



FactoryTalk Batch Administrator Guide

Rockwell Automation Publication BATCH-UM003C-EN-D - November 2023
Supersedes Publication BATCH-UM003B-EN-D - May 2022



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).

Rockwell Automation recognizes that some of the terms that are currently used in our industry and in this publication are not in alignment with the movement toward inclusive language in technology. We are proactively collaborating with industry peers to find alternatives to such terms and making changes to our products and content. Please excuse the use of such terms in our content while we implement these changes.

Summary of changes in FactoryTalk Batch Administrator Guide

This publication contains this new or updated information. This list includes substantive updates only and is not intended to reflect all changes.

Topic	Reason for change
Event File Format on page 86	Functional change
Updated all screen captures	Usability improvement
Use Open Platform Communications on page 17	Clarification

Summary of changes in FactoryTalk Batch Administrator Guide Preface	About this manual.....11 Document organization11 Legal Notices11 Additional resources 12
FactoryTalk Batch components	Chapter 1 FactoryTalk Batch regional settings 15
FactoryTalk Batch communications	Chapter 2 Create a shared directory 17 Use Open Platform Communications (OPC)..... 17 OPC group support 18
FactoryTalk Batch security	Chapter 3 FactoryTalk Batch Server user account configuration requirements ... 19 FactoryTalk Batch system configuration20 System-wide COM security limit settings 20 FactoryTalk Batch Server install configuration 21 FactoryTalk Batch Windows Firewall exceptions..... 21 Add the Batch COM service 22 Set security on folders or files23 Add users and user groups 24 Change permissions on the BATCHCTL share 24 FactoryTalk Security for the Batch Clients..... 25 Tighten security for FactoryTalk Batch Clients 26 FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor security 27 Configure access mode security 28 Configure feature security 28 Modify FactoryTalk Batch Equipment Editor configuration options 29
FactoryTalk Batch default policy settings	Chapter 4 Product policies in the local and network directories.....31 BatchCampaign policy settings 32 BatchView policy settings 32 Equipment Editor policy settings34 Recipe Editor policy settings34
	Chapter 5

FactoryTalk Batch electronic signatures

Command verification policies..... 38

 Command signature requests..... 38

 Set parameter signature request 40

 FactoryTalk Batch Parameter Prompt Acknowledge Signature request 41

 Unit Bind Prompt Acknowledge Signature request..... 42

 Phase Bind Prompt Acknowledge Signature request 42

 Unit Bind Signature request 43

 Remove Batch Signature request..... 44

 Add Event Signature request 45

 Active Step Change Signature request 45

Parameter deviation verification policies 46

 Parameter deviation signature requests 47

 Report parameter deviation signature requests..... 48

General usage phase logic requests 48

 General usage signature requests..... 48

Electronic signature security..... 49

 Configure the ESignature network directory policy setting 50

Chapter 6

FactoryTalk Batch audit trail

Enable auditing for FactoryTalk Batch Equipment Editor or FactoryTalk Batch Recipe Editor 52

Audited FactoryTalk Batch Equipment Editor events..... 52

Audited FactoryTalk Batch Recipe Editor events..... 55

Audit messages 56

 FactoryTalk Diagnostics Viewer 57

 Standard viewer message fields in FactoryTalk Diagnostics Viewer 58

 Extended viewer message fields in FactoryTalk Diagnostics Viewer 58

 FactoryTalk AssetCentre Client 60

 Viewer message fields..... 61

Chapter 7

The FactoryTalk Batch Service Manager

Windows services 63

FactoryTalk Batch Service Manager dialog box..... 64

 Verify/Deploy Area Model dialog box..... 65

 FactoryTalk Batch Server Details dialog box..... 66

 FactoryTalk Batch Server Details dialog box - General tab..... 66

 FactoryTalk Batch Server Details dialog box - Messages tab..... 67

 FactoryTalk Batch Server Details dialog box - PCD Communications tab 67

	FactoryTalk Batch Server Details dialog box - View	
	Communications tab	68
	Start the FactoryTalk Batch Server service.....	68
	Grace periods in FactoryTalk Batch Server	69
	Enable a grace period.....	70
	Add a custom service.....	70
	Chapter 8	
The FactoryTalk Batch Server	FactoryTalk Batch Server service configuration	73
	Specify the startup type	73
	FactoryTalk Batch Server option configuration	74
	Server Options dialog box - Project Settings tab	74
	Set project directories.....	75
	Configure recipe storage.....	76
	Recipe versioning.....	76
	Server Options dialog box - Restart Control tab.....	77
	Redundant servers.....	77
	Server Options dialog box - Batch Reporting tab	79
	Perform end of batch archiving.....	80
	Perform incremental archiving	81
	Perform user-defined end of batch	82
	Never.....	82
	Archiver Event Filters tab	82
	Batch Server tab	83
	Hold propagation.....	84
	Communications timeout.....	85
	Miscellaneous.....	86
	Event File Format.....	86
	Cross Invocation Descriptors and Defaults tab.....	87
	Resource types.....	87
	Descriptors	87
	Default values.....	88
	Hyperlink Descriptors and Defaults tab	88
	Material Policies tab.....	88
	Modify the .NET registry key.....	90
	Use automatic restart control to restart the FactoryTalk Batch Server	90
	Restart the FactoryTalk Batch Server	90
	Perform a manual failover	91
	Configure the FactoryTalk Batch Server for manual failover.....	91
	Configure the material-enabled FactoryTalk Batch Server for	
	manual failover.....	92
	Command handshake timeout.....	93

Area models	Chapter 9	Update an online area model 96
		Verify and deploy an updated area model using the FactoryTalk Batch Service Manager 97
		Review the Compare/Update log 98
		Resolve runtime condition checks 99
		Update global area model 100
		Data server definitions in area model 107
		Update enumerations in area model 108
		Update operation sequences in area model 109
		Phases and classes in area model 116
		Update messages in an area model 122
		Control strategy in an area model 122
		Parameters and reports in area model 124
		Process cells and classes in area model 125
		Resources and classes in area model 126
		Signature templates in area model 128
		Units and classes in area model 128
		Update unit attributes and tags in an area model 131
	Chapter 10	View event log entry properties 135
Windows event log		Modify event log settings 138
	Chapter 11	Cross invocation overview 141
Cross invocation		Cross invocation string configuration 142
		Cross invocation string escape sequences 143
	Chapter 12	Start the FactoryTalk Batch Phase Simulator 145
		FactoryTalk Batch Phase Simulator interface 146
		Phase Configuration dialog box 146
		Change the dwell time 147
		Change the sequence of requests 148
		Indicate a continuous phase 148
		Phase Status dialog box 149
		View phase parameter tag values 149
		View and update report parameter tag values 150
		View request data tag values 150
		Change ownership of a phase 151
The FactoryTalk Batch Phase Simulator		Command a phase 151

Phase commands	152
Acknowledge and clear a request	153
Restore and clear requests.....	153
Unit Tag Status dialog box.....	154
Change the value of a unit tag.....	155

Chapter 13

Network Editor

Open the Network Editor	157
Network Editor interface	157
Hierarchy pane	158
Detail pane.....	158
Results pane.....	159
Menu bar and toolbar	160
Network Editor toolbar.....	160
Status bar	161
Configure your network	161
Select a network model.....	161
Add a server	162
Remove a server.....	162
Synchronize network models.....	162
User Preferences dialog box	162
User Preferences dialog box - General tab	163
Edit colors.....	164
Edit the application font	164
Edit information tips.....	164
User Preferences dialog box - Views tab	165
List View properties.....	165
User Preferences dialog box - Reports tab	166
Set report preferences	167
Change margins.....	167
Set the report fonts and colors.....	168
Column and row preferences in reports	170
Report printing formats	171
Preview reports	172
Print reports	172
Print a range of reports	173

Chapter 14

Server

Abnormal FactoryTalk Batch termination	175
Copied log file filename	175
Abnormal termination batch record entries.....	176
Tag verification log information	177

Tag verification log file 177
Windows Event log tag verification events.....178
FactoryTalk Batch performance chart 178
Open the FactoryTalk Batch performance chart 179

Batch identification

Chapter 15

Create a custom BatchIDCreate.dll181
Change the BatchIDCreate.dll registration 182

**Change the FactoryTalk Batch
Server user account**

Chapter 16

New server user account requirements 183
Configure your FactoryTalk Batch system with the new server user
account 183
 Server account password change locations184
 Change the server user account for the FactoryTalk Batch Server
 service184
 Change the server user account in FactoryTalk Batch COM+
 applications 185
 Change the server user account in My Computer properties.....186

**General troubleshooting errors
Index**

Chapter 17

About this manual

This manual provides administrator instructions for the FactoryTalk Batch components. It is one of a set of related manuals that describe installing, programming, and operating the FactoryTalk Batch system.

To review FactoryTalk Batch release notes and latest information regarding product compatibility refer to the [Product Compatibility and Download Center \(PCDC\)](#).

Document organization

The information in this document is presented in an order that should minimize the effort required for the configuration and administration of a FactoryTalk Batch system.

The following subjects are included:

- Configuring Regional Settings, Communications, Security, Electronic Signatures, and Batch Auditing
- Using the FactoryTalk Batch Service Manager, Windows Event Log, Phase Simulator, and Server
- Configuring Batch IDs and the Batch Server user account.
- General troubleshooting

See also

[FactoryTalk Batch communications](#) on [page 17](#)

[FactoryTalk Batch security](#) on [page 19](#)

[The FactoryTalk Batch Server](#) on [page 73](#)

[Batch identification](#) on [page 181](#)

Legal Notices

Rockwell Automation publishes legal notices, such as privacy policies, license agreements, trademark disclosures, and other terms and conditions on the [Legal Notices](#) page of the Rockwell Automation website.

Software and Cloud Services Agreement

Review and accept the Rockwell Automation Software and Cloud Services Agreement [here](#).

Open Source Software Licenses

The software included in this product contains copyrighted software that is licensed under one or more open-source licenses.

You can view a full list of all open-source software used in this product and their corresponding licenses by opening the `oss_license.txt` file located your

product's OPENSOURCE folder on your hard drive. This file is divided into these sections:

- **Components**
Includes the name of the open-source component, its version number, and the type of license.
- **Copyright Text**
Includes the name of the open-source component, its version number, and the copyright declaration.
- **Licenses**
Includes the name of the license, the list of open-source components citing the license, and the terms of the license.

The default location of this file is:

C:\Program Files (x86)\Rockwell Software\Batch\Batch View HMI Controls\Release Notes\OPENSOURCE\oss_licenses.txt

You may obtain Corresponding Source code for open-source packages included in this product from their respective project web site(s).

Alternatively, you may obtain complete Corresponding Source code by contacting Rockwell Automation via the **Contact** form on the Rockwell Automation website:

<http://www.rockwellautomation.com/global/about-us/contact/contact.page>.

Please include "Open Source" as part of the request text.

Additional resources

This table is a comprehensive documentation list for the FactoryTalk® Batch products from Rockwell Automation.

Installation, Quick Start, and Getting Results Guides

Resource	Description
FactoryTalk Batch Components Installation and Upgrade Guide (BATCH-IN002)	Provides information and procedures for FactoryTalk Batch system installation. Includes information for FactoryTalk Batch Material Manager, FactoryTalk Event Archiver, and associated FactoryTalk Batch Client and Server components.
FactoryTalk Batch View Quick Start Guide (FTBVS-QS001)	Provides information about using FactoryTalk Batch View to create, view, and command control recipes, acknowledge prompts and signatures, view equipment phases and diagnostic information, and view profile information.
FactoryTalk Batch View HMI Controls Quick Start Guide (BATCH-QS001D)	Provides a general overview of FactoryTalk Batch View HMI Controls.
FactoryTalk Batch eProcedure® Getting Results Guide (BWEPRO-GR011)	Explains the basics of FactoryTalk Batch eProcedure.
FactoryTalk Batch Getting Results Guide (BATCH-GR011)	Introduces the basics of automated batch manufacturing and the FactoryTalk Batch product components.

Resource	Description
FactoryTalk Batch Material Manager Getting Results Guide (BWMTR-GR011)	Introduces the basics of FactoryTalk Batch Material Manager.

User Guides

Resource	Description
FactoryTalk Batch Material Editor User Guide (BWMTR-UM001)	Provides access to information and procedural instructions required to configure materials and the containers to hold them. The material data is stored in the material database, which is used to create material-based recipes. This information is intended as a reference for formulators.
FactoryTalk Batch Equipment Editor User Guide (BATCH-UM004)	Provides information on creating and maintaining an equipment database (area model). The area model is available to all other FactoryTalk Batch programs, including the Recipe Editor, Batch View, and Phase Simulator.
FactoryTalk Batch PhaseManager™ User Guide (BATCHX-UM011)	Describes the integration of the FactoryTalk Batch software with the Studio 5000 Logix Designer® application and the Logix 5000™ family of controllers. The integration simplifies the configuration and maintenance of the FactoryTalk Batch automation system, provides better communication between the FactoryTalk Batch Server and the Logix 5000 controller, and significantly reduces the programming effort required to develop the phase logic code that resides in your Logix 5000 controller.
FactoryTalk Batch Recipe Editor User Guide (BATCH-UM006)	Provides instructions on using FactoryTalk Batch Recipe Editor to create and configure master recipes for use in batch automation. The interface is based on IEC 61131-3 sequential function charts to organize recipes graphically into procedures, unit procedures, operations, and phases. Build recipes using either the SFC format or a table-based format.
FactoryTalk Batch View HMI Controls User Manual (FTBVS-UM003)	Provides details about using FactoryTalk Batch View HMI Controls to monitor and interact with the production process within a FactoryTalk View SE Display Client.
FactoryTalk Batch View User Manual (FTBVS-UM002)	Provides information and procedural instructions for using FactoryTalk Batch View in a modern and intuitive portal into a comprehensive batching solution for effective operations, leveraging its own web server using HTML5 technology to provide connectivity into a FactoryTalk Batch Server.
FactoryTalk Event Archiver User Guide (BATCH-UM012)	Provides information and instructions specific to the FactoryTalk Event Archiver. Intended for use by system administrators and production supervisors.

Administrator Guides

Resource	Description
FactoryTalk Batch Administrator Guide (BATCH-UM003)	Provides instructions for configuring security and services, and implementation and use of components not typically accessed or used by batch operators, such as the FactoryTalk Batch Server.
FactoryTalk Batch eProcedure Administrator Guide (BWEPRO-UM011)	Provides procedures specific to FactoryTalk Batch eProcedure, such as implementing security. Included are instructions for tasks specific to FactoryTalk Batch, such as configuring security and services to support FactoryTalk Batch eProcedure. Provides instructions on the implementation and use of components not typically accessed or used by batch operators, such as the FactoryTalk Batch Server.

Resource	Description
FactoryTalk Batch Material Manager Administrator Guide (BWEPRO-UM011)	Provides information and instructions specific to FactoryTalk Batch Material Manager. Intended for use by system administrators and database administrators.

Reference Guides

Resource	Description
FactoryTalk Batch Material Server API Reference Manual (BWMTR-RM001)	Provides access to information regarding the interface between the FactoryTalk Batch Material Server and the FactoryTalk Batch Material Editor and FactoryTalk Batch. It is intended to be used as a reference information by custom interface developers.
FactoryTalk Batch PCD Programming Reference Manual (BATCH-RM004)	Provides information and instructions about the FactoryTalk Batch PCD interface design. It is intended to be used as a reference guide for PCD programmers.
FactoryTalk Batch Server API Reference Manual (BATCH-RM003)	Provides information regarding the interface between the FactoryTalk Batch Server and FactoryTalk Batch View – the Server Application Programming Interface (API). It is intended to be used as a reference guide by custom interface developers.
FactoryTalk Batch System Files Reference Manual (BATCH-RM005)	Provides the technical information for configuration and maintenance of a FactoryTalk Batch system. It can be used as a reference information for implementation engineers and system administrators.
FactoryTalk Batch eProcedure Instruction File Design Reference Manual (BWEPRO-RM001)	Includes information about the building of manual nstruction files for manual phases in the equipment database This information is intended to be used as a reference by instruction file authors.

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

Rockwell Automation recognizes that some of the terms that are currently used in our industry and in this publication are not in alignment with the movement toward inclusive language in technology. We are proactively collaborating with industry peers to find alternatives to such terms and making changes to our products and content. Please excuse the use of such terms in our content while we implement these changes.

FactoryTalk Batch components

This guide contains instructions for tasks specific to FactoryTalk® Batch Batch, such as configuring security and services. Instructions are also provided on the implementation and use of components not normally accessed or used by batch operators, such as the FactoryTalk Batch Server, Simulator, and performance chart.

See also

[FactoryTalk Batch regional settings](#) on [page 15](#)

[FactoryTalk Batch communications](#) on [page 17](#)

[FactoryTalk Batch security](#) on [page 19](#)

[FactoryTalk Batch electronic signatures](#) on [page 37](#)

[FactoryTalk Batch audit trail](#) on [page 51](#)

FactoryTalk Batch regional settings

FactoryTalk Batch components are supported in U.S. English only. Install the English language version of the operating system used with FactoryTalk Batch components to ensure proper operation of FactoryTalk Batch, FactoryTalk eProcedure, or FactoryTalk Batch Material Manager.

IMPORTANT Modifying regional settings is not supported.

See also

[FactoryTalk Batch security](#) on [page 19](#)

[FactoryTalk Batch communications](#) on [page 17](#)

FactoryTalk Batch communications

The FactoryTalk Batch Server uses the Component Object Model (COM) to access data from project directories and to communicate with other programs such as FactoryTalk Batch View, and data servers. All project directories accessed by the FactoryTalk Batch Server must be shared, and the server user account must have permission to access that shared directory.

See also

[Create a shared directory](#) on [page 17](#)

[Use Open Platform Communications \(OPC\)](#) on [page 17](#)

Create a shared directory

A shared directory allows computers on the network to read from and write to a common recipe, area model, or data file. The FactoryTalk Batch installation creates a shared directory named **BATCHCTL** where the FactoryTalk Batch software is installed (the default is **C:\Program Files (x86)\Rockwell Software\Batch**). If you installed your project files in a different directory, configure that directory as a shared directory. To configure a shared directory, administrative privileges are required.

To create a shared directory

1. Navigate to the project file directory, then right-click the folder, and select the sharing option.
2. Use the **File Sharing** dialog box to add each user or user group name that requires access to the project file directory.
3. Repeat for each user or user group, including the user accounts used by the FactoryTalk Batch Server and FactoryTalk Event Archiver.
4. Select **Share**.

See also

[Use Open Platform Communications \(OPC\)](#) on [page 17](#)

Use Open Platform Communications (OPC)

The FactoryTalk Batch Server supports the use of the Open Platform Communications (OPC) protocol. OPC provides a defined set of COM interfaces for data access functions.

The OPC communications protocol is used by the FactoryTalk Batch Server to communicate with process-connected devices via a data server and PC-Based

phases. COM is used to communicate with other high-level applications such as FactoryTalk Batch Material Manager.

See also

[OPC group support](#) on [page 18](#)

OPC group support

Within the OPC protocol, data is accessed through collections of data items referred to as groups. Different data servers have varying levels of OPC support, including maximum numbers of groups, maximum numbers of items per group and performance issues, and so on. Because of this, FactoryTalk Batch support of OPC is flexible in order to achieve successful communication with as many different data servers as possible. To achieve this, the FactoryTalk Batch Server is flexible in its definition of Groups, and allows for different OPC grouping configurations to communicate with the data server(s).

The manner in which the FactoryTalk Batch Server organizes the area model tags into groups is defined by data server definition files (DSDF). These files are located in the Batch\Dataservers directory. FactoryTalk Batch comes with a predefined set of DSDF files. If you need to communicate with a data server for which there is no predefined DSDF file, contact Rockwell Automation Application Support.

See also

[Use Open Platform Communications \(OPC\)](#) on [page 17](#)

FactoryTalk Batch security

The following information explains administering security for the FactoryTalk Batch Server using Windows security and implementing FactoryTalk Security for the FactoryTalk Batch Client components. For information on the administration of your FactoryTalk Security system as a whole, see **FactoryTalk Help**.

To open FactoryTalk Help, select **Start > Rockwell Software > FactoryTalk Help**.

To help meet secure system design requirements, review these publications:

- [Configure System Security Features User Manual](#) (publication SECURE-UM001)
- [System Security Design Guidelines Reference Manual](#) (publication SECURE-RM001)
- [Converged Plantwide Ethernet \(CPwE\) Design and Implementation Guide](#) (publication ENET-TD001)

To learn about implementing CIP Security, see [CIP Security with Rockwell Automation Products](#) (publication SECURE-AT001). See also

[FactoryTalk Batch Server user account configuration requirements](#) on [page 19](#)

[FactoryTalk Batch system configuration](#) on [page 20](#)

[FactoryTalk Security for the Batch Clients](#) on [page 25](#)

[FactoryTalk Batch default policy settings](#) on [page 31](#)

FactoryTalk Batch Server user account configuration requirements

Create a Windows user account for the FactoryTalk Batch Server prior to installing FactoryTalk Batch applications. When installing FactoryTalk Batch and FactoryTalk eProcedure applications, you are prompted to enter this user account.

When creating the server user account, the following configuration requirements must be met.

- **The password must be configured to never expire.**
If the password expired, the batch server service would fail to log on.
- **The server user account must never be disabled or deleted.**
If this account is ever disabled or deleted, the batch server service would fail to log on.

- **The domain user account must have a unique name.**
If the server user account is a domain account, remove any local user accounts with the same name.
- **The Server user account or user group must exist on all workgroup computers.**
If you are using a local account and expect that account to have access to resources on other computers in a workgroup environment, you must create accounts with the same name and password on each computer in the workgroup.

See also

[FactoryTalk Batch system configuration](#) on [page 20](#)

[FactoryTalk Batch security](#) on [page 19](#)

FactoryTalk Batch system configuration

The FactoryTalk Batch components installation program configures the local computer with the necessary COM Security settings to allow remote communication between the FactoryTalk Batch Server and client applications.

The configuration of the local computer during installation depends on whether you are installing a server or client application, a FactoryTalk Batch or eProcedure application and whether the Windows Firewall is enabled or disabled.

See also

[System-wide COM security limit settings](#) on [page 20](#)

[FactoryTalk Batch Server install configuration](#) on [page 21](#)

[FactoryTalk Batch Windows Firewall exceptions](#) on [page 21](#)

[Add the Batch COM service](#) on [page 22](#)

System-wide COM security limit settings

To support DCOM client-server communication, configure the local computer's COM Security Limits with these access, launch, and activation permissions.

Access Permissions

Security Limits for ANONYMOUS LOGON:

- Local Access Permissions
- Remote Access Permissions

Launch and Activation Permissions

Security Limits for ANONYMOUS LOGON:

- Local Launch Permissions
- Local Activation Permissions
- Remote Activation Permissions

See also

[FactoryTalk Batch Server install configuration](#) on [page 21](#)

[FactoryTalk Batch Windows Firewall exceptions](#) on [page 21](#)

[Add the Batch COM service](#) on [page 22](#)

[FactoryTalk Batch system configuration](#) on [page 20](#)

FactoryTalk Batch Server install configuration

The installation of the FactoryTalk Batch Server creates the Windows user group **batchsvr_group** on the FactoryTalk Batch Server. This group is automatically assigned all the user rights and default access permissions required for the FactoryTalk Batch Server to function correctly.

