failure range. • It is a statistical outlier per the Modified Thompson Tau test. • It is outside of a user-defined deviation from the mean. • It has its Bad quality input bit set or its Source and Quality input indicates it has Bad quality. • It has a floating point value that is infinite or not a number (floating point exception).	• P_AlnMulti
CantStart	There are not enough motors that can be started to satisfy Number of Motors to Run.	• P_LLS
CantStop	There are not enough motors that can be stopped.	• P_LLS
DeviceFault	Raised when the Inp_DeviceFault input is true. This alarm is provided for use by devices that generate their own fault signal. If the device fault is configured as a shed fault, the device is commanded to State 0 and a reset is required to command the device to any other state.	• P_D4SD
Diff	Raised when the difference between the two input signals exceeds the configured high difference threshold.	• P_AlnDual
DriveFault	Raised when the drive detects a fault and sets its Faulted status bit. Check the Fault Code and description to determine the cause. A Reset of this object causes a Clear Fault command to be sent to the drive in an attempt to clear the fault.	• P_PF52x • P_PF753 • P_PF755 • P_VSD • P_PF6000 • P_PF7000
EqpFault	Raised when the Inp_CtrlEqpFault input is true, or when equipment feedback signals fail to track the commanded state of the equipment within the configured time. If an equipment fault is configured as a shed fault, the flow is stopped and a reset is required to resume flow.	• P_Dose

Table 3 - P_Alarm Types by Library Objects

Alarm Type	Alarm Description	Library Objects
Fail	Raised when any of the following is true: - The PV quality is bad - The Inp_PVBad input is true - The PV is outside the configured failure limits - The PV is infinite or not a number (floating point exception) - The raw or engineering unit range configuration is invalid	- P_Aln - P_AlnAdv - P_AlnHART
	Raised when the PV is flagged as Bad, or when the PV is flagged as Uncertain and Cfg_FailOnUncertain is 1. The PV can be configured to be flagged as Bad or Uncertain for the following reasons: - Its value has not changed for more than the configured Stuck PV time - It is outside the configured failure range thresholds for more than the Out of Range on-delay time - It is infinite or not a number (floating point exception) - Module Fault input is true - Channel Fault input is true - Out of Specification (measurement uncertain) input is true - Function Check (PV substituted at device) input is true - Maintenance Required input is true - There is a Configuration Error (see Sts_Err and the Err_ bits)	P_AlChan
	Raised when any of the following is true: - Both PV input values are outside the configured failure range thresholds - Both PV input values have bad quality or are infinite or not a number - Selected PV is infinite or not a number (floating point exception) - Raw or engineering unit range configuration is invalid	P_AlnDual
	Raised when any of the following is true: - Number of unrejected PVs is less than the configured required number of good PVs - Calculated PV is infinite or not a number (floating point exception) - Raw or engineering unit range configuration is invalid. A PV can be rejected if: - It is set not to be used (by Maintenance) - It is outside the configured failure range thresholds - It is infinite or not a number (floating point exception) - It has Bad quality - It has Uncertain quality and Cfg_RejectUncertain is true - It is an outlier, either because its deviation is outside the configured threshold from the mean or its deviation from the mean exceeds the Modified Thompson Tau statistical test	P_AlnMulti
	Raised when the device is commanded to a new state and the device feedbacks fail to confirm that the device reached the new state within the configured time (Cfg_FailT). If the Failure is configured as a shed fault, the device is commanded to State 0 and cannot be commanded to another state until reset.	P_D4SD

Table 3 - P_Alarm Types by Library Objects

Alarm Type	Alarm Description	Library Objects
Fail (continued)	Raised when the internal PIDE instruction reports an Instruction Fault. The PIDE instruction reports an Instruction Fault under any of the following conditions: • Process variable (PV) bad quality • Control variable (CV) bad quality • Hand feedback bad quality • Invalid span of PV. PVEUMax -PVEUMin • Setpoint below low setpoint limit or above high setpoint limit. The instruction uses the clamped value for SP • Limits invalid: SPLoLim < PVEUMin, SPHiLim > PVEUMax, or SPHiLim < SPLoLim. If SPHiLim < SPLoLim, the instruction limits the value using SPLLimit • Ratio below low ratio limit or above high ratio limit. The instruction uses the clamped value for Ratio • CV < 0 or CV > 100, CV below low CV limit, or CV above high CV limit when the Loop Mode is Manual and Manual Limiting is enabled. The instruction clamps the value for CV • Interlock CV is < 0 or > 100 and the Interlock CV is active. The instruction clamps the value for CV • Invalid CVEU span. The instruction uses a value of CVEUMax = CVEUMin. • CVLLimit < 0, CVHLimit > 100, or CVHLimit < CVLLimit. If CVHLimit < CVLLimit, the instruction limits CV using CVLLimit • CVRoCLimit < 0. The instruction disables rate of change limiting. • Feedforward < -100 or > 100. The instruction clamps the value for Feedforward • Hand Feedback < 0 or > 100. The instruction clamps the Hand Feedback value • Proportional gain < 0. The instruction uses a value of PGain = 0 • Integral gain < 0. The instruction uses a value of IGain = 0 • Derivative gain < 0. The instruction uses a value of DGain = 0 • Zero-crossing deadband < 0. The instruction disables zero-crossing deadband	• P_PIDE
	Raised when the valve is commanded to a new position and the device feedbacks fail to confirm that the valve reached each required position (see state diagram) within the configured time (Cfg_FailT). If the Failure is configured as a shed fault, the valve is commanded closed and cannot be opened until reset.	• P_ValveMP
FailToStart	Raised when the motor has and is using run feedback, an attempt is made to start the motor, and the run feedback does not indicate that the motor is running within the configured time. If Fail to Start is configured as a shed fault, the motor is stopped and a reset is required to start the motor.	• P_Motor • P_Motor2Spd • P_MotorRev • P_SMC50 • P_SMCFlex
	Raised when the drive has and is using run feedback, an attempt is made to start the drive, and the run feedback does not indicate that the drive is running within the configured time. If Fail to Start is configured as a shed fault, the drive is stopped and a reset is required to start the drive.	• P_PF52x • P_PF753 • P_PF755 • P_VSD • P_PF6000 • P_PF7000
FailToStop	Raised when the motor has and is using run feedback, an attempt is made to stop the motor, and the run feedback does not indicate that the motor stopped within the configured time.	• P_Motor • P_Motor2Spd • P_MotorRev • P_SMC50 • P_SMCFlex
	Raised when the drive has and is using run feedback, an attempt is made to stop the drive, and the run feedback does not indicate that the drive stopped within the configured time.	• P_PF52x • P_PF753 • P_PF755 • P_VSD • P_PF6000 • P_PF7000

Table 3 - P_Alarm Types by Library Objects

Alarm Type	Alarm Description	Library Objects
FullStall	Raised when the valve has and is using Open and/or Closed limit switches, an attempt is made to open or close the valve, and the limit switches indicate that the valve did not move from its original position at all within the configured time. P_ValveSO only: If Full Stall is configured as a shed fault, the valve is de-energized and a reset is required to command the valve to its energized position.	• P_ValveMO • P_ValveSO
Hi	Raised when the PV is above the High threshold. The threshold, deadband, gating, and timing are set in configuration.	• P_Aln • P_AlnAdv • P_AlnDual • P_AlnMulti • P_AlnHART
HiDev	Raised when the amount by which the PV exceeds the setpoint or reference is above the High Deviation threshold. The threshold, deadband, gating, and timing are set in configuration.	• P_AlnAdv • P_DBC • P_PIDE
HiHi	Raised when the PV is above the High-High threshold. The threshold, deadband, gating, and timing are set in configuration.	• P_Aln • P_AlnAdv • P_AlnDual • P_AlnMulti • P_AlnHART
HIHIDev	Raised when the amount by which the PV exceeds the setpoint or reference is above the High-High Deviation threshold. The threshold, deadband, gating, and timing are set in configuration.	• P_PIDE
HiRoC	Absolute value of PV rate of change above High Rate of Change limit. Limit, deadband, and severity are set in configuration.	• P_AlnAdv
HiRoCDec	Absolute value of PV rate of change above High Rate of Change limit and decreasing. Limit, deadband, and severity are set in configuration.	• P_DBC
HiRoCInc	Absolute value of PV rate of change above High Rate of Change limit and increasing. Limit, deadband, and severity are set in configuration.	• P_DBC
IntlkTrip	Raised when an interlock 'not OK' condition causes the output CV to be changed to the configured Interlock CV value or held at its last value. If interlocks are not bypassed, a bypassable interlock or a non-bypassable interlock 'not OK' condition initiates an interlock trip. If interlocks are bypassed, only a non-bypassable interlock 'not OK' condition initiates an interlock trip.	• P_AOut • P_PIDE • P_ValveC • P_AOutHART
	Raised when an interlock 'not OK' condition causes the device to transition from the On state or a pulsing state to the Off state. If interlocks are not bypassed, a bypassable interlock or a non-bypassable interlock 'not OK' condition initiates an interlock trip. If interlocks are bypassed, only a non-bypassable interlock 'not OK' condition initiates an interlock trip.	• P_DOut
	Raised when the motor is running and an interlock 'not OK' condition causes the motor to stop. If interlocks are not bypassed, a bypassable interlock or a non-bypassable interlock 'not OK' condition initiates an interlock trip. If interlocks are bypassed, only a non-bypassable interlock 'not OK' condition initiates an interlock trip.	• P_LLS • P_Motor • P_Motor2Spd • P_MotorRev • P_SMC50 • P_SMCFlex
	Raised when the motor is running, the optional trip function is used, and an interlock 'not OK' condition triggers the trip function to stop the motor. If interlocks are not bypassed, a bypassable interlock or a non-bypassable interlock 'not OK' condition initiates an interlock trip. If interlocks are bypassed, only a non-bypassable interlock 'not OK' condition initiates an interlock trip.	• P_MotorHO
	Raised when an interlock 'not OK' condition causes the device to transition from some other state to State 0. If interlocks are not bypassed, a bypassable interlock or a non-bypassable interlock 'not OK' condition initiates an interlock trip. If interlocks are bypassed, only a non-bypassable interlock 'not OK' condition initiates an interlock trip.	• P_D4SD

Table 3 - P_Alarm Types by Library Objects

Alarm Type	Alarm Description	Library Objects
IntlkTrip (Continued)	Raised when the drive is running and an interlock 'not OK' condition causes the drive to stop. If interlocks are not bypassed, a bypassable interlock or a non-bypassable interlock 'not OK' condition initiates an interlock trip. If interlocks are bypassed, only a non-bypassable interlock 'not OK' condition initiates an interlock trip.	• P_PF52x • P_PF753 • P_PF755 • P_VSD • P_PF6000 • P_PF7000
	Raised when the optional trip function is used and an interlock 'not OK' condition triggers the trip output to the valve. If interlocks are not bypassed, a bypassable interlock or a non-bypassable interlock 'not OK' condition initiates an interlock trip. If interlocks are bypassed, only a non-bypassable interlock 'not OK' condition initiates an interlock trip.	• P_ValveHO
	Raised when an interlock 'not OK' condition causes the valve to transition from some other position to the closed position. If interlocks are not bypassed, a bypassable interlock or a non-bypassable interlock 'not OK' condition initiates an interlock trip. If interlocks are bypassed, only a non-bypassable interlock 'not OK' condition initiates an interlock trip.+	• P_ValveMP
	Raised when the valve is energized and an interlock 'not OK' condition causes the valve to be de-energized. If interlocks are not bypassed, a bypassable interlock or a non-bypassable interlock 'not OK' condition initiates an interlock trip. If interlocks are bypassed, only a non-bypassable interlock 'not OK' condition initiates an interlock trip.	• P_ValveSO
	Raised when an interlock 'not O' condition occurs and the device is not in Position 1. The device can be configured to be commanded to Position 1 when an interlock trip occurs. If interlocks are not bypassed, a bypassable interlock or a non-bypassable interlock 'not OK' condition initiates an interlock trip. If interlocks are bypassed, only a non-bypassable interlock 'not OK' condition initiates an interlock trip.	• P_nPos
	Raised when an interlock 'not OK' condition causes the sequence to perform its configured interlock action. The sequence can: • Command the sequence to Stop; • Hold at the current step; • Transfer control back to the last step configured as an Interlock Fallback Step. If interlocks are not bypassed, a bypassable interlock or a non-bypassable interlock 'not OK' condition initiates an interlock trip. If interlocks are bypassed, only a non-bypassable interlock 'not OK' condition initiates an interlock trip.	• P_Seq
IOFault	Raised when the Inp_IOFault input is true. This input is used to indicate to the instruction that a communication failure has occurred for its I/O. If the I/O Fault is configured as a shed fault, the device is commanded to State 0 and cannot be commanded to another state until reset.	• P_D4SD
	Raised when the Inp_IOFault input is true. This input is used to indicate to the instruction that a communication failure has occurred for its I/O. If the I/O Fault is configured as a shed fault, the device is commanded Off and cannot be commanded to another state until reset.	• P_DOut
	Raised when the Inp_IOFault input is true. This input is used to indicate to the instruction that a communication failure has occurred for its I/O. If the I/O Fault is configured as a shed fault, the device can transition to the Faulted state and remain de-energized until reset.	• P_nPos
	Raised when the Inp_IOFault input is true. This input is used to indicate to the instruction that a communication failure has occurred for its I/O. If the I/O Fault is configured as a shed fault, the valve is commanded to Stop motion and cannot be commanded to either position until reset.	• P_ValveMO

Table 3 - P_Alarm Types by Library Objects

Alarm Type	Alarm Description	Library Objects
IOFault (Continued)	Raised when the Inp_IOFault input is true. This input is used to indicate to the instruction that a communication failure has occurred for its I/O. If the I/O Fault is configured as a shed fault, the device can transition to the Faulted state and remain de-energized until reset.	• P_ValveSO
	Raised when the Inp_IOFault input is true. This input is used to indicate to the instruction that a communication failure has occurred for its I/O. If the I/O Fault is configured as a shed fault, the motor is stopped and not permitted to start until reset.	• P_Motor • P_Motor2Spd • P_MotorRev • P_SMC50 • P_SMCFlex
	Raised when the Inp_IOFault input is true. This input is used to indicate to the instruction that a communication failure has occurred for its I/O. If the I/O Fault is configured as a shed fault, the drive is stopped and not permitted to start until reset.	• P_PF52x • P_PF753 • P_PF755 • P_VSD • P_PF6000 • P_PF7000
	Raised when the Inp_IOFault input is true. This input is used to indicate to the instruction that a communication failure has occurred for its I/O. If the I/O Fault is configured as a shed fault and the optional trip function is used, the trip output is triggered until reset.	• P_MotorHO • P_ValveHO
	Raised when the Inp_IOFault input is true. This input is used to indicate to the instruction that communication with the overload relay has failed. The device faceplate can show the I/O Source and Quality as communication failure flag a "Not Ready" diagnostic.	• P_E1PlusE • P_E3000vld • P_E30vld
	Raised when the Inp_IOFault input is true. This input is used to indicate to the instruction that a communication failure has occurred for its I/O. If the I/O Fault is configured as a shed fault, the output CV is set to the configured Interlock CV or held at its last value until reset.	• P_AOut • P_ValveC • P_AOutHART
	Raised when an interlock 'not OK' condition causes the valve to transition from some other position to the closed position. If interlocks are not bypassed, a bypassable interlock or a non-bypassable interlock 'not OK' condition initiates an interlock trip. If interlocks are bypassed, only a non-bypassable interlock 'not OK' condition initiates an interlock trip.	• P_ValveMP
Lo	Raised when the PV is below the Low threshold. The threshold, deadband, gating, and timing are set in configuration.	• P_Aln • P_AlnAdv • P_AlnDual • P_AlnMulti • P_AlnHART
LockFail	Raised when a device with a locking or sealing feature is commanded to a new position, but the lock/seal feedback failed to confirm the device unlocking before moving or failed to confirm the device locking after moving, within the time allowed. If the Lock Failure is configured as shed fault, the device can transition to the Faulted state and remain de-energized until reset.	• P_nPos
LoDev	Raised when the amount by which the PV exceeds the setpoint or reference is below the Low Deviation threshold. (Since the threshold is a negative number, this amount is the amount the PV falls below the setpoint or reference.) The threshold, deadband, gating, and timing are set in configuration.	• P_AlnAdv • P_DBC • P_PIDE
LoLo	Raised when the PV is below the Low-Low threshold. The threshold, deadband, gating, and timing are set in configuration.	• P_Aln • P_AlnAdv • P_AlnDual • P_AlnMulti • P_AlnHART
LoLoDev	Raised when the amount by which the PV exceeds the setpoint or reference is below the Low-Low Deviation threshold. (Since the threshold is a negative number, this amount is the amount the PV falls below the setpoint or reference.) The threshold, deadband, gating, and timing are set in configuration.	• P_PIDE

Table 3 - P_Alarm Types by Library Objects

Alarm Type	Alarm Description	Library Objects
MinGood	Raised when at least one input signal has been rejected, and the remaining unrejected signals are the minimum number that are configured as required for a good PV. This status / alarm is to warn you that the next input failure can result in a Bad PV quality.	• P_AlnMulti
MotorFault	Raised when the Smart Motor Controller detects a fault and sets its Faulted status bit. Check the Fault Code and description to determine the cause. Issuing a Reset of this object can cause a Clear Fault command to be sent to the Smart Motor Controller in an attempt to clear the fault.	• P_SMC50 • P_SMCFlex
NoneGood	Raised when neither PV input has good quality. This alarm is an indication that both PV inputs have degraded or bad signal quality, and so the resulting PV does not have good quality.	• P_AlnDual
OffFail	Raised when the device is commanded Off, but the device feedback does not confirm that the device is actually Off within the configured failure time (Cfg_OffFailT).	• P_DOut
OneGood	Raised when either of the two PV inputs has degraded or bad quality.	• P_AlnDual
OnFail	Raised when the device is commanded On, but the device feedback does not confirm that the device is actually On within the configured failure time (Cfg_OnFailT). If the Failure is configured as a shed fault, the device is commanded Off and cannot be commanded On until reset.	• P_DOut
OverTol	Raised when the tolerance check is performed and the quantity that is delivered exceeds the setpoint by more than the High Tolerance threshold.	• P_Dose
PosFail	Raised when the device is commanded On, but the device feedback does not confirm that the device is actually On within the configured failure time (Cfg_OnFailT). If the Failure is configured as a shed fault, the device is commanded Off and cannot be commanded On until reset.	• P_nPos
SeqTO	Raised when the overall time in the sequence, since the sequence was started, exceeds the sequence timeout configuration (Cfg_SeqTO).	• P_Seq
StepTO	Raised when the amount of time in the current step of the sequence exceeds the step timeout configuration for that step (Ref_Steps[stepnumber].Cfg_FaultT).	• P_Seq
TgtDisagree	Raised when the Input PV is not in the same state as the Target (Inp_Target). Gating, and timing are set in configuration.	• P_DIn
TransitStall	Raised when the valve is using both open and closed limit switches and neither position is confirmed (the valve position is in transit) for the configured transit stall time.	• P_ValveH0
	Raised when the valve has and is using both open and closed position feedback, an attempt is made to open or close the valve, and the position feedback indicates that the valve moved off the original position but did not reach the target position within the configured transit stall time.	• P_ValveM0 • P_ValveS0
Trip	Raised when the overload relay has tripped, helping prevent the motor from running. The overload relay must be reset before the motor can be started.	• P_E1PlusE • P_E3000vld • P_E30vld
TripFail	Raised if the motor has and is using the optional trip feature, an attempt is made to trip (stop) the motor, and the run feedbacks show that the motor did not stop within the configured fail to trip time.	• P_MotorH0
	Raised if the valve has and is using the optional trip feature, an attempt is made to trip the valve, and the limit switch feedbacks show that the valve did not reach the configured tripped position (opened or closed) within the configured fail to trip time.	• P_ValveH0
UnderTol	Raised when the tolerance check is performed and the quantity that is delivered falls short of the setpoint by more than the Low Tolerance threshold. TIP: In some instances, the Bump function can be used to make up the shortage.	• P_Dose

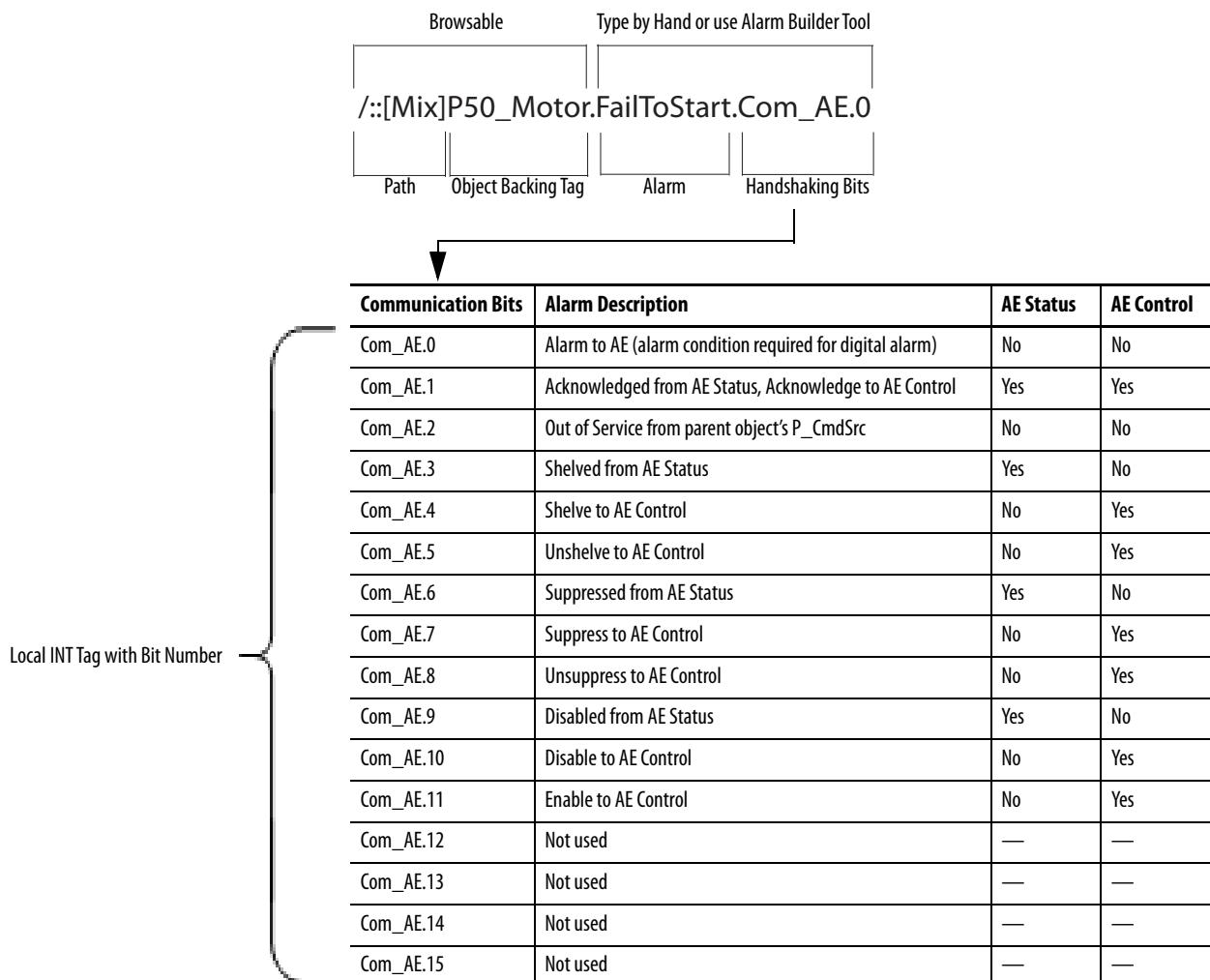
Table 3 - P_Alarm Types by Library Objects

Alarm Type	Alarm Description	Library Objects
Warn	Raised when a motor overload condition is occurring and a trip of the overload relay is imminent. Immediate action must be taken to reduce the load on the motor.	• P_E1PlusE • P_E3000vld • P_E30vld
ZeroFault	Raised if the totalizer fails to clear, or if the totalizer is cleared but then registers flow before flow is commanded to start.	• P_Dose

Alarm with FactoryTalk Alarms and Events Server

FactoryTalk View SE alarms use digital alarms; one digital alarm per P_Alarm instance. This section describes how to configure digital alarm parameters.

As shown in [Figure 7](#), each object with alarms has a P_Alarm instance for each alarm, and each P_Alarm instance has a Local Tag (.Com_AE.x) specifically for communicating all alarm status and commands with the FactoryTalk Alarms and Events server.

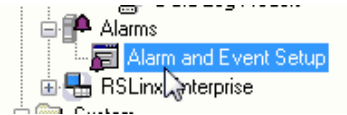
Figure 7 - FactoryTalk View SE Alarm Path

By using an INT (Com_AE) tag for the interface and bits for status and control, the number of tags and elements on scan in the FactoryTalk Alarms and Events server are kept to a minimum.

Adding a Digital Alarm

Complete these steps to add a digital alarm.

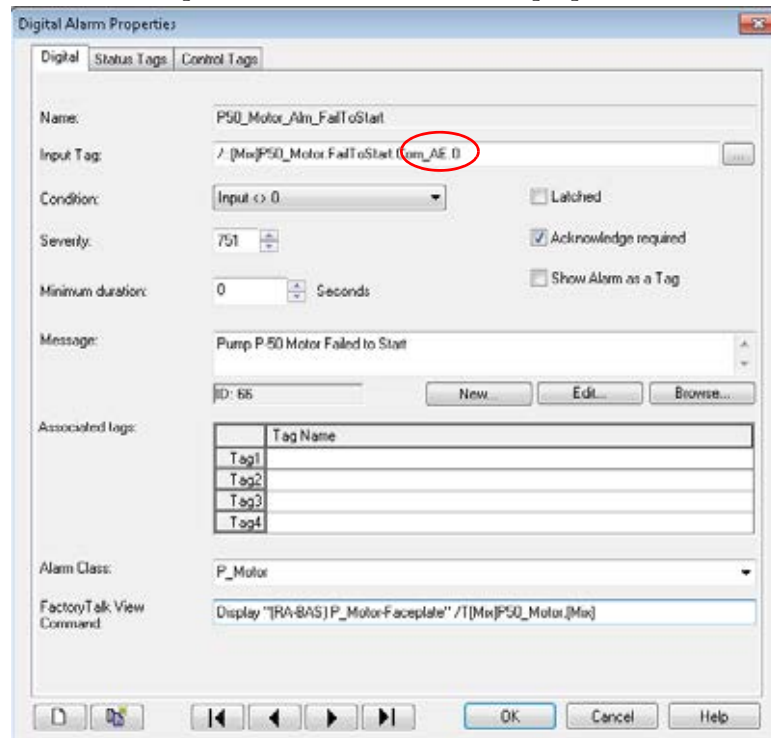
1. Open the FactoryTalk View Studio software.
2. In the Explorer window, click Alarms and double-click Alarm and Event Setup.



The Alarm and Event Setup dialog box appears with the All Alarms tab.

3. Click the New toolbar icon  and choose Digital.

The Digital Alarm Properties dialog box appears empty for a new alarm. The example shows data for instructional purposes.



IMPORTANT In the 'Com_AE' local tag, bit .0 (circled in the example and described on [page 111](#)) is the alarm condition that is required for the digital alarm in the FactoryTalk Alarm and Event setup.

4. Complete the Digital Alarm Properties dialog box.

Table 4 - Digital Alarm Property Descriptions

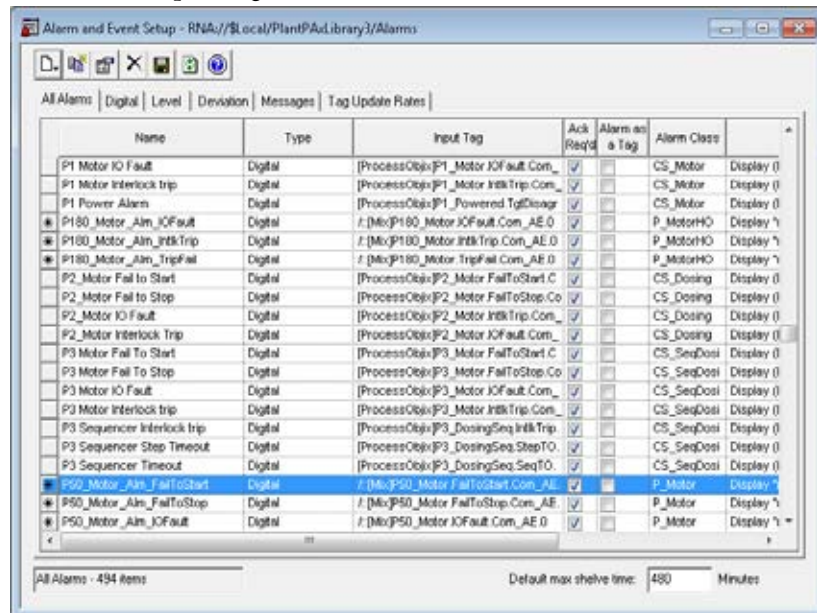
Topic	Description										
Name	Type a unique designator for the object and a short description of the alarm. In the example, 'P50_Motor' identifies the P_Motor object and 'Alm_FailToStart' identifies the alarm. IMPORTANT: A simple method to generate the alarm name, especially in an editing tool like Microsoft Excel, is to remove the communication path (shortcut) and replace the dot (".") between the tag name and the alarm status parameter name with an underscore ('_'). For example, the input tag: /::[Mix]P50_Motor.Alm_FailToStart becomes alarm name: P50_Motor_Alm_FailToStart										
Input Tag	Type bit .0 of the Com_AE local tag in the P_Alarm instance. You can click Browse (...) to select the path and object but not for local tags. Therefore, you must type the last part of the input tag (Com_AE.0).										
Condition	From the Condition pull-down menu, choose Input <> 0.										
Severity	Choose an alarm severity that aligns with the severity in the controller and is shown on the HMI faceplate. Severity values include the following: <table> <tr> <td>Severity on Faceplate</td><td>A & E Severity</td></tr> <tr> <td>1...250 = Low</td><td>1...250</td></tr> <tr> <td>251...500 = Medium</td><td>251...500</td></tr> <tr> <td>501...750 = High</td><td>501...750</td></tr> <tr> <td>751...1000 = Urgent</td><td>751...1000</td></tr> </table> IMPORTANT: The Alarm Builder tool works with Library versions 2.0, 3.0, 3.1, and 3.5. The mapping that is shown above is for version 3.1 and later that has a range from 1...1000 (INT data type). For Add-On Instructions version 3.0 and earlier, the severity values are in the range of 1...4 (SINT data type). When an ACD file that contains version 3.0 Add-On Instructions is updated with version 3.1 or later Add-On Instructions, the Add-On Instruction tags retain their existing severity values (1...4).	Severity on Faceplate	A & E Severity	1...250 = Low	1...250	251...500 = Medium	251...500	501...750 = High	501...750	751...1000 = Urgent	751...1000
Severity on Faceplate	A & E Severity										
1...250 = Low	1...250										
251...500 = Medium	251...500										
501...750 = High	501...750										
751...1000 = Urgent	751...1000										
Minimum duration	Set to zero. The controller handles alarm on-delay and off-delay timing.										
Latched	Leave blank. The controller handles alarm latching (reset required).										
Acknowledge required	Make sure that there is a check in the check box. The controller handles alarm auto-acknowledgment for alarms that are configured as 'Ack Not Required' from the faceplate.										
Show Alarm as a Tag	Leave blank.										
Message	To create an optional text message with embedded variables for each alarm, click the Browse button (ellipsis, ...) beside the Message box. At runtime, these messages appear in Alarm and Event graphic objects, such as the Alarm and Event Summary. The maximum length of an alarm message is 255 characters. During alarm message import, Studio 5000 Logix Designer application verifies the message length and displays a warning if the alarm message exceeds the character limit.										
Associated tags	Optionally define up to four Associated Tags. At runtime, the associated tag values are recorded in the alarm and event history log. The tags 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. 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.										
Alarm Class	Optionally, type the name of a new Alarm Class or select one from the list of classes. Each time that you create an alarm class, the name is added to the list of Alarm Classes. Alarm Classes are used to combine related alarms. These classes are based on characteristics other than priority and severity.										