The installation adds the FactoryTalk Batch Server user account to the following:

- Administrators group
- batchsvr_group
- MasterRecipeAuthor
- MasterRecipeViewer
- Batch COM+ Application
- FactoryTalk Batch Server service
- FactoryTalk Event Archiver service
- FactoryTalk eProcedure Server service (eProcedure Server installation)

The services for the FactoryTalk Batch Server, FactoryTalk Event Archiver, FactoryTalk eProcedure Server, and the Batch COM+ Application require identical user accounts to support the FactoryTalk Batch Server functionality.

See also

[FactoryTalk Batch system configuration](#) on [page 20](#)

[FactoryTalk Batch Windows Firewall exceptions](#) on [page 21](#)

[Add the Batch COM service](#) on [page 22](#)

FactoryTalk Batch Windows Firewall exceptions

If the Windows Firewall is enabled when you install FactoryTalk Batch or FactoryTalk eProcedure applications, the installation program adds the following exceptions to the local computer:

Windows Firewall Exceptions

• batchsvr.exe (Batch server installs)	• NmspHost.exe
• CodeMeter Runtime Server	• OPCEnum.exe
• CodeMeterFWEx1	• Port 1433 TCP (SQL Server installs)
• DaClient.exe	• Port 137 NetBIOS Name Service
• EventClientMultiplexer.exe	• Port 135 TCP
• EventServer.exe	• Port 80 TCP
• Flexsvr.exe	• RDCyHost.exe
• FTCounterMonitor.exe	• RnaDiagnosticsSrv.exe
• FTSPVStudio.exe	• RnaDiagReceiver.exe
• FTSysDiagSvcHost.exe	• RnaDirServer.exe
• iexplore.exe	• RsvcHost.exe
• Imgrd.exe	• VStudio.exe

If the Windows Firewall is enabled, and not externally controlled, the **Windows Firewall Configuration Utility** runs automatically during installation. If this utility does not run automatically, manually run it on each computer in your FactoryTalk Batch system after installing servers or clients.

To manually run the FactoryTalk Windows Firewall Configuration Utility, select **Start > Rockwell Software > Windows Firewall Configuration Utility**.

See also

[FactoryTalk Batch system configuration](#) on [page 20](#)

[System-wide COM security limit settings](#) on [page 20](#)

[FactoryTalk Batch Server install configuration](#) on [page 21](#)

[Add the Batch COM service](#) on [page 22](#)

Add the Batch COM service

To add the COM service, run the **BatchIDCreationCOMPlusInstall** script and then add the server user account to the FactoryTalk Batch Server service, FactoryTalk Event Archiver service, and the Batch COM+ Application. If FactoryTalk eProcedure is part of your system, add the server user account to the eProcedure Server service.

To add the Batch COM service

1. In Windows Explorer, expand **Program Files x86 > Rockwell Software > Batch > Scripts**.
2. Double-click **BatchIDCreationCOMPlusInstall.vbs** to add the FactoryTalk Batch COM service to the computer.
3. Select **Start > Windows Administrative Tools > Component Services > Computers > My Computer > COM+ Applications**.
4. Right-click **Batch > Properties**.
5. Select the **Identity** tab, select **This user**, and enter the user name and password for the FactoryTalk Batch Server.
6. Select **OK** and then close Component Services.

7. Select **Start > Windows Administrative Tools > Services**.
8. Right-click **FactoryTalk Batch Server > Properties**.
9. Select the **Log On** tab and then select **This Account**.
10. Enter the user account name and password for the server.
11. Select **OK**.
 - Add the server user account to FactoryTalk Event Archiver service if you are using FactoryTalk Event Archiver in incremental mode.
 - Add the server user account to the eProcedure Server service if eProcedure is part of your FactoryTalk Batch system.
12. Close **Services**.
13. Restart the computer.

See also

[FactoryTalk Batch Server install configuration](#) on [page 21](#)

[FactoryTalk Batch Windows Firewall exceptions](#) on [page 21](#)

[FactoryTalk Batch system configuration](#) on [page 20](#)

Set security on folders or files

Limit log file and event journal (.evt or .evtj) access to prevent data from being lost. Setting security on the folders or files is one suggested method. Set the security according to company requirements.

Use Notepad to open log files. If a log file is opened with a program other than Notepad, the FactoryTalk Batch Server may not run due to an access error.

IMPORTANT FactoryTalk Batch Server continues writing information to log files and event journals (.evt or .evtj) even when they are open. If a file is opened and then saved, data that may have been written to the event journal or log file while the file is open is lost.

To set security on folders or files

1. In **Windows Explorer**, locate the folder or file.
2. Right-click the folder or file, and then select **Properties**.
3. Select the **Security** tab, select **Edit**, and then select **Add**.
4. From the **Select Users, Computers, Service Accounts, or Groups** dialog box, select **Advanced** and then select **Find Now**.
5. Select a user or group from the Search results list, and then select **OK**.
6. Repeat for each user or group that requires access to the folder or file.



Be sure to add the FactoryTalk Batch Server user account and then configure the account with **Full Control** permissions.

7. Select **OK** to return to the **Properties** dialog box.
8. From the **Group or user name** list, select a user or group, and then select the permissions in the **Permissions** area. Repeat for each user or group.
9. Select **OK**.

See also

[FactoryTalk Batch Server user account configuration requirements](#) on [page 19](#)

Add users and user groups

Set security on folders and files to limit log file and event journal (.evt or .evtj) access.

To add users and user groups:

1. In **Windows Explorer**, locate the folder or file.
2. Right-click the folder or file, and then select **Properties**.
3. Select the **Security** tab, select **Edit**, and then select **Add**.
4. From the **Select Users, Computers, Service Accounts, or Groups** dialog box, select **Advanced** and then click **Find Now**.
5. Select a user or group from the **Search results** list, and then select **OK**.
6. Repeat for each user or group that requires access to the folder or file.



Tip: Be sure to add the FactoryTalk Batch Server user account and then configure the account with **Full Control** permissions.

7. Select **OK** to return to the **Properties** dialog box.
8. From the **Group or user name** list, select a user or group, and then select the permissions in the **Permissions** area. Repeat for each user or group.
9. Select **OK**.

See also

[Set security on folders or files](#) on [page 23](#)

Change permissions on the BATCHCTL share

Change the permissions on the BATCHCTL share to tighten security by removing the **Everyone** group and adding a new group that contains all the Windows users that need to access the share.

IMPORTANT This new group allows applications to access the BATCHCTL share while locking out non-domain users.

Include these user types in the group:

- The FactoryTalk Batch Server and FactoryTalk Event Archiver user.
- Windows users logged in when the FactoryTalk Batch Equipment Editor, FactoryTalk Batch Recipe Editor and FactoryTalk Batch View are launched.

To change permissions on the BATCHCTL share

1. In **Windows Explorer**, locate the Batch folder. The default location is C:\Program Files (x86)\Rockwell Software\Batch.
2. Right-click the Batch folder, and then select **Properties**.

3. Select the **Security** tab and do the following:
 - a. Select **Edit**, select the **Everyone** group, and then select **Remove**.
 - b. After removing the **Everyone** group, select **Add**.
4. From the **Select Users, Computers, Service Accounts, or Groups** dialog box, select **Advanced** and then click **Find Now**.
5. Select a group from the **Search results** list, and then select **OK**.
6. Select **OK** to return to the **Properties** dialog box.
7. From the **Group or user name** list, select a user or group, and then select the permissions in the **Permissions** area. Repeat for each user or group.
8. Select **OK**.

FactoryTalk Security for the Batch Clients


Security settings are shared among FactoryTalk enabled products across a network or on the same computer using FactoryTalk Security in FactoryTalk Batch View, HMI Controls, FactoryTalk Batch Equipment Editor, and FactoryTalk Batch Recipe Editor. FactoryTalk Directory is the infrastructure which contains the policies defined for products and resources.

FactoryTalk Batch product policies are created in both the Local Directory and the Network Directory when the FactoryTalk Services Platform is installed. The FactoryTalk Batch product policies are used to restrict access to the FactoryTalk Batch Client components and their features. For a complete list of FactoryTalk Batch secured resources and their default product policy settings, see FactoryTalk Batch default policy settings.

The following information is specific to configuring FactoryTalk Security for the FactoryTalk Batch Clients. For information on securing your FactoryTalk system as a whole, see **FactoryTalk Help**.

To open FactoryTalk Help, click **Start > Rockwell Software > FactoryTalk Help**.

The FactoryTalk Services Platform installation process:

- **Creates a FactoryTalk Local Directory and a FactoryTalk Network Directory on the computer.** When the install process finishes, both directories are fully configured and ready for use.
 -  **Tip:** Security settings in the Network Directory are completely separate from the security settings in the Local Directory. For example, changes made to the security settings in the Network Directory do not change security settings in the Local Directory.
- **Allows any Windows Administrator user account to log on to either directory.** The installation process automatically adds the local Windows Administrators group to the FactoryTalk Administrators group in each FactoryTalk Directory.
- **Allows any authenticated Windows user to log on to the FactoryTalk Local Directory.** The installation process automatically adds all members of the local Windows Authenticated Users group to the FactoryTalk Local Directory.

- **Allows all new user accounts full access permissions.** The installation process automatically sets system policies to allow all new user accounts added to either directory full access to that directory by default.

See also

[FactoryTalk Batch default policy settings](#) on [page 31](#)

[Tighten security for FactoryTalk Batch Clients](#) on [page 26](#)

Tighten security for FactoryTalk Batch Clients

Restrict access to specific features of your individual FactoryTalk products. Only users with access can use secured product features.

For example, when you set up product policies for FactoryTalk Batch, restrict use of the Abort command to specific users. This prevents automated batch processes from being unintentionally aborted during run time.

To configure security for multiple features in FactoryTalk, use the Feature Security for Product Policies dialog box.

To configure security for the FactoryTalk Batch features, refer to the secured product policies defined for the FactoryTalk Batch Clients.

To tighten security for FactoryTalk Batch Clients:

- Delete the Windows Authenticated Users group from the Local Directory.
- Create new FactoryTalk user accounts and groups in the FactoryTalk Directory. Create user accounts or groups to secure the FactoryTalk Batch Clients.
- Add user accounts and groups created in Windows to the FactoryTalk Directory. Windows accounts added to the FactoryTalk Directory are called Windows-linked user accounts and groups.

IMPORTANT Windows Workgroup User Accounts are not supported in FactoryTalk. If you use Windows workgroups, you cannot administer user accounts centrally for FactoryTalk Batch Client applications. FactoryTalk Directory does not allow multiple user accounts to have the same name and password.

- Remove the All Users group from the FactoryTalk Batch product policies. The FactoryTalk Services Platform installation process adds the All Users group to all FactoryTalk product policies.
- Add user accounts and groups to the FactoryTalk Batch product policies. To secure FactoryTalk Batch resources, add only the necessary user accounts and groups to the corresponding product policies.

See also

FactoryTalk Batch Client security

FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor security

Security for FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor is implemented using the FactoryTalk Security product policies defined in the FactoryTalk Network Directory. There are two product policy options available:

- **Access Modes** governs access to the editors (only FactoryTalk users can be granted access). There are two levels of working access to area models in FactoryTalk Batch Equipment Editor, and recipes in FactoryTalk Batch Recipe Editor.
- **View Only** provides full viewing privileges, but no changes can be saved.
- **Full Edit** provides full editing privileges.
- **Feature Security** governs which FactoryTalk users can secure area models and recipes to the FactoryTalk Network Directory Security Authority Identifier (SAI).
 - The **Area Model: Secure** policy governs which users can secure an area model to a particular SAI in FactoryTalk Batch Equipment Editor.
 - The **Recipe: Secure** policy governs which users can secure a recipe to a particular SAI in FactoryTalk Batch Recipe Editor.

Additionally, there are other **Configuration Options** product policies created in the FactoryTalk Local Directory for the editors.

- To tighten security, remove the **All Users** group from FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor product policies in the FactoryTalk Directory.
- FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor support the use of an external security device, such as a retinal scanner.
- In addition to FactoryTalk Security, configure Windows security on FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor files and folders in Windows Explorer.
- RDB security can also be configured when using RDB recipes that are stored in SQL Server.

See also

[Configure access mode security](#) on [page 28](#)

[Configure feature security](#) on [page 28](#)

[Product policies in the local and network directories](#) on [page 31](#)

[Modify FactoryTalk Batch Equipment Editor configuration options](#) on [page 29](#)

Configure access mode security

Configure the two levels of working access to area models in FactoryTalk Batch Equipment Editor, and recipes in FactoryTalk Batch Recipe Editor.

To configure the access mode security:

1. Open the FactoryTalk Administration Console and log on to the appropriate FactoryTalk Directory.
2. Expand **System > Policies > Product Policies > Batch > Equipment Editor (or Recipe Editor)**.
3. Right-click **Access Modes** and then select **Properties**.
4. Select the policy setting you want to configure and then click the browse button.
5. In the **Configure Securable Action** dialog box:
 - Select a user or group and then click **Remove**.
 - Allow or deny a user or group access to the feature by selecting or clearing the corresponding check box and then clicking **OK**.
 - Add a user account or user group by clicking **Add**. This option displays the **Select User and Computer** dialog box.
6. In the **Select User or Group** dialog box:
 - Select a user, user group, computer or computer group and then click **OK**.
 - Click **Create New** to create and then add a new user, user group or computer group.
7. Click **OK** to close the **Access Modes Properties** dialog box.
8. Restart all FactoryTalk Batch components to update changes made in the FactoryTalk Directory.

See also

[FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor security](#) on [page 27](#)

[Configure feature security](#) on [page 28](#)

[Modify FactoryTalk Batch Equipment Editor configuration options](#) on [page 29](#)

Configure feature security

Configure which users can secure area models and recipes to the FactoryTalk Network Directory Security Authority Identifier (SAI).

To configure feature security:

1. Open the FactoryTalk Administration Console and log on to the appropriate FactoryTalk Network Directory.

2. Expand **System > Policies > Product Policies > Batch > Equipment Editor** (or **Recipe Editor**).
3. Right-click **Feature Security** and then select **Properties**. The **Feature Security Properties** dialog box opens.
4. Select **Area Model: Secure** and then click browse (...). For FactoryTalk Batch Recipe Editor, the policy is **Recipe: Secure**.
5. In the **Configure Securable Action** dialog box you can:
 - Click **Add** to add a user account or user group using the **Select User and Computer** dialog box:
 - Select a user, user group, computer or computer group and then click **OK**.
 - Click **Create New** to create and then add a new user, user group, or computer group.
 - Allow or deny a user or group access to the feature by selecting or clearing the corresponding check box and then clicking **OK**.
 - Select a user or group and then click **Remove**.
6. Click **OK** to close the **Feature Security Properties** dialog box.



Tip: Restart all FactoryTalk Batch components to update changes made in the FactoryTalk Network Directory.

See also

[FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor security](#) on [page 27](#)

[Configure access mode security](#) on [page 28](#)

[Modify FactoryTalk Batch Equipment Editor configuration options](#) on [page 29](#)

Modify FactoryTalk Batch Equipment Editor configuration options

Modify the configuration options in product policies created in the FactoryTalk Local Directory for the editors.

The following example is specific to FactoryTalk Batch Equipment Editor. FactoryTalk Batch Recipe Editor configuration options are configured similarly.

To modify FactoryTalk Batch Equipment Editor configuration options:

1. Open the FactoryTalk Administration Console and then log on to the **FactoryTalk Local Directory**.
2. Expand **System > Policies > Product Policies > Batch > Equipment Editor > Configuration**.
3. Right-click **Options** and then select **Properties**.

4. From the **Options Properties** dialog box, configure the policy settings and then click **OK**.



Tip: Restart all FactoryTalk Batch components to update changes made in the FactoryTalk Directory.

See also

[Configure access mode security](#) on [page 28](#)

[FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor security](#) on [page 27](#)

[Configure feature security](#) on [page 28](#)

[Product policies in the local and network directories](#) on [page 31](#)

FactoryTalk Batch default policy settings

When the FactoryTalk Services Platform is installed, FactoryTalk Security product policies are created in the FactoryTalk Directory for FactoryTalk Batch View clients.

The product policies for FactoryTalk Batch Clients are created in both the Local and Network Directories, except for the **Configuration Options** which are only created in the Local Directory.

See also

[Product policies in the local and network directories](#) on [page 31](#)

Product policies in the local and network directories

The FactoryTalk Batch components have product policies that can be secured with FactoryTalk Security.

Policies are applied to secured objects that can then be configured to require user validation prior to execution. By default, all users have access to FactoryTalk Batch components. To tighten security, remove or restrict the **All Users** group and then add the necessary user accounts to the policy setting. To require user validation for a command, set the associated **Confirmation** policy setting to **True**. Additionally, you can restrict access to resources based on where a user is physically located, such as a computer used to perform actions. For details, see FactoryTalk Help.

The **System\Policies\Product Policies\Batch\Feature Security** setting is used to control whether users can make changes to the area model while the batch server is running.

Security Objects	Product Policy Descriptions
Configure Runtime Area Model Deployment	<p>Allow access to Runtime Area Model Deployment. With this permission users can make changes to the area model and deploy the changes while the area model is online with the batch server.</p> <p>By default, only members of the Administrator group have access.</p>

The product policy configurations are created in the FactoryTalk Local and Network Directories for FactoryTalk Batch. FactoryTalk Batch product policy is located in **FactoryTalk Directory > System > Policies > Product Policies > Batch**.

BatchCampaign policy settings

BatchCampaign policies are provided for backwards compatibility with Rockwell Automation product Batch Campaign Manager. Batch Campaign Manager is no longer licensed or supported by Rockwell Automation and has been released to the market as open source. The source code for Batch Campaign Manager can be downloaded from the [Rockwell Automation Sample Code Library](#).

BatchView policy settings

BatchView product policies determine how clients interact with the different FactoryTalk Batch components. Configure policies in the following areas:

- Batch View Server (modern clients)
- Commands (legacy clients)
- Phase Commands (legacy clients)
- View (legacy clients)

The product policy configurations can be created in either the FactoryTalk Local and Network Directories. FactoryTalk Batch product policy is located in **FactoryTalk Directory > System > Policies > Product Policies > Batch > BatchView**.

Batch View Server

The Batch View Server security policy settings enable the modern FactoryTalk Batch View client and FactoryTalk Batch View HMI User Controls within a FactoryTalk View SE Display Client to control aspects of the Batch Server.

Confirmation Authorization is enabled or disabled for all of these settings as a whole instead of for each setting individually.

- The **Command Batch** section contains settings related to the batch process.
- The **Command Equipment** section contains settings related to the resources and equipment.
- The **Command Prompts** section contains settings related to operator prompts.

Security Objects	Product Policy Descriptions
Command Batch	
• Abort	• Abort a batch
• Add Batch	• Add a batch
• Bind	• Bind a batch
• Change Active Step	• Enables or disables the Active Step Change toggle.
• Clear Failure	• Clears all failures and system-generated messages
• Comment	• All the Operator to enter a comment in the event journal.
• Disconnect	• Disconnect an equipment module
• Force Transition	• Force transition to next step
• Hold	• Hold a batch
• Material Server Reconnect	• Reconnect to the Material Server
• Auto Mode	• Run the batch in automatic mode
• Manual Mode	• Run the batch in manual mode

Security Objects	Product Policy Descriptions
• Semi Auto Mode	• Run the batch in semi-automatic mode
• Override Expression	• Allow the Operator to override the expression by assigning a new value.
• Override Expression Clear	• Allow the Operator to clear the assigned override value.
• Remove	• Remove a batch
• Restart	• Restart a batch
• Resume	• Resume a batch
• Change Parameter	• Change a parameter value
• Start	• Start a batch
• Stop	• Stop a batch
• Tag Verify	• Start the tag verification process
• Timer Complete	• Forces a timer step to complete
• Timer Reset	• Resets the ELAPSED_TIME in a Timer step to 0 (if the Timer step is configured as a COUNT_DOWN timer the REMAINING_TIME will be reset to the setpoint value).
Command Equipment	
• Abort	• Abort an equipment phase
• Acquire	• Acquire ownership of the equipment resource
• Clear Failure	• Clear a failure on the equipment module
• Container Data	• Configure the set of containers from which this equipment module can select material, or into which it can distribute material, or both.
• Disconnect	• Disconnect from the equipment resource
• Hold	• Temporarily stops execution of the batch at a step.
• Auto Mode	• Put the equipment into auto mode giving control of the running batch to the FactoryTalk Batch Server.
• Semi Auto Mode	• Put the equipment into semi-auto mode so you can control the execution of all filter steps in the current batch.
• Release	• Release ownership of the equipment resource
• ReOrder	• Change the priority order for the equipment resource
• Reset	• Resets the resource to the idle state.
• Restart	• Restarts a batch that is currently on the batch list.
• Resume	• Continues processing on a batch that is paused.
• Stop	• Performs an orderly termination of a currently running batch.
• Start	• Starts a batch that is currently on the batch list
Command Prompts	
• Acknowledge	• Acknowledges a prompt.
• Reactivate	• Reactivate past instructions in a phase that is currently RUNNING that does not have pending, reactivated steps.
• Sign	• Complete an electronic signature record for the resource.
• Sign Cancel	• Cancel the electronic signature request for the resource
Confirmation Authorization	
• Disable Confirmation	• When selected, the Batch View Server commands will not require additional confirmation before being sent from the client component to the batch server.

Equipment Editor policy settings

Commands

These policy settings apply to the legacy FactoryTalk Batch View, eProcedure client, and ActiveX controls, which are no longer supported.

Phase Commands

These policy settings apply to the legacy FactoryTalk Batch View, eProcedure client, and ActiveX controls, which are no longer supported.

View

These policy settings apply to the legacy FactoryTalk Batch View, eProcedure client, and ActiveX controls, which are no longer supported.

Equipment Editor product policy settings are used to control access to the Equipment Editor and the area model.

The product policy configurations can be created in either the FactoryTalk Local and Network Directories. FactoryTalk Batch product policy is located in **FactoryTalk Directory > System > Policies > Product Policies > Batch > Equipment Editor.**

Security Objects	Product Policy Descriptions
Recipe Editor – Access Modes	
<ul style="list-style-type: none"> • Full Edit 	<p>Once logged on, the user can edit and save changes to the area model.</p> <p>To tighten security, remove or restrict the All Users group and then add the necessary user accounts.</p> <p>Default = All Users have access.</p>
<ul style="list-style-type: none"> • ViewOnly 	<p>Once logged on, the user can view and edit the Area Model but cannot save changes.</p> <p>Default = All Users have access.</p>
Equipment Editor -- Feature Security	
<ul style="list-style-type: none"> • Area Model: Secure 	<p>Once logged on, Full Edit users can enable Security Authority for an area model. The secured area model contains the same Security Authority Identifier (SAI) as stored in the FactoryTalk Network Directory. Thereafter, the area model can only be opened and worked on in FactoryTalk Batch Equipment Editor under the same SAI. Attempting to open the area model under another SAI results in a security error.</p>

Recipe Editor policy settings

Recipe Editor product policy settings are used to control access to the Recipe Editor and the recipes.