Table 4 - Digital Alarm Property Descriptions

Topic	Description
FactoryTalk View Command	Type a FactoryTalk View command that accesses a faceplate when you double-click an alarm for a particular object on an A&E Alarm Summary dialog box. The command is configured as this example: 'Display {faceplate display name} /TPath object' The name of the display must be inside double-quotes because it contains a space.

- Click OK.

The digital alarm configuration information appears on the Alarm and Event Setup dialog box.

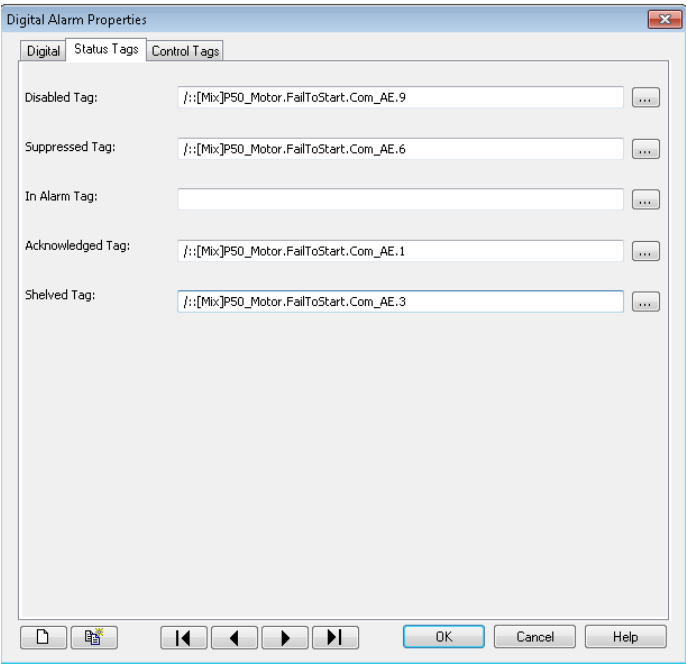


Add Digital Alarm Status Tags

The status tags configuration sends to the controller any shelved/unshelved, disabled/enabled, suppressed/unsuppressed, and acknowledged status updates from the FactoryTalk Alarms and Events Alarm Summary and Alarm Status Explorer dialog boxes.

1. From the Alarms Properties dialog box, click the Status Tags tab.

The Status Tags dialog box appears.



2. Complete the status tag dialog boxes.

Table 5 - Status Tag Dialog Box

Field	Description
Disabled Tag	Type "Path Object.Alarm.Com_AE.9" for the disabled tag. Our example is /::[Mix]P50_Motor.FailToStart.Com_AE.9
Suppressed Tag	Type 'PathObject.Alarm.Com_AE.6 for the suppressed tag. Our example is /::[Mix]P50_Motor.FailToStart.Com_AE.6
In Alarm Tag	Leave blank.
Acknowledged Tag	Type 'Path Object.Alarm.COM_AE.1' for the acknowledged tag. Our example is /::[Mix]P50_Motor.FailToStart.Com_AE.1
Shelved Tag	Type 'Path Object.Alarm.COM_AE.3' for the shelved tag. Our example is /::[Mix]P50_Motor.FailToStart.Com_AE.3

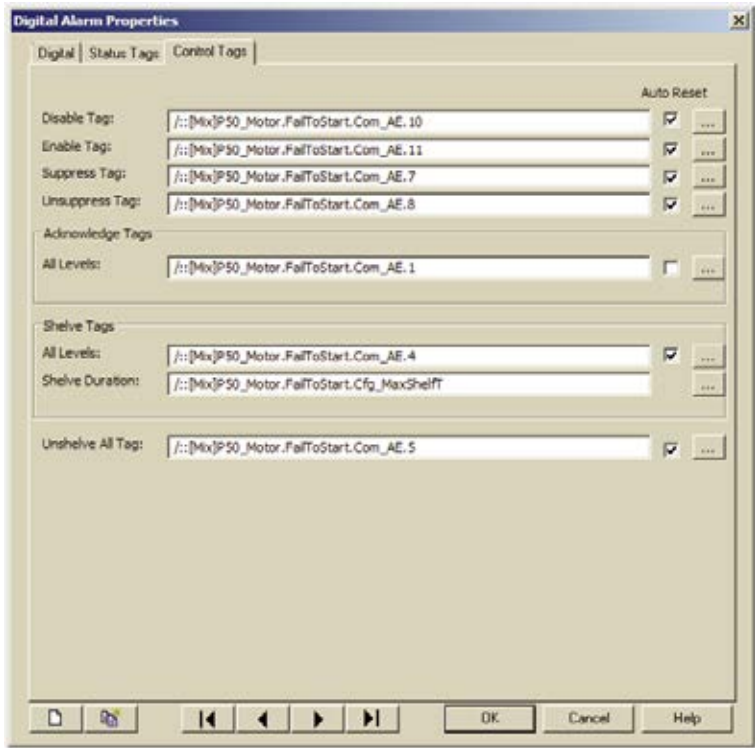
IMPORTANT Even though Com_AE is a Local Tag in the Add-On definition, it's configured to be writable (Read/Write, not Read Only) so the FactoryTalk Alarms and Events server status is sent to the bits identified for the tags that are shown previously.

3. Click OK.

Add Digital Alarm Control Tags

Control Tags let the FactoryTalk Alarms and Events server to access alarm acknowledgment, disable, enable, shelve, and unshelve actions. These actions are performed via faceplates. Suppress and unsuppress actions are performed in controller logic.

- 1. From the Alarms Properties dialog box, click the Control Tags tab.
The Control Tags dialog box appears.



- 2. Complete the Control Tags dialog box.

Field	Description
Disable Tag Auto Reset checked	Type "Path Object.Alarm.Com_AE.10" for the disable tag. Check the Auto Reset checkbox (so the Disable control automatically resets). Our example is ::[Mix]P50_Motor.FailToStart.Com_AE.10
Enable Tag Auto Reset checked	Type "Path Object.Alarm.Com_AE.11" for the enable tag. Check the Auto Reset checkbox (so the Enable control automatically resets.) Our example is ::[Mix]P50_Motor.FailToStart.Com_AE.11
Suppress Tag Auto Reset checked	Type "Path Object.Alarm.Com_AE.7" for the suppress tag. Check the Auto Reset checkbox (so the Suppress control automatically resets). Our example is ::[Mix]P50_Motor.FailToStart.Com_AE.7
Unsuppress Tag Auto Reset checked	Type "Path Object.Alarm.Com_AE.8" for the unsuppress tag. Check the Auto Reset checkbox (so the Unsuppress control automatically resets.) Our example is ::[Mix]P50_Motor.FailToStart.Com_AE.8
Acknowledge Tags All Levels Auto Reset unchecked	Type "Path Object.Alarm.Com_AE.1" for the acknowledge tag. Leave the Auto Reset checkbox blank. Our example is ::[Mix]P50_Motor.FailToStart.Com_AE.1

Field	Description
Shelve Tags All Levels Auto Reset checked	Type "Path Object.Alarm.Com_AE.4" for the shelve tag. Check the Auto Reset checkbox (so the Shelve control automatically resets.) Our example is /::[Mix]P50_Motor.FailToStart.Com_AE.4
Shelve Duration	Type "Path Object.Alarm.Cfg_MaxShelfT" for the shelve duration tag. Our example is /::[Mix]P50_Motor.FailToStart.Cfg_MaxShelfT
Unshelve All Tags Auto Reset checked	Type "Path Object.Alarm.Com_AE.5" for the unshelve tag. Check the Auto Reset checkbox (so the Unshelve control automatically resets.) Our example is /::[Mix]P50_Motor.FailToStart.Com_AE.5

3. Click OK.

Alarm with FactoryTalk View ME Software

This section describes how to configure a library alarm in FactoryTalk View ME software for use with PanelView™ Plus and other FactoryTalk View ME terminals. Because FactoryTalk View ME software does not support features for suppression, shelving, and disabling, this procedure has not changed from the method that is used for the Library Version 2.0.

This method does support acknowledgment of alarms from the FactoryTalk View ME terminal. In addition, because the FactoryTalk View ME faceplates are the same as the FactoryTalk View SE faceplates, operator actions for shelve, unshelve, disable, enable, and acknowledgment are recognized by the P_Alarm Add-On Instruction, and by the FactoryTalk Alarms and Events server.

The missing capabilities in FactoryTalk View ME software include the following:

- Ability to shelve/unshelve and disable/enable from the Alarm Status Explorer (there is none) or Alarm Summary
- Ability to provide lists of shelved, suppressed, or disabled alarms

An operator on a FactoryTalk View ME terminal can shelve alarms, or maintenance personnel can disable alarms (and the P_Alarm Add-On Instruction helps prevent new alarms from being generated when shelved). The shelved or disabled status is reflected on the FactoryTalk View SE Alarm Status Explorer.

The examples use the following attributes:

- Path is /::[Mix]
- Object is P50_Motor
- Alarm is FailToStart

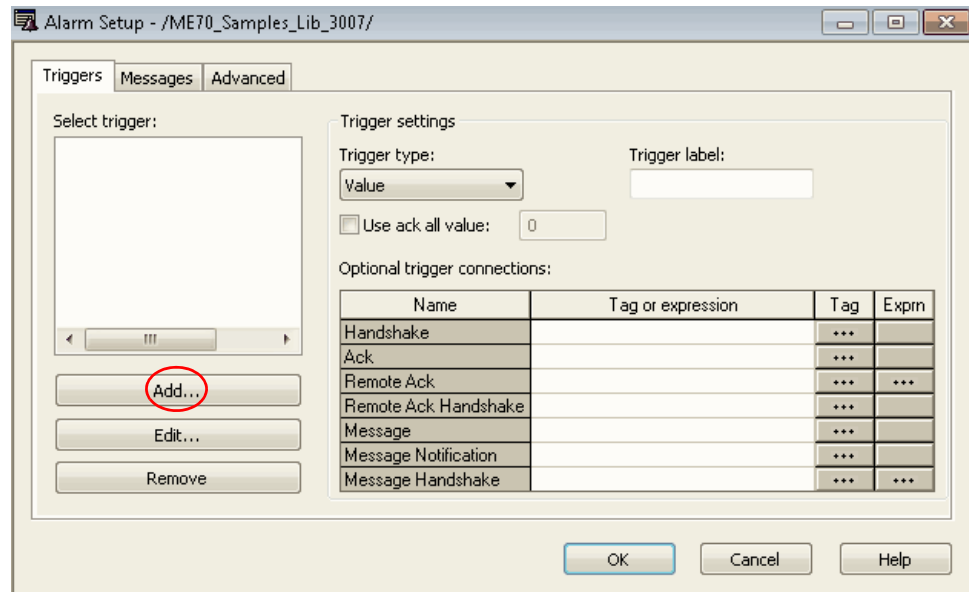
Create a New Alarm Trigger

Complete these steps.

1. Open the FactoryTalk View Studio software.
2. In the Explorer window, click Alarms and double-click Alarm Setup.

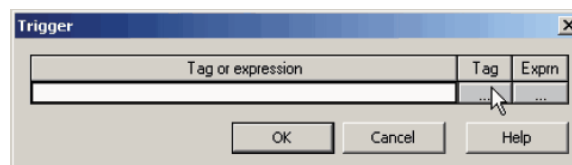


The Alarm Setup dialog box appears with the Triggers tag open.



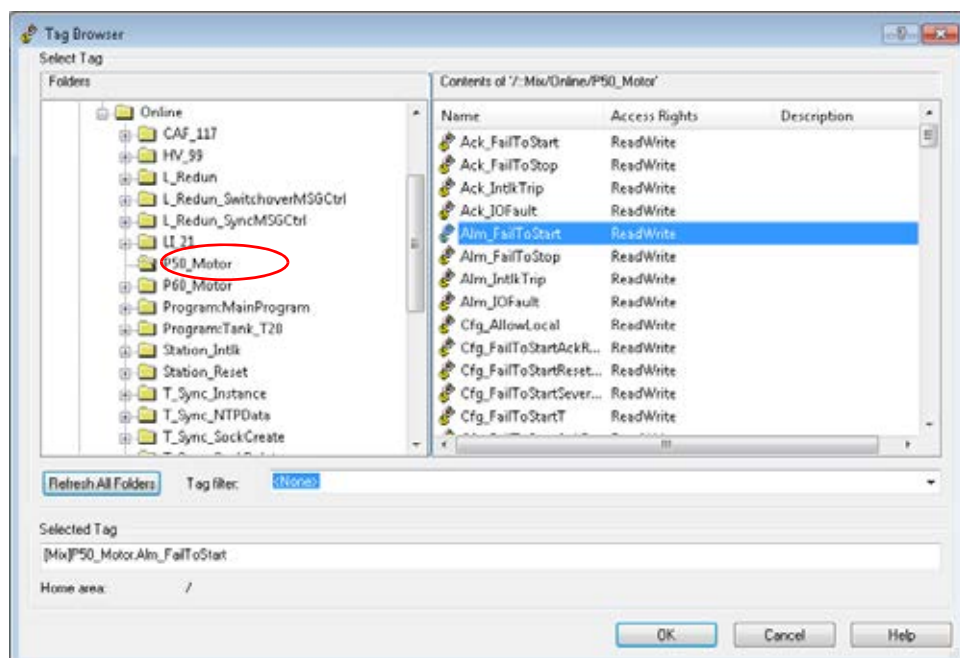
- Click Add.

The Trigger dialog box appears.



- Click Browse (...) under Tag.

The Tag Browser appears.



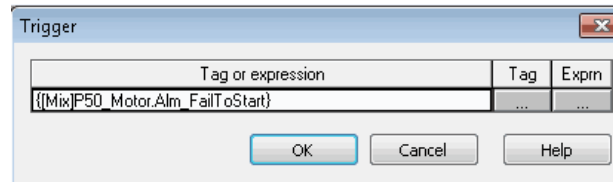
- In the left pane of the Tag Browser, click the folder that represents the Add-On Instruction instance. Our example is P50_Motor.

You can use the folder for the offline controller. If you are connected to the actual hardware controller on a network and the controller application is loaded, you can use the online folder.

- In the right pane of the Tag Browser, double-click the alarm status parameter for the alarm.

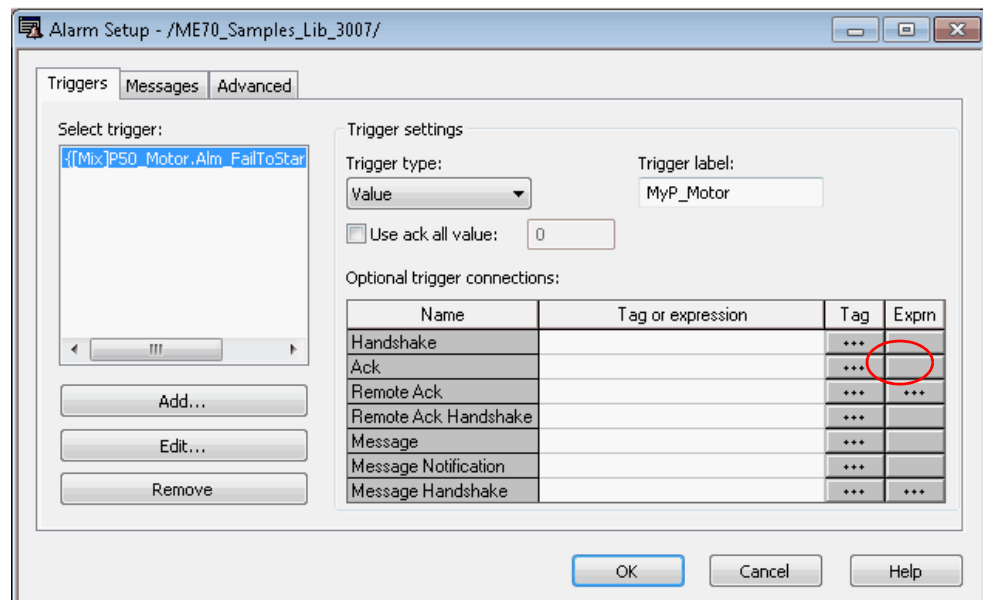
TIP The alarm status parameter name starts with 'Alm_'. For example, the parameter is 'Alm_FailToStart'.

The alarm status parameter appears in the Trigger popup window.



- Click OK.

The alarm status parameter appears in the Select trigger box of the Alarm Setup dialog box.



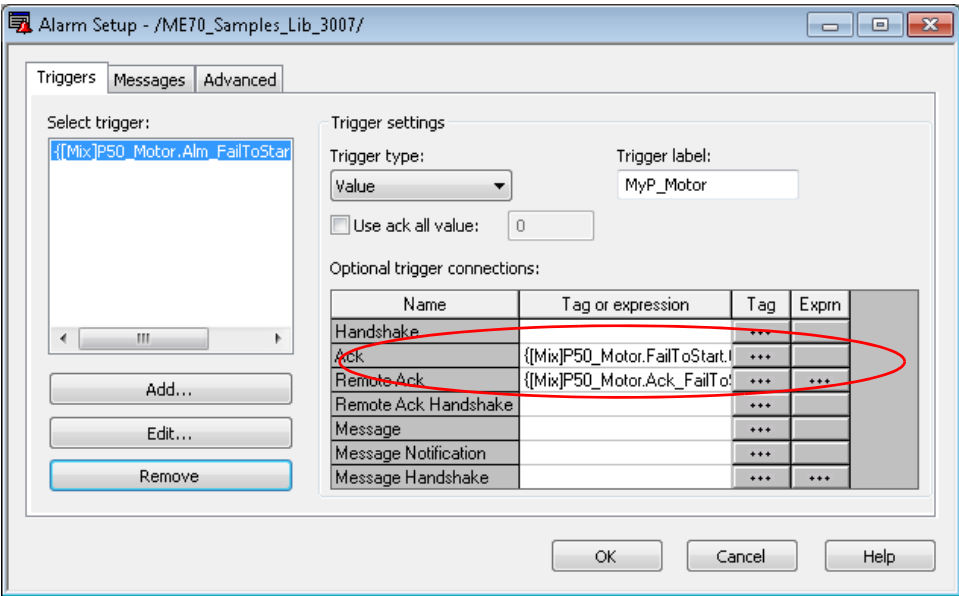
- Click Browse (...) in the Tag column for the Ack row.

'Ack' is an abbreviation for acknowledge.

The Tag Browser appears when you click Browse.

- Use the Tag Browser to choose the acknowledge status parameter.
- Repeat [step 8](#) and [step 9](#) to choose the acknowledge status parameter for the Remote Ack row.

Your setup now looks like the example.



The 'Ack' parameter must be changed so it triggers the Operator Acknowledge Command, which is part of the alarm's Local Tag within the Add-On Instruction. Because this parameter is in a Local Tag, it cannot be browsed. But, the FactoryTalk View ME HMI server can still write to the tag.

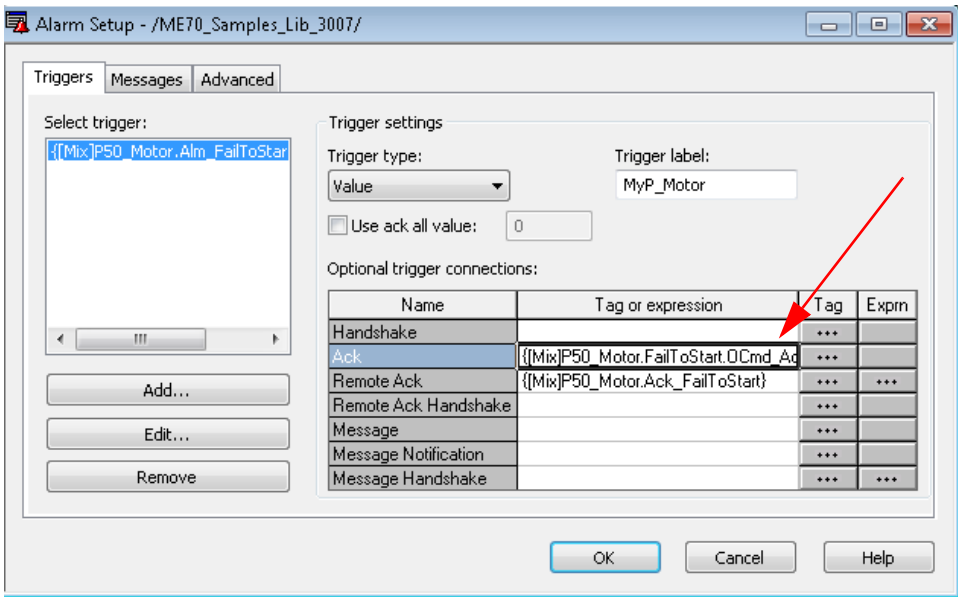
11. To change the 'Ack' parameter, click the tag in the Tag or expression box to access a text cursor and manually type the change.

The format of the expression changes from:

'Path Object.Ack_Alarm'

to:

'Path Object.Alarm.OCmd_Ack'



IMPORTANT Each 'Ack' tag occurrence can be changed in the alarm database. To change the tag occurrence, perform an export of the alarms, use the editing tool to modify the tag, then import the changes. Simply Find and Replace every 'Ack' tag of: .Ack_FailToStart to: .FailToStart.OCmd_Ack

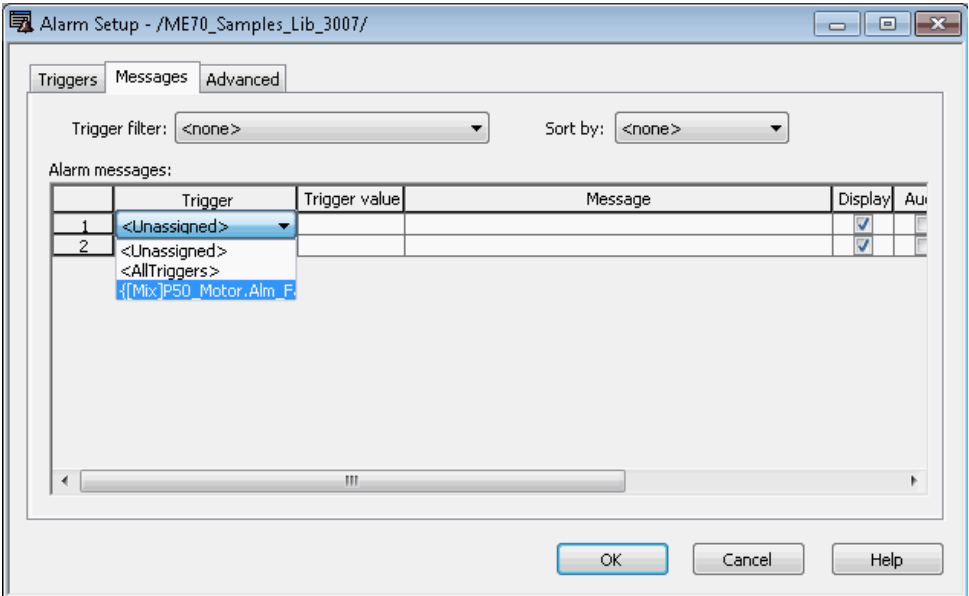
Perform a similar Find and Replace for each type of alarm: FailToStop, IOFault, and so forth.

Be careful when performing the Find and Replace procedure that you do not change the RemoteAck entries.

Configure the Alarm Message

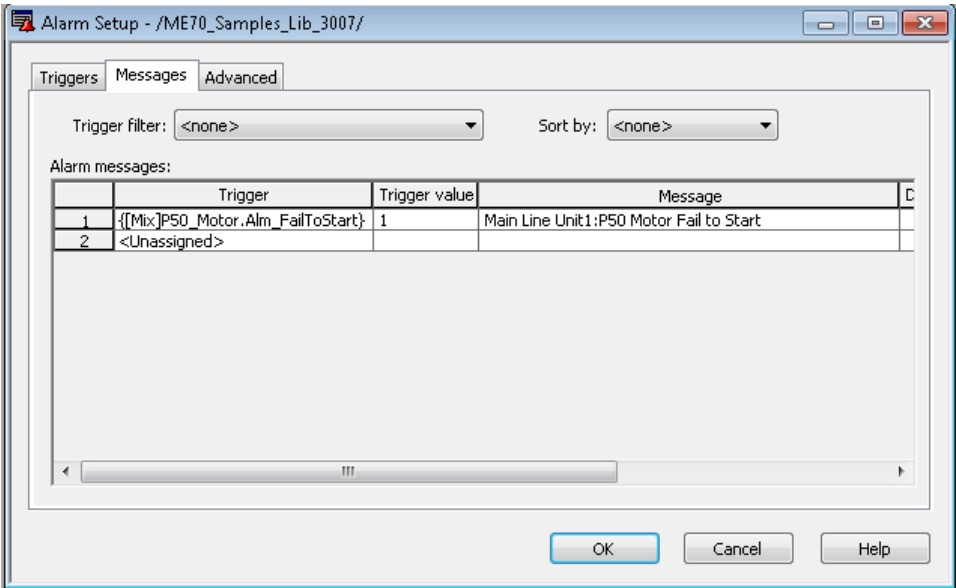
Follow these steps to define a message that appears when the alarm occurs.

- 1. On the FactoryTalk View ME Alarm Setup dialog box, click the Messages tab.



- 2. From the Trigger pull-down menu, choose the Trigger tag that you created.
- Our example is the {[Mix]P50_Motor.
- 3. In the Trigger Value column, type 1.

4. In the Message column, type the message to appear with the alarm.

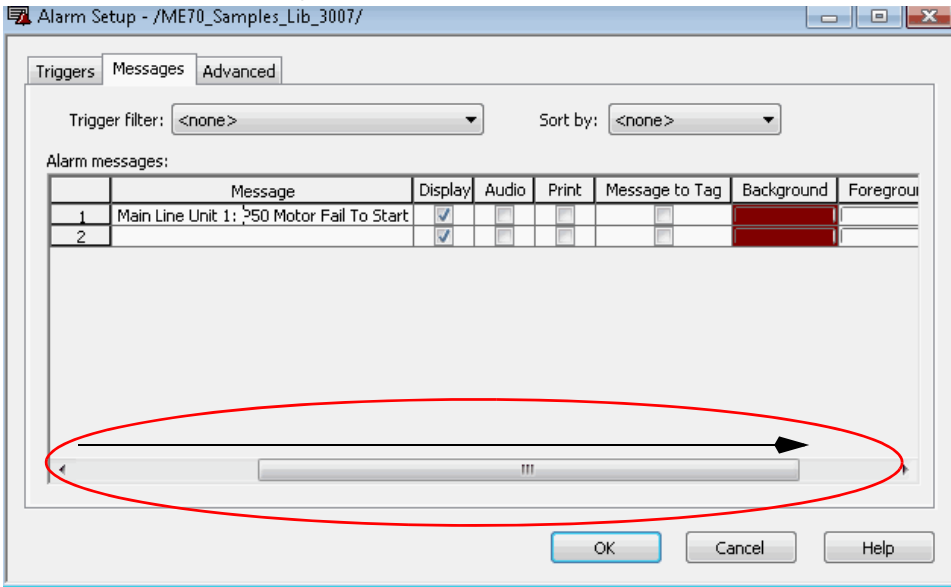


5. Click OK.

Configure Notification Options

Follow these steps for audio or visuals when an alarm occurs.

1. On the Message tab of the Alarm Setup dialog box, click and drag the bottom display bar to the right to show the rest of the columns for the alarm message.



2. Complete the notification options.

Field	Description															
Display	Leave the check in the checkbox so you see the alarm display when there is an alarm occurrence.															
Audio	Check the checkbox if you want a sound to play from the speakers for the HMI terminal when an alarm occurs.															
Print	Check the checkbox if you have a printer that is attached to the HMI terminal (directly or via a network) and you want the alarm to print when the alarm occurs.															
Message to Tag	We recommend that you leave the box blank.															
Background and Foreground	Select the colors that are used to display the alarm on the alarm summary. Colors that are selected must match the alarm severity. <table><tr><th><u>Severity</u></th><th><u>Foreground</u></th><th><u>Background</u></th></tr><tr><td>1...250 (Low)</td><td>White</td><td>Blue</td></tr><tr><td>251...500 (Medium)</td><td>Black</td><td>Bright Yellow</td></tr><tr><td>501...750 (High)</td><td>Black</td><td>Bright Red</td></tr><tr><td>751...1000 (Urgent)</td><td>Black</td><td>Bright Magenta</td></tr></table>	<u>Severity</u>	<u>Foreground</u>	<u>Background</u>	1...250 (Low)	White	Blue	251...500 (Medium)	Black	Bright Yellow	501...750 (High)	Black	Bright Red	751...1000 (Urgent)	Black	Bright Magenta
<u>Severity</u>	<u>Foreground</u>	<u>Background</u>														
1...250 (Low)	White	Blue														
251...500 (Medium)	Black	Bright Yellow														
501...750 (High)	Black	Bright Red														
751...1000 (Urgent)	Black	Bright Magenta														

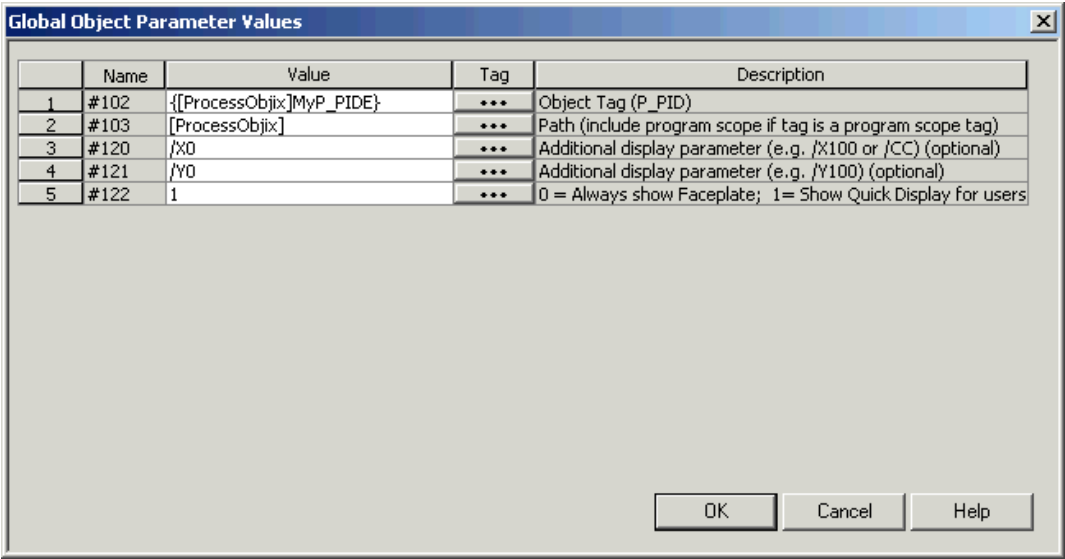
3. Repeat these steps for additional messages.
4. Click OK.

Global Object Configuration

Graphic displays, which consist of display elements (graphic objects), provide an operator with a visual, runtime outlook on a plant activity. The displays show system or process data, and provide operator's with a way to write values to external devices, such as a controller.

/X and /Y Positioning

The Global Object Parameter Values dialog box, which is used for specifying faceplate coordinates, has two optional parameters: #120 and #121.



Parameters #120 and #121 let you specify additional display command switches. These additional switches can be used to specify where the faceplate appears on your screen.

For example, '/x150' in parameter #120 and '/Y50' in parameter #121 specify that the faceplate appears 150 pixels from the left and 50 pixels from the top. These parameters are optional and do not need to be specified.

[Table 6](#) and [Table 7](#) describe the position parameters and how to use them with the #120 and #121 global object parameters, respectively.

Table 6 - Corner and Center Positioning

Global Object Parameter	Position Parameter	Description
#120	/Q1	Top, right corner
	/Q2	Top, left corner
	/Q3	Bottom, left corner
	/Q4	Bottom, right corner
	/CT	Centered on top edge
	/CB	Centered on bottom edge
	/CL	Centered on left edge
	/CR	Centered on right edge
	/CC	Center of screen
#121	Leave blank. See the tip.	

TIP The values of global object parameters #120 and #121 are passed directly to the Display command that opens the faceplate display (.gfx). Nothing inhibits you from putting any valid Display command parameters in these values. For more information, see the FactoryTalk View Site Edition User Guide, publication [VIEWSE-UM006](#), or the online Help for the Display command parameters in the FactoryTalk View Studio software.

Table 7 - Absolute Positioning

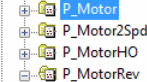
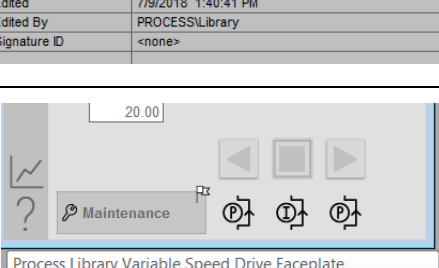
Global Object Parameter	Position Parameter	Description
#120 (SE)	/Xnnn	Left edge of faceplate 'nnn' pixels from the left edge of the screen
#120 (ME)	Numeric value (for example, 100)	
#121 (SE)	/Ynnn	Top edge of the faceplate 'nnn' pixels down from the top edge of the screen
#121 (ME)	Numeric value (for example, 100)	

Parameter #122 controls the display that appears when you click the display element. To expand the row height for parameter #122, type a value and click Enter.

- '0' specifies the full faceplate always appears.
- '1' specifies the full faceplate appears if the user has security code 'C'; and the 'Quick' display appears if you do not have security code 'C'.
- '2' specifies the 'Quick' display always appears.
- If parameter #122 is left blank, the full faceplate always appears.

Maintain Library Releases

Each library object has a revision x.yy.zz where: x is the Major Revision number, yy is the Minor Revision number, and zz is the Maintenance Release. Each release of the Process Library comes with release notes that describe the changes that were made since the last release.

Component	Example																				
The Add-On Instruction in Logix Designer application has revision information visible when the instruction is selected in the Controller Organizer.	 <table border="1" data-bbox="953 541 1421 745"> <tr><td>Description</td><td>Single Speed Motor</td></tr> <tr><td>Revision</td><td>v4.0 .00 Release</td></tr> <tr><td>Revision Note</td><td>See Instruction Help for a summary of changes, and see the Logix Designer Help for more information.</td></tr> <tr><td>Vendor</td><td>Rockwell Automation</td></tr> <tr><td>Data Type Size</td><td>1032 bytes</td></tr> <tr><td>Created</td><td>3/27/2008 3:50:50 PM</td></tr> <tr><td>Created By</td><td>Not Available</td></tr> <tr><td>Edited</td><td>7/9/2018 1:40:41 PM</td></tr> <tr><td>Edited By</td><td>PROCESSLibrary</td></tr> <tr><td>Signature ID</td><td><none></td></tr> </table>	Description	Single Speed Motor	Revision	v4.0 .00 Release	Revision Note	See Instruction Help for a summary of changes, and see the Logix Designer Help for more information.	Vendor	Rockwell Automation	Data Type Size	1032 bytes	Created	3/27/2008 3:50:50 PM	Created By	Not Available	Edited	7/9/2018 1:40:41 PM	Edited By	PROCESSLibrary	Signature ID	<none>
Description	Single Speed Motor																				
Revision	v4.0 .00 Release																				
Revision Note	See Instruction Help for a summary of changes, and see the Logix Designer Help for more information.																				
Vendor	Rockwell Automation																				
Data Type Size	1032 bytes																				
Created	3/27/2008 3:50:50 PM																				
Created By	Not Available																				
Edited	7/9/2018 1:40:41 PM																				
Edited By	PROCESSLibrary																				
Signature ID	<none>																				
The faceplate in FactoryTalk View software has revision information visible when the pointer is paused just inside the lower left corner of the faceplate.	 20.00 Process Library Variable Speed Drive Faceplate Revision 4.0-00 Release (RA-BAS) P_VSD-Faceplate.gfx Copyright © Rockwell Automation, Inc. All Rights Reserved																				

The instruction and faceplate are compatible if they have the same Major and Minor Revision numbers. In general, maintenance releases can be loaded into your application with little impact. We suggest that you review the release notes to make sure that you understand the changes that were made in the release and assess the impact to your application.

Process Library patches are made available between Maintenance Releases via [Knowledgebase Article 654910–Rockwell Automation Library Patch TOC](#).

TIP You are required to log into the Knowledgebase.

The article provides a collection, or table of contents, which directs the user to all patches related to the Rockwell Automation Library of Process Objects. Patches are arranged chronologically by Major Revision. Each subsequent Maintenance Release of the Process Library rolls up all patches that are associated with the prior Maintenance Release. Therefore, the user only needs to check for patches that are related to the version of the Process Library that they are using.

The use of two different Major/Minor revisions of the Library in one application is not supported. When updating major or minor releases, there are several considerations to consider:

- The new revision can have differences in functionality that requires adjustment of your application code written around the library instruction instances.
- There can be differences in the look-and-feel that requires an update of operator training and manuals.
- If you have made any customizations, repeat the customizations on the new release of the library after it is loaded.

It is important that these considerations are weighed properly against the benefits you gained from the new Library release. The release of the Library typically includes documentation on how to upgrade from the previous Major/Minor release.

Customize the Library

The Rockwell Automation Library of Process Objects can be customized for project or customer-specific reasons. However, we recommend that you consider the following:

- Weigh the value of the customization against the value of using the library as issued. For example, you must document any customization so it can be reapplied, if necessary, on top of a subsequent maintenance release.
- Rockwell Automation provides a varied schedule of maintenance releases for the library. If the library object has not been altered, updating the latest maintenance release can be done more easily. If customized, the customization has to be reapplied manually on subsequent releases.

IMPORTANT	There are some library objects, such as Logix Diagnostic objects and Steam Table instructions, which are source protected and cannot be modified. These objects typically are not operator-facing and have logic that must be unchanged to provide the desired functionality.
------------------	---

- Customization could inhibit your ability to leverage the library documentation or standard training for the library.
- The library is supported through Rockwell Automation Technical Support as long as the Add-On Instructions have not been modified from the original deployment. If customized, the library is supported similar to any other application code.

Use the Process Library

This chapter shows how to use Library Add-On Instructions, global objects, and faceplates to build your control application.

Global objects (display elements) provide touch areas from which faceplates are launched, link tag names to the faceplates, and display the process variables and alarms.

The procedures in this section require that you have installed Add-on Instructions, images, displays, and global objects. See [Import Visualization Files on page 13](#) for installation details, if necessary.

The following table describes the topics in this chapter.

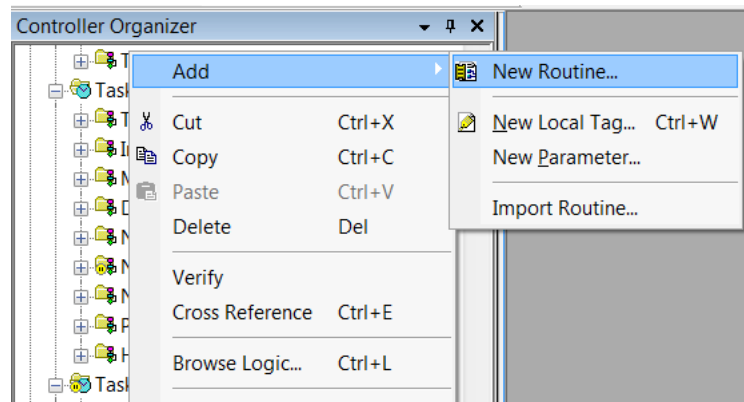
Topic	Page
Create a Logic Instance	130
Create an HMI Instance	133
Device Configuration	139

Create a Logic Instance

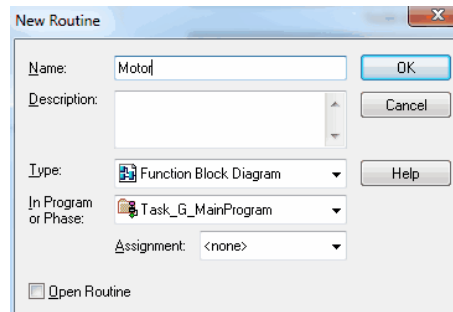
Perform these steps to add an Add-On Instruction to a routine, create a backing tag, and connect I/O.

Add-On Instructions can be used in any of the Logix languages: Ladder Diagram, Function Block Diagram, or Structured Text. In this example, we show how to add an instruction instance to a Function Block Diagram routine.

1. In the Controller Organizer, right-click the Task and choose Add>New Routine.



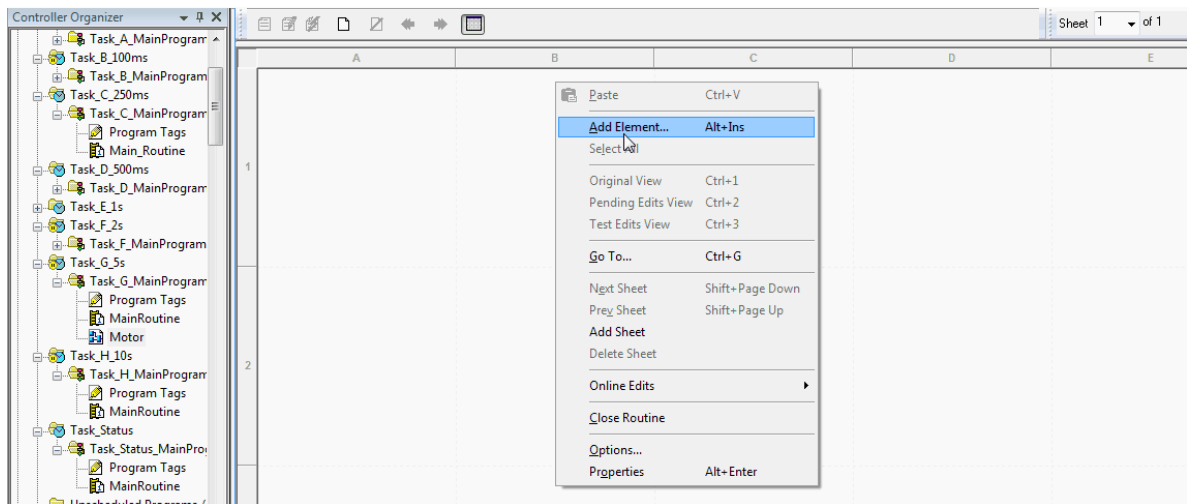
The New Routine dialog box appears.



2. Type a name for the routine.
3. Click the Type pull-down to select a Logic language, such as Function Block Diagram, and click OK.
4. Double-click the routine name in the Controller Organizer

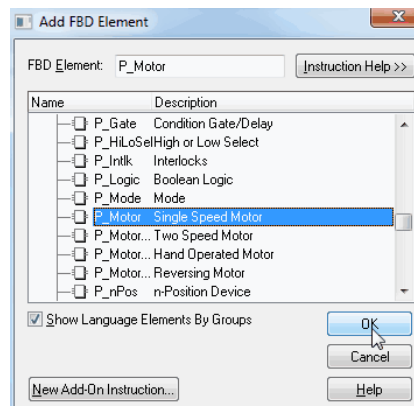
A blank sheet appears in the right pane.

5. Right-click the blank sheet and choose Add Element.



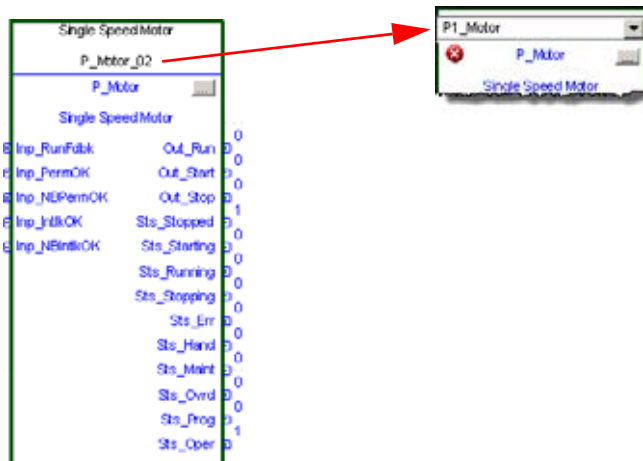
The Add FBD Element window appears.

6. Browse to the Add-On Instruction folder, select the Add-On Instruction, and click OK.

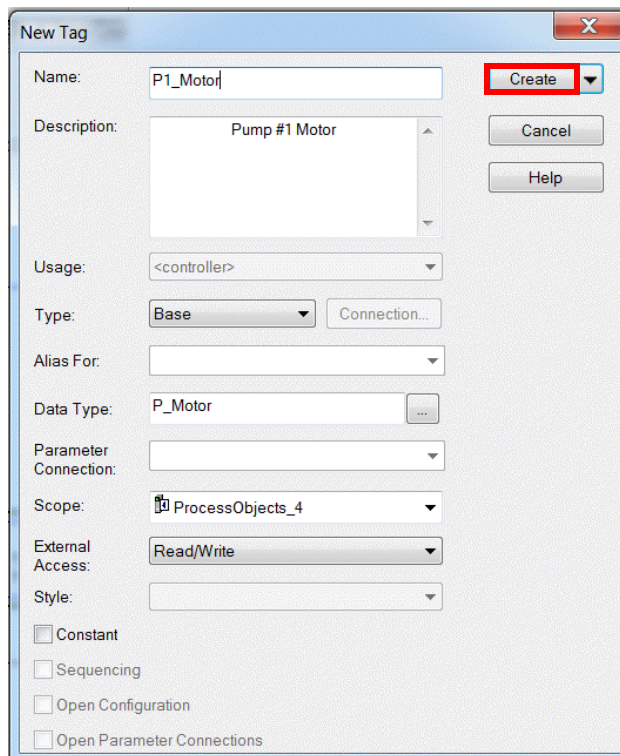


A function block (or the language type you selected) appears.

7. Double-click the name, type a new name (P1_Motor in our example), and press Enter.



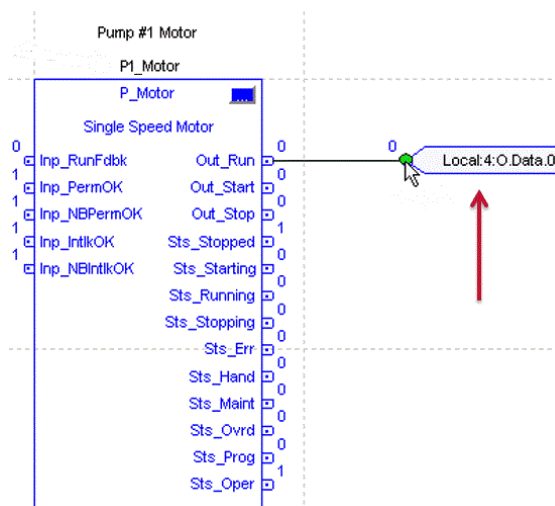
8. Right-click the new name and choose New 'P1_Motor'.
The New Tag dialog box appears.
9. Type a description for your new backing tag and click Create.



TIP The backing tag for the Add-On Instruction can be program-scoped or controller-scoped.

The description and name of the tag appears at the top of the function block.

10. Connect the pins to add I/O.

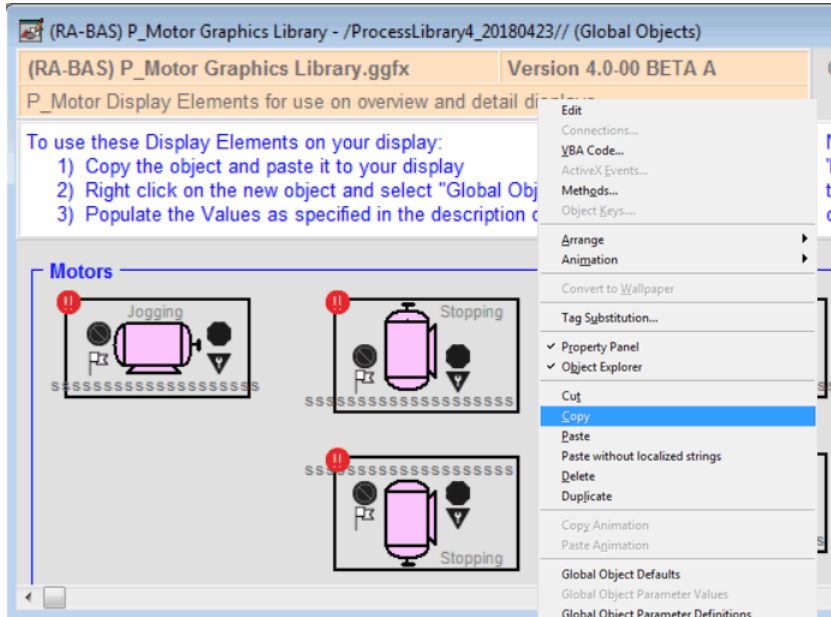


11. Repeat this process for each required Add-On Instruction.

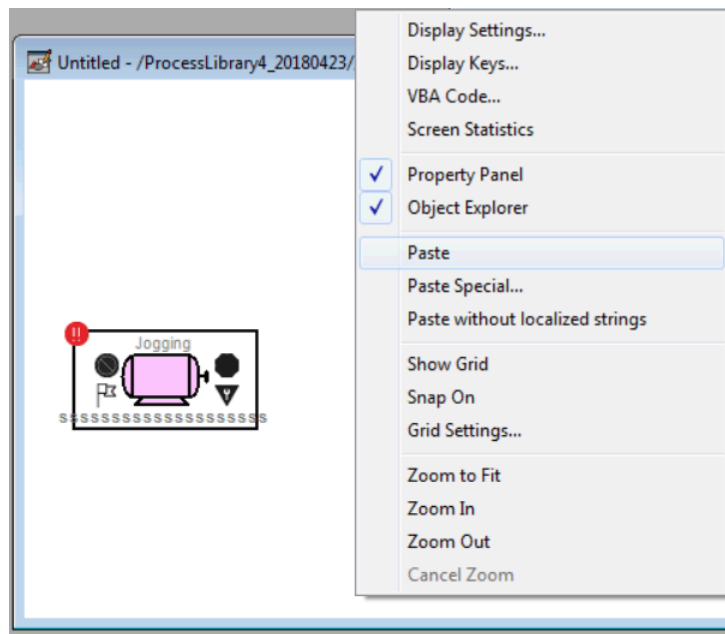
Create an HMI Instance

Follow these steps to configure the HMI instance.

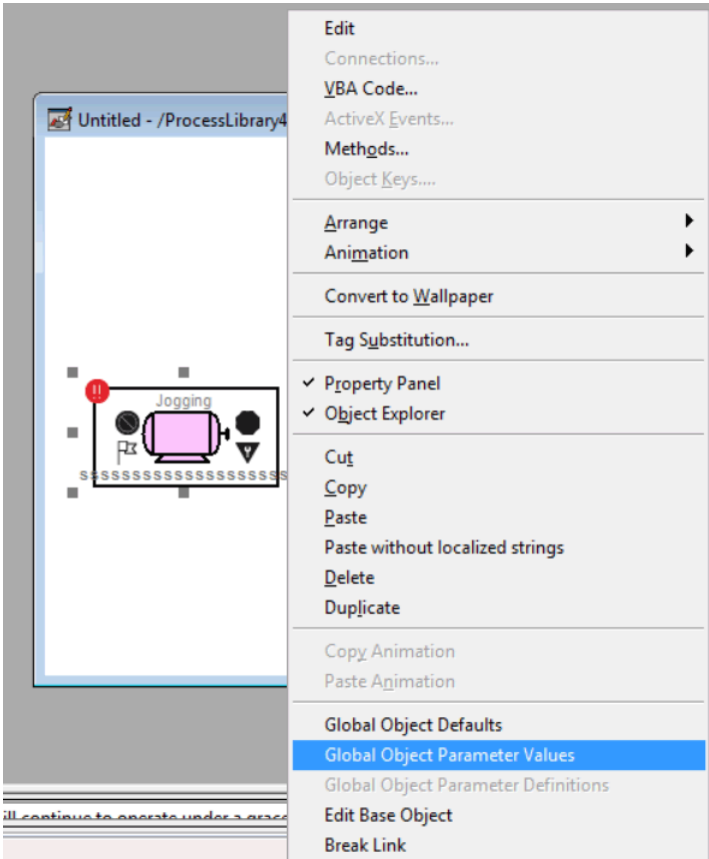
1. In the FactoryTalk® View Studio software program, open the global objects (.ggfx) file that contains the graphics library for the instruction.
2. Right-click and copy a global object.



3. Paste the global object onto the Studio display file you are building.



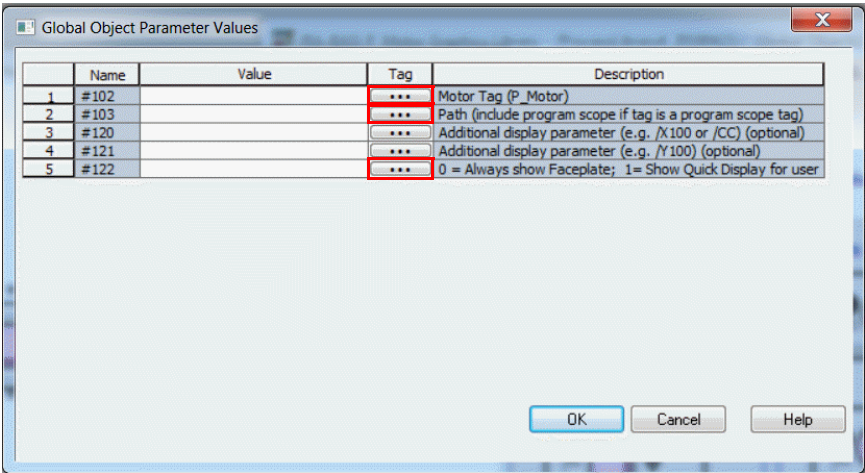
4. In the display file, right-click the global object file and choose Global Object Parameter Values.



The Global Object Parameter Values dialog box appears.

Configure the 1st, 2nd, and 5th parameters.

The third and fourth parameters are 'optional' for FactoryTalk View SE. Errors occur for FactoryTalk View ME if the parameters are not completed. See [page 125](#) for more information.



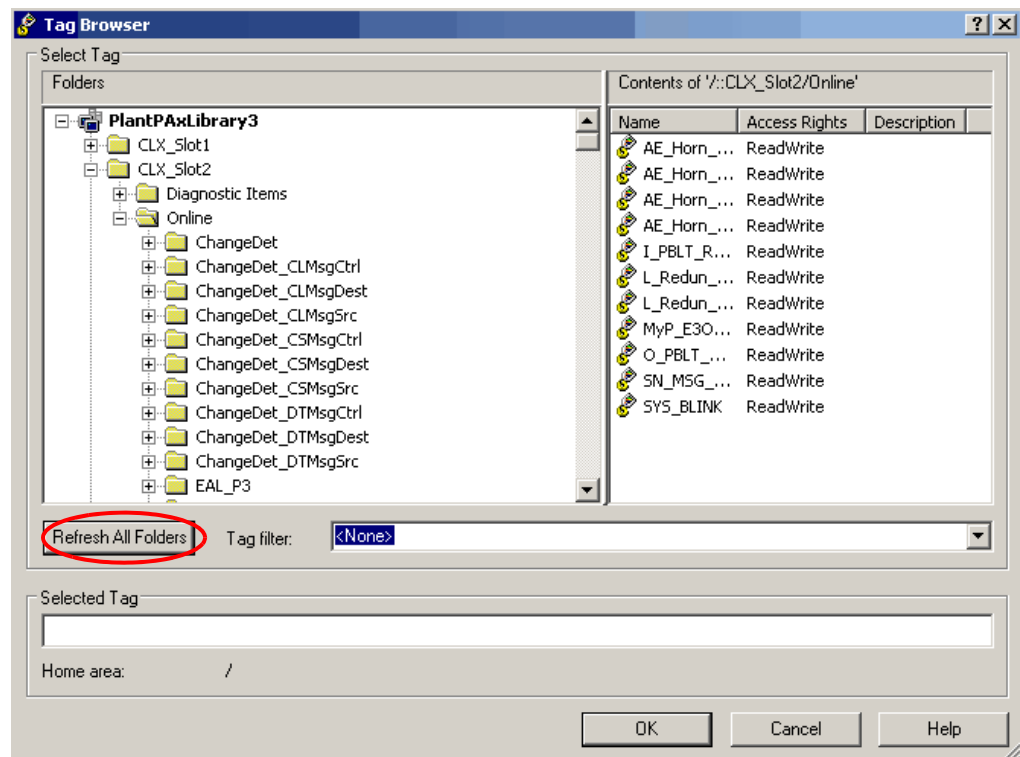
The global object parameters are as follows.

Parameter	Required	(SE) Description	(ME) Description
#102	Y	Object tag to point to the name of the associated object Add-On Instruction in the controller.	
#103	Y	Path that is used for display navigation features to other objects. Include program scope if tag is a program-scope tag.	
#120	N	Additional parameter to pass to the display command to open the faceplate. Typically used to define position for the faceplate.	Optional position parameter for Left offset. Type a number ("nnn") to set the number of pixels the left edge of the display is from the left edge of the screen
#121	N	Additional parameter to pass to the display command to open the faceplate. If you define the X and Y coordinates, separate parameters so that X #120 and Y #121. This functionality lets the same parameters to be used in subsequent display commands that originate from the faceplate.	Optional position parameter for Top offset. Type a number ("nnn") to set the number of pixels the top edge of the display is from the top edge of the screen
#122	Y	The options for the global object display: 0 = Always show faceplate 1 = Show Quick Display for users without Maintenance access (Code C) 2 = Always show Quick Display	

5. To enter the backing tag for the #102 parameter, you can type a tag into the Value column or click the Browse button .

For the following procedures, click Browse to enter the first parameter, the object tag.

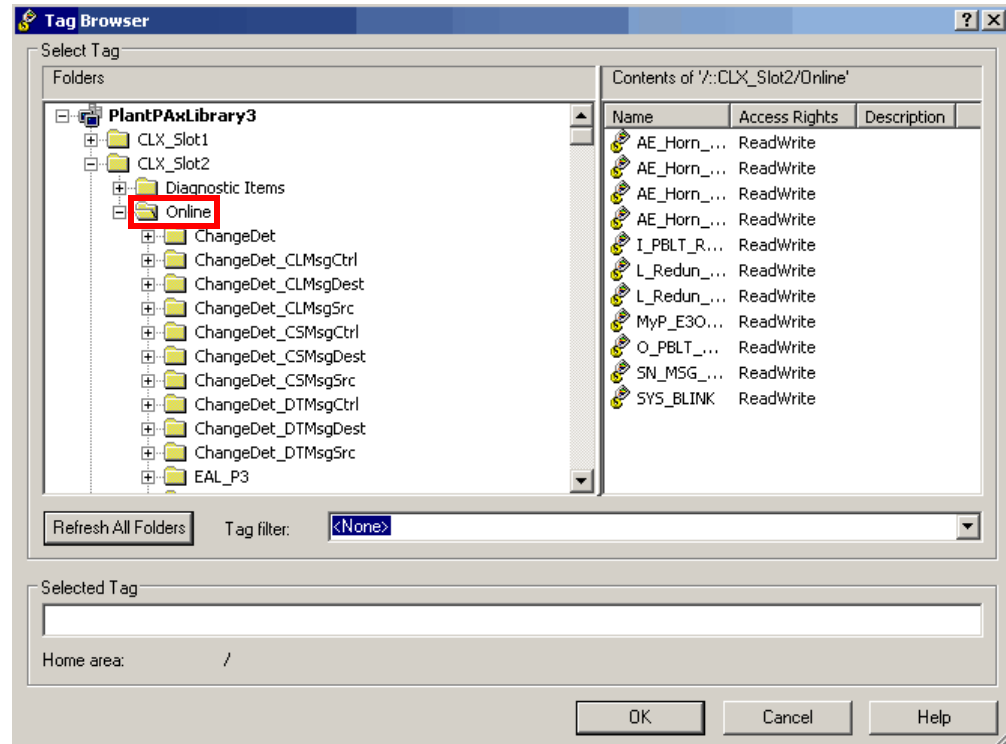
The Tag Browser dialog box appears.



6. Click Refresh All Folders.

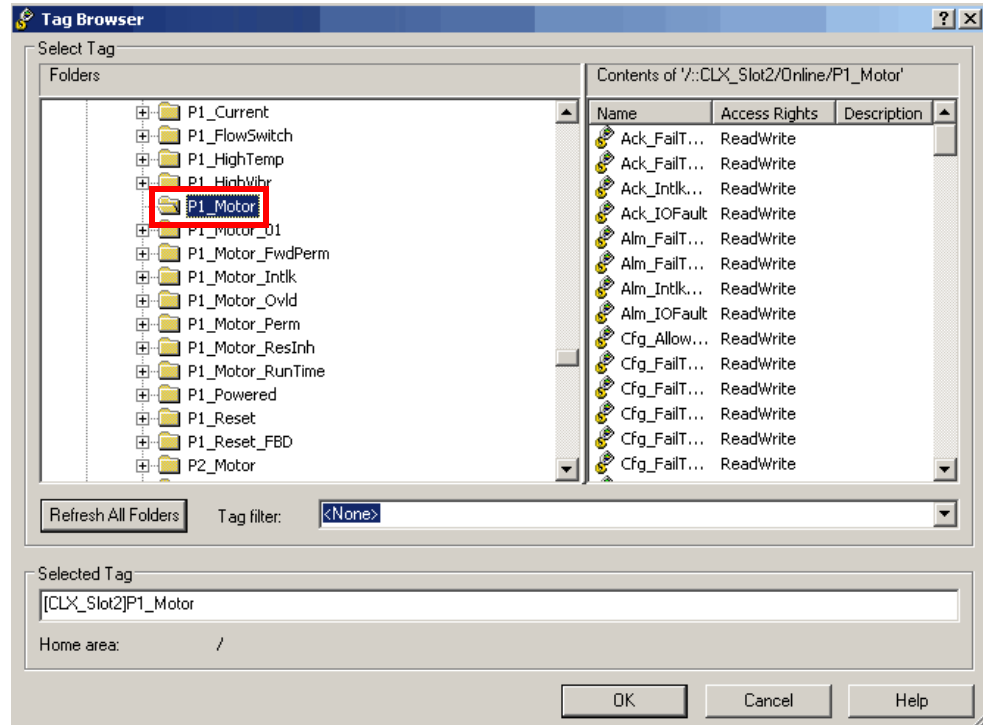
7. Expand the shortcut for the project controller and then click Online.

TIP If your controller shortcut in FactoryTalk View has an associated Logix project (.ACD) file, you can navigate to the tag using the offline folders. You will need to do this if the online controller is not available.



8. To select the tag for the Add-On Instruction, expand and scroll down the Online menu.

IMPORTANT The backing tag for the Add-On Instruction is a folder.