The product policy configurations can be created in either the FactoryTalk Local and Network Directories. FactoryTalk Batch product policy is located in **FactoryTalk Directory > System > Policies > Product Policies > Batch > Recipe Editor**

Security Objects	Product Policy Descriptions
------------------	-----------------------------

Recipe Editor – Access Modes	
<ul style="list-style-type: none"> • Full Edit 	<p>Once logged on, the user can edit and save changes to recipes.</p> <p>To tighten security, remove or restrict the All Users group and then add the necessary user accounts.</p> <p>Default = All Users have access.</p>
<ul style="list-style-type: none"> • ViewOnly 	<p>Once logged on, the user can view recipes, but cannot make changes or save the recipe with a new name using the Save As function.</p> <p>Default = All Users have access.</p>
Recipe Editor -- Feature Security	
<ul style="list-style-type: none"> • Recipe: Secure 	<p>Once logged on, Full Edit users can enable Security Authority for a recipe. The secured recipe contains the same Security Authority Identifier (SAI) as stored in the FactoryTalk Network Directory. Thereafter, the recipe can only be opened and worked on in FactoryTalk Batch Recipe Editor if the current FactoryTalk Network Directory has the same SAI. Attempting to open the recipe under another SAI results in a security error.</p>

FactoryTalk Batch electronic signatures

FactoryTalk Batch electronic signatures are electronically recorded representations of a signature and their associated data. Signature data includes time stamps, security permissions, meanings for signoffs, and comments. When a signature request is created, an entry is recorded in the Event Journal. If the signature request is user generated, a **Signature** dialog box is displayed. Additional journal entries are recorded when a signature request is completed, user canceled, system canceled, or when a signoff is successful or unsuccessful. These electronic signatures include all the components necessary for compliance with 21 CFR Part 11.

A **Signature** dialog box displays context information to the user based on the signature request type. For example, a **Command Signature** dialog box displays **Time Generated**, **Batch ID**, **Procedure ID**, and **Command**.

There are several steps involved in configuring and executing electronic signatures.

IMPORTANT The FactoryTalk Batch Server responds differently when electronic signatures are configured in the area model. Be aware of the behavior changes before a decision is made to implement electronic signatures.

The requested user information in a **Signature** dialog box is defined by the signature template configured in the area model. Signature templates determine the number of signoffs required to complete a signature request. Each signoff has associated security permissions, meaning, and may require a comment. Signature templates are used with:

- Command Verification Policies
- Parameter Deviation Verification Policies
- General Usage Phase Logic Requests
- eProcedure Step Verifications

Signature templates are used to generate signature requests displayed in FactoryTalk Batch View, HMI Controls, and eProcedure.

Signature templates determine the credentials and number of signoffs required for steps defined as part of a Recipe Approval process.

See the *FactoryTalk Batch Equipment Editor User Guide* for information on configuring signature templates.

See also

[Command verification policies](#) on [page 38](#)

[Parameter deviation verification policies](#) on [page 46](#)

[General usage phase logic requests](#) on [page 48](#)

[Electronic signature security](#) on [page 49](#)

Command verification policies

The area model supports several types of command verification policies that can generate signature requests before, during, or when batch execution is complete.

Actions initiated in FactoryTalk Batch View, HMI Controls, eProcedure or through the FactoryTalk Batch Server API, generate signature requests. For information on executes, see the *FactoryTalk Batch Server API Communication Language Reference Guide*.

See also

[Command signature requests](#) on [page 38](#)

[FactoryTalk Batch Parameter Prompt Acknowledge Signature request](#) on [page 41](#)

[Unit Bind Signature request](#) on [page 43](#)

[Active Step Change Signature request](#) on [page 45](#)

Command signature requests

If a command verification policy is associated with a signature template, and a command is issued against a batch, a **Command Signature** request is generated and a dialog box is displayed.

For example: If a HOLD command requires a signature, then when a HOLD command is issued against a batch, an electronic signature request is generated and a dialog box is displayed. For information on command signatures, see **Acknowledge a Command Signature** in the *FactoryTalk Batch View User Guide*.

The following is an explanation of the modifications made to support FactoryTalk Batch electronic signatures:

Command Execute	
Batch Server	When a legal command execute is received, the FactoryTalk Batch Server checks the area model configuration to determine if a signature request is required. If yes, then the command action is queued and a signature request is generated. Exception: ABORT Command is not queued.
Response	An ActionID is returned by the execute used to locate the generated signature.
Completion	When all signature signoffs generated by the queued command are complete, the command is issued to the target procedure. Exception: ABORT command is issued regardless of Signature state.

Command Execute	
Cancellation	When any signature is user canceled, then the queued command is terminated and all other signature requests associated with the same command action are system canceled. Exception: ABORT command cannot be canceled by the user.
Overlap	If a command execute is received by a procedure that has incomplete signature requests for a prior command execute action, then the new command execute fails. A failed execute's response item contains an appropriate error message. Exception: ABORT Command creating a command overlap terminates the original command action causing a system cancel instead of a failure. If ABORT overlaps with an ABORT command, then any incomplete signatures for the first command are system canceled and a new signature request is generated for the second command.
Automatic Cancellation	When a procedure transitions to a state in which a queued command action is not legal, the command execute action is terminated, and its associated incomplete signature requests are system canceled.
Cmd	Commands used with command verification policies are: START STOP AUTO-MODE PAUSE HOLD RESTART MAN-MODE RESUME ABORT RESET CLEAR_FAILURES DISCONNECT SEMI-AUTO-MODE

When a Command Verification Policy is configured for a command, and the same Command is issued in Manual Phase Control against a Phase, a **Command Signature** request is generated.

The following is an explanation of the modifications made to support FactoryTalk Batch electronic signatures:

Phase, Phase2, Phase3 Executes	
Batch Server	When a legal command execute is received, the FactoryTalk Batch Server checks the area model to determine if a signature request is required. If yes, then the command is queued and a signature request is generated. Exceptions: <ul style="list-style-type: none"> • Phase Reset and Phase Disconnect. • Batch is created regardless of whether a Signature is required, or if the Signature is canceled. • ABORT Command is not queued.
Response	An ActionID is returned by the Execute used to locate the generated signature.
Completion	When all Signature Signoffs generated by the queued command are complete, then the Command is issued to the target procedure. Exception: ABORT Command issued regardless of Signature state.
Cancellation	When any Signature is user canceled, then the queued Command is terminated and all other Signature Requests associated with the same command action are System Canceled. Exception: ABORT Command cannot be canceled by the user.

Phase, Phase2, Phase3 Executes

Overlap	<p>If a Phase Execute is received by a procedure that has incomplete Signature Requests for a prior Phase Execute Action, then the new Phase Execute fails. If a Command Execute is received by a phase that has incomplete Signature Requests for a prior Phase Reset or Phase Disconnect, then the new Command Execute fails. The failed Execute's response item contains an appropriate error message.</p> <p>Exception: ABORT Command creating a Command overlap terminates the original Command Action causing a System Cancel instead of a failure. If ABORT overlaps with an ABORT Command, then any incomplete Signatures for the first command are System Canceled and a new Signature Request is generated for the second command.</p>
Automatic Cancellation	<p>When a procedure transitions to a state in which a queued Action would not be legal, the Execute is terminated, and its associated incomplete Signature Requests are System Canceled.</p>
Warm Reboot	<p>Signature Requests generated in response to a Phase Execute Action are System Canceled in the event of a FactoryTalk Batch Server warm boot.</p>

See also

[Command verification policies](#) on [page 38](#)

[Set parameter signature request](#) on [page 40](#)

Set parameter signature request

If a Parameter Change command verification policy is configured in the area model and associated with a signature template, then a **Set Parameter Signature** request is generated when a user attempts to change a recipe parameter value or a formula parameter value.

The following is an explanation of the modifications made to support FactoryTalk Batch electronic signatures:

SetParm Execute

Batch Server	<p>If the target is a recipe phase parameter, the FactoryTalk Batch Server checks the area model to determine if a Signature Request is required for SetParm Execute confirmation. If yes, then a Signature Request is generated. The server also checks for deviations on the recipe phase parameter. If the SetParm Execute target is a formula parameter, then the server checks all recipe phase parameters that reference the formula parameter. If deviations are detected, then the appropriate Signature Request(s) are generated. If one or more signatures are generated, then storing the specified value to the parameter is deferred until all Signature Requests are complete.</p>
Response	<p>An ActionID is returned by the execute that is used to locate the generated signature(s).</p>
Completion	<p>When all generated Signature Requests associated with the SetParm Execute are complete, then the value passed in the SetParm Execute Action is stored into the specific parameter.</p>
Cancellation	<p>When any incomplete Signature associated with the SetParm Execute is cancelled, then the entire SetParm Execute Action is terminated and all Signature Requests associated with the same SetParm Execute Action are System Canceled.</p>
Overlap	<p>If a SetParm Execute is received targeting a parameter that has incomplete Signature Requests for a prior SetParm Execute Action, then the new SetParm Execute fails. The failed Execute's response item contains an appropriate error message. The specified value is not stored in the specified parameter.</p>

SetParm Execute	
Automatic Cancellation	If any incomplete Signature Requests associated with a SetParm Execute is user cancelled, then the Execute Action is terminated and its remaining incomplete Signature Requests are System Cancelled.
Warm Reboot	Signature Requests generated in response to a SetParm Execute Action are restored in the event of a FactoryTalk Batch Server warm boot.

See also

[Command verification policies](#) on [page 38](#)

FactoryTalk Batch Parameter Prompt Acknowledge Signature request

If an Ack Command Verification policy is configured in the area model and associated with a signature template, then a **Recipe Parameter Prompt Signature** request is generated when a prompt is used to retrieve a Recipe Parameter Phase value.

The following is an explanation of the modifications made to support FactoryTalk Batch electronic signatures:

Parameter Value Ack Execute	
Batch Server	If an Ack execute is received in response to an UnacknowledgedEvent generated to retrieve a Recipe Parameter Phase value, then the FactoryTalk Batch Server generates a confirmation Signature prompt if so configured in the area model. The server also performs configured deviation checks on the value that is to be stored in the Recipe Parameter. If deviations are detected then a signature request(s) is generated. If one or more signatures is generated, then storing the specified value is delayed until the signature requests are complete.
Response	An ActionID is returned by the Execute used to locate the generated signature(s).
Completion	When all Signature Signoffs generated by the Execute are completed, then the parameter value is stored.
Cancellation	When any signature generated by a Parameter Value Ack execute is canceled, then the Parameter Value Ack Action is terminated and its associated incomplete Signature Requests are System Cancelled.
Overlap	If an Ack execute response to a parameter value prompt is received while there are existing incomplete Signature Requests generated by a prior Ack to the same prompt, or the Batch has a Remove Action pending, then the new execute fails. The failed execute is populated with an appropriate error message.
Automatic Cancellation	A Signature generated for confirmation of a Parameter Ack Value Action may be System Canceled if any other signature associated with the same Action is user canceled, or if the Batch System UnacknowledgedEvent targeted is terminated by the Ack.
Warm Reboot	Signature Requests generated in response to a Parameter Ack Execute Action are System Cancelled in the event of a FactoryTalk Batch Server warm boot.

See also

[Command verification policies](#) on [page 38](#)

Unit Bind Prompt Acknowledge Signature request

If a Bind Ack Command Verification policy is configured in the area model and it is associated with a Signature Template, then a **Unit Bind Prompt Acknowledge Signature** request is generated when binding a Unit to a step after a prompt has been generated or prompting bind's a Unit to a step.

The following is an explanation of the modifications made to support FactoryTalk Batch electronic signatures:

Unit Bind Execute	
Batch Server	When the FactoryTalk Batch Server receives a legal Ack execute in response to a Unit Bind Unacknowledged Event, it may queue the bind selection delivery pending a Signature Request if such behavior is configured in the area model.
Response	An ActionID is returned by the Execute used to locate the generated signature.
Completion	When all Signature Signoffs generated by an Ack received in response to a Unit Bind UnacknowledgedEvent are completed, then the Unit Requirement binds to the Ack execute response Unit specified if the bind action is still legal.
Cancellation	When any Signature generated by a Unit Bind Ack execute is cancelled, then the Unit Bind Ack Action is terminated and its associated incomplete Signature Requests are System Cancelled.
Overlap	If a Unit Bind Ack execute response to a Unit Bind Prompt is received while there are existing incomplete Signature Requests generated by a prior Unit Bind Ack Action targeting the same prompt, or the Batch has a Remove Action pending, then the new execute fails. The failed execute's response item is populated with an appropriate error message string.
Automatic Cancellation	If the batch system cancels an UnacknowledgedEvent targeted by a Unit Bind Ack Action, then the queued Execute Action is terminated and all incomplete Signature Requests associated with the Action are System Cancelled.
Warm Reboot	Signature Requests generated in response to a Unit Bind Ack Execute Action are System Cancelled in the event of a FactoryTalk Batch Server warm boot.

See also

[Command verification policies](#) on [page 38](#)

Phase Bind Prompt Acknowledge Signature request

If a Bind Ack Command Verification policy is configured in the area model and it is associated with a Signature Template, then a **Phase Bind Signature** request is generated when binding a phase to a material after a prompt is generated or prompting binds a phase to a material.

The following is an explanation of the modifications made to support FactoryTalk Batch electronic signatures:

Recipe Phase Bind Ack Execute	
Batch Server	When the FactoryTalk Batch Server receives a legal Ack execute in response to a Recipe Phase Bind UnacknowledgedEvent, it may queue the bind selection delivery pending a Signature Request if such behavior is configured in the area model.
Response	An ActionID is returned by the Execute used to locate the generated signature.
Completion	When all Signature Signoffs generated by an Ack received in response to a Recipe Phase Bind UnacknowledgedEvent are completed, then the Recipe Phase is bound using the Ack execute response data specified if the bind action is still legal.

Recipe Phase Bind Ack Execute	
Cancellation	When any Signature generated by a Recipe Phase Bind Ack Execute is cancelled, then the Recipe Phase Bind Ack Action is terminated and its associated incomplete Signature Requests are System Cancelled.
Overlap	If an Ack execute response to a recipe Phase Bind Prompt is received while there are existing incomplete Signature Requests generated by a prior Ack to the same prompt, or the Batch has a Remove Action pending, then the new execute fails. The failed execute's response item is populated with an appropriate error message string.
Automatic Cancellation	If the batch system cancels an UnacknowledgedEvent targeted by a Recipe Phase Bind Ack Action, then the queued Execute Action is terminated and all incomplete Signature Requests associated with the Action are System Cancelled.
Warm Reboot	Signature Requests generated in response to a Recipe Phase Bind Ack Execute Action are System Cancelled in the event of a FactoryTalk Batch Server warm boot.

See also

[Command verification policies](#) on [page 38](#)

Unit Bind Signature request

If a Bind Command Verification policy is configured in the area model and it is associated with a Signature Template, then a **Unit Bind Signature** request is generated when a user attempts to bind a Unit to a step.

The following is an explanation of the modifications made to support FactoryTalk Batch electronic signature:

Bind Execute	
Batch Server	When the FactoryTalk Batch Server receives a legal Bind execute, the Area Model configuration is checked to determine if a confirmation signature is required for the Bind request. If one is required, then a Signature Request is generated using the specified Signature Template and the Bind action is queued pending Signature completion.
Response	An ActionID is returned by the Execute used to locate the generated signature.
Completion	When all Signature Signoffs associated with a Bind Execute Action are completed, then the Unit Requirement targeted by the Bind Execute Action is bound to the specified Unit if the bind action is still legal.
Cancellation	When any Signature generated by a Bind Execute Action is cancelled, then the Bind Action is terminated and its associated incomplete Signature Requests are System Cancelled.
Overlap	If a Bind Execute is received while a previous Bind Execute Action is pending for the same target Unit Requirement, then the new Bind Execute fails. The failed execute's response item is populated with an appropriate response string. For purposes of determining the legality of overlapping executes, Command, Step, Remove, and BindPhase Executes are treated as being Bind Executes. This means, for example, that if a Step Execute Action is pending Signature completion on a procedure, then a Bind Execute to the same procedure fails because of the overlap with the Step execute.
Automatic Cancellation	If any step associated with the Unit Requirement being targeted by a queued Bind Execute Action becomes an active recipe element, then the Bind Execute Action is terminated and its associated incomplete Signature Requests are System Cancelled. This happens because the Bind action is not legal for a Unit Requirement associated with any active steps.

Bind Execute	
Warm Reboot	Signature Requests generated in response to a Bind Execute Action are restored in the event of a FactoryTalk Batch Server warm boot.

See also

[Command verification policies](#) on [page 38](#)

Remove Batch Signature request

If a Remove/Reset Command Verification policy is configured in the area model and it is associated with a Signature Template, then a **Remove Batch Signature** Request is generated when a user attempts to remove a batch from the Batch List or reset a phase run under Manual Phase Control.

The following is an explanation of the modifications made to support FactoryTalk Batch electronic signatures:

Remove Execute	
Batch Server	When the FactoryTalk Batch Server receives a legal Remove execute, the area model configuration is checked to determine if a confirmation Signature is required for the batch removal request. If one is required, then a Signature Request is generated using the specified Signature Template and the batch removal action is queued pending completion of the Signature Request. Manual Phase Control - When the server receives a legal Phase Reset Execute, it checks the area model configuration to determine if a confirmation signature is required. If so, then a Signature Request is generated using the specified Signature Template and the phase reset action is queued pending of the Signature Request completion.
Response	An ActionID is returned by the Execute used to locate the generated signature.
Completion	When all Signature Signoffs associated with a Remove Execute are completed, then the batch specified in the Remove execute is removed from the batch list if the Remove instruction is still legal. When all Signatures associated with a Phase Reset are complete, then the Manual Phase Control batch specified in the Phase Reset Execute is removed from the Batch List if the Phase Reset instruction is still legal.
Overlap	If a Remove Execute is received targeting a batch for which a queued Remove Execution Action already exists pending Signature completion(s), then the new Remove Execute fails. The failed execute's response item is populated with an appropriate response string. If a Phase Reset Execute is received targeting a batch for which a queued Phase Reset Execution Action already exists pending Signature completion(s), then the new Phase Reset Execute fails. For purposes of determining the legality of overlapping of executes, Remove Executes are treated as Phase Reset Executes. This means that if a Remove Execute action is pending Signature completion on a Manual Phase Control batch, then a Phase Reset issued via a Phase Execute to the same batch fails because of the Remove Execute overlap.
Cancellation	When an incomplete Signature associated with a Remove Execute Action is cancelled, then the queued Remove Execute Action is terminated and its associated incomplete Signature Requests are System Cancelled. When an incomplete Signature associated with a Phase Reset Execute Action is cancelled, then the queued Phase Reset Execute Action is terminated and its associated incomplete Signature Requests are System Cancelled.

Remove Execute	
Automatic Cancellation	A Remove Execute Action that is queued pending Signature Completion(s) is terminated if the target batch changes to an illegal state for Batch Removal, resulting in System Cancellation of its incomplete Signature Request(s). A Phase Reset Execute Action that is queued pending Signature Completion(s) is terminated if the target batch changes to an illegal state for Batch Removal, resulting in System Cancellation of its incomplete Signature Request(s).
Warm Reboot	Signature Requests generated in response to a Phase Reset or a Remove Execute Action are System Cancelled in the event of a FactoryTalk Batch Server warm boot.

See also

[Command verification policies](#) on [page 38](#)

Add Event Signature request

If a Comment Command Verification policy is configured in the area model and it is associated with a Signature Template, then an **Add Event Signature Request** is generated when a user attempts adding a comment to a batch. For information on the AddEvent execute, see the *FactoryTalk Batch Server API Communication Language Reference Guide*.

The following is an explanation of the modifications made to support FactoryTalk Batch electronic signatures:

AddEvent Execute	
Batch Server	When the FactoryTalk Batch Server receives a legal AddEvent, AddEvent_Phase, or AddEvent_CI execute, it checks the Area Model configuration to determine if a confirmation Signature is required for the AddEvent action. If one is required, then a Signature Request is generated using the specified Signature Template and the AddEvent action is queued pending completion of the Signature Request. An AddEvent is not generated when all batches are selected.
Response	An ActionID is returned by the execute used to locate the generated signature.
Completion	When all Signature Signoffs associated with an AddEvent execute are completed, then the AddEvent action is handled if it is still legal.
Cancellation	When any Signature generated by an AddEvent execute action is cancelled, then the AddEvent execute action is terminated and its associated incomplete Signature Requests are System Cancelled.
Overlap	AddEvent executes targeted at the same batch are permitted to overlap. If the batch has a Remove Action pending, then AddEvent execute fails.
Warm Reboot	Signature Requests generated in response to a AddEvent execute action are restored in the event of a FactoryTalk Batch Server warm boot.

See also

[Command verification policies](#) on [page 38](#)

Active Step Change Signature request

If a Step Command Verification policy is configured in the area model and associated with a signature template, then an **Active Step Change Signature request** is generated when performing an active step change.

The following is an explanation of the modifications made to support FactoryTalk Batch electronic signatures:

Step Execute	
Batch Server	When the FactoryTalk Batch Server receives a legal Step execute, the Area Model configuration is checked to determine if a confirmation signature is required for the Active Step Change action. If one is required, then a signature request is generated using the specified signature template and the Active Step Change action is queued pending signature request completion.
Response	An ActionID is returned by the execute used to locate the generated signature.
Completion	When all signatures associated with a Step execute action are completed, then the queued Step execute action is performed if it is still legal.
Cancellation	When any signature generated by a Step execute action is cancelled, then the Active Step Change action is terminated and its associated incomplete Signature Requests are System Cancelled.
Overlap	When a Step execute is received targeting a procedure that already has a pending Step execute action, then the new Step execute fails. The failed execute's response item is populated with an appropriate error message. For purposes of determining overlapping legality executes, Command and Remove execute actions are treated as being Step execute actions. This means, for example, that if a Step execute action is pending signature completion, then an AUTO command issued via a Command execute to the same procedure fails because of the overlap with the Step execute action.
Warm Reboot	Signature requests generated in response to a Step execute action are not restored in the event of a FactoryTalk Batch Server warm boot.

See also

[Command verification policies](#) on [page 38](#)

Parameter deviation verification policies

FactoryTalk Batch parameter deviations provide automatic detection of deviations for recipe and report parameters, automatic prompting for electronic signatures, and automatic recording of these events in the FactoryTalk Batch Event Journal.

If a phase class parameter or report parameter is configured with a Verification method and associated with a signature template, then a Signature Request is generated. If the signature is user generated, then a Signature dialog box opens to the user when the defined parameter limit values are exceeded. If the Signature Request is not user generated, then the Signature prompt begins to flash and the Signature Request displays in the Signature List. For information on enabling parameter limit tags, see the *FactoryTalk Batch Equipment Editor User Guide*.



Tip: The FactoryTalk Batch Server does not allow batch creation if the Verification configuration in the recipe is different from the area model Verification configuration.

See also

[Parameter deviation signature requests](#) on [page 47](#)

[Report parameter deviation signature requests](#) on [page 48](#)

Parameter deviation signature requests

The following is an explanation of modifications made to support parameter deviation signature requests:

Parameter Deviations	
Batch Server	<p>When an attempt is made to change a recipe parameter value, the FactoryTalk Batch Server validates the new value against the correct limits and then executes the correct verification policy. There are two ways a recipe parameter can be changed, SETPARAM and ACK executes.</p> <p>After checking for deviations, the server carries out the parameter's configured verification policy. A verification policy of "None" has no action, meaning that no electronic signatures or failures are generated even if a deviation exists. When the verification policy is "Signature Template", the server initiates an action as defined by the signature template. When the verification policy is "Not Allowed", the server fails the parameter change. The SETPARAM or ACK execute response will be: "FAIL: Value is outside the permitted limits."</p> <p>If a parameter value is changed on a recipe parameter, the limit values must remain within the high and low range defined in the area model. The server will not allow a batch to be created containing a recipe parameter that violates the limit values defined area model.</p> <p>When a phase is run in Manual Phase Control, the server reads the recipe and report parameter limits from the area model phase class configuration. If a phase is executing within a recipe, then the server reads the recipe and report parameter limits from the recipe step configuration.</p>
Response	An ActionID is returned by the execute used to locate the generated signature(s).
Completion	The server does not change the parameter value sent in the ACK or SETPARAM execute until all required signatures are complete. Completion of a signature generated for Parameter Deviation acknowledgment results in execution of a queued SetParm or ParamValueAck execute action if the signature was the last remaining incomplete signature associated with the action.
Cancellation	Cancellation of a signature generated for acknowledgment of a Parameter Deviation results in termination of its associated SetParm or ParamValueAck execute action and termination of any incomplete signature requests associated with the same action.
Overlap	<p>If a SetParm execute is received targeting a parameter that has incomplete signature requests for a prior SetParm execute action, then the new SetParm execute fails. A failed execute's response item contains an appropriate error message.</p> <p>If an Ack execute response to a recipe phase bind prompt is received while there are existing incomplete signature requests generated by a prior Ack to the same prompt, or the batch has a Remove action pending, then the new execute fails. The failed execute's response item is populated with an appropriate error message string.</p>
Automatic Cancellation	Parameter deviation signature requests are system cancelled if any other incomplete signatures associated with the same Action ID are cancelled. The server does not change the parameter value sent in the ACK or SETPARAM if any of the required signatures are cancelled.
Warm Reboot	A FactoryTalk Batch Server warm boot restores parameter deviation signature requests generated by a SetParm execute action, but not those generated by an Ack execute action.

See also

[Parameter deviation verification policies](#) on [page 46](#)

[Report parameter deviation signature requests](#) on [page 48](#)

Report parameter deviation signature requests

The following is an explanation of how report parameter deviation signature requests are handled by the FactoryTalk Batch Server:

Report Parameter Deviations	
Batch Server	<p>The FactoryTalk Batch Server validates all real and integer report parameters according to the report parameter's verification method as part of the upload process. In all cases, except one, the upload is completed and recorded in the event journal regardless of whether a deviation is detected. The exception occurs for a verification policy of "Not Allowed". In this case, the request fails and the value is not uploaded.</p> <p>When the verification policy is "None", no electronic signatures or failures are generated, even if a deviation exists.</p> <p>When the verification policy is "Signature Template", the server initiates any required signature prompts as defined by the signature template. The report upload is completed without failure – even if a signature is pending.</p> <p>If a report limit value is not known, the server processes all the other report parameters, recording deviations, and report values then fails the upload request.</p>
Response	An ActionID is returned by the execute used to locate the generated signature.
Completion	Completion of a signature generated for report deviation acknowledgment does not result in any additional action.
Cancellation	Report deviation signature requests cannot be cancelled.
Automatic Cancellation	There is no scenario under which a report deviation signature request is system cancelled.
Warm Reboot	A FactoryTalk Batch Server warm boot restores report parameter deviation signature requests.

See also

[FactoryTalk Batch electronic signatures](#) on [page 37](#)

[Parameter deviation verification policies](#) on [page 46](#)

[Parameter deviation signature requests](#) on [page 47](#)

General usage phase logic requests

A **General Usage Signature** request is created when a General Usage Phase Logic request is sent to the FactoryTalk Batch Server during the execution of an automated phase. Each signature template configured in FactoryTalk Batch Equipment Editor has a unique template index. To generate a signature the Phase Logic request specifies the signature template index.

For information on configuring general usage phase logic requests, see **Download Batch Parameters** in the *FactoryTalk Batch PCD Programmer Reference Guide*.

See also

[General usage signature requests](#) on [page 48](#)

General usage signature requests

The following is an explanation of how general usage signature requests are handled by the FactoryTalk Batch Server:

General Usage Signatures	
Batch Server	<p>General usage signatures are generated by Phase Logic requests received from Equipment Phases. The equipment phase specifies the Signature Template ID from which the Signature is generated, and the Phase Logic Request is considered complete when the Signature Request is complete.</p>

General Usage Signatures	
Response	If a Phase Logic Request for a General Usage Signature specifies an invalid Signature Template ID, then the Phase Logic Request is handled according to standard Phase Logic Request failure protocols. If the request is successful in generating the Signature, then no response is made until the Signature is either complete or cancelled.
Completion	When the Signature for a General Usage Signature generated by a Phase Logic Request is complete, then the Phase Logic Request is considered complete. The Phase Logic Request completion is handled via the standard Phase Logic Request communication protocol.
Cancellation	When the Signature for a General Usage Signature generated by a Phase Logic request is cancelled, the Phase Logic request is considered failed. The Phase Logic request failure is handled via the standard Phase Logic request communication protocol.
Overlap	The Phase Logic Request protocol does not support overlapped Phase Logic requests; hence, overlapped general usage signatures are not an issue. Note that parallel recipe phases can create parallel requests, but no parallelism is possible from within the scope of a single equipment phase.
Automatic Cancellation	If the Phase Logic request from which a general usage signature is generated is aborted, then the general usage signature is System Cancelled if it is in the incomplete state.
Warm Reboot	Signature requests generated in response to a General Usage Signature Phase Logic request are System Cancelled in the event of a FactoryTalk Batch Server warm boot.

See also

[General usage phase logic requests](#) on [page 48](#)

Electronic signature security

FactoryTalk Security is used to support FactoryTalk Batch Electronic Signature Signoffs. The **ESignature Network Directory** policy setting is created in the FactoryTalk Local Directory during FactoryTalk Batch installation with a default value of **True**. The value of the **ESignature Network Directory** policy setting determines whether the users and groups defined in the FactoryTalk Local Directory or the users and groups defined in the FactoryTalk Network Directory are used to assign Security Permissions to Electronic Signature Templates and validate Electronic Signature Request Signoffs.

Security Permissions for Signature Signoffs are specified when the Signature Template is created in FactoryTalk Batch Equipment Editor. The Security Permissions defined in the Signature Template are required to complete the Signoff(s) generated in the associated Signature Request. Each Signoff requires a UserID and a password. When a signature requires multiple signoffs, each signoff must be made using a unique UserID and password.

Signature templates are also used for Recipe Approval process signoffs, and must be configured before recipe approvals can be enabled, configured, and executed.

For more information on configuring signature templates, see the *FactoryTalk Batch Equipment Editor User Guide*.

See also

[FactoryTalk Batch electronic signatures](#) on [page 37](#)

Configure the ESignature network directory policy setting

[Configure the ESignature network directory policy setting](#) on [page 50](#)

The ESignature Network Directory policy setting must be set to True to enable signature signoff permission validations across a network.

IMPORTANT Changing the FactoryTalk Directory Server after creating Signature Templates causes Signoffs to fail unless the FactoryTalk System folder is backed up and restored on the new FactoryTalk Directory Server.

To configure the ESignature network directory policy setting:

1. Open the Local Directory in FactoryTalk Administration Console.
2. In the **Explorer** tree, navigate to **System > Policies > Product Policies > Batch > Equipment Editor > Configuration > Options**.
3. Right-click **Options**, and select **Properties**. The **Options Properties** dialog box displays the available policy settings.
4. Select the **ESignature Network Directory** policy setting and select **True** to use network security.
5. Select **OK**.
6. Restart the server and client applications.

See also

[FactoryTalk Batch electronic signatures](#) on [page 37](#)

[Electronic signature security](#) on [page 49](#)

[FactoryTalk Batch default policy settings](#) on [page 31](#)

FactoryTalk Batch audit trail

FactoryTalk Batch auditing provides a detailed audit trail of user-initiated changes from FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor.

FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor track all changes in the current editing session for each user-initiated action. When changes are saved, each entry in the list generates an audit message. If changes are made, but are not saved, no audit message is generated.

Audit messages are forwarded from the FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor to FactoryTalk Diagnostics, a Rockwell Software product that enables application logging and auditing, or to FactoryTalk AssetCentre (if installed).



Tip: Batch audit messages are generated in two formats: Engineer Audit and Secure Audit. The audit messages formatted for the local log are categorized as Engineer, and audit messages formatted for FactoryTalk AssetCentre are categorized as Secure. If FactoryTalk AssetCentre is installed, a Secured logging destination is enabled and messages can be routed to both the local and FactoryTalk AssetCentre log.

The **FactoryTalk Directory System** folder contains Audit settings that apply to all FactoryTalk-enabled products in a system. The FactoryTalk Audit Policy Setting, **Audit changes to configuration and control system**, must be enabled to display audit messages in either the FactoryTalk Diagnostics Viewer or the FactoryTalk AssetCentre client. This FactoryTalk policy setting overrides the registry settings used to enable and disable auditing in FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor. (For more information see FactoryTalk Help, located in the FactoryTalk Administration Console.)

See also

[Enable auditing for FactoryTalk Batch Equipment Editor or FactoryTalk Batch Recipe Editor](#) on [page 52](#)

[Audited FactoryTalk Batch Equipment Editor events](#) on [page 52](#)

[Audited FactoryTalk Batch Recipe Editor events](#) on [page 55](#)

[Audit messages](#) on [page 56](#)

Enable auditing for FactoryTalk Batch Equipment Editor or FactoryTalk Batch Recipe Editor

Use the FactoryTalk Administration Console to modify the policy settings for auditing. These policy settings are located in both the FactoryTalk Network Directory and Local Directory. The Local Directory audit policy overrides the Network Directory audit policies.



Tip: The FactoryTalk Audit Policy Setting, **Audit changes to configuration and control system**, must be enabled to display audit messages in FactoryTalk Diagnostics or FactoryTalk AssetCentre.

To enable auditing for FactoryTalk Batch Equipment Editor or FactoryTalk Batch Recipe Editor:

1. Open the FactoryTalk Administration Console and log on to the Network Directory or Local Directory.
2. Navigate to **System > Policies > System Policies**.
3. Right-click **Audit Policy**, and select **Properties**.
4. Select **Audit changes to configuration and control system** and set it to **Enabled**. Auditing is enabled the next time FactoryTalk Batch Equipment Editor or FactoryTalk Batch Recipe Editor runs.



Tip: Restart all FactoryTalk Batch components to update changes made in the FactoryTalk Diagnostics.

See also

[FactoryTalk Batch audit trail](#) on [page 51](#)

[Audited FactoryTalk Batch Equipment Editor events](#) on [page 52](#)

[Audited FactoryTalk Batch Recipe Editor events](#) on [page 55](#)

[Audit messages](#) on [page 56](#)

Audited FactoryTalk Batch Equipment Editor events

This section describes the set of audit messages generated by FactoryTalk Batch Equipment Editor. Not all user actions are audited; for example, **View** menu changes and **Options** menu changes do not generate audit messages. The following events are audited when the area model is saved.

Message Type	Event
Area Model	Invoke FactoryTalk Batch Equipment Editor Open an existing area model Create a new area model Save an area model Edit/update an area model property Delete an area model Add a signature template Delete a signature template Update a signature template Enable/disable recipe approvals Enable/disable recipe versioning Update a command verification policy

Message Type	Event
Area Model Security	Security authority is configured for an area model Security authority identifier (SAI) is set or removed Security authority configuration change is accepted by user Security authority changes are saved to the area model
Data Server	Add an OPC server Delete an OPC server Edit/update an OPC server property Add a Logix5000 CIP server Delete a Logix5000 CIP server Edit a Logix5000 CIP server Add an FTD server Delete an FTD server Edit/update an FTD server property
Enumeration Set	Create an enumeration set Delete an enumeration set Edit/update an enumeration set property
Enumeration	Add an enumeration Delete an enumeration Edit/update an enumeration property
Resource Class	Add a resource class Delete a resource class Edit/update a resource class property
Resource	Add a resource Delete a resource Edit/update a resource property
Process Cell Class	Add a process cell class Copy a process cell class Delete a process cell class Edit/update a process cell class property
Process Cell	Add a process cell Copy a process cell Delete a process cell Edit/update a process cell property
Unit Attribute	Add a unit attribute Delete a unit attribute Edit/update a unit attribute property
Unit	Add a unit Copy a unit Edit a unit Delete a unit Edit/update a unit property
Global Unit Binding Requirements	Add global unit attribute Edit global unit attribute Delete global unit attribute
Unit Flow Path	Link units with a flow path Unlink units with a flow path

Message Type	Event
Equipment Phase Class (Recipe Phase)	Add an equipment phase class Copy an equipment phase class Delete an equipment phase class Edit/update an equipment phase class property Edit/update an equipment phase class control strategy set of parameters Edit/update an equipment phase class optional material parameters property Edit/update an equipment phase class control strategy set of reports Edit/update an equipment phase class control strategy
Equipment Phase	Add an equipment phase Copy an equipment phase Delete an equipment phase Edit/update an equipment phase property Edit/update an equipment phase's server property Reset an equipment phase's tags to default values Share an equipment phase between units
Recipe Parameter	Add a recipe parameter Delete a recipe parameter Edit/update a recipe parameter property
Recipe Approval Step	Add a recipe approval step Delete a recipe approval step Edit/update a recipe approval step
Report Parameter	Add a report parameter Delete a recipe parameter Edit/update a recipe parameter property
Phase Message	Add a phase message Delete a phase message Edit/update a phase message property
Synchronize	Synchronize a phase
Tag Class	Add a tag class Delete a tag class Edit/update a tag class property
Unit Attribute Tag	Add a unit attribute tag Delete a unit attribute tag Edit/update a unit attribute tag property
Container Association	Add a container association Delete a container association Edit/update a container association property
Instruction Configuration	Edit/update an instruction configuration property
Phase Tag	Edit/update a phase tag property

See also

[Enable auditing for FactoryTalk Batch Equipment Editor or FactoryTalk Batch Recipe Editor on page 52](#)

[Audit messages on page 56](#)

Audited FactoryTalk Batch Recipe Editor events

This section describes the set of audit messages generated by FactoryTalk Batch Recipe Editor in response to recipe modifications and operations.

Message Type	Event
Recipe Object	<ul style="list-style-type: none"> Open an existing recipe Create a new recipe Save a recipe, including master recipes and recipe formulations Save all recipes, including master recipes and recipe formulations Remove a recipe (User-supplied comment when deleting) Import a recipe (User-supplied comment when importing) Export a recipe (User-supplied comment when exporting) Rebuild the recipe directory (User-supplied comment when rebuilding recipe) Delete an existing recipe, including master recipes and recipe formulations Update an existing recipe, including master recipes and recipe formulations
Recipe Print	Print a procedure, unit procedure, or operation
Recipe Header	Edit/update recipe header properties, except the recipe name
Recipe Step	<ul style="list-style-type: none"> Add a recipe step, insert step, or parallel recipe step Edit/update a recipe step Delete a recipe step
Recipe Approval Step	<ul style="list-style-type: none"> Signoff on an approval step Revert an approval step
Recipe Versioning	<ul style="list-style-type: none"> Check in a recipe (create a new version of a recipe) Check out a recipe (create a work-in-progress (WIP) recipe) Obsolete a recipe Save an obsolete recipe Reinstate an obsolete recipe Save a reinstated recipe Version conflict on recipe import Version conflict on recipe directory rebuild, including auto-rebuild of the directory when FactoryTalk Batch Recipe Editor opens Delete a recipe due to version conflict resolution during import or recipe directory rebuild Remove signoff(s) due to verification failure for versioned and non-versioned recipes
Recipe Security	<ul style="list-style-type: none"> Change a recipe security state Edit/modify a security authority identifier (SAI) Change a FactoryTalk Services Platform SAI host computer
Recipe Text Box	<ul style="list-style-type: none"> Add a text box Delete a text box Edit a text box (only if the comment changes) Associate a text box Disassociate a text box
Recipe Formula Parameter	<ul style="list-style-type: none"> Add a recipe formula parameter to a step Delete a recipe formula parameter from a step Edit/update a recipe formula parameter of a step
Recipe Formulation	<ul style="list-style-type: none"> Add a recipe formulation Delete a recipe formulation Rename a recipe formulation Change the recipe formulation description

Message Type	Event
Recipe Formulation Parameter	Edit/update a recipe formulation parameter
Formula Value	Edit/update a formula value's non-origin properties Edit/update a formula value's origin property Edit/update a formula value's value property for deferred parameters
Report Limit	Add a report limit Delete a report limit Edit/update a report limit
Bind Requirement	Add/update a bind requirement Delete a bind requirement
Bind Preference	Add/update bind preference Delete a bind preference
Recipe Transition	Add a transition to a recipe Edit/update a recipe transition Delete a transition to a recipe
Link Group	Add a recipe link group Delete a recipe link group Edit/update a recipe link group
Equipment Requirement	Add an equipment requirement to a recipe Delete an equipment requirement of a recipe Edit/update procedure-only equipment requirement properties, except downstream units Edit/update procedure-only equipment requirement's downstream units property Edit/update operation and unit procedure equipment requirements
Recipe Verification	Verify a single recipe Verify all recipes
Material Loop	Add a material loop to a recipe Undo a material loop
Recipe Link	Add a recipe link Delete a recipe link

See also

[Enable auditing for FactoryTalk Batch Equipment Editor or FactoryTalk Batch Recipe Editor on page 52](#)

[Audit messages on page 56](#)

Audit messages

When auditing is enabled for FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor, view the audit messages through the FactoryTalk Diagnostics Viewer or the FactoryTalk AssetCentre client.



Tip: Batch audit messages are generated in two formats: Engineer Audit and Secure Audit. The audit messages formatted for the local log are categorized as Engineer, and audit messages formatted for FactoryTalk AssetCentre are categorized as Secure. If FactoryTalk AssetCentre is installed, a Secured logging destination is enabled and messages can be routed to both the local and FactoryTalk AssetCentre log.

See also

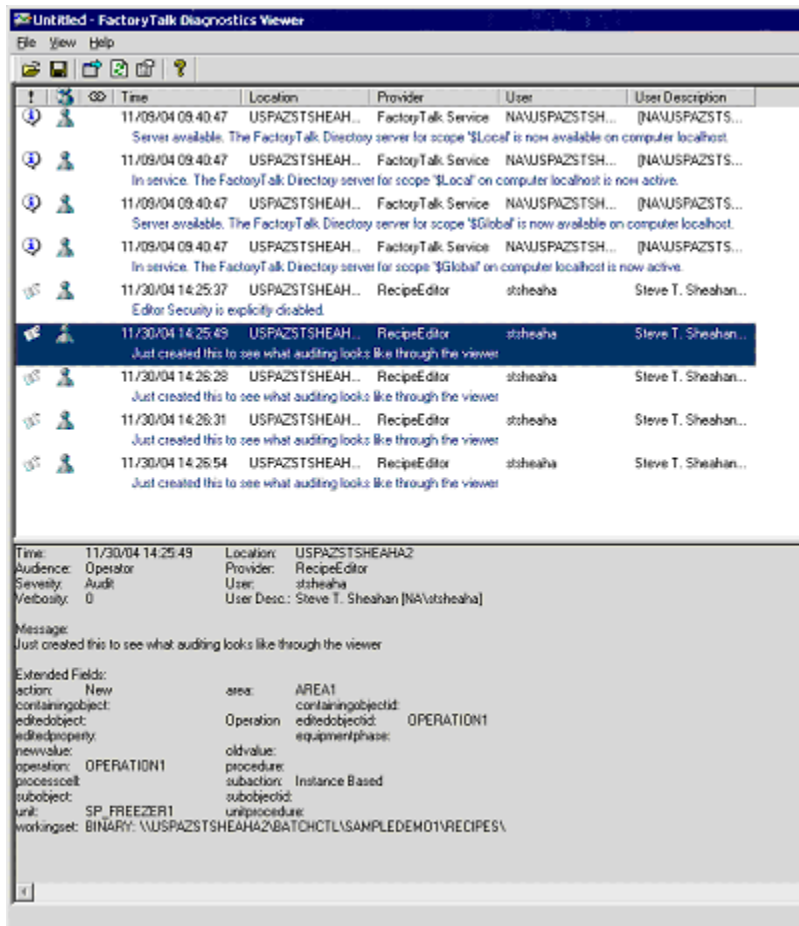
[FactoryTalk Diagnostics Viewer](#) on [page 57](#)

[FactoryTalk AssetCentre Client](#) on [page 60](#)

FactoryTalk Diagnostics Viewer

For information about using or customizing the FactoryTalk Diagnostics Viewer, refer to FactoryTalk Diagnostics Viewer online help.

The illustration shows a FactoryTalk Batch audit record in the FactoryTalk Diagnostics Viewer. The audit record was generated by the creation of a new operation.



See also

[FactoryTalk Batch audit trail](#) on [page 51](#)

[Standard viewer message fields in FactoryTalk Diagnostics Viewer](#) on [page 58](#)

[Extended viewer message fields in FactoryTalk Diagnostics Viewer](#) on [page 58](#)

Standard viewer message fields in FactoryTalk Diagnostics Viewer

The following is a detailed list of the default FactoryTalk Diagnostic Viewer standard message parameters.

Field	Data Type	Description	Used In
Message	Variant	Application and event-specific message associated with the event being reported.	If provided by the user, applies to all FactoryTalk Batch Recipe Editor and FactoryTalk Batch Equipment Editor audit messages.
Severity	Enum	This field specifies the severity of the audit message. This value, along with the Audience parameter, defines where this message is routed.	This is RNADiagSeverity_Info. This is not configurable.
Audience	Enum	This field specifies the audience for the audit message. This value, in conjunction with the Severity parameter, defines where this message is routed.	This is RNADiagAudience_Op. This is not configurable.
Time	Date	Date and time, specified as (GMT/UTC)	Represented in the TimeStamp field of the audit message.
Location	BSTR	Name of the host computer.	Represented in the MachineName field in the audit message.
Provider	BSTR	Name of the client application or component.	Not used.
User	BSTR	Name of the person logged into the editor.	Represented in the UserName field of the audit message.
User Description	BSTR	Fully resolved user name.	Represented in the UserDescription field of the audit message.
Args	Variant	Used for Visual Basic Compatibility.	Not used.

See also

[FactoryTalk Diagnostics Viewer](#) on [page 57](#)

[Extended viewer message fields in FactoryTalk Diagnostics Viewer](#) on [page 58](#)

Extended viewer message fields in FactoryTalk Diagnostics Viewer

Extended fields are provided in the FactoryTalk Diagnostics Viewer.

Field	Data Type	Description	Used In
Verbosity	Long	This value field is arbitrary; values from 1 through 100 are valid. Use to further define or filter messages.	Not used.
Action	BSTR	The updating/editing action made by the user.	All audit messages.
Area	BSTR	For Equipment objects, the area name the object belongs to. This field is used when needed.	Some Equipment audit messages.
ContainingObject	BSTR	The object type that contains the EditedObject. This is used to help define exactly what object is being edited/updated.	Some Equipment audit messages.
ContainingObjectID	BSTR	The sub-property or object property field of the equipment or recipe object affected by the action.	Some Equipment and Recipe audit messages.