9. Click OK.
10. To configure the #103 parameter, copy the path part of #102 into the Value column:
 - No { } (curly braces)
 - No tag
 - Just /:[] (area and shortcut)

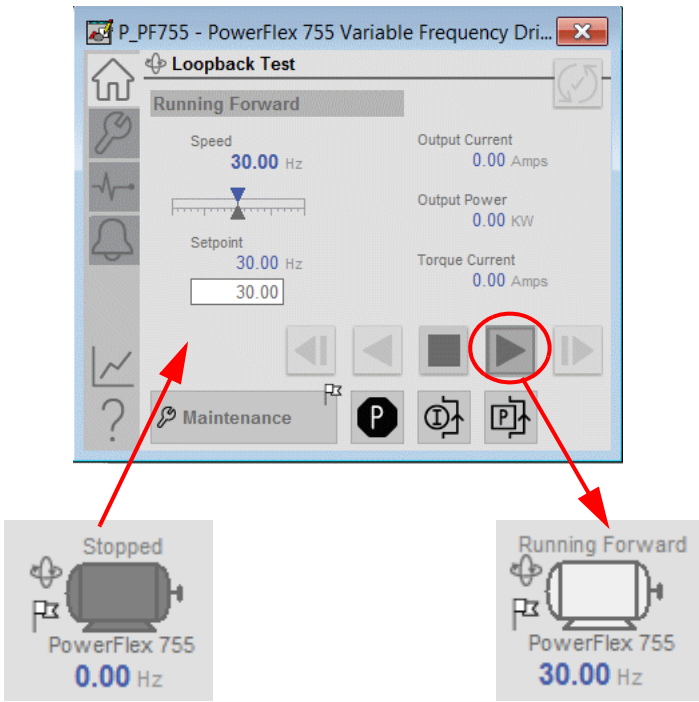
IMPORTANT If the Tag is Program scoped, include the scope in the Path. For example, if the Tag is "[CLX_Slot2]PROGRAM:Unit1.P1_Motor}", then the Path is "[CLX_Slot2]PROGRAM:Unit1." **Be sure to include the period after the program name.**

11. Click the Value column for the fifth parameter and enter a value (0, 1, 2) per the desired display.

The following image is an example of parameter values.

Global Object Parameter Values				
	Name	Value	Tag	Description
1	#102	{[CLX_Slot2]P1_Motor}	...	Motor Tag (P_MotorHO)
2	#103	[CLX_Slot2]	...	Path (include program scope if tag is a program scope tag)
3	#120		...	Additional display parameter (e.g. /X100 or /CC) (optional)
4	#121		...	Additional display parameter (e.g. /Y100) (optional)
5	#122	2	...	0 = Always show Faceplate; 1 = Show Quick Display for users

- 12. Click OK.
- 13. Click Save.
- 14. On an HMI display, access the display that contains the global object.
- 15. Log in as a user with operating privileges.
- 16. To access a faceplate, click the global object.



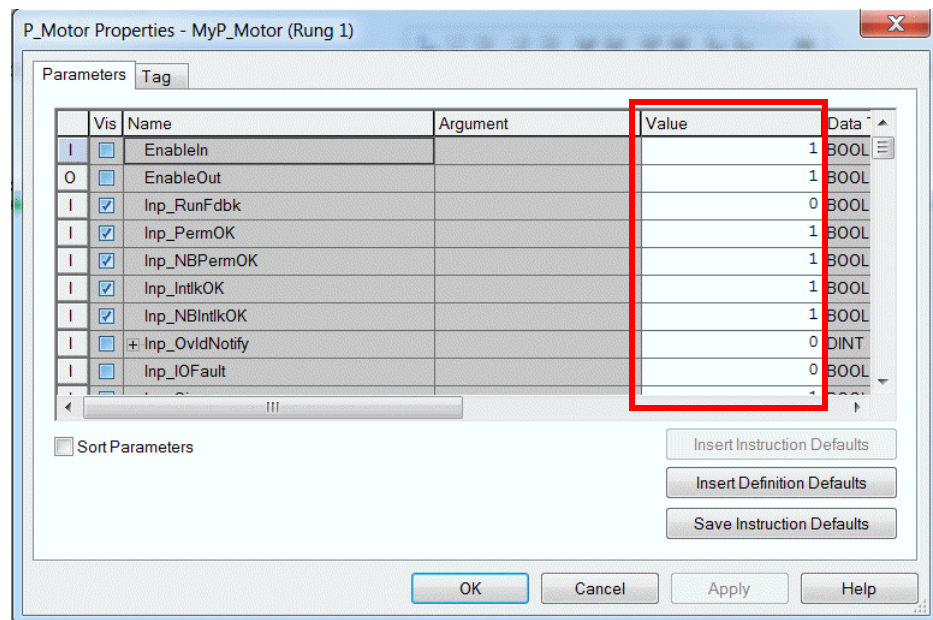
- 17. To operate the device, click the command buttons on the faceplate.

TIP See [Appendix C](#) to change the color on a display element or faceplate.

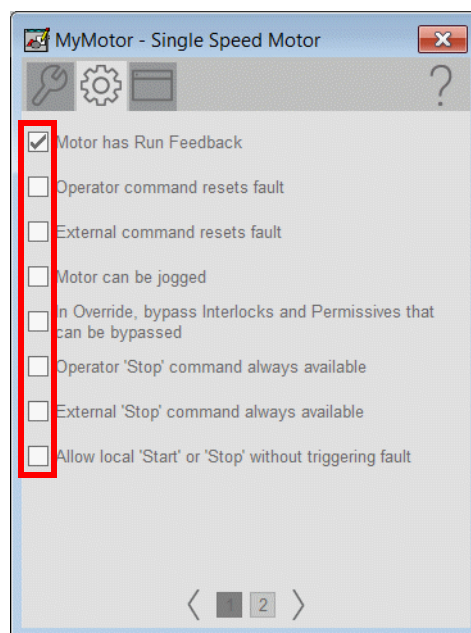
Device Configuration

You can configure the device parameters by doing either of the following:

- Type values in the Parameters tab of the Studio 5000 Logix Designer® application



- Make selections and enter data for each option on the Engineering tab of the instruction HMI faceplate



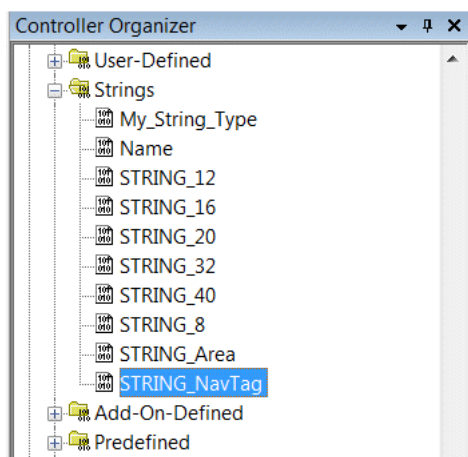
Modify Navigation and Area Tags

You can change the size of all navigation tags and area names globally without having to change the definitions of each Add-On Instruction. By changing the definition of these STRING types, you can use longer names for backing tags and area names.

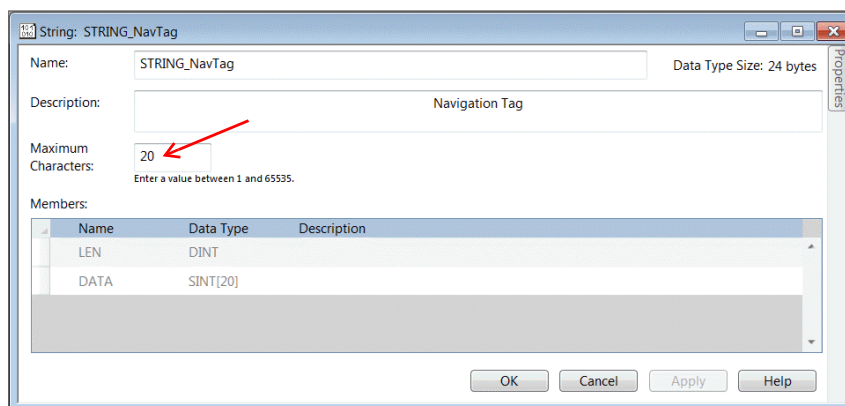
IMPORTANT Changing the definition of a string type clears the values of all strings of that type. Decide on string sizes early in the project.

Complete these steps.

1. In the Controller Organizer, click '+' to expand the Strings folder.
2. Double-click STRING_NavTag.



A dialog box appears that provides access to a text box for modifying the number of characters in the string.



3. Type a number (1...40) into the Maximum Characters text box.

You can also use the Up and Down counter arrows.

Any length works, but lengths from 4...40 in multiples of 4 (4, 8, 12, ...36, 40) work the best. Controller tag names have a 40-character limit.

You can also shorten tag names to save some controller memory.

IMPORTANT We suggest that your device tag names are no longer than 28 characters. This limit helps prevent corresponding tags ('_Intlk', '_Perm', '_Reslnh', '_RunTime', and '_Valve Stats') from exceeding the 40-character tag name limit.

Increasing the length of tag names requires corresponding changes to faceplate global object files in FactoryTalk View software. You are responsible for modifications to the Nav Tag string entry and string display fields in FactoryTalk View software.

Notes:

Online Configuration Tool

The Online Configuration Tool is a standalone, Microsoft Excel-based spreadsheet. The multi-tabbed spreadsheet lets you access configuration tags of Library objects (Add-On Instruction parameters and local tags) for multiple instances of a library instruction simultaneously by using OPC.

IMPORTANT The Configuration Tool is supported **only** in 32-bit editions of Microsoft Excel software. The tool uses RSLinx® Classic OPC/DA for its online communication, and RSLinx Classic OPC Server is a 32-bit application. (The tool works fine on 64-bit Microsoft Windows operating systems, but it requires a 32-bit installation of Microsoft Excel/Microsoft Office.)

This tool lets you make bulk changes more easily, especially for local configuration tags like strings, instead of modifying each tag separately for each Add-On Instruction instance.



WARNING: The spreadsheet is used to modify parameters by using an online controller. However, the controller **must not** be in a production environment controlling machinery or processes. The spreadsheet works best with the controller in Program mode, so we strongly suggest it be used that way to avoid unintended control changes to equipment in service.

Before You Begin

The spreadsheet uses RSLinx Classic software OPC, so you need a suitably licensed copy (OEM, Gateway) of the software. You cannot use the Lite version of RSLinx Classic software.

We suggest that you make a back-up copy of your application before completing the following steps. When using the spreadsheet, all of your Add-On Instruction instances (backing tags) are to be created in your Studio 5000 Logix Designer® application project (.acd file) and downloaded to your controller.

Configure Parameters By Using a Spreadsheet

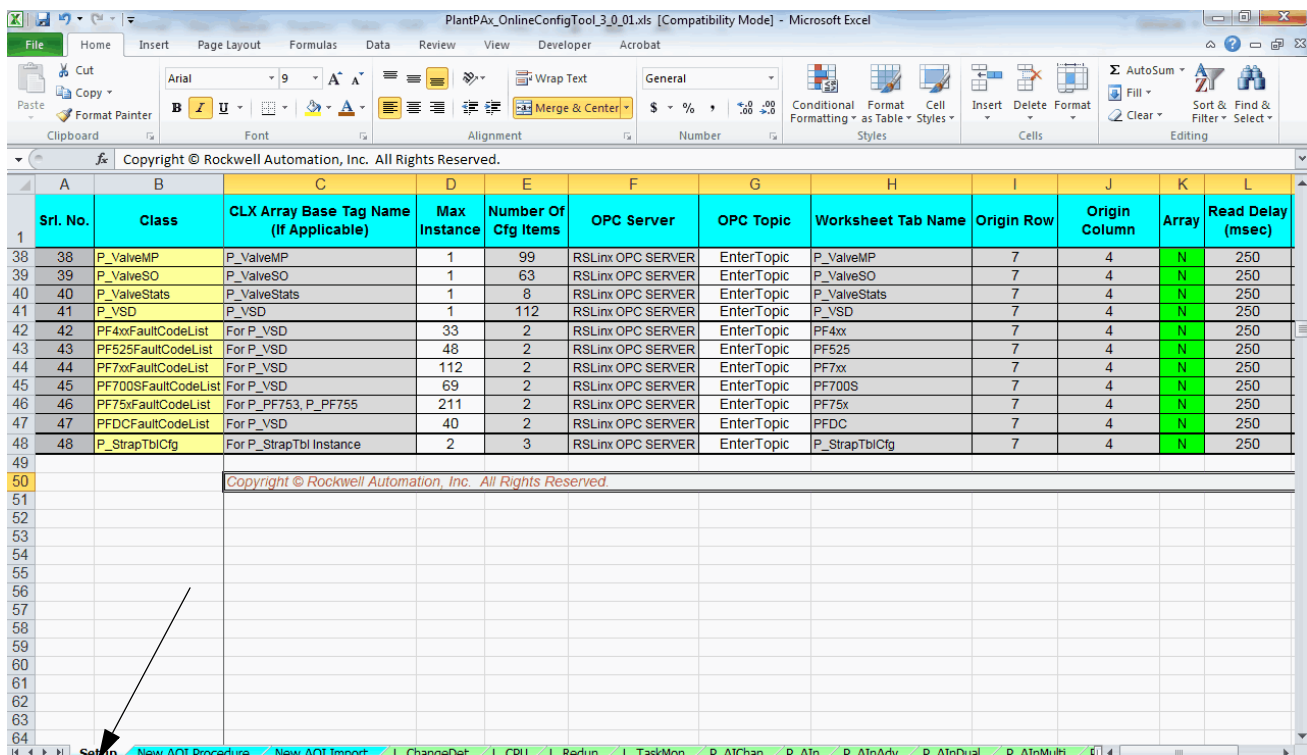
The following procedure is for using a spreadsheet to upload (save) and download (restore) the configuration (.Cfg) parameters from library instances in an online controller.

Local tags also can be configured through the HMI faceplates or in Logix Designer application. Open the instruction logic of the Add-On Instruction instance and then open the Data Monitor on a local tag.

IMPORTANT See the Warning on [page 143](#) that applies for the controller that is being used before starting these steps.

1. Download the Rockwell Automation® Library from the Product Compatibility and Download Center at <http://www.rockwellautomation.com/rockwellautomation/support/downloads.page>.
2. After the Library is downloaded, open Tools & Utilities and double-click PlantPAx® Online Configuration Tool.
3. Open the Excel spreadsheet.

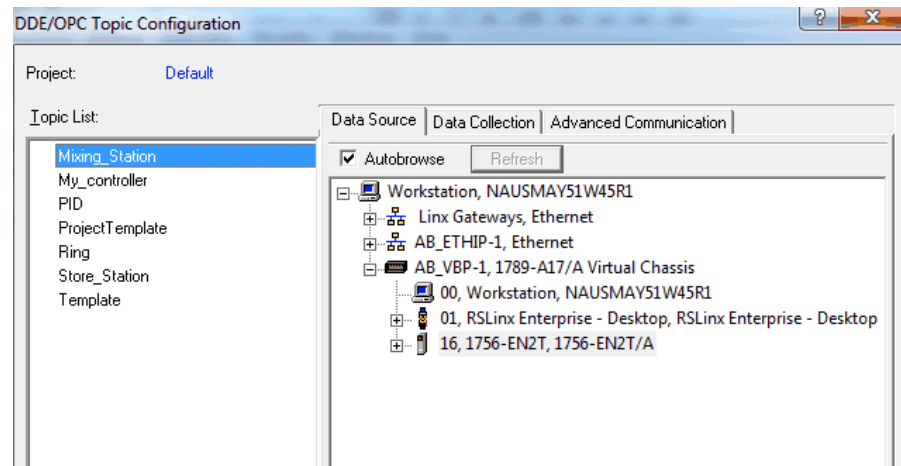
The spreadsheet opens in the default Setup folder. (Folders are shown at the bottom of the screen).



4. Save the file with a filename that matches the name of the controller.
5. Open RSLinx Classic software.

6. From the DDE/OPC menu, choose Topic Configuration.

The DDE/OPC Topic Configuration dialog box appears.



7. In the left pane, find the Topic that points to your controller or create one.
8. Open your new spreadsheet file.
Allow use of the active content if you get a warning message.
9. Type the Topic name in all rows of column G of the Setup sheet.
Be sure that you are going to the correct controller.
10. Open an Add-On Instruction instance by clicking the respective tab name at the bottom of the screen.
11. In column C, start in row 10 and type the backing tag names for your Add-On Instruction instances.

	A	B	C	D	
1		P_DIn: Discrete Input	Read From CLX:		
2			Send To CLX:		
3					
4			Description:		Description f
5			Usage:		
6			Data Type:		
7				(Origin)	
8	Unit	Tag Description	TagName	Instance	
9			Default Values:	0	
10			FS_ 101	1	
11			TSM 102	2	
12				3	
13				4	
14				5	

TIP You can export the tags for your application from Studio 5000 Logix Designer to a .CSV file, then open the file in Excel. Sort the tags by Data Type, then copy the tag name from the export sheet into Column C of the configuration sheet.

If your Add-On Instruction instances are in Program-scope tags, you can get to them as well. The TagName syntax is:

Program:<program_name>.<tagname>

12. On the Setup sheet, type the number of instances of each Add-On Instruction in column D.

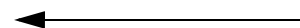
IMPORTANT Make sure that you switched to the Setup sheet for [step 12](#). Do **not** change the 'Number of Cfg Items' in Column E or any of the other data on the Setup sheet.

The following example shows two instances that are entered on the Setup sheet because we entered two instance tags in column C in the P_DIn sheet.

	A	B	C	D	E
	Srl. No.	Class	CLX Array Base Tag Name (If Applicable)	Max Instance	Number Of Cfg Items
4	3	L_Redun	L_Redun	1	37
5	4	L_TaskMon	L_TaskMon	1	15
6	5	P_AlChan	P_AlChan	1	48
7	6	P_Aln	P_Aln	1	96
8	7	P_AlnAdv	P_AlnAdv	1	148
9	8	P_AlnDual	P_AlnDual	1	138
10	9	P_AlnMulti	P_AlnMulti	1	145
11	10	P_AOut	P_AOut	1	46
12	11	P_D4SD	P_D4SD	1	85
13	13	P_DIn	P_DIn	2	24
14	14	P_DoseFM	P_DoseFM	1	73
15	15	P_DoseWS	P_DoseWS	1	72
16	16	P_DOut	P_DOut	1	69
17	17	P_Fanout	P_Fanout	1	110

13. Return to the Add-On Instruction sheet, in our example its P_DIn, and click the Read From ControlLogix® box near the top of the spreadsheet.

C	D
Read From CLX:	☐
Send To CLX:	☐
Description:	
Usage:	
Data Type:	
TagName	(Origin)
Default Values:	0
FSL_101	1
TSH_102	2



14. Click Yes to the overwrite message to populate the row for that instance.
15. Make your modifications, for example change the strings, and click the Send To ControlLogix to send the new configuration to the controller.
16. Check the data in the controller and save the project (to a new .acd file is best).

Make sure that you upload tag values when you save the project.

PlantPAx Alarm Builder Tool

The PlantPAx® Alarm Builder tool expedites the process of creating the FactoryTalk® View SE and FactoryTalk View ME alarms that work with the Rockwell Automation® Library of Process Objects. With this tool you can:

- Define a project that contains multiple controller ACD files and associated FactoryTalk View HMI applications.
- Organize Logix code, Logix tags, and FTVIEW SE HMI displays in a Process Tree organizer. The tree structure creates AE alarm groups, select tags to use in the AE and ME alarm builders, and SILAlarm import files.
- Tag Data Editing, Import, Export:
 - Edit Logix tag data in off-line controller ACD files.
 - Export and import Logix tag data to/from text files.
 - Create Microsoft Excel workbook for online OPC tag data read/writes.
- Create AE XML import files by using tag data from controller files.
- Create ME XML import files by using tag data from controller files.

Before You Begin

The Rockwell Automation Library of Process Objects contains the ‘PlantPAx Configuration Tools for Tags, Alarms, and Historian’ software. This software includes The PlantPAx Alarm Builder tool. The Rockwell Automation Library of Process Objects is available from the Product Compatibility and Download Center at

<http://www.rockwellautomation.com/rockwellautomation/support/downloads.page>.

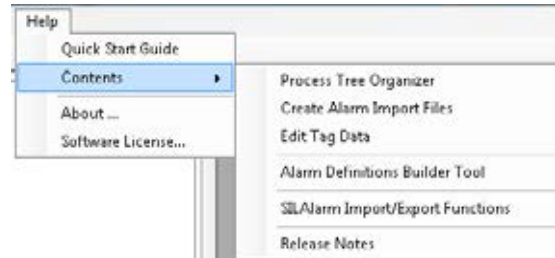
You must also have Studio 5000 Logix Designer® application or RSLogix 5000® software installed. Logix Designer services, which are part of the software installation package, convert a controller ACD file to an XML file. The XML conversion lets the Alarm Builder software to read the data and create alarm tags.

Additional documentation resources provide information on how to configure alarm tags, data types, and other functions that Alarm Builder can perform.

In the Help menu on the PlantPAx Alarm Builder and Tag Data Edit Tool window, click Quick Start Guide to access this document.

In the Help menu, click Contents to access these documents:

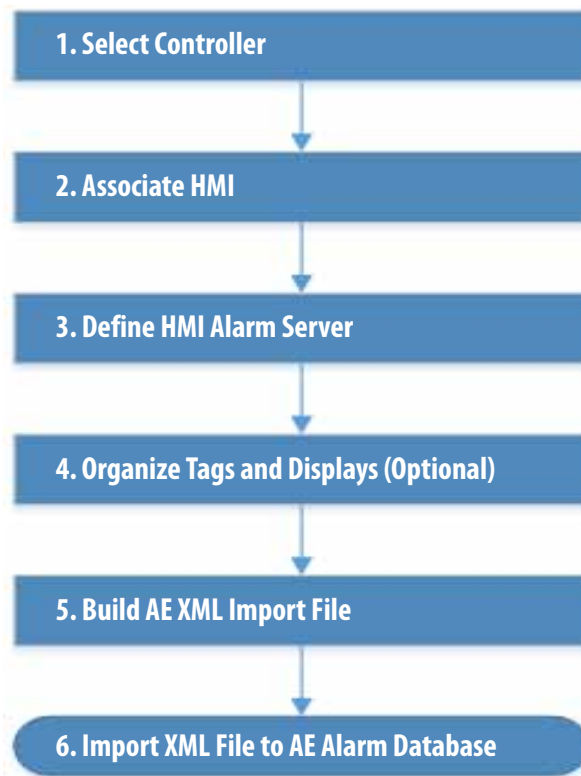
- Process Tree Organizer
- Create Alarm Import Files
- Edit Tag Data
- Alarm Definitions Builder Tool
- SILAlarm Import/Export Functions
- Release Notes



Build AE Alarms

The diagram outlines the procedures for creating FactoryTalk View SE software alarms. The procedures in this section are in the same order as the headings in the diagram.

Figure 8 - Alarm Builder AE Workflow



1. Select Controller

This section describes how to associate a Logix controller to FactoryTalk View HMI servers and data servers so the Alarm Builder tool can create HMI alarm tags. This tool obtains server information from specified FactoryTalk View directories, which use default FactoryTalk View installation settings. The HMI server and data server information can be entered manually, if necessary.

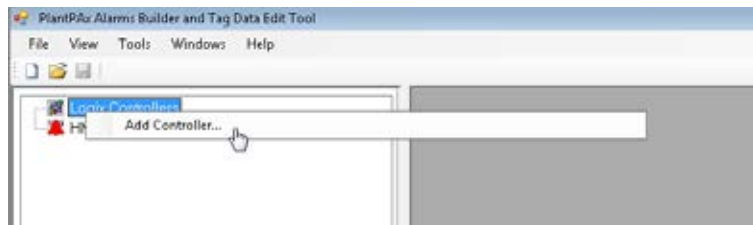
Follow these steps to add a controller to the Alarm Builder software tool.

1. From the Tools & Utilities folder in the downloaded library files, open the Alarm Builder tool.

A Quick Start PDF file opens in a separate Adobe PDF Reader window. Review the Quick Start guide and close the Adobe window.

The PlantPAx Alarm Builder and Tag Data Edit Tool window appears.

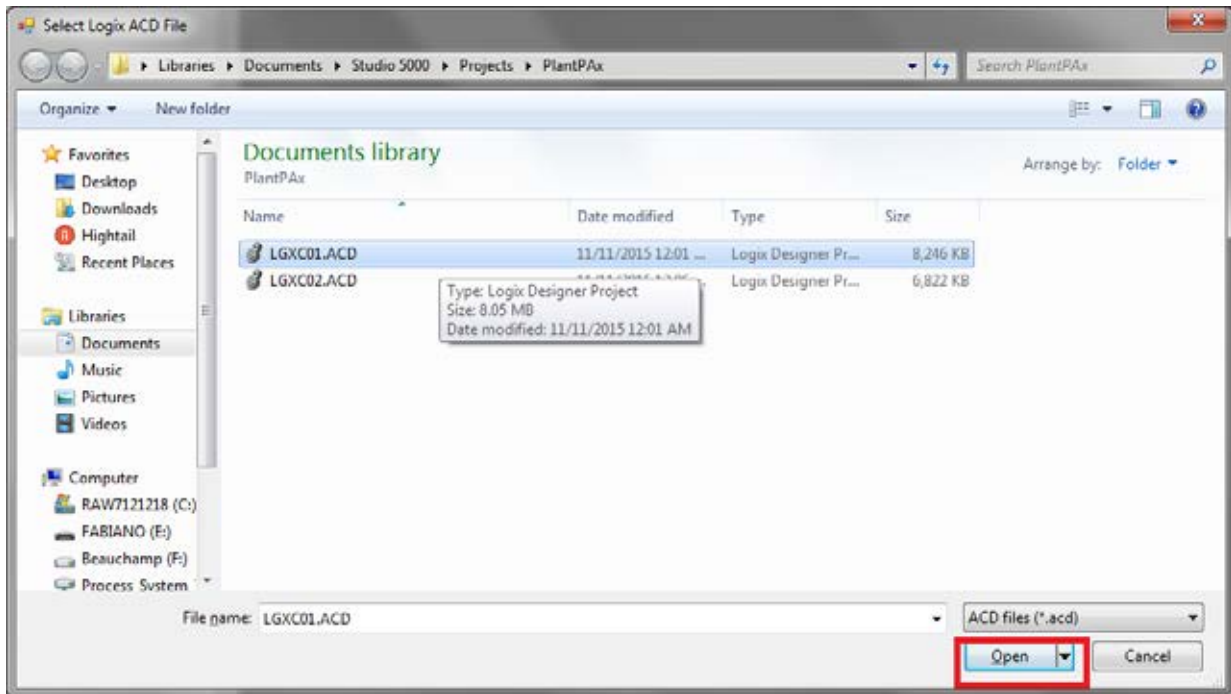
2. Right-click Logix Controllers and choose Add Controller.



The Select Logix ACD File dialog box appears.

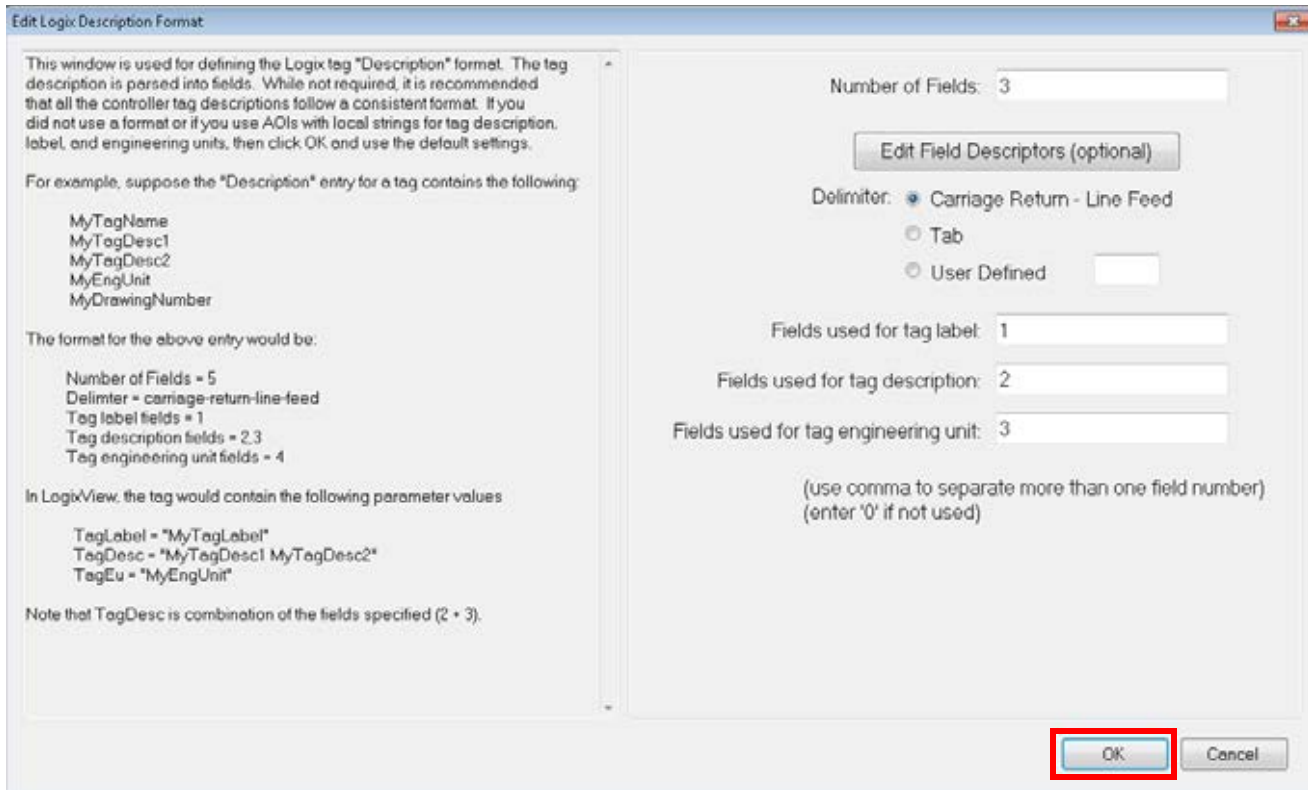
3. Navigate to the folder that stores the controller project file.

4. Select the ACD file and click Open.



A message window displays if the tool cannot determine the Logix Designer version of the selected ACD file.

5. Type the Logix Designer application version and click OK if prompted. Otherwise, proceed to step 6.
6. On the Edit Logix Description Format dialog box, click OK to use the defaults if the following conditions apply:
 - You do not use a format to describe controller tags
 - You use Add-On Instructions with local strings that describe the tag label and engineering units



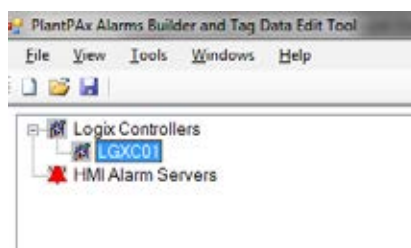
7. If the preceding conditions do **not** apply, type data into the text boxes to describe the tags and click OK.

A dialog box appears to update severity values if you are replacing Version 3.0 Add-On Instructions with Version 3.1 or later instructions.

See [Replace Logix Tag Severity Values on page 152](#) for details.

TIP The conversion of the controller project file to an XML format can take several minutes for each controller file.

When the conversion is complete, the controller file is added to the configuration tree under Logix Controllers.



8. Repeat [step 2](#) through [step 6](#) to add multiple controllers to the project.

Multiple controllers can be associated to different HMI alarm servers. For example, you can have an AE alarm server for several controllers in a SE project and a ME alarm server for skid-mounted controllers.