Field	Data Type	Description	Used In
EditedObject	BSTR	The Equipment or Recipe object affected by the action.	All audit messages.
EditedObjectID	BSTR	The most common way to identify the EditedObject.	All audit messages.
EditedProperty	BSTR	The property affected by the action.	Some Recipe, Equipment and Area audit messages.
EditedProperty1	BSTR		
EditedProperty2	BSTR	Preference priority	
EquipmentPhase	BSTR	For Equipment objects, the equipment phase name the object belongs to. This field is used when needed.	Some Equipment audit messages.
NewValue	BSTR	The value of a property after it was changed.	Updating audit events only.
NewValue1	BSTR	The value of a property after it was changed: <ul style="list-style-type: none"> • Phase Name • Attribute Name • Expression text 	
NewValue2	BSTR	Priority number	
OldValue	BSTR	The value of a property before it was changed.	Updating audit events only.
OldValue1	BSTR	The value of a property before it was changed: <ul style="list-style-type: none"> • Phase Name • Attribute Name • Expression text 	Updating audit events only.
OldValue2	BSTR	Priority number	Updating audit events only.
Operation	BSTR	For Recipe objects, the operation name the object belongs to. This field is used when needed.	Some Recipe audit messages.
Procedure	BSTR	For Recipe objects, the procedure name the object belongs to. This field is used when needed.	Some Recipe audit messages.
ProcessCell	BSTR	For Equipment objects, the process cell name the object belongs to. This field is used when needed.	Some Equipment audit messages.
SubAction	BSTR	An updating/editing action to a list that is a property of the EditedObject. This can be Add or Delete.	Updating audit events for properties that are lists only.
SubObject	BSTR	The object type referenced by the EditedObject.	Updating audit events for properties that are lists only.
SubObjectID	BSTR	The value of a property after it was changed.	Updating audit events for properties that are lists only.
Unit	BSTR	For Equipment objects, the unit name the object belongs to. This field is used when needed.	Some Equipment audit messages.
UnitProcedure	BSTR	For Recipe objects, the unit procedure name the object belongs to. This field is used when needed.	Some Recipe audit messages.
Working Set	BSTR	For Recipe objects, the computer name and database where the recipes are stored.	Some Recipe audit messages.

See also

[FactoryTalk Diagnostics Viewer](#) on [page 57](#)

[Standard viewer message fields in FactoryTalk Diagnostics Viewer](#) on [page 58](#)

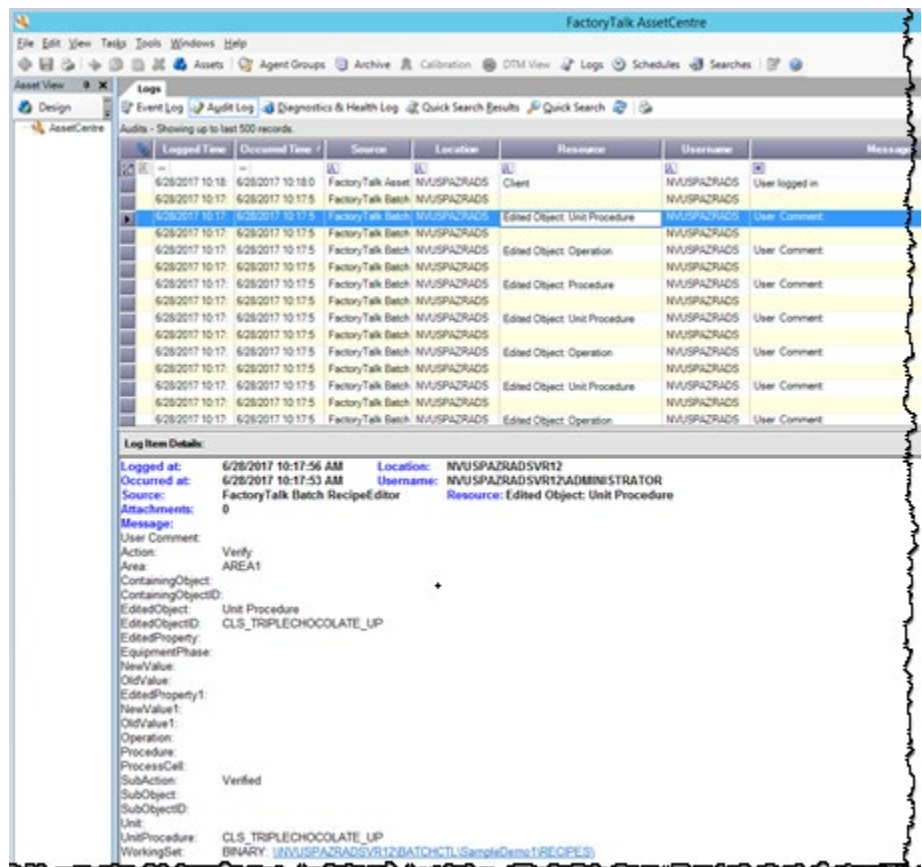
FactoryTalk AssetCentre Client

When FactoryTalk AssetCentre is installed, and auditing is enabled for FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor, view the audit messages through the FactoryTalk AssetCentre Client.



Tip: To use FactoryTalk AssetCentre for auditing, Change Management (FactoryTalk AssetCentre add-on solution) must be installed.

The graphic illustrates an audit log as viewed with the FactoryTalk AssetCentre Client. The FactoryTalk Batch audit messages are written to the **Message Summary** field. Only the first 100 characters of the **Message Summary** field are displayed on a line. The entire message is displayed in the Details window. The **EditedObject** audit message field is used to populate the **Resource** field in the Viewer.



See also

[FactoryTalk Batch audit trail](#) on [page 51](#)

[Viewer message fields](#) on [page 61](#)

Viewer message fields

The audit messages for FactoryTalk AssetCentre are formatted as closely as possible as those used in FactoryTalk Diagnostics Viewer messages. The **Resource** field for audit messages contains the value of the **Edited Object** field of a diagnostics message.

The **Message** field contains these fields:

- User comment
- Action
- Area
- ContainingObject
- ContainingObjectID
- EditedObject
- EditedObjectID
- EditedProperty
- EquipmentPhase
- NewValue
- OldValue
- Operation
- Procedure
- ProcessCell
- SubAction
- SubObject
- SubObjectID
- Unit
- UnitProcedure
- WorkingSet

Each entry consists of a Label and its associated data. Each entry also contains tab characters to line up the data for readability, and is followed by a new line character.

See also

[FactoryTalk AssetCentre Client](#) on [page 60](#)

[Enable auditing for FactoryTalk Batch Equipment Editor or FactoryTalk Batch Recipe Editor](#) on [page 52](#)

The FactoryTalk Batch Service Manager

The FactoryTalk Batch Service Manager is used with the FactoryTalk Batch Server, FactoryTalk Batch View Server, and FactoryTalk Event Archiver, when FactoryTalk Event Archiver is configured as a Windows service. The Service Manager manually starts and stops the FactoryTalk Batch Server and manually starts, pauses, continues or stops FactoryTalk Event Archiver services. The Service Manager accesses the **Batch Server Statistics** dialog box, which gives access to COM server information, COM client status, error information, and other FactoryTalk Batch Server information.



Tip: To command the FactoryTalk Batch Server, FactoryTalk Batch View Server, or FactoryTalk Event Archiver services, you must have local administrator privileges on the computer where the server and/or FactoryTalk Event Archiver services are installed. If you do not have local administrator privileges, you will have view-only privileges.

See also

[Windows services](#) on [page 63](#)

[FactoryTalk Batch Service Manager dialog box](#) on [page 64](#)

[FactoryTalk Batch Server Details dialog box](#) on [page 66](#)

[Start the FactoryTalk Batch Server service](#) on [page 68](#)

[Add a custom service](#) on [page 70](#)

Windows services

The FactoryTalk Batch Server, FactoryTalk Batch View Server, and FactoryTalk Event Archiver operate as Windows services. During FactoryTalk Batch installation, the user account specified during the installation is assigned to the **Log On As: This Account:** option, found in the (Administrative Tools) **Services** dialog box.

Running as a Windows service allows the FactoryTalk Batch Server to run in the absence of an interactive Windows logon. Logging on or logging off Windows during operation does not disrupt the server execution.

Configure the FactoryTalk Batch Server to start automatically, giving server control to Windows Services, or control the FactoryTalk Batch Server manually using the FactoryTalk Batch Service Manager. The Service Manager also controls the type of boot method that the server uses, and runs the server in demo mode.



Tip: Specifying the boot method or demo mode in the FactoryTalk Batch Service Manager does not alter the Batchsvr.ini file and has no effect on the server when automatically starting.

See also

[The FactoryTalk Batch Service Manager on page 63](#)

FactoryTalk Batch Service Manager dialog box

The **FactoryTalk Batch Service Manager** dialog box is organized into three areas:

- Connection
- Service state
- Server

The **Connection** area contains:

Item	Definition
Computer	Displays the selected computer.
Select Computer	Selects the computer where the service is located.
Service	Allows you to select a service from a list of batch services for the selected computer.

The **Service state** area contains two sections, one that contains buttons, the other that has a traffic light graphic.

The buttons allow you to select a service state:

Item	Definition
Stop	Stops the selected service.
Pause	Pauses the selected service. This button is not enabled for the batch server.
Start/ Continue	Starts the selected service or continues a service that has been paused.

The traffic light graphic displays the current service status with an accompanying color:

Service Status	Color
STARTING	None
RUNNING	Green
STOPPED	Red
PAUSED	Yellow
NOT CONNECTED	None
START PENDING	None



Tip: The state of the FactoryTalk Batch Service Manager (information from the **Connection** and **Server** groups) is stored from the last time the Service Manager was run. These values are written when the FactoryTalk Batch Server is closed.

The **Server** area options include:

Item	Definition
Allow Demo Mode	The FactoryTalk Batch Server runs with full functionality without a license for a two-hour period. After two hours, the server stops running.
Allow Grace Period	The FactoryTalk Batch Server starts when a valid activation is not found. The grace period is seven days, after which a valid FactoryTalk Activation license is required.

Item	Definition
Cold Boot	The FactoryTalk Batch Server starts in a completely initialized condition. All restart data is erased and all recipe content is removed from the Batch List.
Warm Boot	The FactoryTalk Batch Server starts and attempts to restore the set of batches that were on the batch list when the server previously terminated. The restart files are altered so only one warm restart may be attempted.
Warm All Boot	Restarts the FactoryTalk Batch Server only if it is able to restore all of the batches to the batch list. Restart does not alter the restart files so another warm-all or warm restart may be attempted. This is the default setting for automatic restarts.
Verify/Deploy Area Model	Opens the Verify/Deploy Area Model dialog box.
Server Details	Opens the Batch Server Details dialog box.

See also

[FactoryTalk Batch Server Details dialog box](#) on [page 66](#)

Verify/Deploy Area Model dialog box

The **Verify/Deploy Area Model** dialog box displays the name and file date of the current area model, a list of updated area models, the status of any comparison or load operation, and the ability to compare or deploy updates to a runtime area models.

The **Verify/Deploy Area Model** dialog box contains the following:

Field	Definition
Server Name	The name of the server.
Running Area Model	The runtime area model filename.
Running Area Model Date	The file date of the area model file at the time of FactoryTalk Batch Server startup or last file load.
Updated Area Model	A list of updated area model files by filename.
Updated Area Model Date	The date of the selected area model in the Updated Area Model list.
Verify Area Models	Compares the running area model with the selected updated area model. In the Status area, displays the comparison results, a runtime conditions check, and all changes that are permitted or not permitted to load. Also displays the log file name and location for viewing further comparison and verification details.
Deploy Updated Area Model	If changes are permitted, deploys the selected updated area model. In the Status area, displays the name of the original and the successfully deployed area model, comparison results, a runtime conditions check, and a load report. Also displays the log file name and location for viewing further deployment details. Note: If the operator attempting to deploy an updated area model does not have the required FactoryTalk permissions, the button is disabled.
View Log	Opens the compare.update log (.log) text file. Note: A log file can be viewed only if Microsoft Excel or Notepad is installed on the system. The log file name is based on the operation, updated area model file name, date, and time stamp. To avoid duplicates, the time stamp is written in hours, minutes, seconds, and micro seconds (HHMMSSmmmm). For example: COMPARE_ICE_CREAM1_UPDATED.CFG.2017-09-25_T1135460629.LOG

See also

[The FactoryTalk Batch Service Manager on page 63](#)

[FactoryTalk Batch Server Details dialog box on page 66](#)

[Verify and deploy an updated area model using the FactoryTalk Batch Service Manager on page 97](#)

[Update an online area model on page 96](#)

FactoryTalk Batch Server Details dialog box

The **FactoryTalk Batch Server Details** dialog box displays an overview of the current system’s status and the current running area model. The area model file referenced is specified in the **Server Options** dialog box accessed from FactoryTalk Batch Equipment Editor. The **FactoryTalk Batch Server Details** dialog box is organized into five tabs.

See also

[FactoryTalk Batch Server Details dialog box - General tab on page 66](#)

[FactoryTalk Batch Server Details dialog box - Messages tab on page 67](#)

[FactoryTalk Batch Server Details dialog box - PCD Communications tab on page 67](#)

[FactoryTalk Batch Server Details dialog box - View Communications tab on page 68](#)

FactoryTalk Batch Server Details dialog box - General tab

The **General** tab lists data on the number of batches and prompts being currently handled by the FactoryTalk Batch Server, as well as the Windows time on the server, the time the batch server started, and the name and creation date of the current area model.

The **General** tab shows:

Field	Definition
Batch Count	The number of batches loaded in the FactoryTalk Batch Server.
Prompts Count	The number of prompts currently existing within all batches.
Version	The version number of the FactoryTalk Batch Server.
Time	The time as specified by Windows on the FactoryTalk Batch Server.
Started at	The time the FactoryTalk Batch Server started.
Area Model File Name	The name of the currently active area model.
Area Model Date	The file date of the currently active area model at the time of FactoryTalk Batch Server startup or the last update.

See also

[FactoryTalk Batch Service Manager dialog box on page 64](#)

[FactoryTalk Batch Server Details dialog box on page 66](#)

FactoryTalk Batch Server Details dialog box - Messages tab

The **Messages** tab is used to monitor FactoryTalk Batch Server log information. The messages displayed in the three sections are the most recent messages written to the server log by the server.

The **Messages** tab shows:

Field	Definition
Info	The most recent Info -type entry in the log file. Info entries are log entries recorded as a part of the normal execution.
Warning	The most recent Warning -type entry in the log file. Warning entries indicate that a minor irregularity the system detected. The irregularity was handled by the system or the system is not sure of the occurrence's implications.
Severe	The most recent Severe -type entry in the log file. Severe entries indicate that the system encountered a severe error condition. The system handled the situation as best it could.

See also

[FactoryTalk Batch Service Manager dialog box](#) on [page 64](#)

[FactoryTalk Batch Server Details dialog box](#) on [page 66](#)

FactoryTalk Batch Server Details dialog box - PCD Communications tab

The **PCD Communications** tab is used to monitor the FactoryTalk Batch Server communication status and any data servers that are communicating with the server. Verification of all tags, phases, and equipment operation sequences associated with the current area model is also performed.

The **PCD Communications** tab displays these components:

Field	Definition
Data Server Status	Lists of all configured data servers in the area model and the communication status between FactoryTalk Batch and these data server(s).
Tag Verify Status	Indicates if the verification function is READY, IN PROGRESS, or COMPLETED.
Tag Verify Verified	Indicates the number of tags, phases, and equipment operation sequences that the server processed during tag verification.
Tag Verify Bad	Indicates the number of tags, phases, and equipment operation sequences that the server was unable to verify. Bad tags, phases, and equipment operation sequences indicate either the data server cannot supply the data requested by the FactoryTalk Batch Server or the data was not of the expected type.
Tag Verify Total	Indicates the total number of tags, phases, and equipment operation sequences.
Start	Starts the verification process.
Stop	Stops the verification process.

See also

[FactoryTalk Batch Service Manager dialog box](#) on [page 64](#)

[FactoryTalk Batch Server Details dialog box](#) on [page 66](#)

FactoryTalk Batch Server Details dialog box - View Communications tab

The **View Communications** tab is used to monitor the number of Dynamic Data Exchange (DDE), Component Object Model (COM) and Open Platform Communications (OPC) conversations taking place between the FactoryTalk Batch Server and Client software.

The **View Communications** tab shows:

Item	Definition
DDE Connections	The number of DDE connections attached to the FactoryTalk Batch Server.
COM Connections	The number of COM connections attached to the server.
OPC Connections	The number of OPC connections attached to the server.
Items	The number of connection items the server currently supports.

See also

[FactoryTalk Batch Service Manager dialog box](#) on [page 64](#)

[FactoryTalk Batch Server Details dialog box](#) on [page 66](#)

Start the FactoryTalk Batch Server service

The FactoryTalk Batch Service Manager communicates with the Windows Services of the selected computer to determine available services. There may be a noticeable delay as communications are established, marked by the cursor displayed as an hourglass. If the FactoryTalk Batch Service Manager cannot communicate with the selected computer's Windows Services, a message box is displayed stating that the FactoryTalk Batch Service Manager failed to connect to the specified computer.

The FactoryTalk Batch Phase Simulator must be running to test FactoryTalk Batch in a demo environment. For more information, see **The FactoryTalk Batch Phase Simulator** in the *FactoryTalk Batch Administrator Guide*.



IMPORTANT The FactoryTalk Batch Server cannot start if **Security Authority** is enabled and there is a mismatch between the Security Authority Identifiers (SAIs) in the area model and the FactoryTalk Network Directory. For information on your recovery options, see **Troubleshooting** in the *FactoryTalk Batch Equipment Editor User Guide*.

To start the FactoryTalk Batch Server service:

1. Select **Start > Rockwell Software > Batch Service Manager**. The **Batch Service Manager** opens.
2. Select **Select Computer**. The **Select Computer** dialog box opens.

IMPORTANT Administrator privileges are required to perform this procedure.

3. In the **Enter the object name to select** area, enter the name of the computer where the FactoryTalk Batch Server is installed (or select the **Advanced** button to search for a computer).
4. Select **OK** to close the **Select Computer** dialog box.
5. From the **Service** list, select **FactoryTalk Batch Server**.

6. (optional) Select **Allow Demo Mode** if you want to run FactoryTalk Batch in demo mode. If you run FactoryTalk Batch in the demo mode, be aware that the Server stops after 2 hours of operation.
7. Select the method to use for restarting the server after a service halt.
 - **Cold Boot:** Restarts the server in a cold state. All journal data or recipe content is erased upon startup. If Security Authority is enabled, SAIs in a secured area model and any secured recipes must match the current Network FactoryTalk Directory SAI.
 - **Warm Boot:** Restarts the server, restoring the set of batches that were on the batch list when the server previously terminated. No validation for Security Authority is performed.
 - **Warm All Boot:** Restarts the server only if it is able to restore all of the batches to the batch list. No validation for Security Authority is performed.
8. Select the **Start/Continue** button to start the service. Wait for the status to change to RUNNING and the light is green.
9. Select the **Server Details** button. The **FactoryTalk Batch Server Details** dialog box opens.
10. Select the **PCD Communications** tab. The **Data Server Status** box displays the status of the conversation. Make sure that conversation is GOOD.
11. Select **Start**. The tag verification process begins. When tag verification is COMPLETED, select **OK**, or select **Stop** to end the verification process and then select **OK**.
 -  Tip: If the number of COM servers exceeds the viewable area within the **Data Server Status** box, a scroll bar displays, enabling you to scroll through the list.
12. Select **Close**. The **Batch Service Manager** dialog box closes.
 -  Tip: If software component conversation becomes *LOST* while running in Demo mode, make sure that the simulator is running and try starting the server(s) again.

Grace periods in FactoryTalk Batch Server

When the FactoryTalk Batch Server starts, it performs a license check. If a valid activation is found, a successful license check is returned to the server. As a result of the license check, the server determines which applications have a license.

If its normal license check process fails, the server requests grace period activation from FactoryTalk Activation. The manner in which the grace period activation is requested depends on the area model and why the license check process failed. The server will log the request for, and receipt of, the grace period license.

When grace period activation is requested for multiple activations by one process, the grace period is ended and the timer reset when any of the multiple activations can be successfully obtained.

Every four hours the FactoryTalk Activation software attempts to check out each activation requested as a grace period activation by the FactoryTalk

Batch Server. If any one of these activations is successfully checked out, the grace period for the server ends.



Tip: If the FactoryTalk Batch Server is started in the Demo mode, an exception is made to the license and unit checks. Even if no licenses are found, the server will run for two hours and then shut down.

See also

[Start the FactoryTalk Batch Server service](#) on [page 68](#)

[Enable a grace period](#) on [page 70](#)

Enable a grace period

Use the **Allow Grace Period** option on the **Batch Service Manager** dialog box to start the FactoryTalk Batch Server in the event that it cannot find a valid activation license. The grace period is seven days.

To enable a grace period:

1. Select **Start > Rockwell Software > Batch Service Manager**. The **Batch Service Manager** opens.
2. Select **Select Computer**. The **Select Computer** dialog box opens.
3. In the **Enter the object name to select** area, enter the computer name where the batch server is installed (or click the **Advanced** button to search for a computer).
4. Select **OK** to close the **Select Computer** dialog box.
5. From the **Service** list, select **FactoryTalk Batch Server**.
6. Select **Allow Grace Period** enable grace period activation.
7. Select **Start/Continue** to start the service. Wait for the status to change to RUNNING.

The FactoryTalk Batch Server searches for valid activations first. If the appropriate activation is not found, the server will start under a grace period activation. The FactoryTalk Batch Server checks for valid activation licenses every four hours until it either finds one or the grace period expires.

8. Select **Close**. The **Batch Service Manager** dialog box closes.

See also

[Grace periods in FactoryTalk Batch Server](#) on [page 69](#)

[Start the FactoryTalk Batch Server service](#) on [page 68](#)

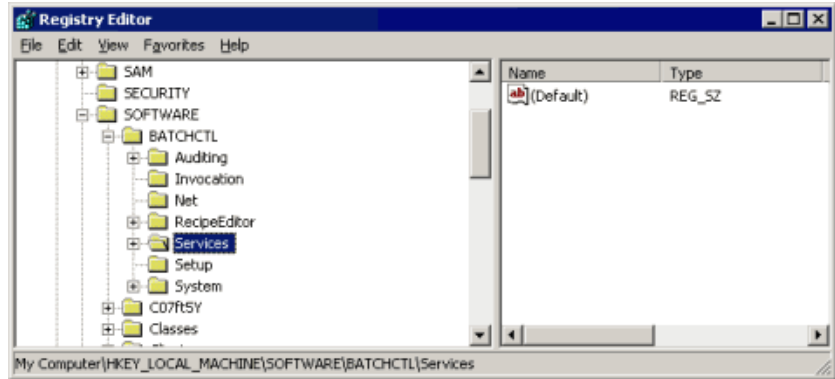
Add a custom service

Define a custom service to command the FactoryTalk Batch Server and FactoryTalk Event Archiver services using the FactoryTalk Batch Service Manager. To make a service available to the Service Manager, you must enter the file name of the executable for the service in a service registry key. This key is located at:

HKEY_LOCAL_MACHINE\SOFTWARE\BATCHCTL\Services

To add a custom service:

1. In the Registry Editor, select the **HKEY_LOCAL_MACHINE\SOFTWARE\BATCHCTL\Services** subkey.



2. From the **Edit** menu, select **New > Key**. A new key folder is added.
3. Enter the **Key Name** without the file extension.