Replace Logix Tag Severity Values

Use the procedures in this section if you have upgraded an ACD file with Rockwell Automation Library Add-On Instructions, Version 3.0, to Version 3.1 or later and want to change the severity values. The severity values for the Version 3.1 or later instructions are now a range as shown in [Table 8](#).

Table 8 - Logix Tag Severity Values

Version 3.0 and Earlier		Version 3.1 or Later		Definitions	Alarm Color
Logix Severity	A&E Severity	Logix Severity	A&E Severity		
1	1	1...250	1...250	Low	Blue
2	251	251...500	251...500	Medium	Yellow
3	501	501...750	501...750	High	Red
4	751	751...1000	751...1000	Urgent	Magenta

The Version 3.1 instruction range of 1...1000 (INT data type) is consistent with the default AE severity value range.

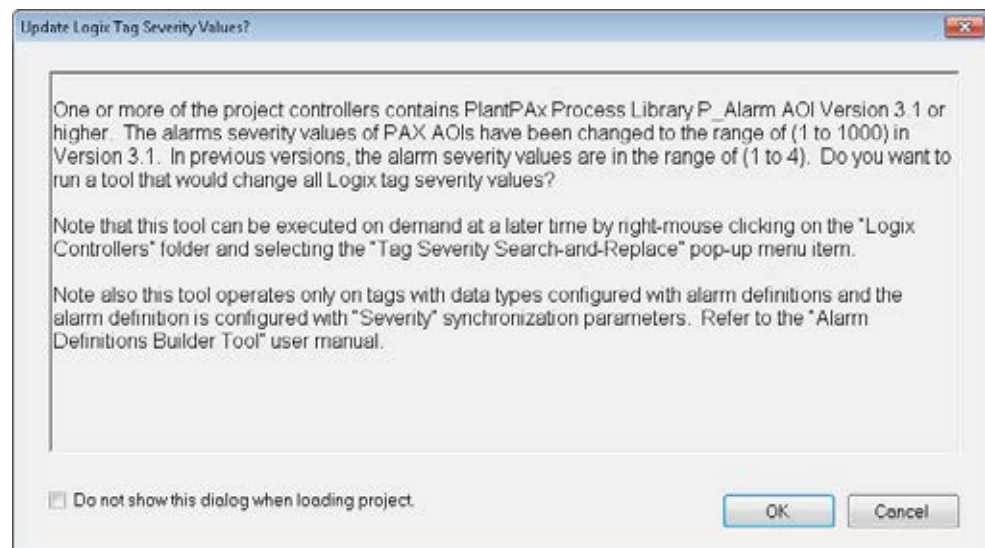
The Version 3.0 and earlier instructions have a severity range of 1...4 (SINT data type). When an ACD file that contains Version 3.0 instructions is updated with Version 3.1 instructions, the tags retain their existing severity values (1...4).

IMPORTANT For FactoryTalk View software, version 8 and later, the AE severity can be configured by using a Logix tag address. The alarm server reads the severity from the Logix tag during runtime. This function lets the severity value be changed from the HMI faceplate; no number mapping is required.

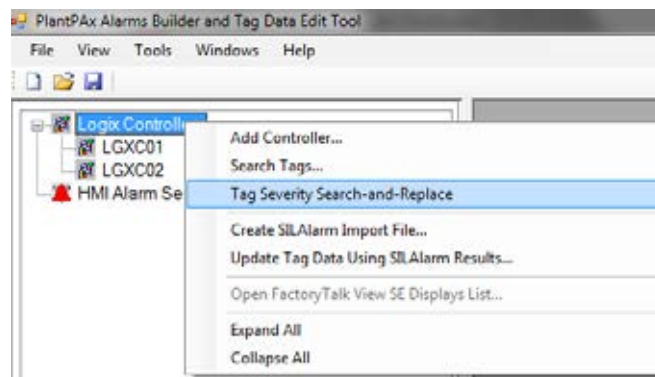
For FactoryTalk View software, version 7 and earlier, the AE severity value is a fixed numeric value. It cannot be changed from the tag faceplate; it can be changed only from the AE tag database.

If you load a controller that has the P_Alarm Add-On Instruction with Version 3.1 or later, an Update or Logix Tag Severity Values dialog box appears. The dialog box has a tool to change severity tag values.

1. Do one of the following:
 - a. Click Cancel to run the tool later.



IMPORTANT To reaccess the prompt, right-click Logix Controllers and choose Tag Severity Search-and-Replace from the pull-down menu.



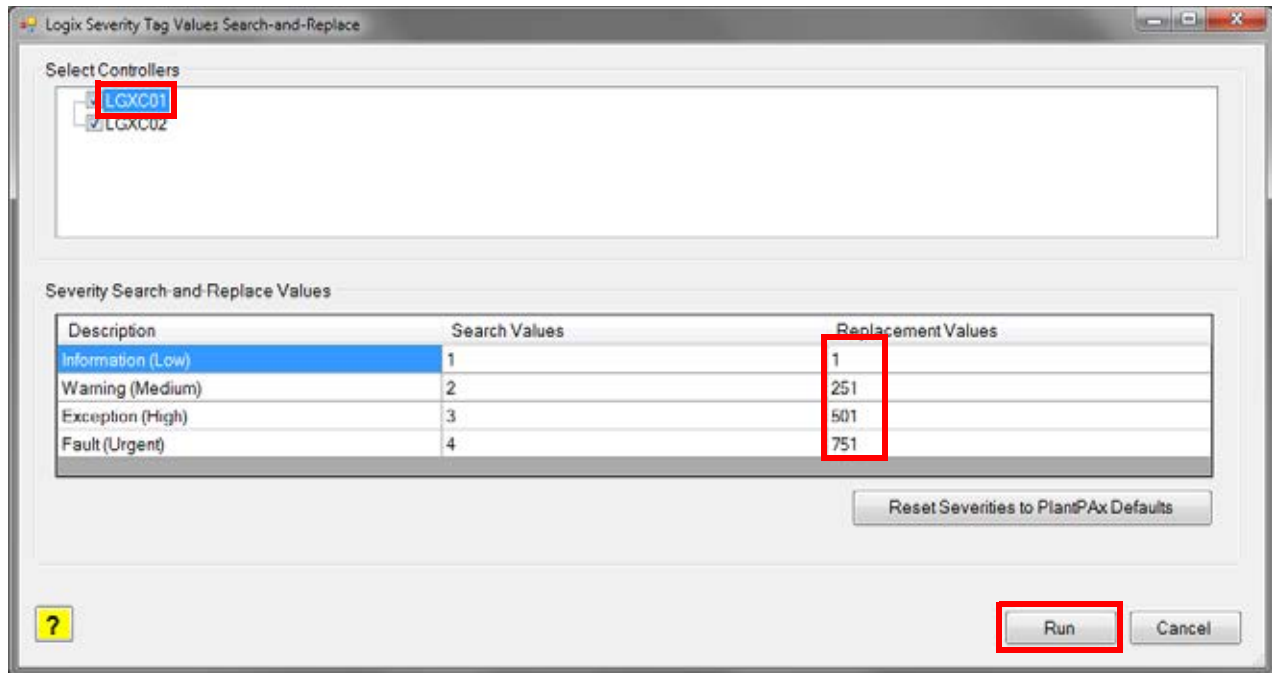
- b. To replace the severity values, click OK.

The Search-and-Replace Tool dialog box appears.

2. In the top box, select a controller.

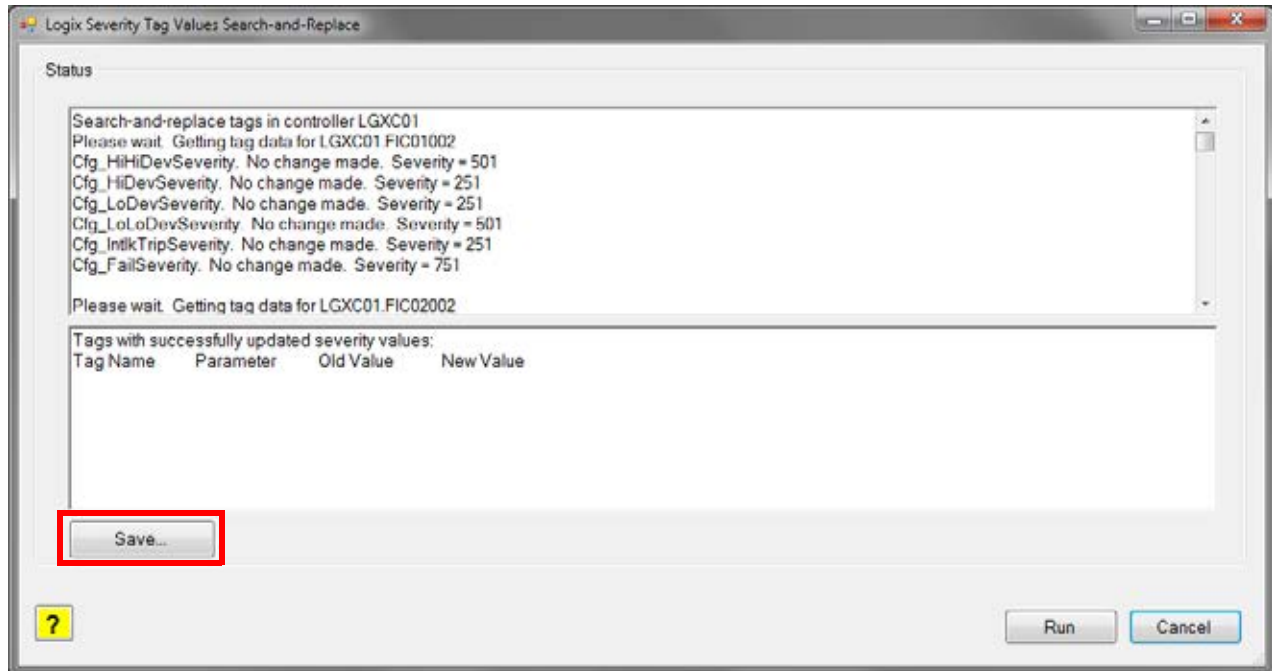
Controllers that contain Version 3.1 or later of the P_Alarm Add-On Instruction are automatically selected.

3. In the bottom box, change the values by typing in the respective cells.
4. Click Run.



Two status boxes appear in the Logic Severity Tag Values Search-and-Replace window during the replacement operation.

- The top table provides status and error messages.
- The bottom table provides a listing of tag values that are changed, if any.

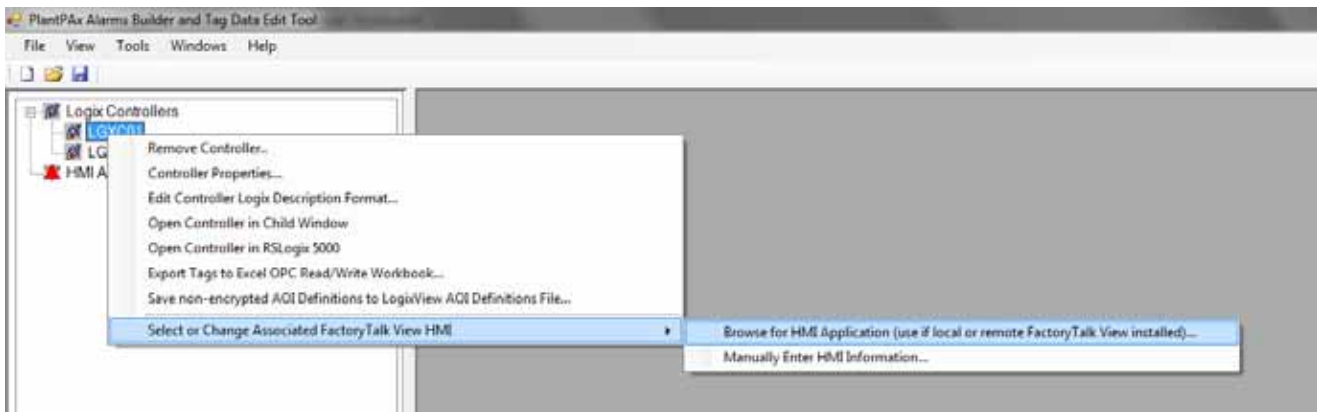


5. Click Save.

2. Associate HMI

The HMI association provides the information (data area name, device shortcut name) needed in the AE tag addresses.

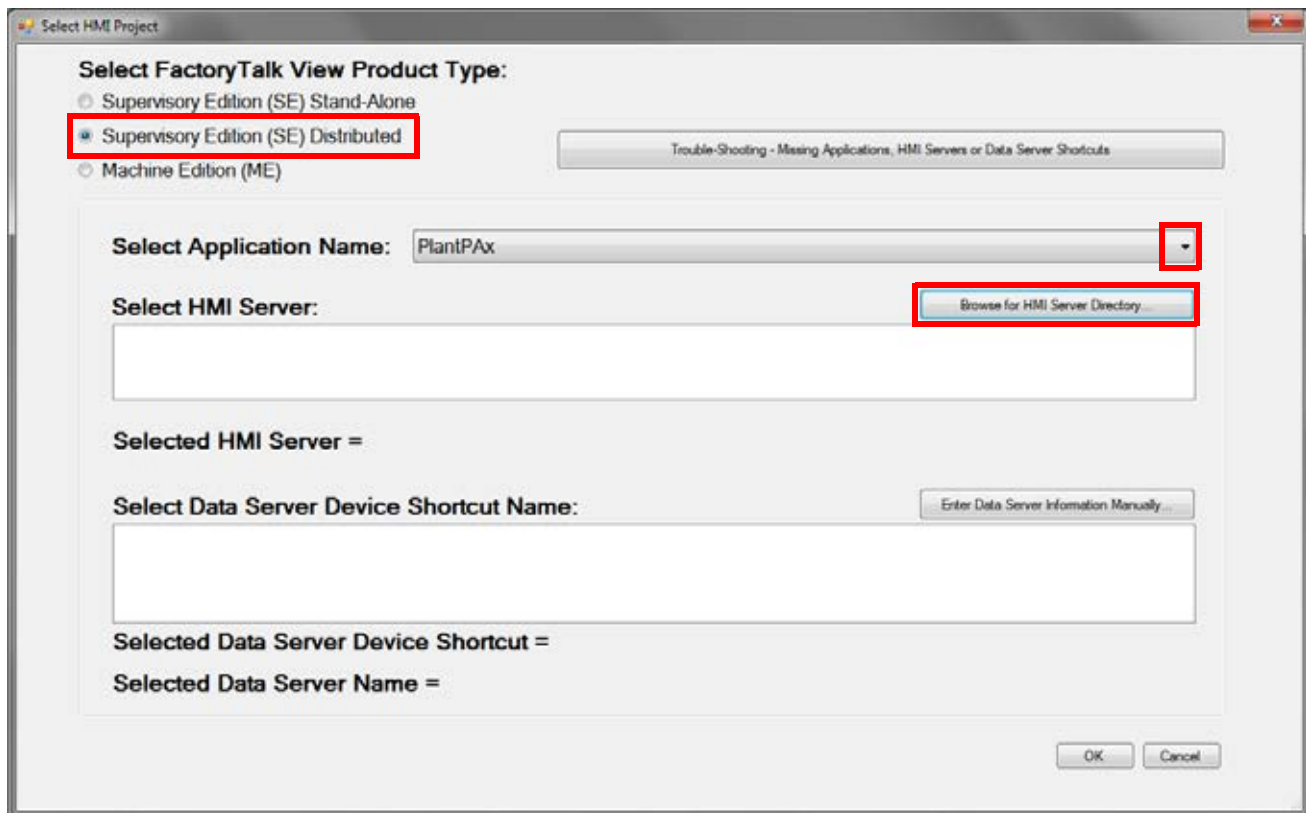
1. If you have multiple controllers, click the '+' to expand the Logic Controllers node.
2. Right-click a controller file, and from the pull-down menu choose Select or Change Associated FactoryTalk View HMI>Browse for HMI Application (use if local or remote FactoryTalk View software installed).



The Select HMI Project dialog box appears.

3. Click the Supervisory Edition (SE) Distributed product type.
4. From the Select Application Name pull-down menu, choose the application name.

- Click Browse for HMI Server Directory.

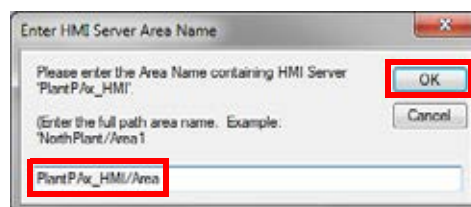


- Navigate to and select the HMI Server and click OK.

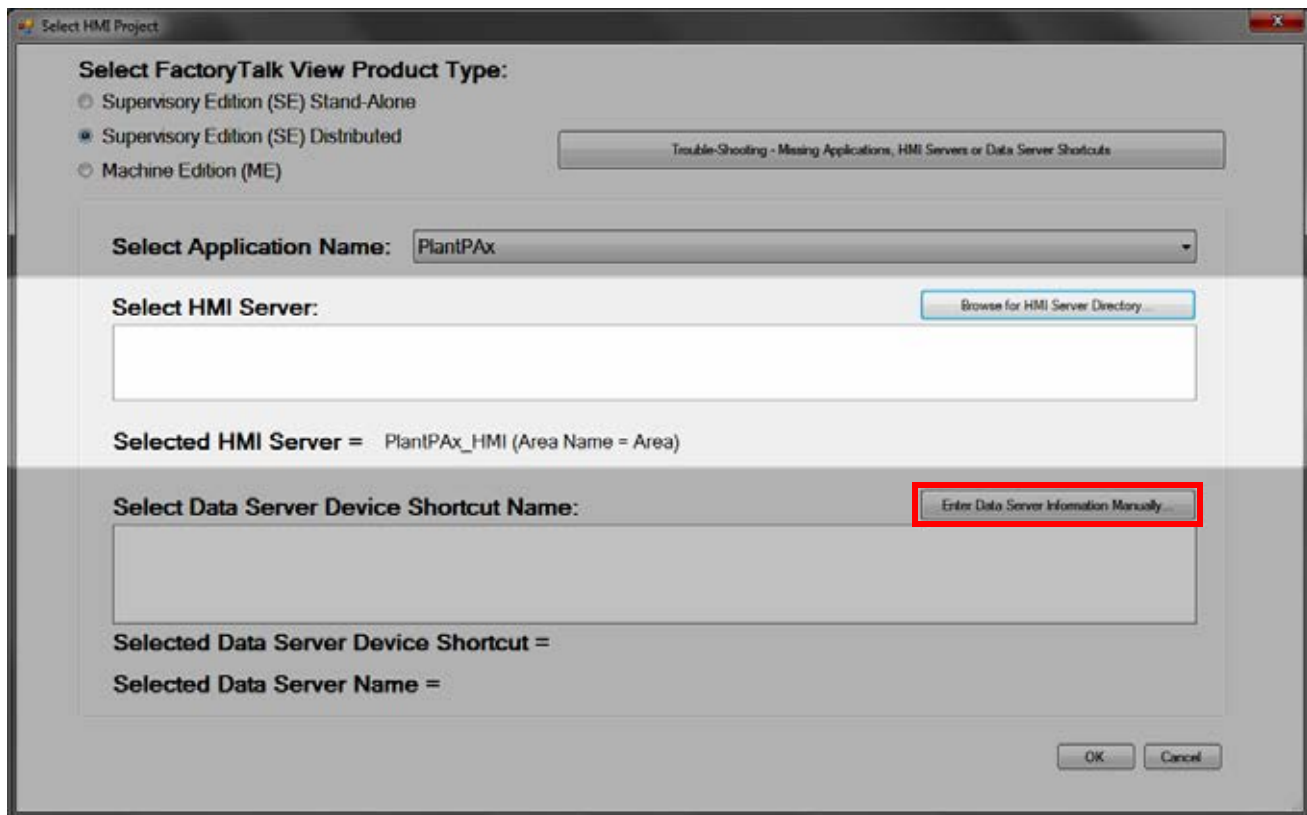


The Area Name dialog box appears.

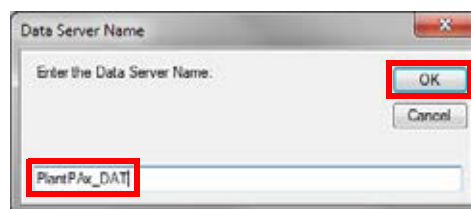
- Type the path and the area name (for example: the HMI project name) and click OK.



The HMI Server name appears below the Select HMI Server box.



8. Click Enter Data Server Information Manually.
9. Type the Data Server name (for example: the Data area in the HMI project) and click OK.



The Shortcut Name dialog box appears.

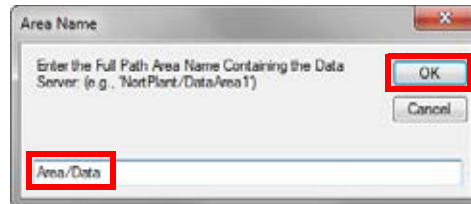
10. Type the shortcut name, which is typically the controller name, and click OK.



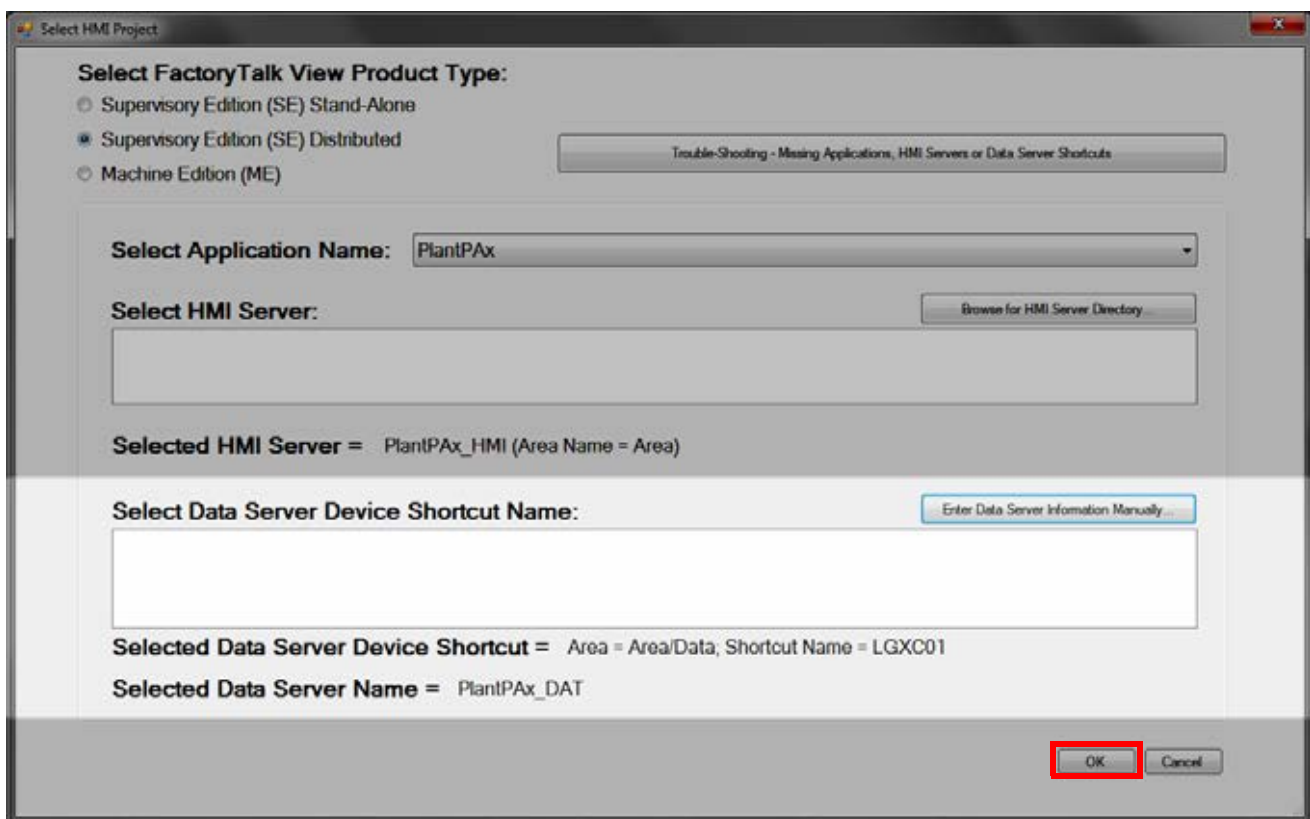
The area name dialog box appears.

11. Type the area name and click OK.

IMPORTANT Do not add a leading slash (for example: '/Area/Data').

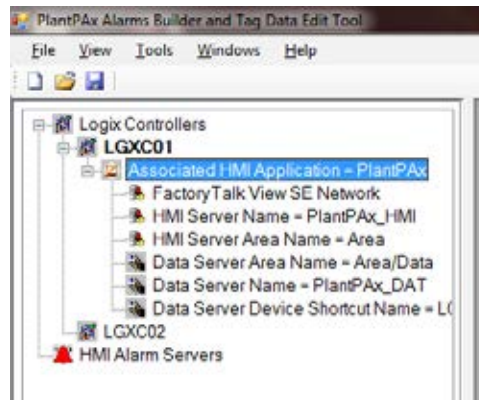


The Selected Data Server Device Shortcut and Selected Data Server Name appear on the Select HMI Project dialog box.



12. On the Select HMI Project dialog box, click OK.

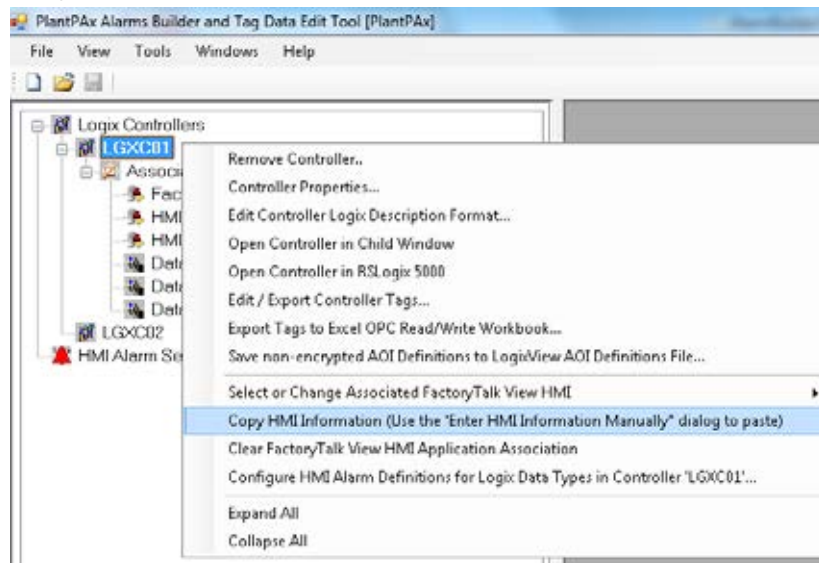
The associated HMI application with its valid information appears under the selected controller.



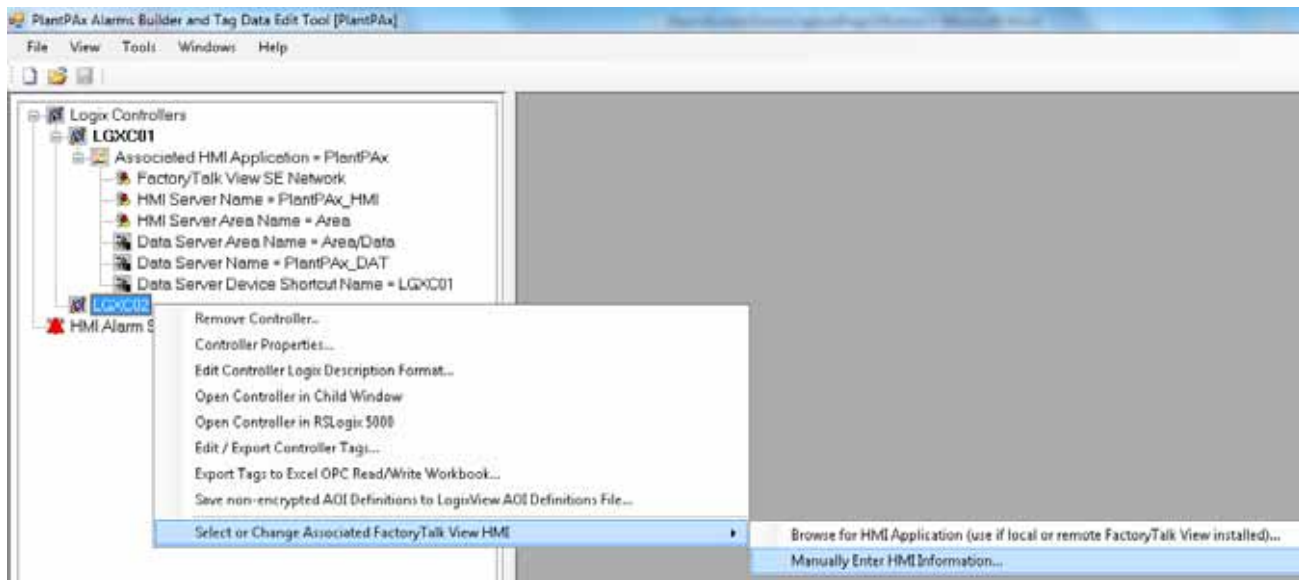
Configure Other Controllers

In this section, you can configure other controllers with the same servers.

1. Right click LGXC01 and select 'Copy HMI Information.'

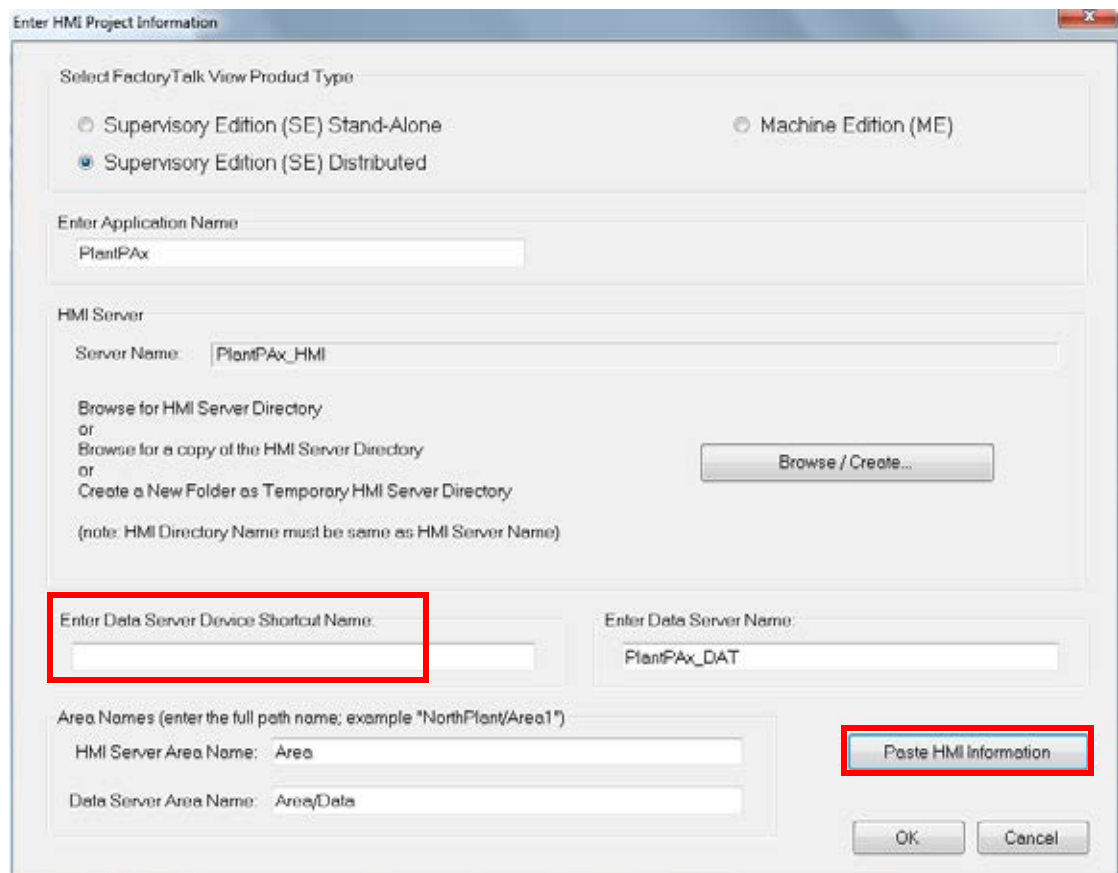


- Right-click LGXC02 and select 'Select or Change Associated FactoryTalk View HMI' > 'Manually Enter HMI Information' to open the HMI information configuration window.



The Enter HMI Project Information dialog box appears.

- Click 'Paste HMI Information'.



The HMI Server Area Name and the Data Server Area Name appear.

4. Type the 'Data Server Device Shortcut Name' (LGXC02 in the example) as controller name.
5. Click OK

3. Define HMI Alarm Server

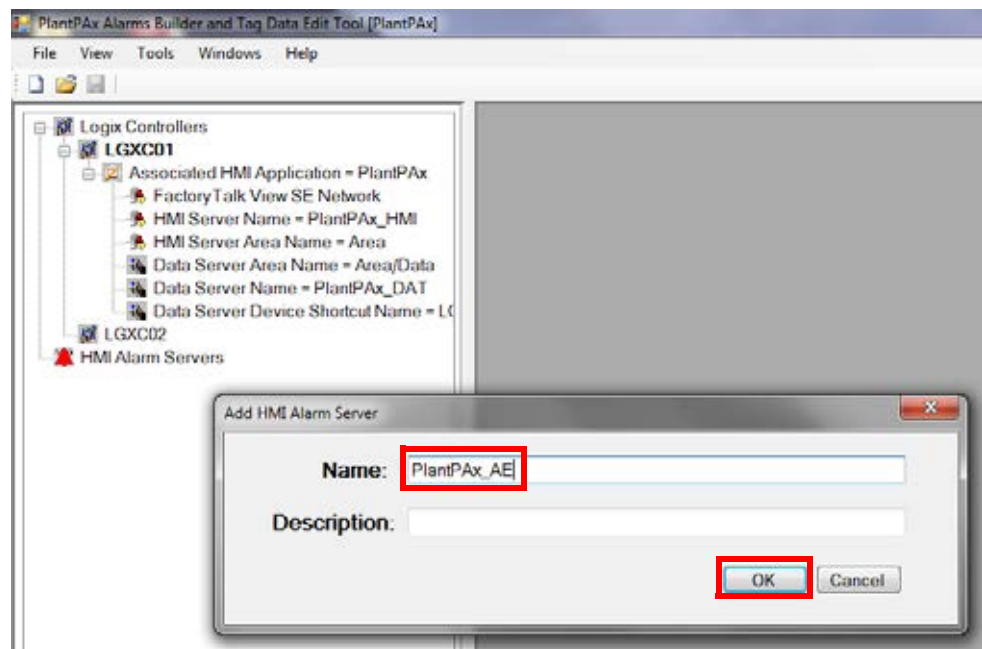
This section shows how to add an alarm server to organize the controllers that you want to create AE alarm polling tags. This alarm server has no association to the AE alarm server that you configure in FactoryTalk View Studio software.

Follow these steps.

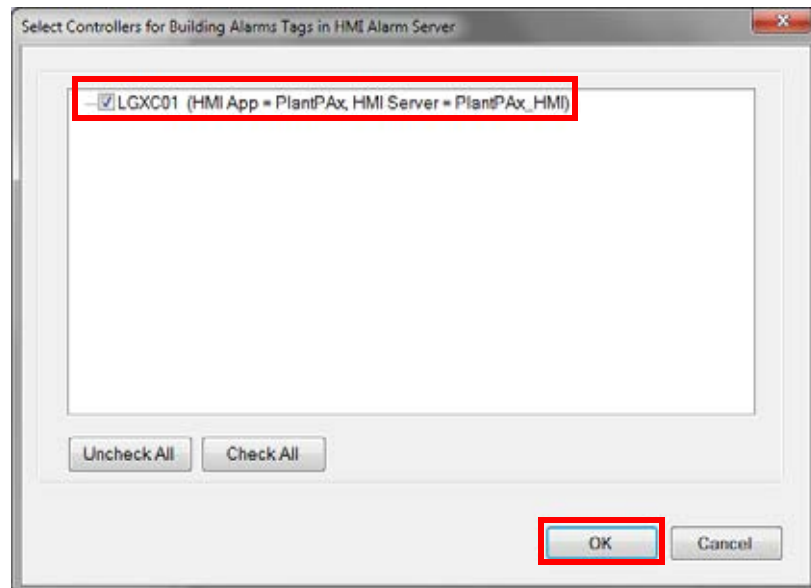
1. In the Alarm Builder configuration tree, right-click HMI Alarm Servers and choose Add Alarm Server.

The Add HMI Alarm Server dialog box appears.

2. Type a name and a description (optional).
3. Click OK.



The Select Controllers for Building Alarms Tags in HMI Alarm Server window appears.



4. To select the controller that you added, check the checkbox and click OK.

The alarm server name appears under HMI Alarm Servers in the configuration tree.

4. Organize Tags and HMI Displays (Optional)

The Alarm Builder Process Tree Organizer can be used to organize Logix tags, Logix code, and SE HMI displays in user-defined folders. The Process Tree folder hierarchy is used for creating AE alarm groups when building the AE XML import file. AE alarm groups are available with FactoryTalk View software, version 8.1 or later.

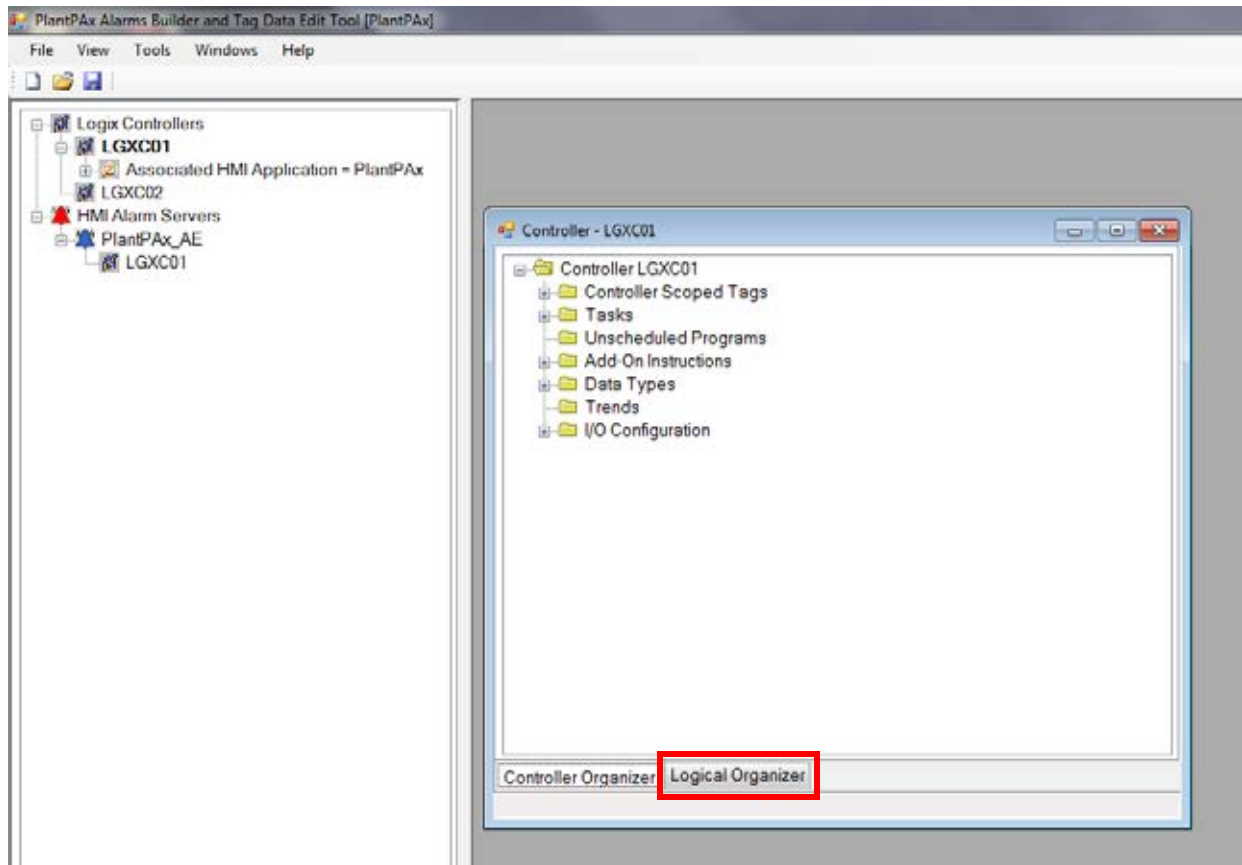
This section provides instructions on the use of the controller Logical Organizer to configure the Process Tree folders. Other methods of configuring the Process Tree folders are available – refer to the Alarm Builder 'Process Tree Organizer' user manual for details.

It is not required to configure the Process Tree to build the AE XML import file. However, we recommend that the Process Tree is configured when controller Logical Organizers have already been previously configured. Use Logix Designer Version 24 or later to configure controller Logical Organizers.

This section also provides instructions on how to organize SE HMI displays in the Process Tree folders. The AE XML import file builder tool provides an option to configure the AE alarm tag View command to open an associated HMI display automatically from the runtime AE alarm summary

Create Process Tree Folders from Controller Logical Organizers

1. Click the 'Controllers & HMI' tab at the bottom of the window.
2. If you have multiple controllers, click the '+' to expand the Logic Controllers node.
3. To open the controller in a child window, double-click a controller node.

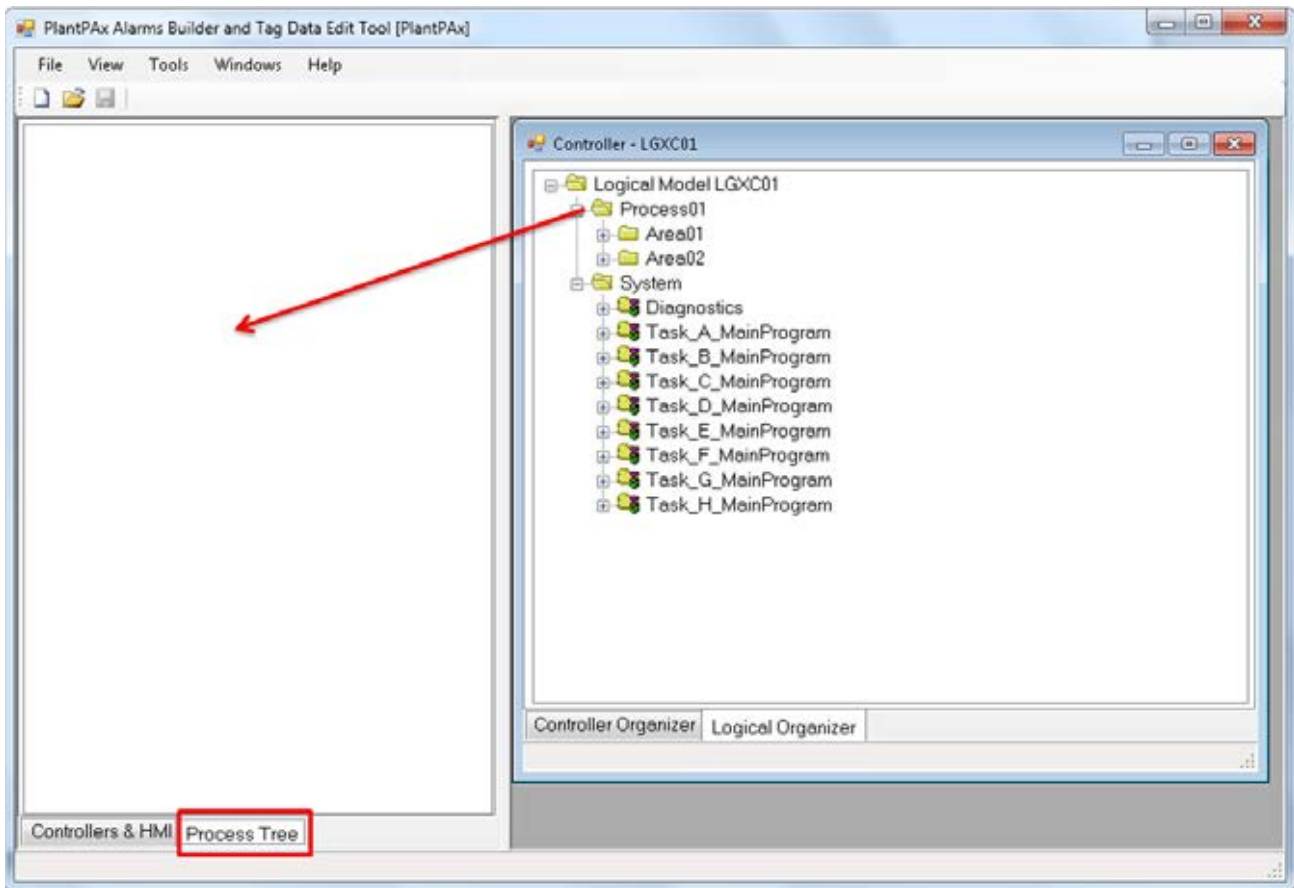


4. Click the 'Logical Organizer' tab in the controller child window. The

IMPORTANT This tab is available only for controller files that are created with Logix Designer application Version 24 or later.

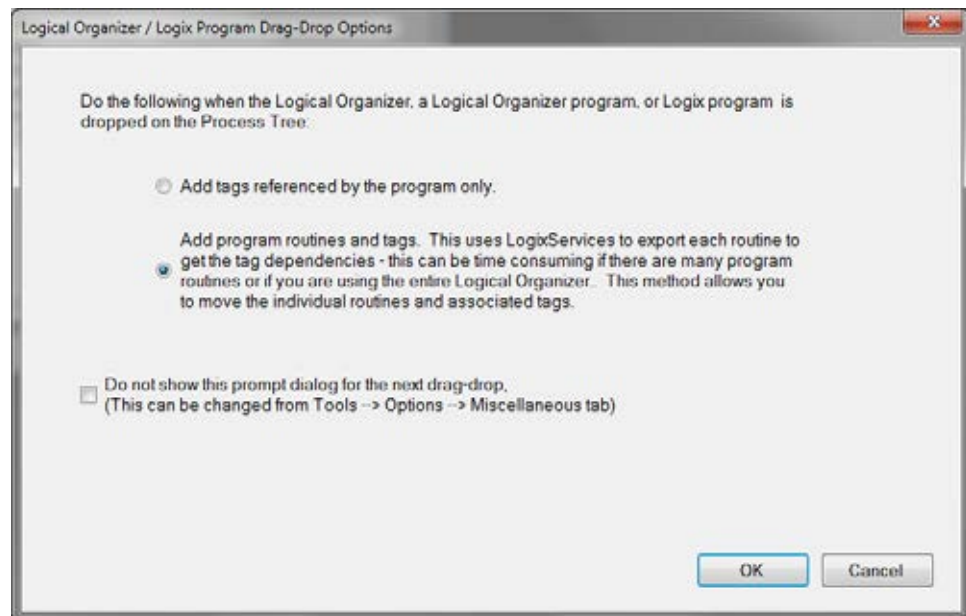
Logical Organizer contains all controller programs and program folders structure that is configured by using the Logix Designer application.

5. Click the Process Tree tab at the bottom the main Alarm Builder window.
6. Click the Process01 folder in the Logical Organizer tree and drag-and-drop the folder to the Process Tree tab area.



The Logical Organizer/Logix Program Drag-Drop Options dialog box appears.

7. Select the 'Add program routines and tags' option.



The Process01 folder and all its child folders are added to the Process Tree. Program folders contain 'Tags' subfolders. The 'Tag' subfolders contain tags that the program references. Process Tree folders can be reorganized by using drag-and-drop if desired.

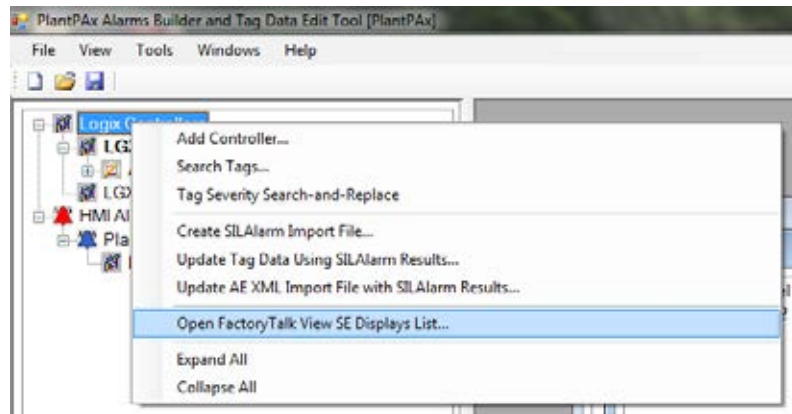
TIP The use of the 'Add program routines and tags' option can be very time consuming if the Process01 folder contains many routines. Use the 'Add tags reference by the program only' option instead for faster processing. The major difference between the two options is that in the second option, tags to be reorganized and moved to different process tree folders by moving routines. Tags have to be moved individually if the first option is used. See the 'Process Tree Organizer' user manual for details. This user manual can be found in Help>Contents on the PlantPAx Alarm Builder and Tag Data Edit Tool window.

8. Click OK.
9. Repeat [step 2](#) through [step 7](#) for each controller in the project.

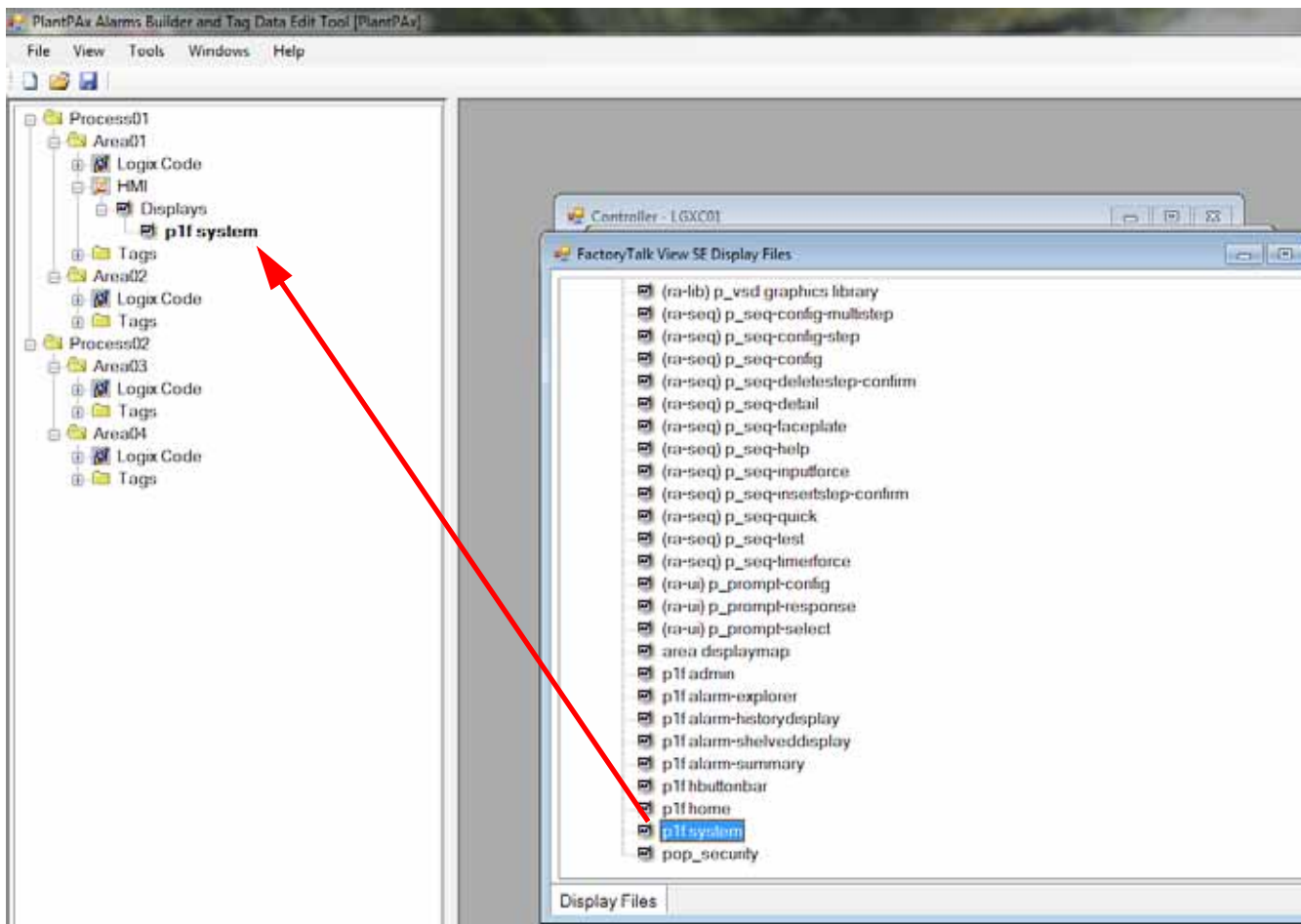
Organize SE HMI Displays in Process Tree Folders

1. Right-click the Logix Controllers tree node in the Controllers & HMI tab, and choose Open FactoryTalk View SE Displays List.

A child window opens with a FactoryTalk View SE displays list that of HMI servers.



2. Click the Process Tree.
3. To organize in the SE Displays Files child window, click an HMI display.



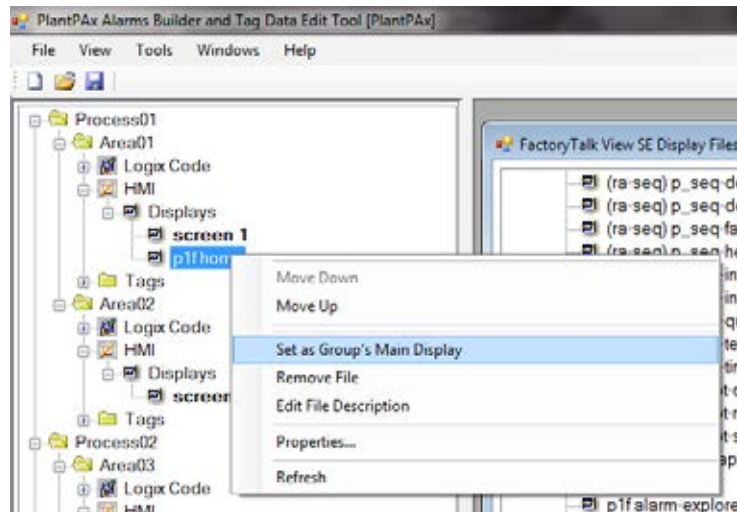
4. Drag and drop the file on the desired Process Tree folder.

HMI displays subfolders are automatically added under the target process folder for organizing HMI displays.

The process tree folder tags are associated with the HMI display.

Move display files by drag-and-drop.

Right-click a display file to display a context-sensitive menu.



Multiple displays can be added to the same process folder – this feature is for future use. If a process folder contains multiple displays, only the one set as the 'main display' for the group that is associated with the process folder tags. The main display has a bold font. Use the popup menu to change the main display.

5. Build AE XML Import File

IMPORTANT Make sure to configure the Logix tags correctly before building the AE XML file. It is especially important to make sure the HasAlarm values are configured correctly. If the HasAlarm is configured as '0' and the default build options are used, then an AE tag is NOT added to the AE XML file. If you later change the Logix HasAlarm value to '1', there is NO alarm annunciation because the AE tag does not exist.

Logix tag values can be configured by using the following software and tools:

- Logix Designer application to make tag edit changes.
- Online Configuration Tool ([See Appendix A](#)).
- Alarm Builder tag data editing tools. See the Alarm Builder 'Edit Tag Data' user manual for details.

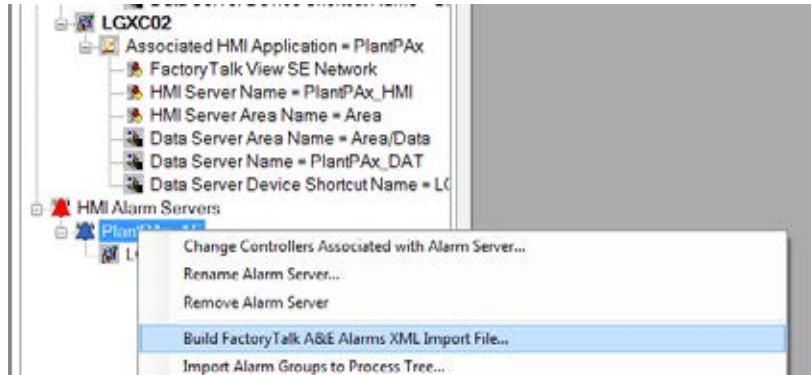
If tag changes are made by using the Logix Designer application or the Online Configuration Tool while a project is loaded in Alarm Builder, then do the following:

1. If online changes are made by using the Logix Designer application or the Online Configuration Tool, then use the Logix Designer application to upload and save the Controller changes to the Controller ACD file.
2. If offline changes are made by using the Logix Designer application, then save the changes and exit the Logix Designer application.
3. In Alarm Builder, click File and choose Open to reload the project (for example, select the same project that is currently loaded in the Open Project File browser).

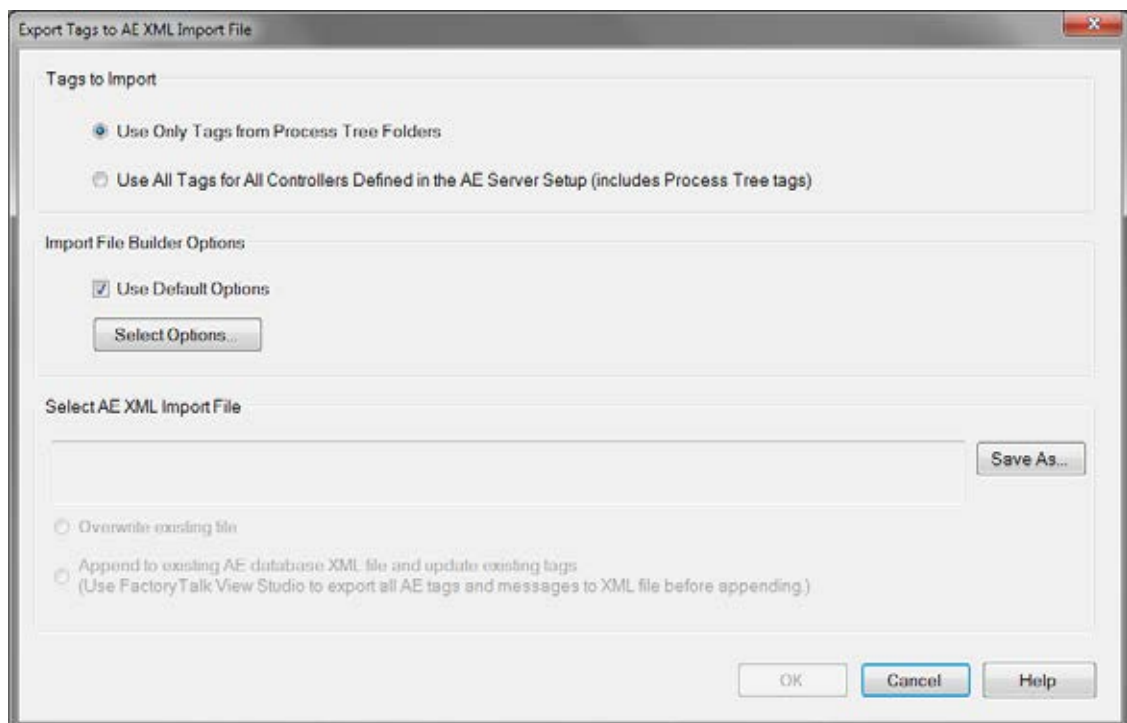
A reload of the project rebuilds the controller XML files for any controller ACD files that have been changed. This rebuilding helps verify that controller XML file contents match the controller ACD file. Alarm Builder uses the controller XML file, it is not able to read the controller ACD file directly.

After the project is loaded with properly configured Logix tag values, then do the following to build the AE XML import file:

1. Right-click the alarm server name and choose Build FactoryTalk A&E Alarms XML Import File.



The Export Tags to AE XML Import File dialog box appears.

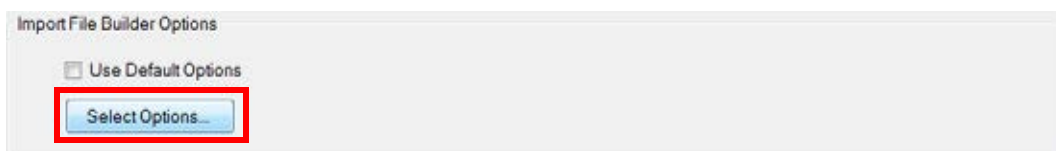


2. If you used the controller the Logical Organizer tree to configure the Alarm Builder process tree (see [4. Organize Tags and HMI Displays \(Optional\) on page 163](#)), then select the 'Use Only Tags from Process Tree Folders' option under Tags to Import.

Otherwise, accept the default.



3. We recommend that you uncheck Use Default Options under Import File Builder Options.



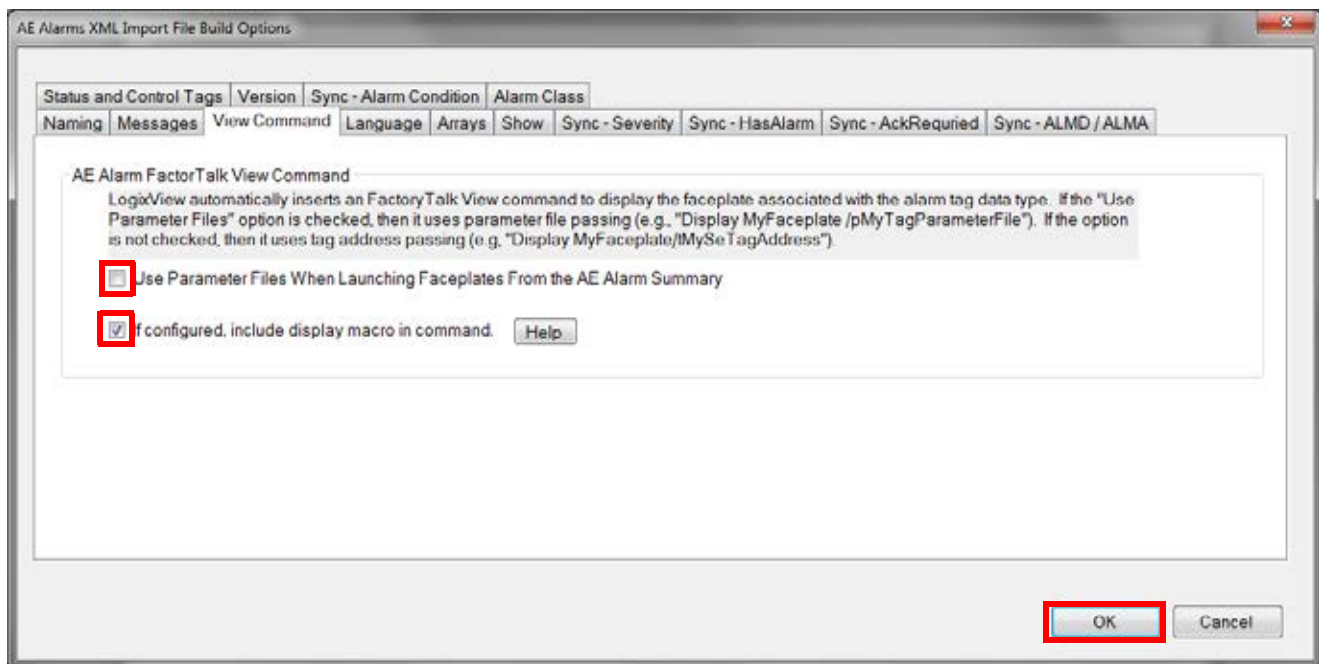
The following rules are used when Use Default Options is checked:

- AckRequired parameter for the AE tag is set to True
- Severity value for the AE tag is set from the alarm severity value of the Logix tag. For FactoryTalk View software, version 8 or later, AE Severity is set to Logix tag severity address if configured in the Alarm Builder data type definition.
- AE tag is added to the XML file only if the HasAlarm value for the Logix tag is '1'.