See also

[The FactoryTalk Batch Service Manager](#) on [page 63](#)

The FactoryTalk Batch Server

The FactoryTalk Batch Server is the FactoryTalk Batch component that controls and coordinates system information, phases, and recipes. The FactoryTalk Batch Server allows integration with process-connected devices and third-party software packages. The FactoryTalk Batch Server must be started and remain active during all batch functions.

See also

[FactoryTalk Batch Server service configuration](#) on [page 73](#)

[FactoryTalk Batch Server option configuration](#) on [page 74](#)

[Modify the .NET registry key](#) on [page 90](#)

FactoryTalk Batch Server service configuration

FactoryTalk Batch Server configuration can be verified and customized as needed. Configuration includes specifying the startup method (Automatic or Manual), specifying the restart type (Warm, Warm All, or Cold), and setting the server options.



Tip: The FactoryTalk Batch Server service is configured to run as a Windows Server 2012 user. Changing the user account in which the server runs is done through Services.

See also

[The FactoryTalk Batch Server](#) on [page 73](#)

[Specify the startup type](#) on [page 73](#)

Specify the startup type

Verify the FactoryTalk Batch Server startup and account types using the Services Administrative Tool. You must be logged on with local administrative privileges to configure the server.

To specify the startup type:

1. Select **Administrative Tools > Services**.
2. From the list of services, double-click **Batch Server**.
3. In the **Startup type** area, select **Automatic** to allow the batch server to start automatically upon system startup.
4. Select the **Log On** tab.
5. From the **Log On As** area, select **This account**.
6. Select **Browse**. The **Select Users** dialog box opens.

7. Select the **User** to be used for the batch server, and then select **OK**.
8. In the **Password** box, enter the password for the batch server account.
9. In the **Confirm password** box, enter the password.
10. Select **OK** to close the **Batch Server Properties** dialog box.
11. Close the **Services** dialog box.

See also

[FactoryTalk Batch Server service configuration](#) on [page 73](#)

FactoryTalk Batch Server option configuration

Configure the FactoryTalk Batch Server and FactoryTalk Event Archiver using the **Server Options** dialog box in FactoryTalk Batch Equipment Editor. The configuration information for the server indicates how it communicates with FactoryTalk Batch clients and the data server(s). Configure FactoryTalk Event Archiver to specify if and how electronic batch record files (**.evt** or **.evtj**) are inserted into the ODBC-compliant database.

Changing items such as FactoryTalk Batch Server information, area model, and the error logging directory take effect when the server is restarted. Changing items such as the recipe directory and batch journal directory take effect the next time the server accesses these items. None of the items in FactoryTalk Event Archiver configuration require the server to be restarted. These changes take effect the next time the server needs that configuration item.

IMPORTANT FactoryTalk Batch does not support the use of mapped drives. Do not use mapped drives for project directories.

See also

[Server Options dialog box - Project Settings tab](#) on [page 74](#)

[Server Options dialog box - Restart Control tab](#) on [page 77](#)

[Server Options dialog box - Batch Reporting tab](#) on [page 79](#)

Server Options dialog box - Project Settings tab

The default project settings for a computer on which the FactoryTalk Batch Server is installed are universal naming convention (UNC) names for all except the **Initialization File Name**. If the server is installed on another computer, then the **Initialization File Name** would use UNC names.

The screenshot shows the 'Server Options' dialog box with the 'Project Settings' tab selected. The 'Batch Server' sub-tab is active. The 'Initialization File Name' field contains the UNC path: \\WIN-9A610F378BJ\BATCHCTL\BIN\BATCHSVR.INI. Under 'Project Directories', the 'Primary Journal' field contains \\WIN-9A610F378BJ\BATCHCTL\SAMPLEDEMO1\JOURNALS\, 'Error Logging' contains \\WIN-9A610F378BJ\BATCHCTL\SAMPLEDEMO1\LOGS\, and 'Equipment Database' contains \\WIN-9A610F378BJ\BATCHCTL\SAMPLEDEMO1\RECIPES\IC. The 'Store Recipes Using' section has 'Binary Files' selected, with a 'Recipe Directory' field containing \\WIN-9A610F378BJ\BATCHCTL\SAMPLEDEMO1\RECIPES\. Other options include 'Microsoft SQL Server Database' (with Node: WIN-9A610F378BJ and Database: MASTERRECIPES) and 'XML Files'. There is also an unchecked checkbox for 'Enable Recipe Versioning' with a descriptive text.

See also

[Set project directories](#) on [page 75](#)

[Configure recipe storage](#) on [page 76](#)

[Recipe versioning](#) on [page 76](#)

Set project directories

Set directories for the following: Primary Journal, Secondary Journal, Error Logging, Instructions (if you are using FactoryTalk *e*Procedure), and Equipment Database.

IMPORTANT The **Primary Journal** path must be defined or the FactoryTalk Batch Server will not start. The **Primary Journal** path must be on the server computer in the BATCHCTL share. In addition, if the **Primary Journal** or **Secondary Journal** path is invalid, the server will not start. If an invalid path is used, an error message indicating the invalid path is written to the **batchsvr.log** file.

To set project directories:

1. In FactoryTalk Batch Equipment Editor, select **Options > Server Options**.
2. Select the **Project Settings** tab.
3. Select **Browse** to open the **Select Directory** dialog box.
4. Select the appropriate directory, and then select **OK** to return to the **Server Options** dialog box. The selected UNC directory path name is inserted into the appropriate box.

See also

[Server Options dialog box - Project Settings tab](#) on [page 74](#)

[FactoryTalk Batch Server option configuration](#) on [page 74](#)

Configure recipe storage

Recipes can be stored either as binary files, in a SQL Server Database, or as XML files

To configure recipe storage

1. Open FactoryTalk Batch Equipment Editor.
2. On the menu bar, select **Options** to open the **Server Options** dialog box
3. Under **Store Recipes Using**, select one of these options for storing recipes:
 - **Binary Files (default format):** Stores recipes in separate files according to the recipe level and specified directory:
 - Stores procedures as .bpc files
 - Stores unit procedures as .upc files
 - Stores operations as .uop files
 - **Microsoft SQL Server Database:** Stores recipes in a SQL Server database.



Tip: If SQL Server is not installed on the same computer as the client, perform a custom installation of SQL Server and install the **Client Tools Connectivity** option. This displays a list of available SQL Server databases in the **Server Options** dialog box.
 - **XML Files:** Stores recipes in separate .xml files:
 - Stores procedures as .pxml files
 - Stores unit procedures as .uxml files
 - Stores operations as .oxml files
4. Select **OK** to apply the configuration settings.

Recipe versioning

Below the **Store Recipes Using** area is the **Enable Recipe Versioning** check box. Check this box to enable Recipe Versioning, a system-enforced naming convention that stores and protects recipe revisions. By default the box is unchecked and recipe versioning is disabled.

IMPORTANT FactoryTalk Full Edit access to FactoryTalk Batch Equipment Editor is required to enable and disable recipe versioning.

See also

[Configure recipe storage](#) on [page 76](#)

[Set project directories](#) on [page 75](#)

Server Options dialog box - Restart Control tab

The **Restart Control** tab allows you to specify a restart method to initiate in the event of a server computer failure. Selecting **Cold Restart** will restart the FactoryTalk Batch Server in an uninitialized state and all recipe content is erased. Selecting **Warm Restart** (the default) forces the server (or backup server) to restart in the same state it was in prior to the failure, without any loss of journal data or recipe content. Selecting **Warm All Restart** allows the server to restart only if it is able to restore all of the batches to the batch list. Warm All Restart is the default setting used by the automatic restart function.

The **Restart Control** tab also allows you to specify primary and secondary path names for the storage of files containing restart information. (See Specify the startup type for more information.)

IMPORTANT You must define a valid **Primary Restart** path or the FactoryTalk Batch Server will not start, and the following error message is logged in the Windows Server event log file: Unable to start because RestartDirectoryPath key not specified in **Batchsvr.ini**. In addition, if the **Primary Restart** or **Secondary Restart** path is invalid, the server will not start.

See also

[Redundant servers](#) on [page 77](#)

[Define redundant servers](#) on [page 78](#)

Redundant servers

FactoryTalk Batch can be configured to store batch journals and batch restart information in two separate locations. If there is a hardware failure on the primary FactoryTalk Batch Server computer, secondary server performs a manual failover. The second computer must be capable of running the server. Define redundant servers by specifying the primary and secondary restart path names. These path names indicate where system files are written to should the system experience a failure. If the secondary path name is not specified, FactoryTalk Batch does not create secondary files.



Tip: Archiver recovery – The **eventdir.txt** and the **archque.txt** are not updated as secondary files. As part of the manual failover procedure, manually copy the files from the primary directory to the secondary directory.

Install the area model file and all recipe files and, if using eProcedure, all instruction files on both the primary server and the secondary server.

See also

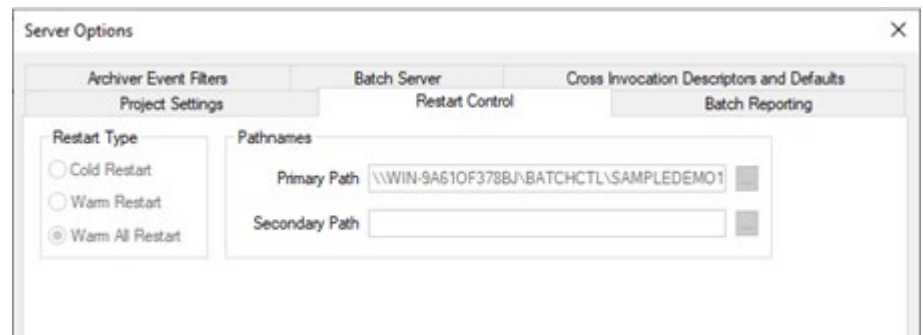
[Perform a manual failover](#) on [page 91](#)

Define redundant servers

FactoryTalk Batch can be configured to store batch journals and batch restart information in two separate locations.

To define redundant servers:

1. From the FactoryTalk Batch Equipment Editor **Options** menu, click **Server Options**.
2. Select the **Restart Control** tab.



3. In the **Restart Type** area, select an option, or accept the default, **Warm Restart**.
4. Select **Primary Path**. The **Select Directory** dialog box opens.
5. Select the **MyProject\restart** folder located in the **BATCHCTL** share on the primary server's hard drive (the default is **c:\Program Files (x86)\Rockwell Software\Batch\MyProject\restart**), and then select **OK**.
6. Select **Secondary Path**. The **Select Directory** dialog box opens.
7. Locate and select the **MyProject\restart** folder located in the **BATCHCTL** share on the secondary server's hard drive, and then select **OK**.
8. Repeat steps 1 through 7 on the redundant server computer, selecting the **Batch\MyProject\restart** folder on the redundant server as the Primary Path. The Secondary Path can remain blank or point to another redundant server. Additionally, you must configure a data server to communicate with the process controller on the redundant server computer. The redundant server computer needs to communicate with the process controller that is running the phase logic.



Tip: You must also verify that the primary and secondary journal project directories are defined.

See also

[Redundant servers](#) on [page 77](#)

[Server Options dialog box - Restart Control tab](#) on [page 77](#)

[Set project directories](#) on [page 75](#)

Server Options dialog box - Batch Reporting tab

Use the **Batch Reporting** tab to select the reporting application for generating batch reports and archiving Batch data. **Never** is the default reporting application.

Server Options

Archiver Event Filters | Batch Server | Cross Invocation D

Project Settings | Restart Control

Reporting Application

Archiver (End of Batch) Executable File: C:\PROGRAM FILES (X86)\ROCKWELL

Archiver (Incremental) Initialization File:

User-Defined (End of Batch)

None (Queue Maintained)

Never (No Queue)

Archiver

Database Connect String: ODBC;

Maximum Records Per Transaction: 75 Table Name: BATCHHIS

Delete files after archive Maximum Log File Size: 10000

Incremental Period: 5 Min Sec

OK Cancel

Reporting Application	Description
Report Editor	As of version 10.00, the Report Editor functionality is no longer supported or available. If you are currently using the Report Editor functionality, it is recommended that you transition to a different reporting solution.

Reporting Application	Description
Archiver (End of Batch)	Defers control of batch reporting to FactoryTalk Event Archiver and only performs batch reporting upon removal of the batch from the Batch List. See instructions, Perform end of batch archiving on page 80 .
Archiver (Incremental)	Defers control of batch archiving to FactoryTalk Event Archiver and performs batch reporting at user-defined intervals during batch execution. See instructions, Perform incremental archiving on page 81
User Defined (End of Batch)	Defers control of batch archiving to FactoryTalk Event Archiver and to perform batch reporting at user-defined intervals during batch execution. See instructions, Perform user-defined end of batch on page 82
None (Queue Maintained)	Select this option if no batch reporting or archiving application is to be used.
Never (No Queue)	Does not maintain a queue. This is the default selection. IMPORTANT: The Never setting causes the archiver queue file to not get written to. For a custom archiver and when writing to this queue file, use a setting other than Never and manually maintain the size of the archive queue file.

Perform end of batch archiving

Selecting **Archiver (End of Batch)** defers control of batch reporting to FactoryTalk Event Archiver and only performs batch reporting upon removal of the batch from the Batch List. The End of Batch runs in the security context of the FactoryTalk Batch Server. When using End of Batch archiving, FactoryTalk Event Archiver must be disabled as a Windows service and cannot be controlled using the FactoryTalk Batch Service Manager. (See the *FactoryTalk Event Archiver User Guide* for information on disabling FactoryTalk Event Archiver as a service.)



Tip: If Event Journal Signatures are enabled, specify **Archiver (End-of-Batch)** archiving to ensure that all event journal records are properly archived.

To perform end of batch archiving:

1. Select **Archiver (End of Batch)**.
2. In the **Archiver** area, select **Delete files after archive** to allow deletion of the **.evt** or **.evtj** files after FactoryTalk Event Archiver has stored the data in the ODBC database.

IMPORTANT Deleting files after archiving is not a reversible action and is not recommended.

3. In **Table Name**, type the name of the table that contains the Batch data. The default is **BATCHHIS**.
4. In **Database Connect String** enter either:
 - **ODBC;DSN=DataSourceName;**
 - **ODBC;**



Tip: If the DSN is not specified the first time FactoryTalk Event Archiver is started the FactoryTalk Event Archiver setup will determine the connection string and automatically update this value.

To change the default **Table Name** from **BATCHHIS**:

- a. Create a new table.
- b. Update the stored procedure named **BHInsert** to use the new table name. (See the *FactoryTalk Event Archiver User Guide* for instructions on creating a new table.)

See also

[Server Options dialog box - Batch Reporting tab](#) on [page 79](#)

Perform incremental archiving

Use **Archiver (Incremental)** to defer control of batch archiving to FactoryTalk Event Archiver and to perform batch reporting at user-defined intervals during batch execution. Specify the Incremental period between active archiving in **Minutes** or **Seconds**. The minimum setting is 5 seconds. The maximum is 9999 minutes. Setting the value outside of this range causes FactoryTalk Event Archiver to run in 5 minute incremental periods.

To update the incremental period after making changes, stop and restart the Incremental Archiver.

Configure the Incremental Archiver with a Windows user account that has administrative privileges to log on to the system. When using incremental archiving, configure FactoryTalk Event Archiver to run as a Windows service with an automatic startup type. Use the FactoryTalk Batch Service Manager to monitor and control FactoryTalk Event Archiver service. (See the *FactoryTalk Event Archiver User Guide* for information on enabling FactoryTalk Event Archiver as a service.)

To perform incremental archiving:

1. Select **Archiver (Incremental)**.
2. In the **Archiver** area, select **Delete files after archive** to enable the deletion of the **.evt** or **,evtj** files after FactoryTalk Event Archiver has stored the data in the ODBC database.

IMPORTANT Deleting files after archiving is not a reversible action and is not recommended.

3. In **Table Name**, type the name of the table that contains the batch data. (The default is **BATCHHIS**.)
4. In **Incremental Period**, type a value and select **Minutes** or **Seconds**.

See also

[Server Options dialog box - Batch Reporting tab](#) on [page 79](#)

Perform user-defined end of batch

Select **User-Defined (End of Batch)** to define a batch reporting application that is not listed in the Reporting Application area. Specify paths and names of the application's .ini and .exe files.

To perform user-defined end of batch:

1. Select **User-Defined (End of Batch)**.
2. Select the **Executable File** browse button to locate the application's executable file. The dialog box opens.
3. In the **Select Archiver Executable File** dialog box, select the appropriate .exe file, and then select **Open**.
4. Select the **Initialization File** browse button to locate the application's initialization file.
5. In the **Select Archiver Initialization File** dialog box, select the appropriate .ini file, and then select **Open**.

See also

[Server Options dialog box - Batch Reporting tab](#) on [page 79](#)

Never

Does not maintain a queue. This is the default selection.

IMPORTANT The **Never** setting causes the archiver queue file to not get written to. For a custom archiver and when writing to this queue file, use a setting other than **Never** and manually maintain the size of the archive queue file.

See also

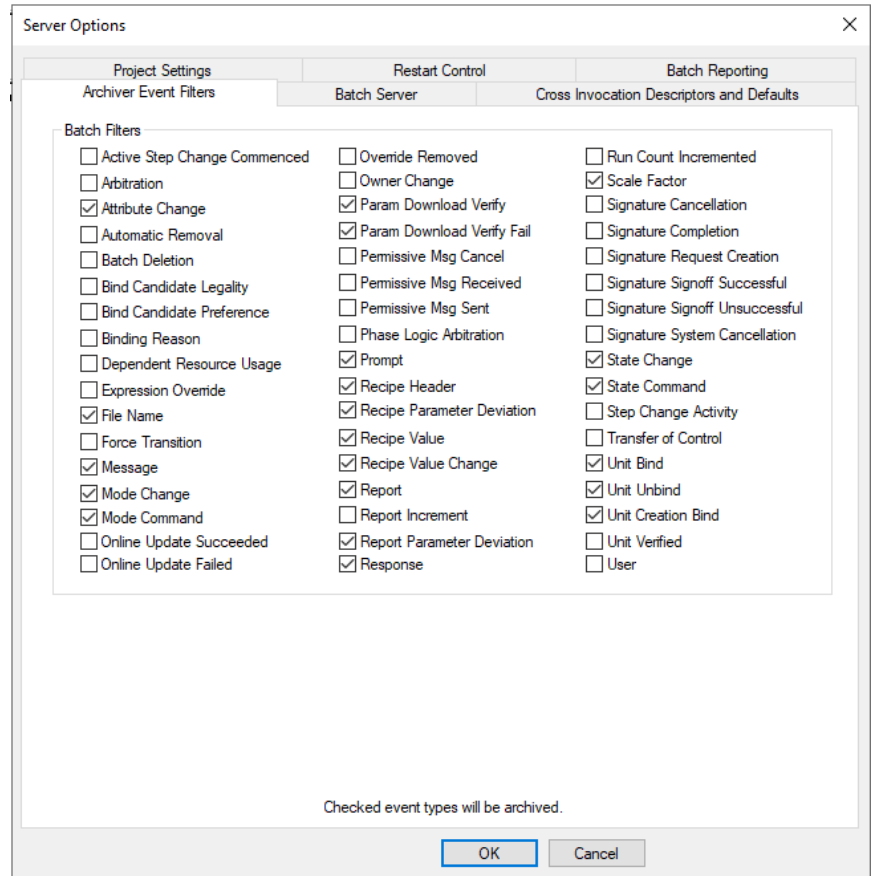
[Server Options dialog box - Batch Reporting tab](#) on [page 79](#)

Archiver Event Filters tab

The options listed under the **Archiver Event Filters** tab determine what electronic batch record data FactoryTalk Event Archiver inserts into the **BATCHHIS** table. To enable an event filter, select the appropriate check box. To disable an event filter, clear the appropriate check box.

To enable or disable an event filter:

1. Select the **Archiver Event Filters** tab.



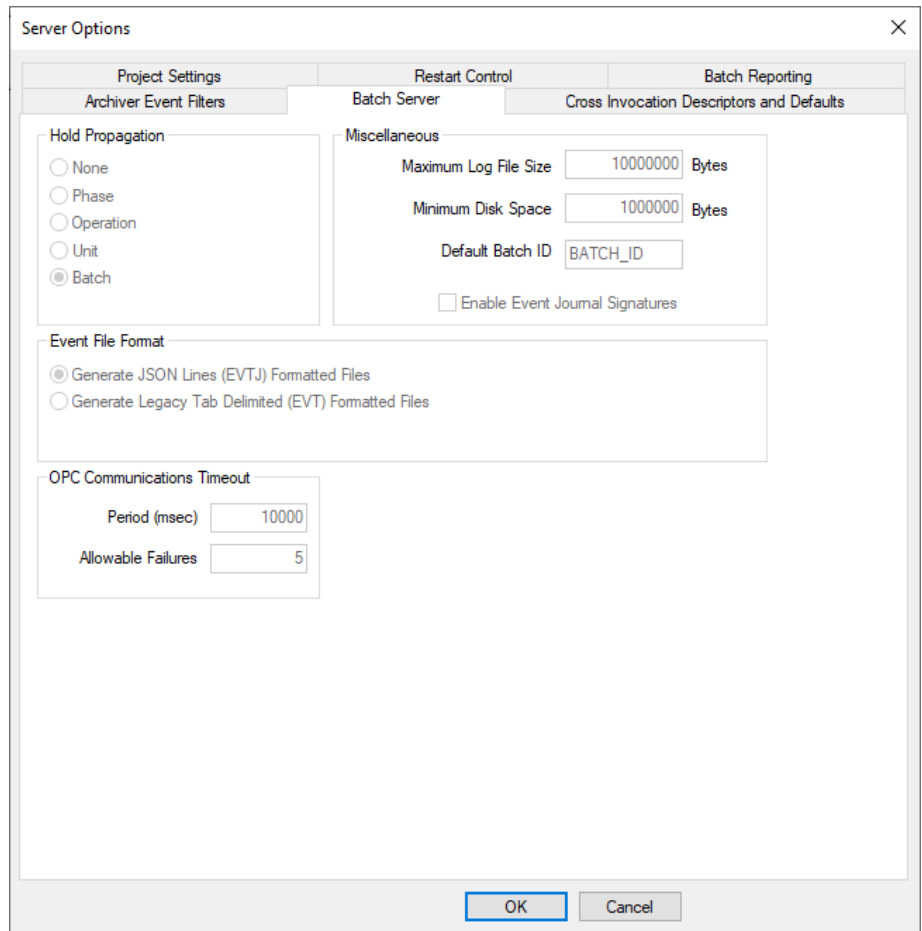
2. Select the appropriate check boxes to indicate the event type data for the FactoryTalk Event Archiver collection. Clear the check boxes for the data that is not to be collected.

See also

[FactoryTalk Batch Server option configuration](#) on [page 74](#)

Batch Server tab

The **Batch Server** tab allows you to select and/or configure FactoryTalk Batch Server options (including the hold propagation type), watchdog communications timeout settings, FactoryTalk Event Archiver log file size, minimum disk space required to add a batch, default batch ID, and whether to enable/disable event journal signatures.