IMPORTANT Make sure to configure the Logix tags correctly before building the AE XML file. It is especially important to make sure the HasAlarm values are configured correctly. If the HasAlarm is configured as '0', then an AE tag is NOT created. If you later change the Logix HasAlarm value to '1', there is NO alarm annunciation because the AE tag does not exist.

- The local string tag (Cfg_Desc) value from the Logix tag is added to the alarm message for the AE tag
- AE tag name equals the Logix tag name and alarm element name. For example, if the Logix tag is 'TI123' and the alarm element is 'Alm_HiHi', then the AE alarm name is 'TI123_Alm_HiHi'
- View command from the AE tag is configured to display the corresponding Logix tag faceplate by using the parameter file passing. The parameter file that contains the tag address is automatically added to the HMI project 'par' folder
- For FactoryTalk View software, version 8.0 and older, the Alarm Class parameter in the AE tag is configured with the path of the Logix tag (for example, Controller\Task\Program) or the parent process tree full path name of the tag (for example, 'Area1\Unit1\Heater'). For FactoryTalk View software, version 8.1 or later, the Alarm Class is configured by using the alarm definitions. Alarm groups are automatically configured.
- AE status and control tags for data types are automatically configured by using the PlantPAx P_Alarm Add-On Instruction

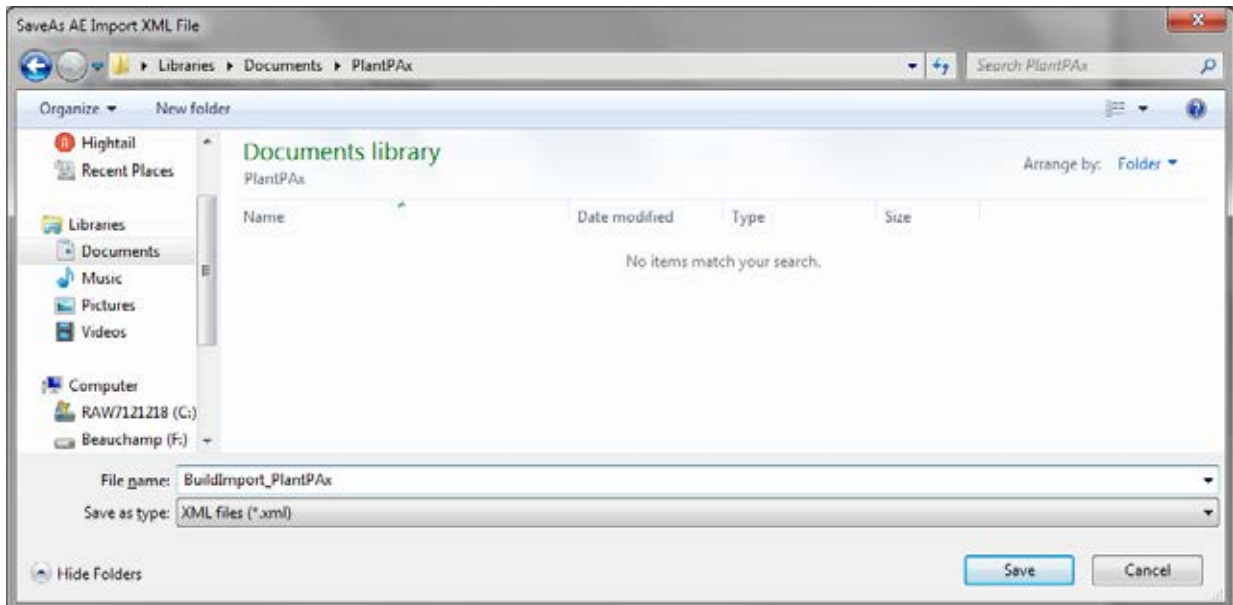
- All other AE parameters are configured based on the Logix data type alarm definition
 - The AE XML file format version uses the installed FactoryTalk View software version. For example, if FactoryTalk View software version 7 is installed, then it uses the AE XML file format compatible with version 7. If FactoryTalk View software is not installed, then the user is prompted for the format version to use.
 - Builder options that are not described in the preceding paragraphs use the configured options (click Select Options to view all available configuration options).
4. Click Select Options.
 5. Click the View Command tab.
 6. Uncheck 'Use Parameter Files when launching Faceplates ...'.
 7. Check 'If configured, include display macro in command'.



With this option, the view command is automatically appended with 'mcrAE_Display DisplayName'. 'DisplayName' is the HMI display name that is associated to the AE tag. 'mcrAE_Display' is a user-defined macro. You can configure the macro to open the HMI display directly or to access a faceplate that has a button to open a display. See the 'Create Alarm Import Files' user manual for details on the macro. This user manual can be found in Help>Contents on the PlantPAx Alarm Builder and Tag Data Edit Tool window.

8. Click OK.
9. In the Export Tags to AE XML Import File dialog box, click Save As.

10. Navigate to the folder where you want to save the XML file, specify the XML file name, and click Save.



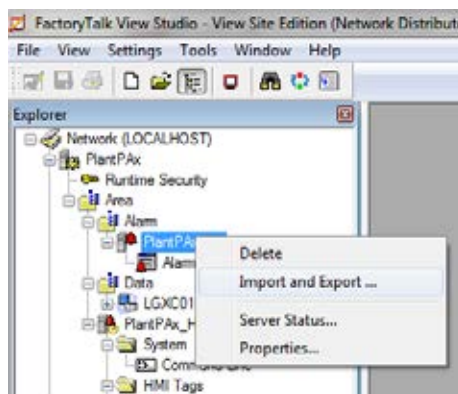
11. In the Export Tags to AE XML Import File dialog box, click OK to build the AE XML import file.

6. Import XML File to AE Alarm Database

This section describes how to use FactoryTalk View Studio software to implement the XML file to import alarms into the PlantPax system.

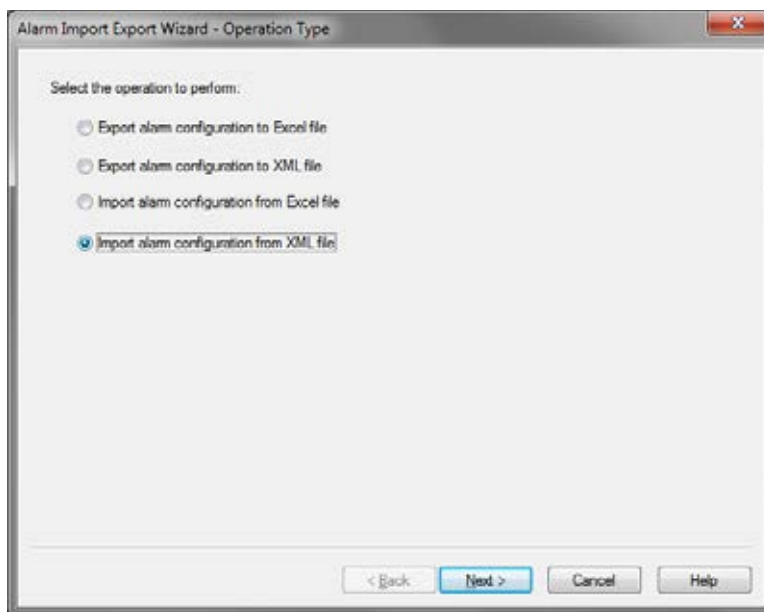
Follow these steps after opening the FactoryTalk View Studio software.

1. From the FactoryTalk View Studio Explorer tree configuration, right-click the AE alarm server and choose Import and Export.



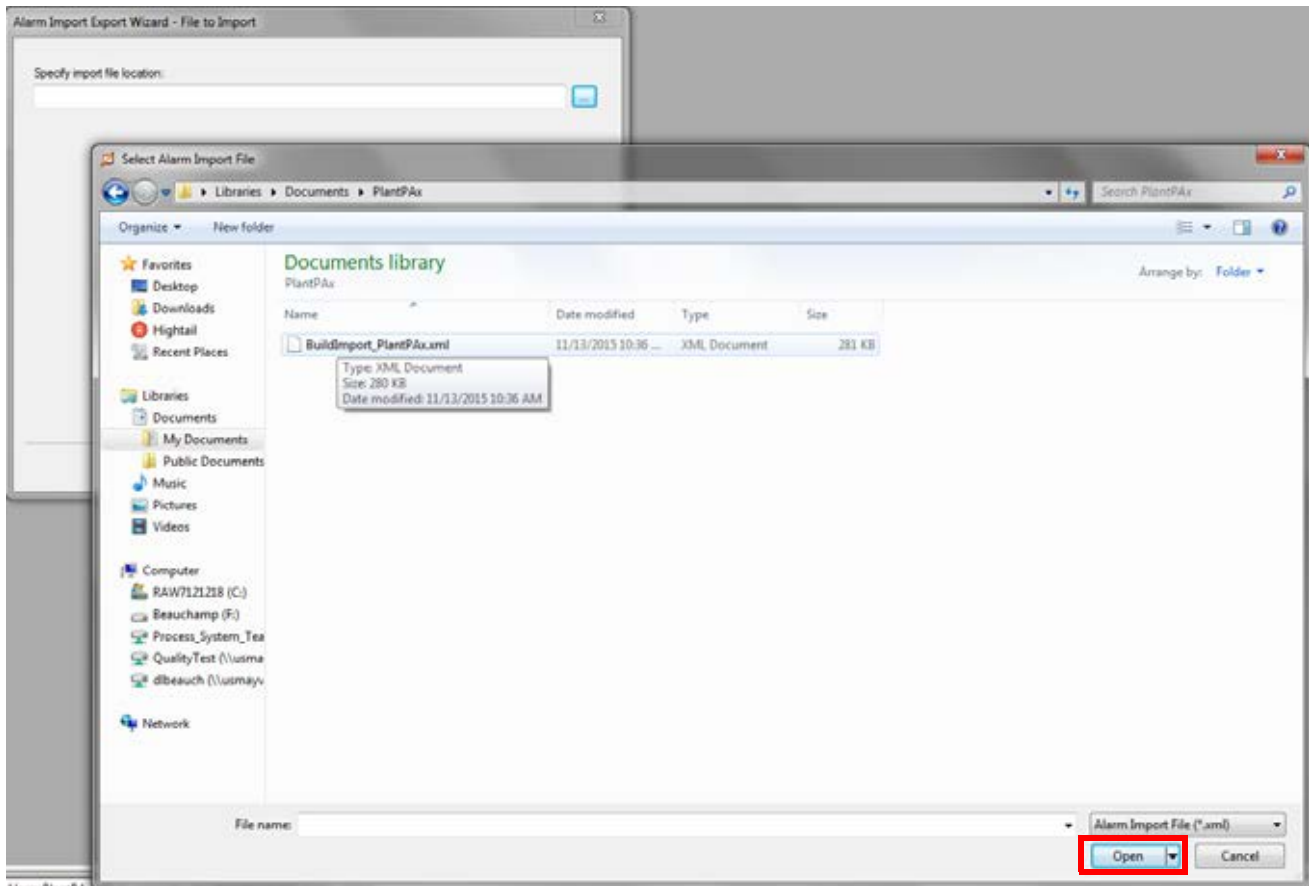
The Alarm Import Export Wizard - Operation Type dialog box appears.

2. Select 'Import alarm configuration from XML file' and click Next.

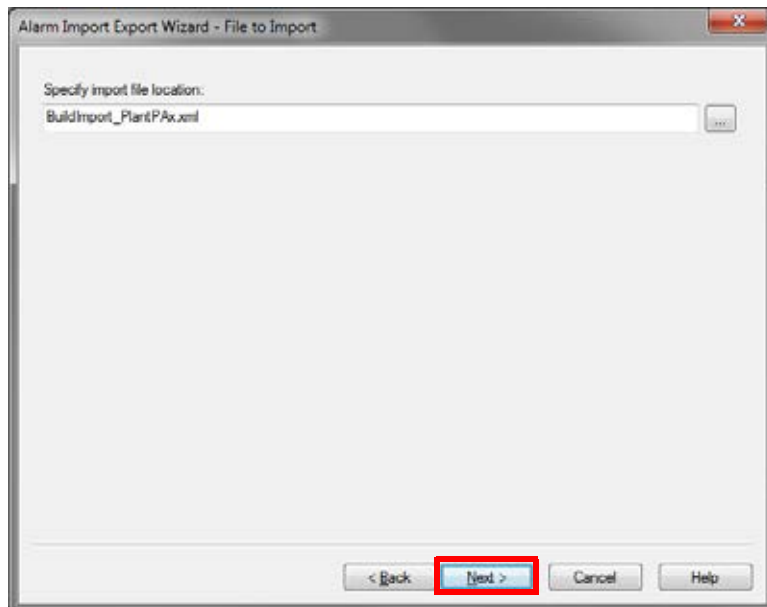


3. Click  and navigate to the XML file you created in the alarm builder tool.

4. Select the file and click Open to close the file dialog.



5. In the Alarm Import Export Wizard - File to Import dialog box, click Next.

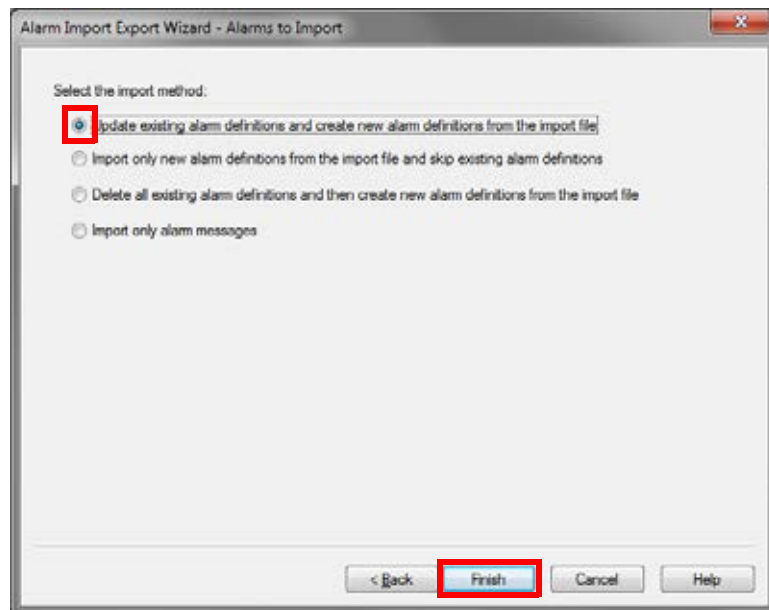


6. In the Alarm Import Export Wizard - Alarms to Import dialog box, select the import method.

Select the first option 'Update existing alarm definitions and create alarm definitions from the import file' or the third option 'Delete all existing alarm definitions and then create alarm definitions from the import File'.

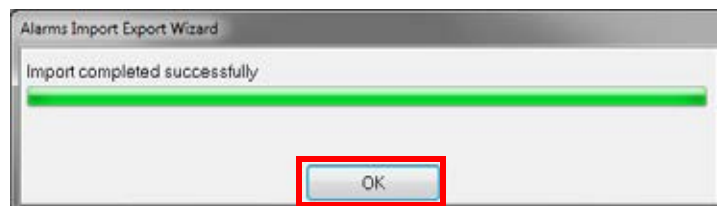
The second option ("Import only new alarm definitions from the import file and skip existing alarm definitions.") is not recommended because PlantPAx HMI Alarm Builder Tool creates messages for each new alarm definition. If you import only new alarm definitions, the newly imported alarm definitions point to the wrong messages or to non-existing messages.

Do not use the fourth option 'Import only alarm messages' with the XML file. The purpose of PlantPAx HMI Alarm Builder Tool is to create alarm definitions that correspond with alarm messages. The import of only alarm messages is not useful.



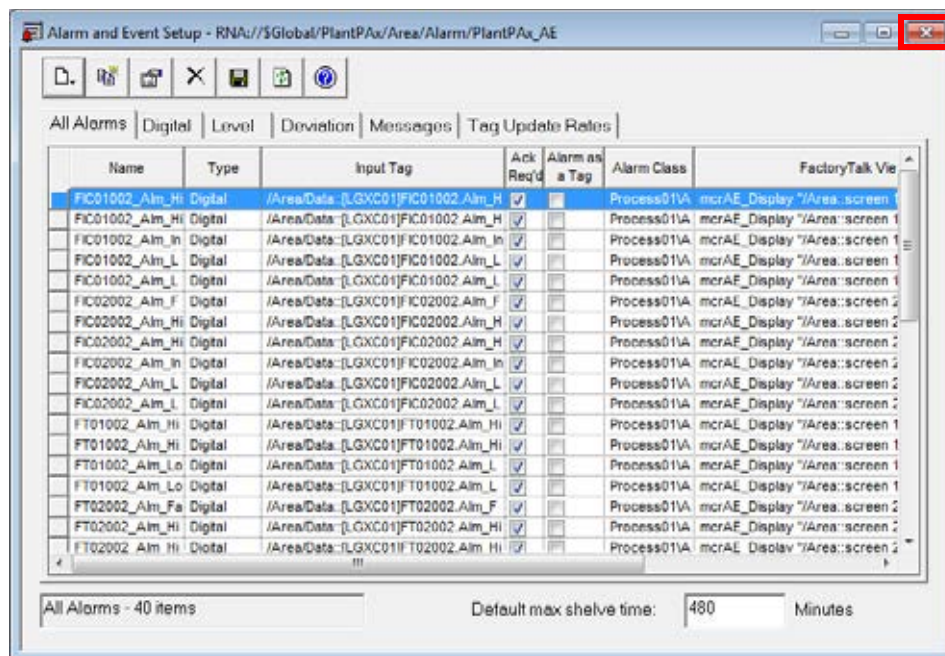
7. Click Finish.

Depending on how many tags are being imported, the computer speed, and if you are updating existing tags, the import can take several minutes. A dialog box appears when the import is completed.



8. Click OK. The imported alarms are now in the alarm server database.

9. In the Explorer tree configuration, double-click Alarm and Event Setup. The Alarm and Event Setup tag database window appears to let you view the alarm definitions.

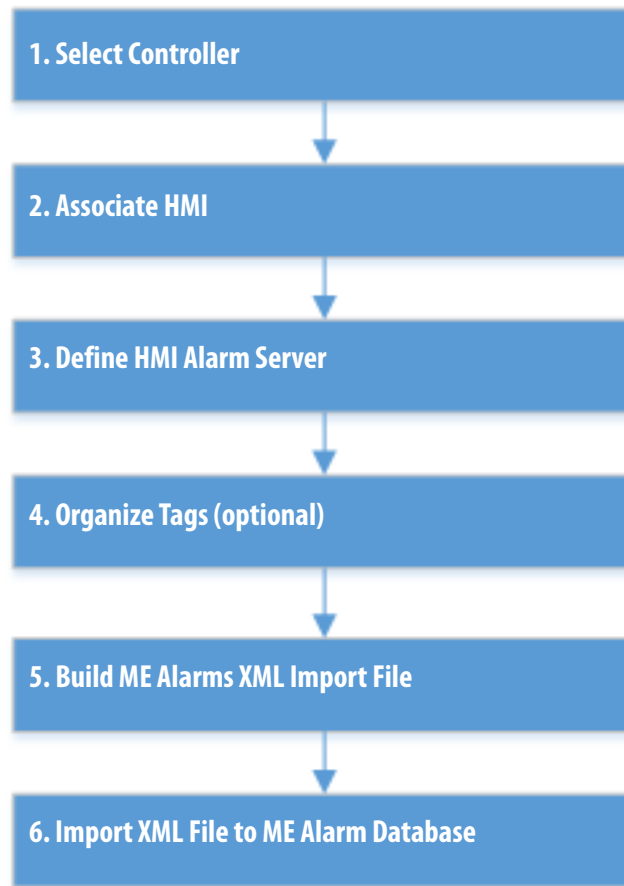


10. To close the window, click the 'X' in the upper, right corner.

Build ME Alarms

The diagram outlines the procedures for creating FactoryTalk View ME software alarms. The procedures in this section are in the same order as the headings in the diagram.

Figure 9 - Alarm Builder ME Workflow



1. Select Controller

Complete the steps that start on [page 149](#).

2. Associate HMI

Complete the steps that start on [page 156](#). However, select Machine Edition (ME) for the product type on the Select HMI Project dialog box.

IMPORTANT If the controllers in the alarm server are associated with FactoryTalk View SE software projects, the ME XML import file building tools can be used as long as the data server device-shortcut name for the controller is the same for SE and ME. If the device shortcut names are different, you must create separate Alarm Builder projects.

3. Define HMI Alarm Server

Complete the steps that start on [page 162](#).

4. Organize Tags (Optional)

The ME alarm builder provides the following options:

- The addition of all controller alarm tags to the ME alarms XML import file
- The addition of tags only from the Alarm Builder Process Tree Organizer folders

To configure a Process Tree, complete [step 1](#) ... [step 4](#) on pages 165...166.

5. Build ME Alarms XML Import File

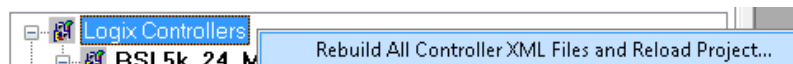
IMPORTANT Make sure to configure the Logix tags correctly before building the ME alarms XML import file. It is especially important to make sure the HasAlarm values are configured correctly. If the HasAlarm is configured as '0' and the default build options are used, then an ME alarm trigger is **not** added to the ME XML files. If the Logix HasAlarm value is changed later to '1', there is **no** alarm annunciation because the ME alarm trigger does not exist.

Logix tag values can be configured by using the following software and tools:

- Studio 5000 Logix Designer application to make tag edit changes
- Online Configuration Tool (see Appendix A on [page 143](#))
- Alarm Builder tag data editing tools

Complete these steps if tag changes are made by using the Logix Designer application or the Online Configuration Tool while a project is loaded in Alarm Builder.

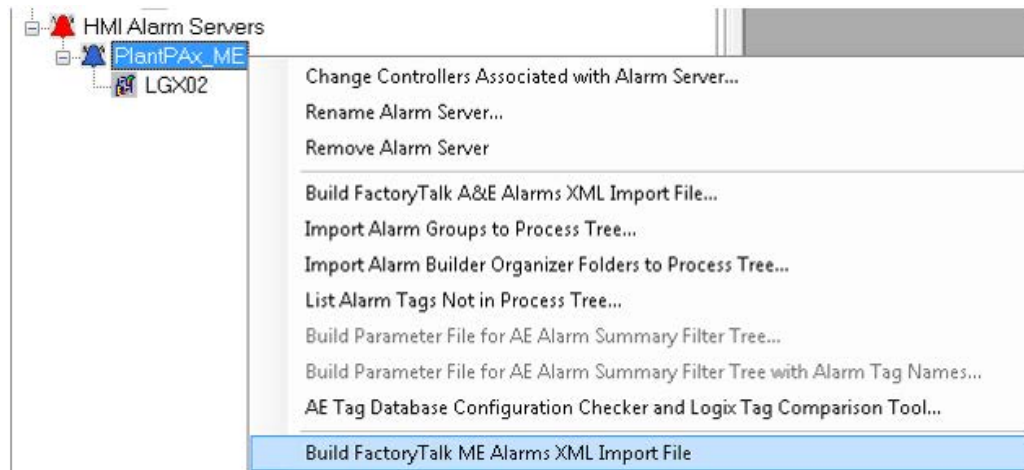
1. If online changes are made by using either tool, use the Logix Designer application to upload and save the controller changes to the controller ACD file.
2. If offline changes are made by using the Logix Designer application, save the changes and exit the Logix Designer application.
3. In the Alarm Builder tree, right-click Logix Controllers and choose Rebuild All Controller XML Files and Reload Project.



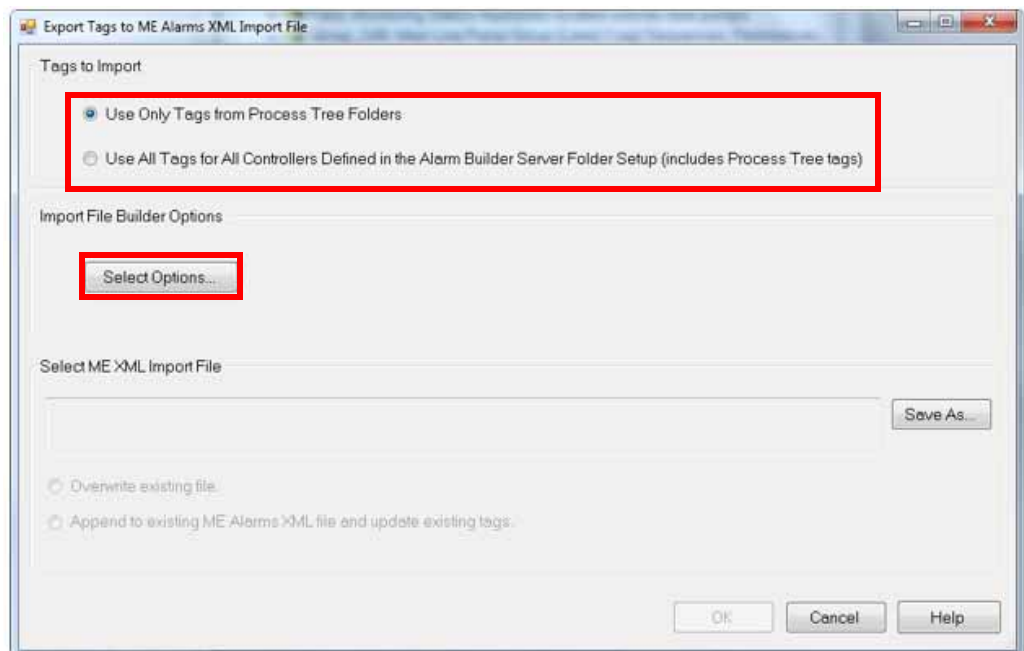
The rebuild helps to make sure the controller XML file contents match the controller ACD file. Alarm Builder uses the controller XML file, it is not able to read the controller ACD file directly.

To build the ME alarms XML import file after the project is loaded, complete these steps.

1. 1.Right-click the alarm server name and choose Build FactoryTalk ME Alarms XML Import File.

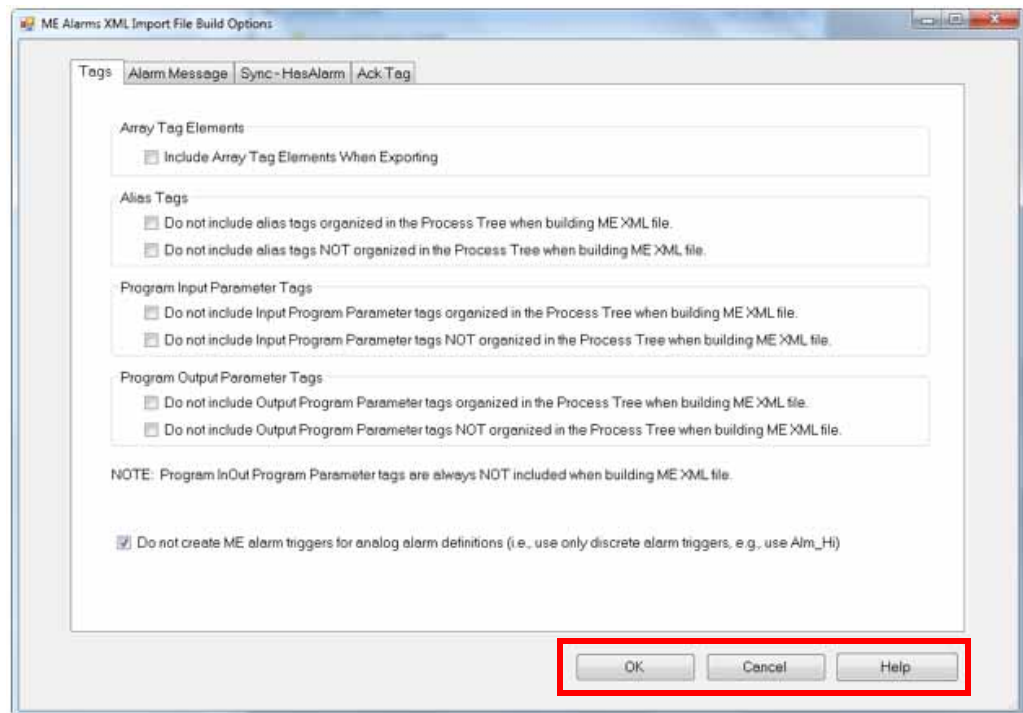


The Export Tags to ME Alarms XML Import File dialog box appears.

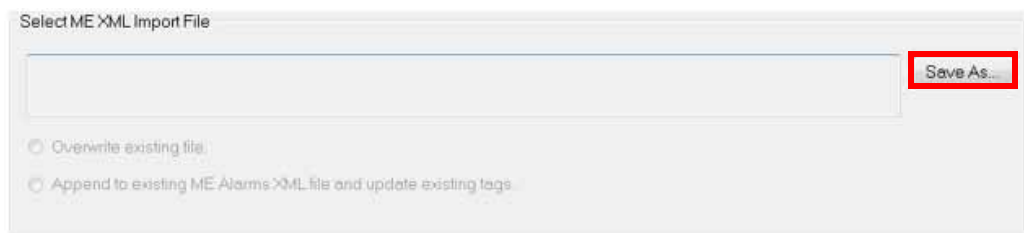


2. Select one of the following options under Tags to Import:
 - a. Use Only Tags from Process Tree Folders if you used the controller in the Logical Organizer to configure the Alarm Builder process.
See [4. Organize Tags and HMI Displays \(Optional\) on page 163](#).

- b. Otherwise, Use All Tags for All Controllers Defined in the Alarm Builder Server Folder Setup (includes Process Tree tags).
3. Click Select Options.
4. Review the build options and, if necessary, use these functions:
 - Click Help for option descriptions
 - Click OK to save option changes
 - Click Cancel



5. Click Save As.



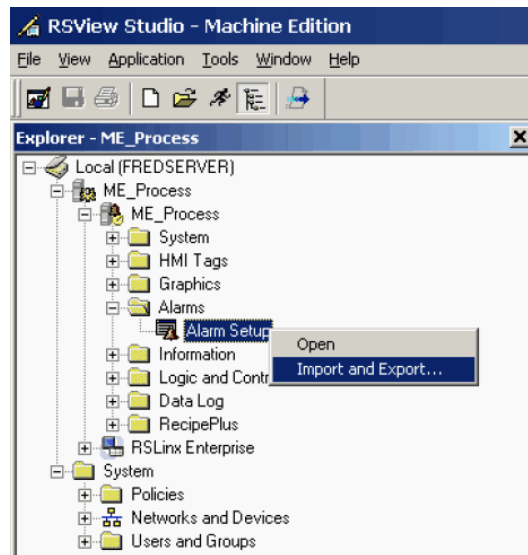
6. Navigate to the folder where you want to save the XML file and specify the XML file name.
7. Click Save.
8. In the Export Tags to ME Alarms XML Import File dialog box, click OK to build the ME Alarms XML import file.

6. Import XML File to ME Alarm Database

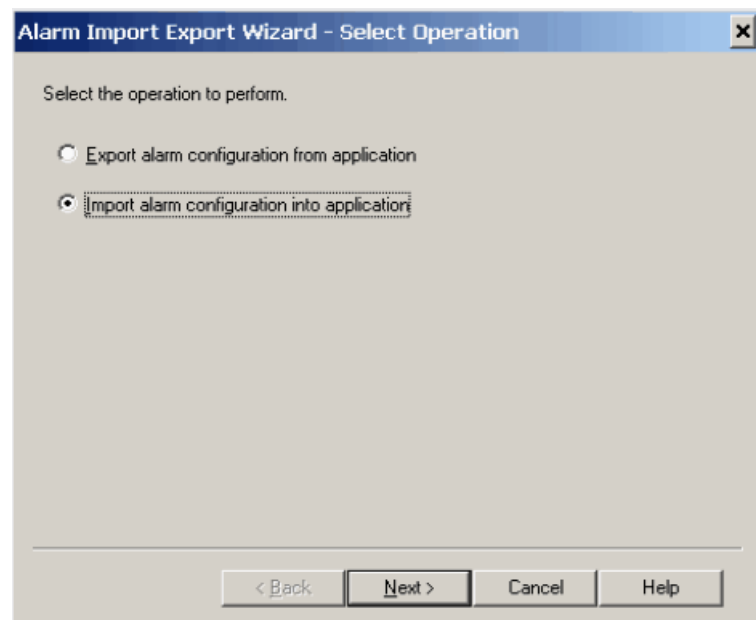
This section describes how to use FactoryTalk View Studio software to implement the XML file to import alarms into the PlantPAx system.

Follow these steps after opening the FactoryTalk View Studio software.

1. From the FactoryTalk View Studio Explorer tree configuration, right-click the alarm server and choose Import and Export.

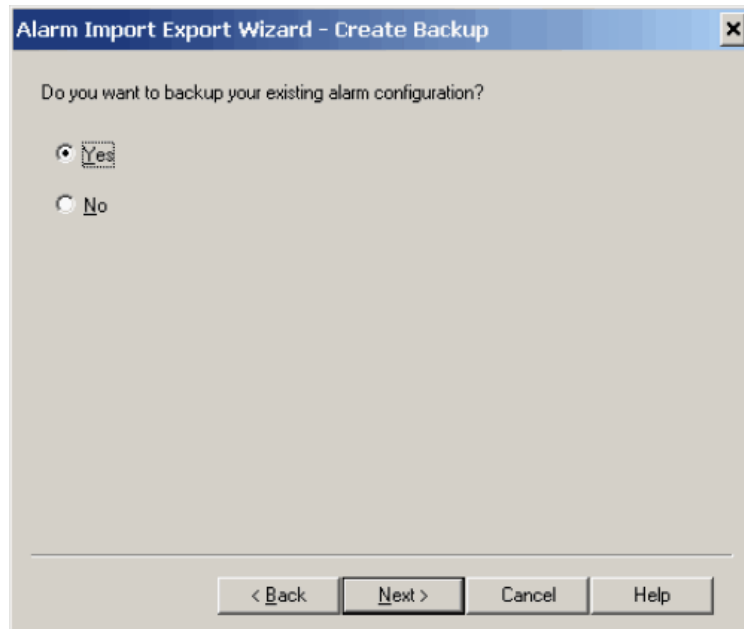


The Alarm Import Export Wizard - Select Operation dialog box appears.

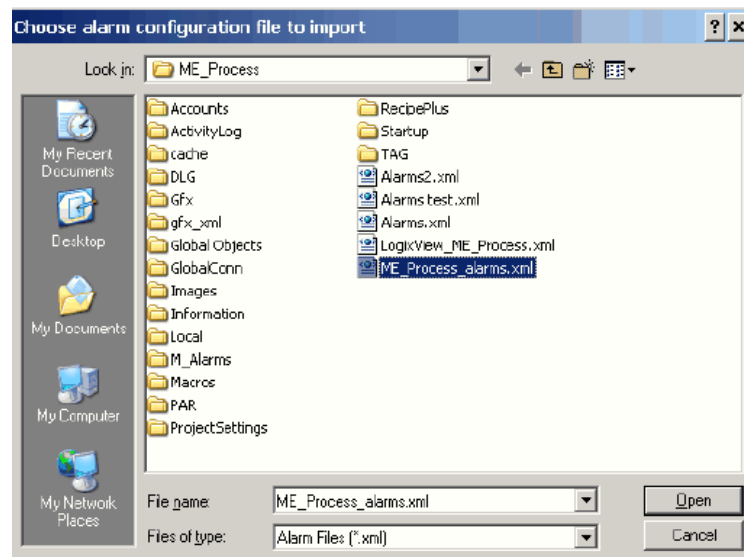


2. Click Import alarm configuration into application and click Next.

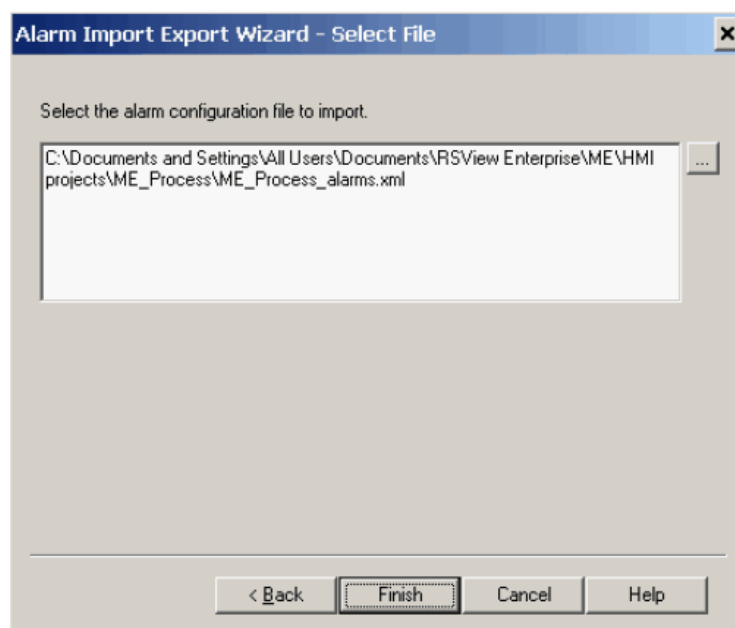
We recommend that you make a backup until you are familiar with the import process.



3. Type a name for the backup file and click Next.
4. To find where you saved the file, click browse (ellipsis, ...) on the alarm wizard dialog box.
5. Select the alarm configuration file and click Open.



The Alarm Import Export Wizard - Select File dialog box appears.



6. Click Finish.

Color Change Utility

This customizing tool lets you create a color palette to change the colors for FactoryTalk® View software display elements (global objects) and faceplates.

The Color Change Utility uses three types of files:

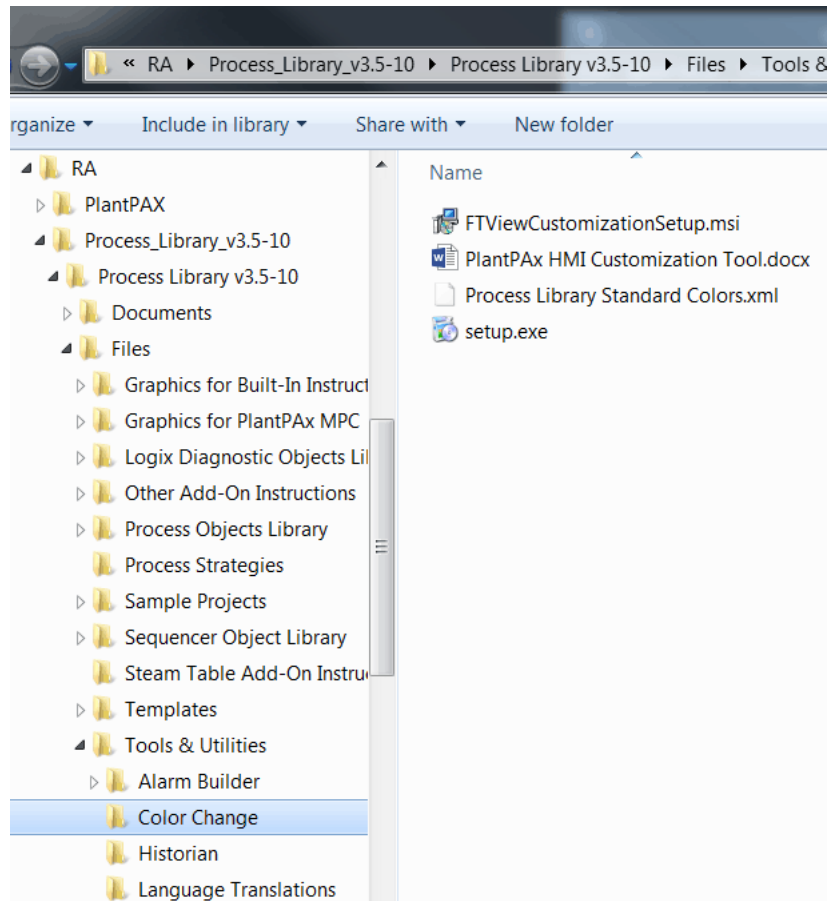
- **FactoryTalk View Graphics .xml file:** This file is exported from the FactoryTalk View graphic (display or global object) in the View Studio software program. Once changes are made, it is imported into the View Studio software program to change the colors in the display or global object.
- **Color Association File:** This .xml file matches a color instance in the FactoryTalk View Graphics .xml file to the color palette entry. There is one Color Association File (CAXML) for each FactoryTalk View Graphics .xml file. The utility creates and maintains the CAXML file.
- **.Color Palette:** This .xml file defines the colors for an application. The utility creates and maintains the .xml file. There is one color palette file for all FactoryTalk View Graphics .xml files that are being customized. If you want to change the color, it is done in the color palette.

TIP We suggest that you make a copy of the color palette .xml file if you plan to use the color tool.

Install Tool File

Obtain the Color Change Utility as part of the Library of Process Objects download from the Product Compatibility and Download Center at <http://www.rockwellautomation.com/rockwellautomation/support/downloads.page>.

Access the utility from the Process Library download. Choose RA>Process Library vX.X>Process Library vX.X>Files>Tools & Utilities>Color Change and double-click FTViewCustomizationSetup.msi.



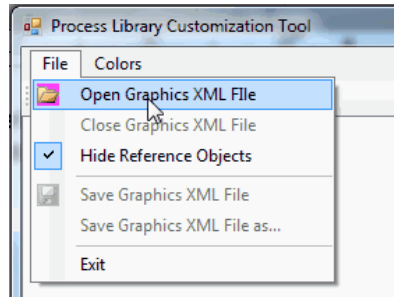
This file installs the program and adds a shortcut to the Start menu under 'PlantPax®'.

Use the Utility with Library Objects

The download includes .xml exports for all global objects and display files in the library (for FactoryTalk View SE software). Make sure that you also download the CAXML and Process Library Standard Colors .xml files.

Follow these steps to change colors in the process library.

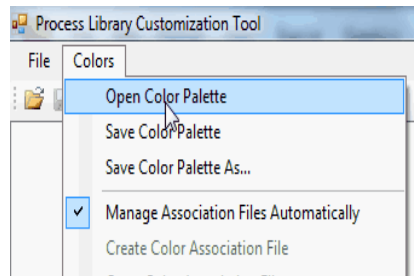
1. From the Process Library Customization Tool File menu, click Open Graphic XML File.



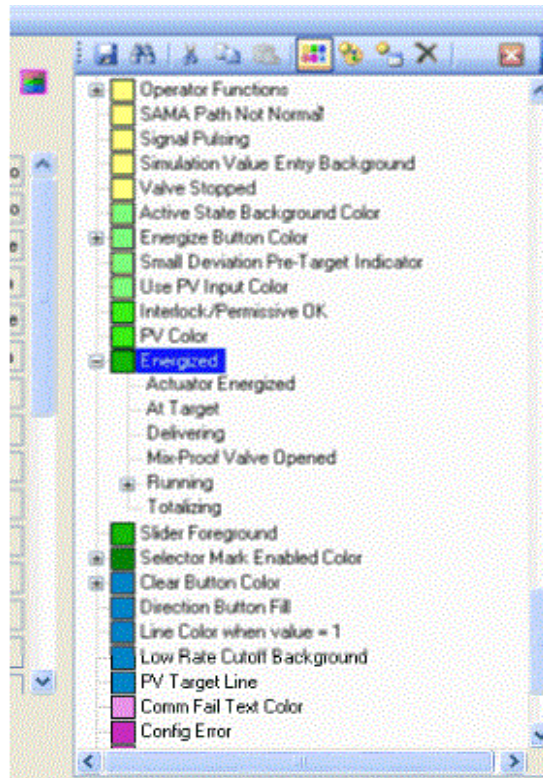
The Open Graphics XML Files dialog box appears.

Multiple global object and display files can be opened simultaneously from the file open dialog box.

2. Click the Colors tab and choose Open Color Palette.

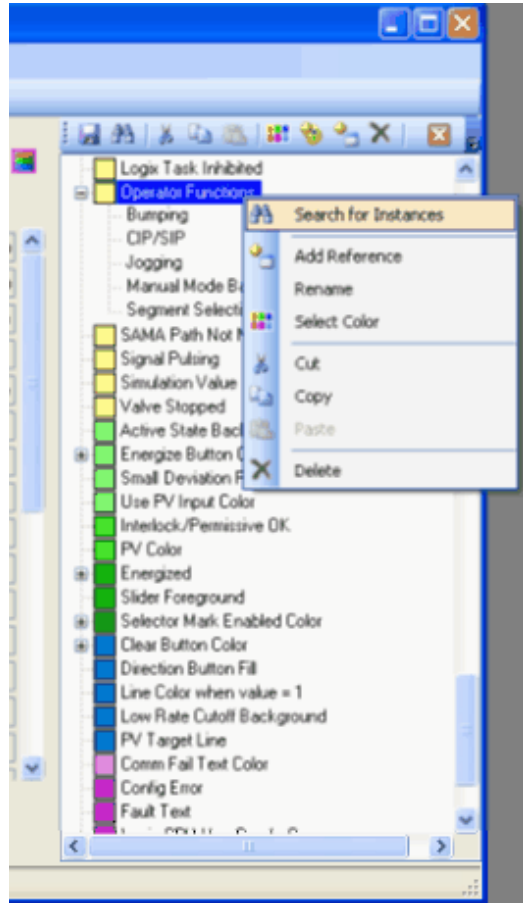


3. Select the colors that you want to change in the palette.

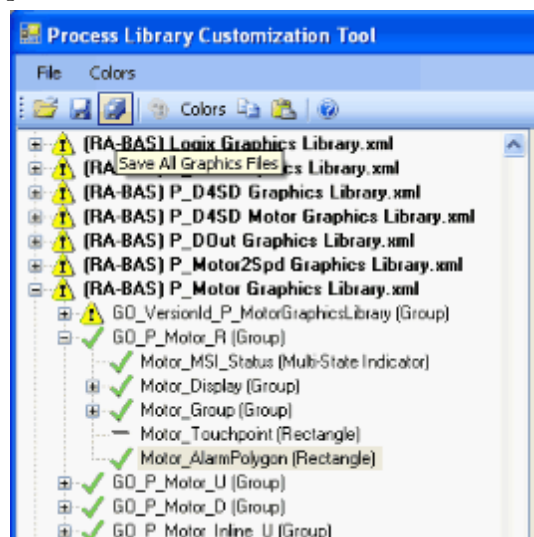


4. To select a new color, click the Choose Color  icon.
5. Repeat [step 4](#) to change each color.

6. To see where a color is used, right-click a color and choose Search for Instances.



7. To save all graphic files (along with their association files) and the color palette file, click Save All .



8. Import the files into the FactoryTalk View software program.

There are bulk import files for the displays (BatchImport_Displays_PlantPAx)Library.xml) and global objects (BatchImport_Global_OlantPAx)Library.xml).

Modifying the Color Palette

The color palette appears in a tree format that shows a parent-child relationship between colors. 'Base Colors' are shown with a color box next to them.

'Reference Colors' reference either a Base Color or another Reference Color.

By changing a Base Color, all Reference Colors under it change. For example, you can create a generic Base Color, called 'Energized', and then reference it with the Reference Color, called 'Running'.

Do not delete Color palette entries unless they are known to be unused. To see if a color palette entry is being used, right-click the color and choose 'Find Color Instances'.

Any color palette entry (Reference or Base Color) can be moved to reference another color. This action is done by simply dragging the color to be moved and dropping it on the new color to reference. When a color that has references is moved, all of its references move as well.

To make a Reference Color a Base Color, right-click the Reference Color and select 'Make Base Color' from the context menu.

Color palette entries are stored with an integer code. That integer code is used in the association file. Renaming a color palette entry does not break any existing associations. Multiple color palette entries can have the same name, but this practice is not recommended.

Follow these color palette considerations:

- Once a color palette entry is deleted and the palette is saved, the only way to restore associations is to recreate them manually.
- Object names in FactoryTalk View software usually have a number on the end. Names are considered to be similar if they are the same after the ending number is removed.
- Button icons are not associated with the color palette for these reasons:
 - Future versions of the library can change these icons to images
 - Button icons are global objects and do not have multiple definitions, the colors are changed only in one file
 - Additional color palette entries for individual button icons can increase palette maintenance

Use the Utility with Other FactoryTalk View Software Files

The color palette must be applied to FactoryTalk View software files that are not part of the Rockwell Automation® Library. Graphic elements in the file must be associated to the color palette. You must create associations and save them in a color association file. When opening an .xml graphics file, if the file already has an association file (CAXML), it is automatically opened as well. If an association file does not exist, it is created.

Follow these steps to create associations.

1. From the Process Library Customization Tool File menu, click Open Graphic XML File.

The Open Graphics XML Files dialog box appears.

2. Select an object from the tree on the left, and its colors appear in the center of the screen.
3. To associate a color from the palette, select the palette color and drag it to the text box next to the color display box.

Once all colors for an object are associated with the color palette, a check appears next to the object in the tree.

Colors that are used for the object only are displayed. For example, if an object is configured as 'Transparent', its background color does not show up in the utility. Also, instances of global objects from display files do not appear in the object tree. The tree can be configured to show instances of global objects. These objects do not have any color instances because their parent global objects control their colors.

4. Copy and paste functions have been included to allow quick creation of color associations. To use these functions, right-click the graphic object in the tree on the left and a menu appears.
 - **Copy Color Associations:** Use this function to copy the color associations for the object. If the object is a group, the color configuration for all group members is copied.
 - **Paste Color Associations (this Object only):** Use this function to paste the previously copied color associations to the selected object. This option is not available if the selected object is a group that has members with color associations.
 - **Paste Color Associations (to all group members):** Use this function to paste the previously copied color associations to the new object and all of its members. This option is available only if the source and destination objects are groups with members that have similar names and object types.
 - **Copy and Paste Color Associations to Similar Objects with Names like 'Xxxx#':** This option copies the selected object and searches objects with a similar name and object type. Color associations are copied to all objects with similar names and types in any of the currently open graphics files. If the objects are groups, then the group members must have similar names and object types. Be careful when you use this feature to help prevent unwanted changes.

Notes:

View Security Permission Tags

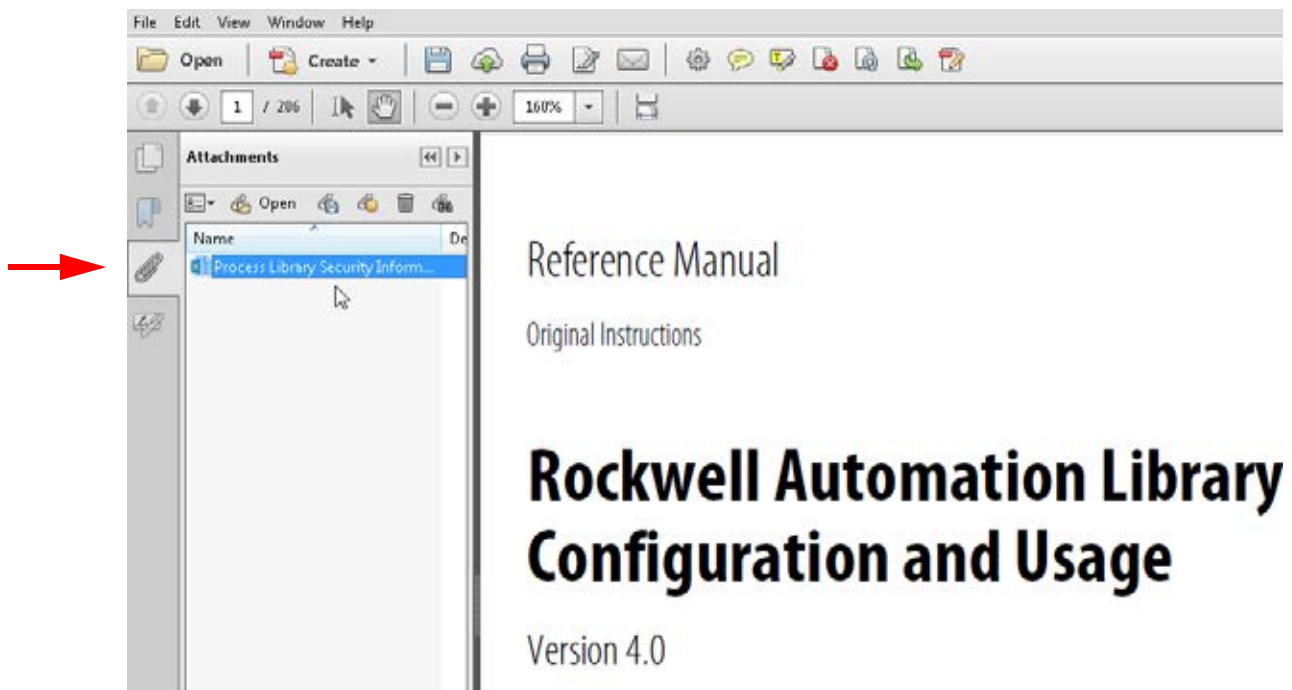
Process Library Security Information tags are attached to the PDF in a Microsoft Excel spreadsheet. These tags are used for configuring and maintaining the faceplate privileges for plant personnel.

For example, an operator who works in packaging is configured for HMI security for this role type and responsibilities. Another operator can be configured for HMI permissions on the conveyor line, and so on.

Tag descriptions are provided to streamline configuration for different role types and responsibilities.

How Do I Access an Attachment?

To view an attachment to the PDF, click the Attachments link (paper clip) in the left pane. Then, double-click the respective file.



As a precaution when you open programs or files, select one of the choices in the popup window and click OK.

Notes:

A

- add**
 - KPI to template 39
- additional resources** 8
- Add-On Instruction**
 - add to routine 130
 - import to controller project 10
- alarm**
 - add digital 112
 - annunciation 33
 - associate HMI 156
 - build AE 148
 - workflow 148
 - builder tool 147
 - colors for severity 123
 - configure
 - message 122
 - notification options 123
 - considerations 103
 - create display 51
 - digital alarm
 - control tags 116
 - status tags 115
 - FactoryTalk
 - ME configuration 118
 - SE configuration 111
 - hierarchy 29
 - import XML file to database 173
 - ME build 178
 - server organization 162
 - severity values 152
 - types 103
- annunciation**
 - alarm 33
- application**
 - HMI 25
- area** 73
 - create object values 69
 - import tags 82
 - user groups 85
- assign**
 - HMI security 90
 - security
 - user groups 78

B

- backing tag**
 - create 130
- build**
 - AE alarms 148
 - workflow 148
 - AE XML import file 169
 - alarm tool 147
 - ME alarms 178
 - XML import file 179
 - workflow
 - alarms builder AE 148
 - alarms builder ME 178

C

- change**
 - string size 70
- client file**
 - create 96
- code**
 - security list 79
 - security version differences 81
- color**
 - change utility 185
 - palette 185
 - palette modification 190
- configuration**
 - common 99
 - device
 - via faceplate 139
 - via Logix Designer parameters tab 139
 - FactoryTalk ME alarms 118
 - global objects 125
 - HMI security 69
 - mode 103
 - object area values 69
 - object tag manually 71
 - online tool 143
 - parameters, using a spreadsheet 144
- considerations**
 - alarms 103
 - ladder diagram 101
 - prescan 101
 - programming 99
- controller**
 - select for AE alarms builder 149
 - select for ME alarms builder 178
- coordinates**
 - x and y positioning 125
- create**
 - alarm display 51
 - area overview displays 41
 - equipment detail displays 42
 - four-monitor header 60
 - level 2 navigation 44
 - level 3 navigation 48
 - OWS client file 96
 - single monitor header 56
 - user groups 76
- custom**
 - display elements and faceplates 185
 - faceplate 133
 - library 129
- customize**
 - HMI template 37

D

- database**
 - alarm, import XML file to 173

device

- configuration
 - via faceplate 139
- configure
 - via Logix Designer parameters tab 139

diagram

- AE workflow 148
- ME workflow 178

digital alarm

- add 112
- control tags 116
- status tags 115

display

- configuration 125
- create alarms 51
- create area overview 41
- create equipment detail 42
- elements customization 185
- hierarchy 29
- import elements 15
- level 3 navigation 48
- SE HMI, organize 163, 167

download

- HMI template 26
- library 9
- tag configuration 144

F**faceplates**

- customization 133, 185
- import 16
- tag configuration 144

FactoryTalk

- ME alarms 118
- SE alarms 111
- set up clients for languages 24
- use color change utility with 191
- users 90

features

- HMI template 28

files

- visualization 13

G**global objects**

- configuration 125

groups

- area users 85
- user creation 76

H**header**

- four monitors 60
- single monitor 56

hierarchy

- display and alarm 29

HMI

- associate in AE alarms 156
- configure project 25
- customize template 37
- download template 26
- KPI additions 39
- language switching 23
- modify objects 37
- modify template 37
- multi-monitor 25
- navigation
 - utilities 32
- security configuration 69
- template features 28
- template structure 27
- templates 10
- view multiple areas 64

I**images**

- import 13

import

- Add-On Instructions 10
- AE XML import file 169
- area tags 82
- display elements 15
- faceplates 16
- L5X file 10
- library images 13
- macro 17
- visualization files 13
- XML file
 - to AE alarm database 173
 - to ME alarm database 182

install

- library 9
- Windows languages 23

K**KPI**

- template additions 39

L**L5X file**

- import 10

ladder diagram

- considerations 101

languages

- FactoryTalk View ME runtime device 24
- set up FactoryTalk View SE clients 24
- switching 23
- Windows, install 23

levels

- create navigation 44

library

- customization 129
- customize 128
- download 9
- getting started 9
- import Add-On Instructions 10
- install elements 9
- macros 35
- maintain releases 127
- use with color change utility 187

line

- of sight 36

list

- security codes 79

local tag

- configuration 144

Logix

- code, organize 163
- organize tag 163
- tag, change severity values 152

M**macros**

- HMI and library 35
- import 17
- rename template 67
- template 35

ME

- alarm build 178
- security differences 95

mode

- configuration 103

modify

- color palette 190
- HMI objects 37
- HMI template 37

monitor

- multi header 60
- single header 56

multi-monitor

- header 60
- HMI 25

multiple

- header
 - template 32
- view HMI areas 64

N**navigation**

- level 2 displays 44
- level 3 displays 48

O**object tag**

- manual configuration 71

objects

- HMI modification 37

online

- configuration tool 143

organize

- controllers with alarms 162
- Logix
 - code 163
 - tags 163
- SE HMI displays 163, 167

OWS

- client file 96

P**palette**

- color modification 190
- color utility 185

PlantPAx configuration tool 73**positioning**

- x and y coordinates 125

prescan

- considerations 101

privileges

- HMI security 69

programming

- considerations 99

purpose statement 7**R****releases**

- maintain library 127

rename

- template macros 67

resources

- additional 8

routine

- add Add-On Instruction 130

S**security**

- assign user groups 78
- HMI privileges 69
- line of sight 36
- list of codes 79
- ME group differences 95
- version differences 81

select

- ME controller, alarms builder 178
- SE controller, alarms builder 149

severity values

- change 152
- Version 3.0 or earlier 152
- Version 3.1 or later 152

sight

- line of 36

single header

- template 32

size

- change string 70

spreadsheet

- upload tags 144
- use to configure parameters 144

statement

purpose 7

string

change size 70

structure

HMI template 27

T**tag**

download configuration 144

faceplate configuration 144

import area 82

upload configuration 144

template

customize HMI 37

HMI 10, 26

HMI features 28

HMI modifications 37

HMI structure 27

macros 35

multiple header 32

rename macros 67

single header 32

tool

alarms builder 147

area string configuration 73

online configuration 143

U**upload**

tag configuration 144

usage

color utility 187

user

area groups 85

FactoryTalk 90

groups

security 78

groups creation 76

utility

color change 185

color palette usage 187

HMI navigation 32

V**version**

difference

security codes 81

view multiple areas

HMI 64

visualization files

import 13

X**x and y coordinates**

positioning 125

XML

build AE import file 169

build ME alarms import file 179

import file

to AE alarm database 173

to ME alarm database 182

Rockwell Automation Support

Use the following resources to access support information.

Technical Support Center	Knowledgebase Articles, How-to Videos, FAQs, Chat, User Forums, and Product Notification Updates.	https://rockwellautomation.custhelp.com/
Local Technical Support Phone Numbers	Locate the phone number for your country.	http://www.rockwellautomation.com/global/support/get-support-now.page
Direct Dial Codes	Find the Direct Dial Code for your product. Use the code to route your call directly to a technical support engineer.	http://www.rockwellautomation.com/global/support/direct-dial.page
Literature Library	Installation Instructions, Manuals, Brochures, and Technical Data.	http://www.rockwellautomation.com/global/literature-library/overview.page
Product Compatibility and Download Center (PCDC)	Get help determining how products interact, check features and capabilities, and find associated firmware.	http://www.rockwellautomation.com/global/support/pcdc.page

Documentation Feedback

Your comments will help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete the How Are We Doing? form at http://literature.rockwellautomation.com/idc/groups/literature/documents/du/ra-du002_-en-e.pdf.

Rockwell Automation maintains current product environmental information on its website at <http://www.rockwellautomation.com/rockwellautomation/about-us/sustainability-ethics/product-environmental-compliance.page>.

Allen-Bradley, ControlLogix, FactoryTalk, Logix 5000, PanelView, PlantPAx, NetBIOS, Rockwell Software, Rockwell Automation, RSLinx, and Studio 5000 Logix Designer are trademarks of Rockwell Automation, Inc.

Trademarks not belonging to Rockwell Automation are property of their respective companies.

Rockwell Otomasyon Ticaret A.Ş., Kar Plaza İş Merkezi E Blok Kat:6 34752 İçerenköy, İstanbul, Tel: +90 (216) 5698400

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters

Americas: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA, Tel: (1) 414.382.2000, Fax: (1) 414.382.4444

Europe/Middle East/Africa: Rockwell Automation NV, Pegasus 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

Publication PROCES-RM002I-EN-P - May 2019

Supersedes Publication PROCES-RM002H-EN-P - January 2019

Copyright © 2019 Rockwell Automation, Inc. All rights reserved. Printed in the U.S.A.