See also

[Hold propagation on page 84](#)

[Communications timeout on page 85](#)

[Miscellaneous on page 86](#)

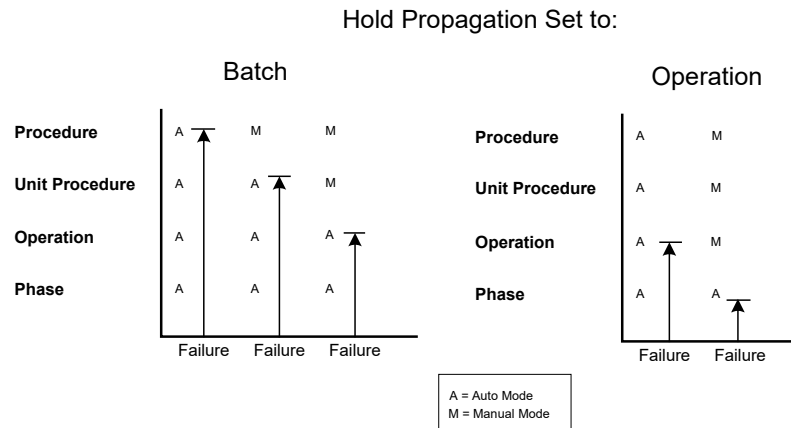
Hold propagation

The **Hold Propagation** area allows you to indicate the hold propagation type to use when the FactoryTalk Batch Server detects a failure caused by a watchdog timeout, a handshake timeout, or a phase failure (PHASE_F > 0). A Hold command associated with a failure propagates up through the recipe hierarchy as high as the mode and selected option allows.

Hold Propagation Type	Description
Phase	Only the affected phase is held, allowing the balance of the phase's unit and all other units to continue running.

Hold Propagation Type	Description
Operation	Only the affected phase and the related operation are held, allowing the balance of the operation's unit and all other units to continue running.
Unit	The phase and the related unit are held and all other units continue running.
Batch	(Default) Holds the entire batch when a failure is detected by the FactoryTalk Batch Server.

This diagram shows how a step's mode can affect the hold propagation. Depending on the step's mode, the batch may be held at a lower level than the selected hold propagation option.



See also

[Batch Server tab](#) on [page 83](#)

Communications timeout

The **Communications Timeout** area allows you to configure the FactoryTalk Batch Server's watchdog timer.

The **Period (msec)** box allows you to set the watchdog timer setpoint in milliseconds. When the timer expires, the FactoryTalk Batch Server checks communication with the process-connected device's data server and performs the set/reset test on the data server watchdog timer. In the event of a failed conversation, the server attempts to reconnect. The default is **10000**.

The **Allowable Failures** box allows you to indicate how many Set/Reset retries the FactoryTalk Batch Server should perform before it declares that conversation with the server is BAD and issues a HOLD command. The default is **5**.



Tip: The **recommended configuration** for Batch watchdog and PCD watchdog is that the PLC watchdog timer setpoint must be configured to have a value that is **5 % or less** of the **total** value of the FactoryTalk Batch Server watchdog timer's period value and allowable failures value.

Example: If the server's watchdog timeout period is set to 10000 milliseconds (10 seconds), and the allowable failures is set to 5, then set the PLC watchdog timer setpoint to 47500 milliseconds (47.5 seconds):

$$.95 (5 \times 10000) = 47500$$

IMPORTANT When the watchdog times out in the PCD, the PLI should be designed to set the Phase Failure Tag to a nonzero value that represents an understandable message in the Phase Failure Enumeration Set such as **PCD1 Batch Watchdog Timed Out**.

(See the *FactoryTalk Batch Equipment Editor User Guide* for details on editing enumerations and enumeration sets.)

See also

[Batch Server tab](#) on [page 83](#)

Miscellaneous

The **Miscellaneous** area allows you to define options relating to the batchsvr.log and the batchSvr.ini file.

The **Maximum Log File Size** box allows you to input the maximum size (in bytes) for the log file. When the log file reaches the maximum specified size, all new data is appended to the top of the log and old information is overwritten.

The **Minimum Disk Space** box allows you to input the minimum disk space (in bytes) that must be available before a batch can be added to batch list. This helps to avoid data loss due to inadequate disk space.

The **Default Batch ID** box allows you to specify a default batch ID; this batch ID automatically populates the Batch ID box when each batch is created.

Enable Event Journals Signatures turns event journal signatures on or off. The Event Journal Signature function is a tool for detecting any unauthorized changes to electronic batch records (.evt or .evtj files). (See the *FactoryTalk Batch System Files Reference Guide* for more information about Event Journal Signatures.)



Tip: Do not edit the batchsvr.log or batchsvr.ini files directly unless you are specifically instructed to do so by Rockwell Automation Technical Support. The **batchsvr.ini** file contains data items that determine the configuration of the FactoryTalk Batch Server and how the server operates in relation to the other FactoryTalk Batch components.

See also

[Batch Server tab](#) on [page 83](#)

[Batch identification](#) on [page 181](#)

Event File Format

.evt (Electronic Batch Record): Contains a complete record of events for one batch. Created by the FactoryTalk Batch Server during batch execution. Two formats of event files, .evt and .evtj, are available.

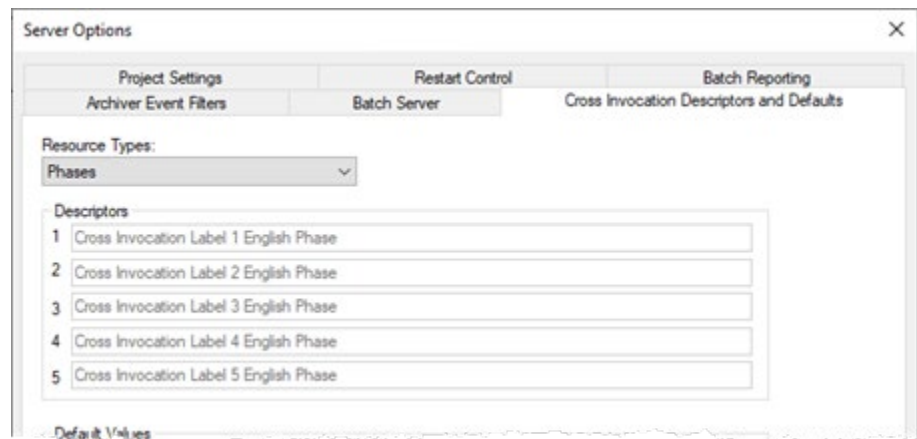


Tip: The tab delimited event journal format (*.evt) will not be updated or enhanced. It has been replaced by the enhanced JSON Lines based format (*.evtj) and the option to use legacy EVT format will be removed in a future release. We recommend systems using FactoryTalk Batch version 16.00 or later move to the new EVTJ format.

Cross Invocation Descriptors and Defaults tab

.evtj (Electronic Batch Record JSON): Contains a complete record of events for one batch. Created by the FactoryTalk Batch Server during batch execution. Two formats of event files, .evt and .evtj, are available.

The **Cross Invocation Descriptors and Defaults** tab allows you to specify cross invocation labels and set default cross invocation strings for different resource types. Use FactoryTalk Batch Equipment Editor to define up to five cross invocation strings for each resource within the area model. Each time a selection changes in FactoryTalk Batch View and HMI Controls, the cross invocation strings associated with the equipment resource are retrieved from the FactoryTalk Batch Server and evaluated. The contents of these strings can indicate a shortcut menu item and subitems, the caption(s) for the shortcut menu item(s), and the data that is passed to the automation server.



See also

[Resource types](#) on [page 87](#)

[Cross invocation](#) on [page 141](#)

Resource types

Use the **Resource Types** list to select a resource type for cross invocation labels and strings. Select one of these from the list: **Equipment Modules**, **Process Cells**, **Resources**, or **Units**.

See also

[Cross Invocation Descriptors and Defaults tab](#) on [page 87](#)

Descriptors

The **Descriptors** area allows you to enter up to five cross invocation labels for each resource type. These labels are used within the resource's **Edit** dialog box.

See also

[Cross Invocation Descriptors and Defaults tab](#) on [page 87](#)

Default values

Use the **Default Values** area to assign default URLs to each hyperlink label defined for the selected resource type. Enter the URL address for the hyperlink in the box that corresponds to the correct hyperlink label. The label in box **1** of the **Descriptors** area corresponds to the URL address in box **1** of the **Default Values** area. Default URLs can be overwritten when a specific resource type instance is edited.



Tip: If a hyperlink value is not defined, either as a default or when the specific resource type instance is created, the hyperlink label is not visible in *eProcedure*.

See also

[Hyperlink Descriptors and Defaults tab](#) on [page 88](#)

Hyperlink Descriptors and Defaults tab

This feature is used in conjunction with FactoryTalk *eProcedure* only.

The **Hyperlink Descriptors and Defaults** tab specifies the hyperlink labels and default values used with different resource types. The labels and default values are used in the resource's **Edit** dialog box. You can define up to five hyperlink labels and default values for each resource within the area model. (Refer to the *FactoryTalk eProcedure Administrator Guide* for additional hyperlink information.)

The **Resource Types** list allows you to select a resource type for the hyperlink labels and URLs you define. Select one of the following from the list: **Phases**, **Process Cells**, **Resources**, or **Units**.

The **Descriptors** area allows you to enter up to five hyperlink labels for each resource type. These labels are used within the resource's **Edit** dialog box.

The **Default Values** area allows you to assign default URLs to each hyperlink label defined for the selected resource type. Enter the URL address for the hyperlink in the box that corresponds to the correct hyperlink label. The label in box **1** of the **Descriptors** area corresponds to the URL address in box **1** of the **Default Values** area. Default URLs can be overwritten when a specific resource type instance is edited.



Tip: If a hyperlink value is not defined, either as a default or when the specific resource type instance is created, the hyperlink label is not visible in *eProcedure*.

See also

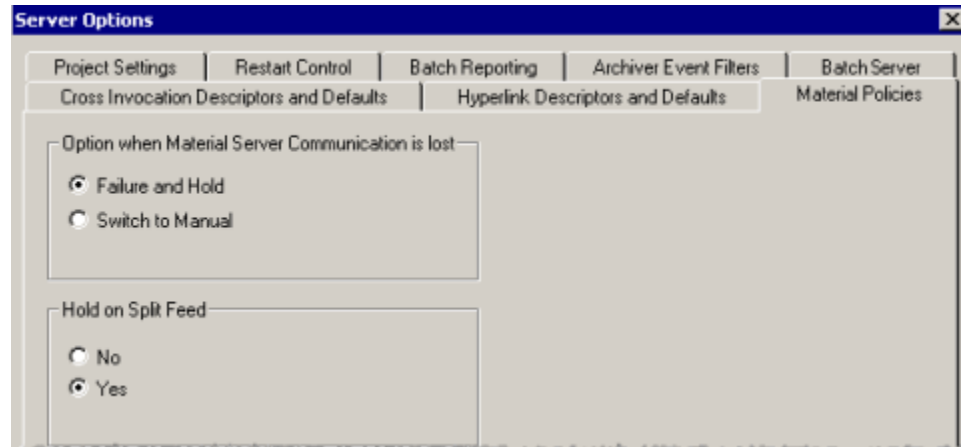
[FactoryTalk Batch Server option configuration](#) on [page 74](#)

Material Policies tab

These features are used in conjunction with FactoryTalk Batch Material Manager only.

- Option When Material Server Communication is lost

- Hold on Split Feed



Option when Material Server Communication is lost

When communication with the Material server is lost, these two options are available: **Failure and Hold** or **Switch to Manual**.

By default, the FactoryTalk Batch Server is configured to **Failure and Hold**. In this case, batches are placed in the HELD state when communication with the Material server is lost. The FactoryTalk Batch Server automatically resumes communication with the Material server when it becomes available.

To run batches even when communication is lost, configure the FactoryTalk Batch Server to **Switch to Manual**. In Manual mode, the server prompts the operator for information normally supplied by the Material Server. Restore the communication manually when the Material server becomes available. (For more information on restoring communication manually, see the *FactoryTalk Batch Material Manager Administrator Guide*.)

Hold on split feed

Select **Yes** to hold batches or **No** to continue to run batches in the event of a split feed.

The default is **Yes** and the batch is HELD when a split feed occurs.

When selecting **No** create a loop in the recipe so that the binding process can select another container for material additions when a split feed occurs. Add a material loop to the recipe with the **Create Material Loop** button in FactoryTalk Batch Recipe Editor. (For information regarding material loops, see the *FactoryTalk Batch Recipe Editor User Guide*.)

See also

[FactoryTalk Batch Server option configuration](#) on [page 74](#)

Modify the .NET registry key

The .NET registry key contains the FactoryTalk Batch Server computer name and is used by remote clients to locate the server computer. During a FactoryTalk Batch Client computer installation, the setup program prompts you for the server computer name and then places the name into the .NET key. In the event that you change the server computer name, or if you are switching control to another FactoryTalk Batch Server computer, you need to manually modify the .NET registry key on all remote client computers to point to the server computer.

To modify the .NET registry key:

1. Open the Registry Editor.
2. Select the HKEY_LOCAL_MACHINE on Local Machine window. Navigate to the **SOFTWARE\BATCHCTL\Net** subkey.
3. Double-click the **Server** registry value.
4. Modify the value data: Server = <MachineName>, where <MachineName> is the computer name on which the FactoryTalk Batch Server is installed.

See also

[The FactoryTalk Batch Server](#) on [page 73](#)

Use automatic restart control to restart the FactoryTalk Batch Server

The FactoryTalk Batch Server maintains a detailed record of every recipe's state as the recipe is running. If the server computer were to lose power or otherwise fail, use the automatic restart control feature to restart the server. Upon restarting the server on a failed server node, the following functions are placed into the state that existed prior to termination of the server:

- Control Recipes
- Semi-Auto Phase Control
- Arbitration

See also

[Restart the FactoryTalk Batch Server](#) on [page 90](#)

Restart the FactoryTalk Batch Server

If the FactoryTalk Batch Server computer experiences a system failure, then restarting the computer starts the server. However, if the FactoryTalk Batch Server service fails, then you need to restart the service using the FactoryTalk Batch Service Manager.

To restart the FactoryTalk Batch Server:

1. Open FactoryTalk Batch View. Batches on the batch list prior to the failure are still on the batch list. All batches that had phases in an

active state now have transitions in the HELD state, and are in MANUAL mode with a failure.

2. Select the batch, and then select **Auto** to place a batch in AUTOMATIC mode.
3. Select **Clear All Failures** to clear the failures.
4. Select the batch, and then select **Restart Batch**. The previously active phases return to a RUNNING state and the batch completes.

See also

[Use automatic restart control to restart the FactoryTalk Batch Server](#) on [page 90](#)

[The FactoryTalk Batch Service Manager](#) on [page 63](#)

Perform a manual failover

FactoryTalk Batch can be configured to store batch journals and batch restart information in two separate locations. The use of a secondary server allows you to perform a manual failover if there is a hardware failure on the primary FactoryTalk Batch Server. The second computer must be capable of running the server.

You must install your area model file and all recipe files (and all instruction files if using eProcedure) on both the primary server and the secondary server.



Tip: Archiver recovery - As part of the manual failover procedure you must manually copy the `eventdir.txt` and the `archque.txt` from the primary directory to the secondary directory.

See also

[The FactoryTalk Batch Server](#) on [page 73](#)

[Define redundant servers](#) on [page 78](#)

Configure the FactoryTalk Batch Server for manual failover

When the FactoryTalk Batch Server fails, complete these steps.



Tip: If you are using FactoryTalk Batch Material Manager, refer to the procedure for material-enabled configuration.

To configure the FactoryTalk Batch Server for manual failover:

1. Start the process-connected device communication software on the secondary server. The secondary server computer needs to communicate with the process controller that is running the phase logic.
2. Ensure that the secondary server's project directories are assigned to local directories in the **Server Options** dialog box located in FactoryTalk Batch Equipment Editor.
3. Use the FactoryTalk Batch Service Manager to start the FactoryTalk Batch Server on the secondary server computer, select **Warm** or **Warm**

- All** as the startup type. The secondary server allows the batches to be restarted based upon information written by the primary server.
4. (optional) If you are also using *eProcedure*, use the FactoryTalk Batch Service Manager to start the *eProcedure* Server on the secondary server computer.
 5. For each computer running FactoryTalk Batch View, enter the secondary server's computer name as the NET registry key value.
 6. Stop and restart each FactoryTalk Batch View instance and any other software that communicates directly with the FactoryTalk Batch Server. This causes the instance to connect with the secondary server.



Tip: The existing batches are placed in the MANUAL mode with transitions in the HELD state. Place the batches in AUTOMATIC mode, clear the failures, and then restart the batch.

See also

[Perform a manual failover](#) on [page 91](#)

[Modify the .NET registry key](#) on [page 90](#)

[Configure the material-enabled FactoryTalk Batch Server for manual failover](#) on [page 92](#)

Configure the material-enabled FactoryTalk Batch Server for manual failover

Upon failure of the primary FactoryTalk Batch Server computer, if this is a FactoryTalk Batch Material Manager system, complete these steps:

To configure the material-enabled FactoryTalk Batch Server for manual failover:

1. On the Material server computer, start the Network Editor.
2. Select the network model for the primary FactoryTalk Batch Server.
3. Right-click BATCHSERVER in the tree and select Remove Server.
4. From the File menu, select **Synchronize**.
5. Select the network model for the Material server.
6. Right-click any <Undefined> FactoryTalk Batch Server in the tree, and then click Add Server.
7. Type a name in the Alias box.
8. Select the Computer Name box, and then select the computer to use as the secondary server.
9. From the File menu, select **Synchronize**.
10. Start the process-connected device communication software on the secondary server. The secondary server computer needs to communicate with the process controller running the phase logic.
11. Ensure that the project directories for the secondary server are assigned to local directories. Use the Server Options dialog box in FactoryTalk Batch Equipment Editor.
12. Use the FactoryTalk Batch Service Manager to start the FactoryTalk Batch Server software on the secondary server computer. Select Warm

- or Warm All as the startup type. The secondary server allows the batches to be restarted based upon information written by the primary server.
13. If you are also using *eProcedure*, use the FactoryTalk Batch Service Manager to start the *eProcedure* Server on the secondary server computer.
 14. For each computer running the FactoryTalk Batch View, enter the computer name of the secondary server as the NET registry key value.
 15. Stop and restart each FactoryTalk Batch View instance and any other software that communicates directly with the FactoryTalk Batch Server. This causes the FactoryTalk Batch View instance to connect with the secondary server.



Tip: The existing batches are placed in the MANUAL mode with transitions in the HELD state. Place the batches in AUTOMATIC mode, clear the failures, and then restart the batch.

See also

[Modify the .NET registry key](#) on [page 90](#)

Command handshake timeout

The FactoryTalk Batch Server uses a command handshake protocol ensuring that commands issued to phases are handled in a serial manner and are not lost or overwritten. The protocol has a configurable timeout period. When a command handshake timeout condition occurs, an error is generated in the phase object that represents the phase, and the batch is placed on HOLD based upon the configured hold propagation settings. This allows the operator to disconnect the server from the affected phase should the timeout period expire. Without this feature, the server must be shut down and restarted if the phase stops responding.

When a command handshake timeout period occurs, the step representing the troubled phase in the **SFC** display turns red and a failure is generated for the phase. The message PHASE NOT RESPONDING is shown on the status bar when the step is selected. It is also displayed in the **Phase List Display** area of the **Phase Control** window when the phase is selected.

A **System Message** event type is logged into the electronic batch record. The description includes the fact that a time-out occurred as well as the type and ID of the command that timed out. In addition, the event is recorded in the FactoryTalk Batch Server log file with a severity status of SEVERE and the description is the type and ID of the timed-out command.

If a communication error is detected while the command handshake is in process, the command handshake timer is stopped. The timer is reset and restarted if communication is successfully restored. If the command handshake completes after the timeout period, then the commands that are pending the handshake completion are processed. However, the error must be cleared before the batch can be restarted.

In order to restart a batch, you must issue the CLEAR_FAILURES command. If the command handshake completes successfully, the error is cleared and the batch can be restarted. Otherwise, the error is not cleared and the batch is not restarted. The operator should check the execution status of the process-connected device.

The **Batchsvr.ini** file contains an item used to configure the length of time, in seconds, that the FactoryTalk Batch Server waits for a command handshake before issuing a timeout. If the item is not included in the **Batchsvr.ini** file, it defaults to **60**. The valid range is 5 to 600.

[XMAN] CommandTimeOut=90

This value is read when the FactoryTalk Batch Server starts. If you change the value, stop and restart the server.

See also

[The FactoryTalk Batch Server](#) on [page 73](#)

Area models

An area model is a definition of the manufacturing facilities capabilities with which the FactoryTalk Batch server interacts. The information for a facility is stored in a configuration file you build using the FactoryTalk Batch Equipment Editor and saved as a file with a `cfg` extension (such as, `SouthPlant.cfg`).

The area model contains information about the process equipment used to create batches. It includes all the equipment configured for one area of a specific physical plant. An area model corresponds to a single FactoryTalk Batch Server. The area model configuration file (`.cfg`) is used by FactoryTalk Batch Recipe Editor, FactoryTalk Batch Server, and Phase Simulator.

Create the area model in this order:

- Enumerations
- Data servers
- Resources
- Process cells
- Units
- Phase classes
- Phases

In some cases, you will need to re-configure upper-level equipment after defining the lower-level equipment.

Online and runtime area models

Once the area model is configured it is ready to be brought online and used during runtime to produce batches. If changes to the area model are required while it is running you can deploy modifications to the runtime area model without disrupting running batches.

See also

[Compare area models](#) on [page 96](#)

[Review the Compare/Update log](#) on [page 98](#)

[Verify and deploy an updated area model using the FactoryTalk Batch Service Manager](#) on [page 97](#)

Update an online area model

Area model configurations are composed of a set of elements that define the functions of the area. A subset of these elements cannot be changed when the model is running. After modifying an online area model you must verify it to confirm that the elements modified are permitted to be updated and can be deployed at runtime.

When you make updates to the area model when the batch server is running you use the **Verify/Deploy** feature in either the FactoryTalk Batch Equipment Editor or the FactoryTalk Batch Service Manager to check that the changes made will not adversely affect the runtime area model.

Verify Area Model

- In this step, the changes to the area model are compared to the currently loaded area model configuration file. The purpose of this comparison is to determine whether the set of differences between the two area models, combined with the current state of the FactoryTalk Batch Server, permit deploying of the new area model configuration file.
- The results of this comparison are written into the Compare/Update log file created when the request is processed. After reviewing the content of the log file you update the area model as needed to resolve any runtime conditions before you continue on to deploy the permitted changes to the area model. These conflicts are created when that portion of the area model that was planned to be updated was in use at the time the deployment occurred. These conditions could be resolved by running the verification again at a later stage in the process.

Deploy Area Model

- After verification is complete and all errors have been addressed the updated area model can be deployed to the currently running batch server. The Update log file is created that records the details of the updates performed.

See also

[Review the Compare/Update log](#) on [page 98](#)

[Resolve runtime condition checks](#) on [page 99](#)

Verify and deploy an updated area model using the FactoryTalk Batch Service Manager

To prepare an online area model for deployment, an operator can verify that changes made to the area model are permitted.



Tip: You will be asked to login to FactoryTalk to perform this procedure. You must be able to login with a user account that has been granted the **Configure Runtime Area Model Deployment** permission in the product policy configuration stored in the FactoryTalk Directory System\Policies\Product Policies\Batch\Feature Security location. Members of the Administrator group have this permission by default.

Then, the operator can deploy changes to the runtime area model without having to stop the FactoryTalk Batch Server. If the set of differences is permitted by the FactoryTalk Batch Server, the updated area model is successfully deployed.



Tip: An area model can also be verified and deployed from the FactoryTalk Batch Equipment Editor. For more information, see the *FactoryTalk Batch Equipment Editor User Guide*.

To verify an updated area model using the FactoryTalk Batch Service Manager

1. Open the FactoryTalk Batch Service Manager.
2. Select **Verify/Deploy Area Model**.
3. If prompted, log on to FactoryTalk.
4. Select an area model from the **Updated Area Model** list.
5. Select **Verify Area Model** to compare the original area model with the updated area model.
6. If the verification is not successful, a message displays. Select **View Log** to view the reason for the differences that are not permitted.
7. Select **Close**.

To deploy an updated area model using the FactoryTalk Batch Service Manager

1. Open the FactoryTalk Batch Service Manager.
2. Select **Verify/Deploy Area Model**.
3. If prompted, log on to FactoryTalk.
4. Select an area model from the **Updated Area Model** list.
5. Select **Deploy Area Model**.

The **Status** area displays whether the deployment is allowed. If it is not successful, a message displays. Select **View Log** to view the details. If the operator attempting to deploy an updated area model does not have the required FactoryTalk permissions, this button is disabled.

6. Select **Close**.

Review the Compare/Update log

The Compare/Update Log records both supported and unsupported modifications to the area model configuration file. When reviewing this log file, be aware of these conventions:

Change type categories

Requested changes to area model configuration data when inspecting a new area model configuration data fall into one of four categories:

- **PERMITTED:** Configuration changes in this category are always supported.
- **PERMITTED WITH CONDITIONS:** Configuration changes in this category are supported unless a current condition in the FactoryTalk Batch Server (for example, a Batch reference to an object, a pending arbitration request, and so on) prohibits the change. These conditions are referred to as *runtime permissives* and are transient conditions. These changes prevent deployment if the condition is true when the update is deployed.
- **NOT PERMITTED:** Configuration changes in this category are always unsupported and the area model will not be able to be deployed. These should be removed before deploying the area model update.
- **IGNORED:** Configuration changes in this category are not detected and do not prevent deployment of the area model. However, they will not be in force until a new batch process is started.

Log file sections

In the Compare/Update log file that is generated by a Verify/Deploy Area Model request, the information used to determine whether the requested change in Area Model data is permitted or not is captured in three sections:

- **CHANGES PERMITTED TO LOAD [count=XXX]:** This section of the log file contains one record for each area model change detected that is either **PERMITTED** or **PERMITTED WITH CONDITIONS**.
- **CHANGES NOT PERMITTED TO LOAD [count=XXX]:** This section of the log file contains one record for each area model change detected that is **NOT PERMITTED**. If one or more of these changes are detected, then the new area model configuration file cannot be deployed.
- **RUNTIME CONDITIONS CHECK:** This section of the document will contain one record for each **PERMITTED WITH CONDITIONS** area model change that is not currently permitted due to a runtime permissive condition. The log record describes the condition that is preventing the change. If no runtime condition checks fail, then this section of the log file will contain a single record stating **All runtime conditions pass**. Once all runtime conditions pass and if the count of **CHANGES NOT PERMITTED TO LOAD** equals zero, then the area model update can be deployed.

See also

[Resolve runtime condition checks](#) on [page 99](#)

[Update global area objects, classes, and attributes](#) on [page 100](#)

[Update data server definitions in an area model](#) on [page 107](#)

[Update enumerations and enumeration sets in an area model](#) on [page 108](#)

[Update operation sequences and operation sequence classes in an area model](#) on [page 109](#)

Resolve runtime condition checks

When the Compare/Update log indicates that there are no NOT PERMITTED changes between the loaded and new area model configuration files, but the update is being blocked by runtime condition checks, the text of the records in the RUNTIME CONDITIONS CHECK section of the log file provide information about the transient conditions that are preventing the new area model from being deployed.

In general, minimizing the number of batches on the batch list and minimizing Operator ownership (arbitration) of resources maximizes the likelihood of avoiding runtime condition checks preventing an area model deployment.

Here are some examples of runtime condition check failure log entries and possible resolutions:

- Phase Class 'MIX' cannot have parameter(s) added. Batch 'MARVELOUS_MARMALADE_932' (Create ID:1064) affected due to active step 'MIX:2' referencing Phase Class.
To resolve this runtime permissive either wait until the step MIX:2 is no longer an active step in the batch or use active step change functionality to deactivate the step MIX:2.
- Unit Attribute Tag Class 'TURBIDITY' cannot be deleted. Batch 'BASIC_BREAD_7' (Create ID:9352) affected due to transition 'MIXER.TURBIDITY > 451' referencing Phase Class.
To resolve this runtime permissive, it would be necessary to wait until the batch BASIC_BREAD_7 was removed from the batch list.
- 'Resources Needed' property of Unit 'REACTOR_105' cannot be modified. REACTOR_105 is currently owned and/or requested.
To resolve this runtime permissive, it would be necessary to inspect the arbitration information for Unit REACTOR_105 to determine the Owners and Requestors of the Unit. All arbitration requests for this resource would then need to be cleared (batches advanced or removed and operator requests released).

See also

[Update global area objects, classes, and attributes](#) on [page 100](#)

[Update data server definitions in an area model](#) on [page 107](#)

[Update enumerations and enumeration sets in an area model](#) on [page 108](#)

[Update phases and phase classes in an area model](#) on [page 116](#)

[Update process cells and process cell classes in an area model](#) on [page 125](#)

Update global area model

The area portion of the area model organizes a hierarchy of objects into object classes and instances of those classes.

Permitted changes

You can make the following changes without impacting the operation of the area model because these changes are not going to interfere with process operation:

- Creating a data server
- Modifying a revision number
- Creating an enumeration set
- Creating or deleting a signature template
- Modifying a command policy
- Creating or deleting a resource class
- Creating a resource
- Creating a tag class
- Creating or deleting a process cell class
- Creating a process cell
- Unsharing a unit from a process cell
- Creating a unit class
- Creating a phase class
- Creating an operation class
- Enabling recipe versioning

Permitted with conditions

Some changes to an area model have to be assessed before they can be permitted. If the object being modified is required for a running process it cannot be changed until that process is complete.

Review this list for an overview of the edits that require additional planning before deployment:



Tip: Text preceded by % are placeholders for variables. In your environment the message will include the appropriate information from your area model.

- Deleting an enumeration set.

An enumeration set cannot be deleted if it is referenced by the recipe document parameters or reports in batches on the batch list. If you attempt to delete an enumeration set that has references, this error is recorded in the Compare/Update log:

```
"Enumeration Set '%s' cannot be deleted. Enumeration Set is referenced in Batch '%s' (CreateID:%ld) by Recipe Parameter(s) and/or Recipe Report(s) in recipe document %s."
```

- Modifying a security authority identifier.

The security authority identifier can be changed, but it must be an exact match to the FactoryTalk Directory being used by the Batch Server, otherwise the deploy will fail without an error. This behavior protects the integrity of the batch process.

If the loaded area model has a security authority identifier configured, it cannot be removed. If you attempt to remove the security authority identifier, this error is recorded in the Compare/Update log:

```
"Online Area Model Change not permitted to turn off securing of Area Model to FactoryTalk Directory."
```

- Deleting a resource.

A resource cannot be deleted if the resource is owned or requested by any entity other than the Operator. This error is recorded in the Compare/Update log:

```
"Resource '%s' cannot be deleted. %s is owned or requested by one or more non-Operator entities."
```

- Deleting a tag class (either a unit attribute, parameter tag class or report tag class).

A tag class cannot be deleted in these situations:

- If the tag class is a unit attribute that is referenced by a transition expression within a batch. This error is recorded in the Compare/Update log:

```
"Unit Attribute Tag Class '%s' cannot be deleted. Batch '%s' (Create ID:%ld) affected due to transition '%s' referencing Phase Class."
```

- If the tag class is a unit attribute that is referenced by a binding requirement or binding preference of a unit requirement within a batch. This error is recorded in the Compare/Update log:

```
"Unit Attribute Tag Class '%s' cannot be deleted. Batch '%s' (Create ID:%ld) contains a Unit Requirement '%s' with a Binding Requirement or Preference referencing Unit Attribute Tag Class."
```

- Deleting a process cell.

A process cell cannot be deleted if the process cell is owned or requested by any entity other than the Operator. This error is recorded in the Compare/Update log:

```
"Process cell '%s' cannot be deleted. %s is owned or requested by one or more non-Operator entities."
```

- Deleting a unit class.

A unit class cannot be deleted in these situations:

- If the unit class is referenced by a class-based unit requirement with a batch on the batch list. This error is recorded in the Compare/Update log:

```
"Unit Class '%s' cannot be deleted. Batch '%s' (Create ID:%ld) contains a class based Unit Requirement, '%s', that references the Unit Class."
```
- If the unit class is referenced by an instance-based unit requirement within a batch on the batch list. This error is recorded in the Compare/Update log:

```
"Unit Class '%s' cannot be deleted. Batch '%s' (Create ID:%ld) contains an instance based Unit Requirement, '%s', that references a Unit of the Unit Class."
```
- Creating a unit

A unit cannot be created if adding the unit causes a previously common Phase Class, Unit Class, or Unit Attribute of the parent Unit Class to no longer be common to all instances of the class and there are one or more class-based UPC recipes in existing batches that reference the Phase Class\Unit Class\Unit Attribute. This error is recorded in the Compare/Update log:

```
"Unit '%s' cannot be created. Associated Unit Class is referenced by one or more class based UPC steps inside of a BPC batch and new Unit would reduce the set of common phase classes and/or common operation classes and/or common unit attributes."
```

- Deleting a unit

A unit cannot be deleted in these situations:

- If the unit is owned or requested by any entity other than the Operator. This error is recorded in the Compare/Update log:

```
"Unit '%s' cannot be deleted. %s is owned or requested by one or more non-Operator entities."
```
- If the unit is referenced by a instance-based unit requirement within a batch. This error is recorded in the Compare/Update log:

```
"Unit '%s' cannot be deleted. Batch '%s' (Create ID:%ld) contains a instance based Unit Requirement '%s' referencing the Unit to be deleted."
```
- If the unit is bound to by a class-based unit requirement within a batch. This error is recorded in the Compare/Update log:

```
"Unit '%s' cannot be deleted. Batch '%s' (Create ID:%ld) contains a class based Unit Requirement '%s' bound to the Unit to be deleted."
```
- If there is a manual phase control batch or manual operation sequence control batch executing against the unit. This error is recorded in the Compare/Update log:

```
"Unit '%s' cannot be deleted. Manual Phase/Operation
Control Batch '%s' (Create ID:%ld) is running within
the Unit to be deleted."
```

- **Deleting a phase class**

A phase class cannot be deleted in these situations:

- If there is a manual phase control batch using an instance of the phase class. This error is recorded in the Compare/Update log:


```
"Phase Class '%s' cannot be deleted. Manual Phase
Control Batch '%s' (CreateID:%ld) is using an instance
of the Phase Class."
```
- If the phase class is referenced by any phase step in a batch. This error is recorded in the Compare/Update log:


```
"Phase Class '%s' cannot be deleted. Batch '%s' (Create
ID:%ld) affected due to step '%s' referencing Phase
Class."
```
- If the phase class is referenced by a unit attribute binding requirement within a batch. This error is recorded in the Compare/Update log:


```
"Phase Class '%s' cannot be deleted. Batch '%s' (Create
ID:%ld) contains a Unit Requirement '%s' with a Binding
Requirement or Preference referencing Phase Class."
```

- **Creating or sharing a phase**

A phase cannot be created or shared if it would result in zero legal bind choices for one or more unbound class-based unit requirements within batches. This error is recorded in the Compare/Update log:

```
"Area Model Changes not permitted. Batch '%s' (Create
ID:%ld) has a Unit Requirement '%s' for which no legal
bind targets would exist."
```

- **Deleting a phase**

A phase cannot be deleted in these situations:

- If deletion of the phase would result in zero legal bind choices for one or more unbound class-based unit requirements within batches. This error is recorded in the Compare/Update log:


```
"Area Model Changes not permitted. Batch '%s' (Create
ID:%ld) has a Unit Requirement '%s' for which no legal
bind targets would exist."
```
- If the phase is owned or requested by any entity other than the operator. This error is recorded in the Compare/Update log:


```
"Phase '%s' cannot be deleted. %s is owned or requested
by one or more non-Operator entities."
```
- If the phase is bound to a phase step within a batch that references the unit from which the phase was removed. This error is recorded in the Compare/Update log:


```
"Phase '%s' cannot be removed from Unit '%s'. Batch
'%s' (Create ID:%ld) affected due to step '%s' being
bound to removed phase."
```

- If the phase is in use by a manual phase control batch within a unit from which the phase was removed. This error is recorded in the Compare/Update log:

```
"Phase '%s' cannot be removed from Unit '%s'. Phase is
in use by Manual Phase Control Batch '%s'
(CreateID:%ld)."
```

- If deletion of the phase will make the phase's phase class non-common to a unit class and that unit is referenced by a class-based unit requirement without a "Require Phase" bind requirement specifying the affected phase class, then it is not permitted to delete the phase. This error is recorded in the Compare/Update log:

```
"Phase '%s' cannot be removed from Unit '%s'. Batch
'%s' (CreateID:%ld) affected by Phase Class '%s'
becoming non-common to Unit Class '%s'."
```

- Unsharing a phase

A phase cannot be unshared in these situations:

- If unsharing of the phase would result in zero legal bind choices for one or more unbound class-based unit requirements within batches. This error is recorded in the Compare/Update log:

```
"Area Model Changes not permitted. Batch '%s' (Create
ID:%ld) has a Unit Requirement '%s' for which no legal
bind targets would exist."
```

- If the phase is bound to a phase step within a batch that references a unit from which the phase was removed. This error is recorded in the Compare/Update log:

```
"Phase '%s' cannot be removed from Unit '%s'. Batch
'%s' (Create ID:%ld) affected due to step '%s' being
bound to removed phase."
```

- If the phase is in use by a manual phase control batch within a unit from which the phase was removed. This error is recorded in the Compare/Update log:

```
"Phase '%s' cannot be removed from Unit '%s'. Phase is
in use by Manual Phase Control Batch '%s'
(CreateID:%ld)."
```

- If unsharing of the phase will make the phase's phase class non-common to a unit class and that unit is referenced by a class-based unit requirement without a "Require Phase" bind requirement specifying the affected phase class, then it is not permitted to delete the phase. This error is recorded in the Compare/Update log:

```
"Phase '%s' cannot be removed from Unit '%s'. Batch
'%s' (CreateID:%ld) affected by Phase Class '%s'
becoming non-common to Unit Class '%s'."
```

- Deleting an operation class.

An operation class cannot be deleted in these situations:

- If there is a manual operation sequence control batch using an instance of the operation class. This error is recorded in the Compare/Update log:


```
"Operation Class '%s' cannot be deleted. Manual Operation Control Batch '%s' (CreateID:%ld) is using an instance of the Operation Class."
```
- If the operation class is referenced by any operation step in a batch. This error is recorded in the Compare/Update log:


```
"Operation Class '%s' cannot be deleted. Batch '%s' (Create ID:%ld) affected due to step '%s' referencing Operation Class."
```
- If the operation class is referenced by a unit attribute binding requirement within a batch. This error is recorded in the Compare/Update log:


```
"Operation Class '%s' cannot be deleted. Batch '%s' (Create ID:%ld) contains a Unit Requirement '%s' with a Binding Requirement or Preference referencing Operation Class."
```
- Creating or sharing an operation.

If creation or sharing of the specified operation would result in zero legal bind choices for one or more unbound class-based unit requirements within batches, then the operation cannot be created. This error is recorded in the Compare/Update log:

```
"Area Model Changes not permitted. Batch '%s' (Create ID:%ld) has a Unit Requirement '%s' for which no legal bind targets would exist."
```
- Deleting an operation

An operation cannot be deleted in these situations:

 - If deletion of the operation would result in zero legal bind choices for one or more unbound class-based unit requirements within batches. This error is recorded in the Compare/Update log:


```
"Area Model Changes not permitted. Batch '%s' (Create ID:%ld) has a Unit Requirement '%s' for which no legal bind targets would exist."
```
 - If the operation is owned or requested by any entity other than the Operator. This error is recorded in the Compare/Update log:


```
"Operation '%s' cannot be deleted. %s is owned or requested by one or more non-Operator entities."
```
 - If the operation is bound to by an operation step with in a batch that references a unit from which the operation was removed. This error is recorded in the Compare/Update log:


```
"Operation '%s' cannot be removed from Unit '%s'. Batch '%s' (Create ID:%ld) affected due to step '%s' being bound to removed Operation."
```
 - If the operation is in use by a manual operation sequence control batch within a unit from which the operation was removed. This error is recorded in the Compare/Update log:

```
"Operation '%s' cannot be removed from Unit '%s'.
Operation is in use by Manual Operation Control Batch
'%s' (CreateID:%ld)."
```

- If deletion of the operation will make the operation's operation class non-common to a unit and that unit is referenced by a class-based unit requirement without a "Require Operation" bind requirement specifying the affected operation class. This error is recorded in the Compare/Update log:

```
"Operation '%s' cannot be removed from Unit '%s'. Batch
'%s' (CreateID:%ld) affected by Operation Class '%s'
becoming non-common to Unit Class '%s'."
```

- Unsharing an operation

An operation cannot be unshared in these situations:

- If unsharing of the operation would result in zero legal bind choices for one or more unbound class-based unit requirements within batches. This error is recorded in the Compare/Update log:

```
"Area Model Changes not permitted. Batch '%s' (Create
ID:%ld) has a Unit Requirement '%s' for which no legal
bind targets would exist."
```

- If the operation is bound to by an operation step referencing a unit from which it was removed within a batch. This error is recorded in the Compare/Update log:

```
"Operation '%s' cannot be removed from Unit '%s'. Batch
'%s' (Create ID:%ld) affected due to step '%s' being
bound to removed Operation."
```

- If the operation is in use by a manual operation sequence control batch within a unit from which the operation was removed. This error is recorded in the Compare/Update log:

```
"Operation '%s' cannot be removed from Unit '%s'.
Operation is in use by Manual Operation Control Batch
'%s' (CreateID:%ld)."
```

- If unsharing of the operation will make the operation's operation class non-common to a unit and that unit is referenced by a class-based unit requirement without a "Require Operation" bind requirement specifying the affected operation class. This error is recorded in the Compare/Update log:

```
"Operation '%s' cannot be removed from Unit '%s'. Batch
'%s' (CreateID:%ld) affected by Operation Class '%s'
becoming non-common to Unit Class '%s'."
```

Not permitted changes

These edits are not permitted for area objects when the area model is running:

- Changing the name of the area model config file.
- Modifying global binding requirements.
- Deleting a data server.



Tip: Any request to change the name of an existing object deletes the existing object and creates a new object. This may be permitted depending on the current condition of the batch process.

Data server definitions in area model

Ignored

These edits are ignored for area objects when the area model is running:

- Enabling recipe approvals.
- Modifying recipe approvals.

A FactoryTalk Batch data server is a set of data that defines how the Batch Server can communicate with phases, sequences, and tags in process-connected devices through the OPC, FactoryTalk LiveData, or Common Industrial Protocol (CIP). These data server references provide runtime information from the process-connected device to FactoryTalk Batch View or the HMI User Controls. Data servers are used to pass data between the FactoryTalk Batch Server and the process-connected device.

A minimum of one data server must be defined in the area model. Define a data server before adding phases or operation sequences to the area model.

Permitted changes

You can make the following changes without impacting the operation of the area model because these changes are not going to interfere with process operation:

- Creating a data server.
- Modifying these data server properties:
 - Adding or modifying a watchdog item name for a LiveData or OPC data server.
 - Adding or modifying a watchdog item access path for a LiveData or OPC data server.
 - Adding or modifying a watchdog write item name for a LiveData or OPC data server.
 - Adding or modifying a watchdog write item path for a LiveData or OPC data server.
 - Adding or modifying the path to a Logix Designer project file.
 - Adding or modifying the revision of a Logix Designer project file.
 - Adding or modifying the controller type of a Logix Designer project file.

Not permitted changes

- Deleting a data server
- Modifying these data server properties:

- Changing the **Name** of the data server.
- Changing the number of times the FactoryTalk Batch Server tries to contact the controller used by the Logix5000 CIP data server before dropping the connection. (**Retries**)
- Changing how often the FactoryTalk Batch Server attempts to contact the controller used by the Logix5000 CIP data server before stopping the connection attempt. (**Timeout**)
- Changing the **FactoryTalk Linx Controller Shortcut** used by the Logix5000 CIP data server.
- Changing the **Watchdog Protocol** defined for a LiveData or OPC data server.
- Changing the **On Control** scan rate for a LiveData or OPC data server.
- Changing the **Off Control** scan rate for a LiveData or OPC data server.
- Changing the **Default Item Access Path** for an OPC data server.
- Changing the **Location** from either **Remote** or **Local** for an OPC data server.
- Changing the defined name (either Universal Naming Convention (UNC) name or Domain Names System (DNS) name) of a **Remote** OPC data server.

See also

[Update enumerations and enumeration sets in an area model](#) on [page 108](#)

[Update phases and phase classes in an area model](#) on [page 116](#)

[Update process cells and process cell classes in an area model](#) on [page 125](#)

[Update resources and resource classes in an area model](#) on [page 126](#)

Update enumerations in area model

Enumerations are system objects that are defined as a variable data type consisting of a numeric value (called an ordinal value) and an associated text string. Enumeration sets contain groupings of related enumerations. An area model can contain customer enumerations and enumeration sets.

Permitted changes

You can make the following changes without impacting the operation of the area model because these changes are not going to interfere with process operation:

- Creating an enumeration set.

Permitted with conditions changes

Some changes to an area model have to be assessed before they can be permitted. If the object being modified is required for a running process it cannot be changed until that process is complete.

Review this list for an overview of the edits that require additional planning before deployment:



Tip: Text preceded by % are placeholders for variables. In your environment the message will include the appropriate information from your area model.

- Deleting an enumeration set.

An enumeration set cannot be deleted if it is referenced by the recipe document parameters or reports in batches on the batch list. If you attempt to delete an enumeration set that has references, this error is recorded in the Compare/Update log:

```
"Enumeration Set '%s' cannot be deleted. Enumeration Set is referenced in Batch '%s' (CreateID:%ld) by Recipe Parameter(s) and/or Recipe Report(s) in recipe document %s."
```

- Modifying an enumeration set.

Before modifying an enumeration set make sure there are no references to the enumeration set from the recipe document parameters or reports in batches on the batch list. If the enumeration set is referenced, this error is recorded in the Compare/Update log:

```
"Enumeration Set '%s' cannot be modified. Enumeration Set is referenced in Batch '%s' (CreateID:%ld) by Recipe Parameter(s) and/or Recipe Report(s) in recipe document %s".
```

Not permitted changes

- Modifying the ordinal value of an enumeration.
- Deleting an enumeration.

See also

[Update operation sequences and operation sequence classes in an area model](#) on [page 109](#)

[Update phases and phase classes in an area model](#) on [page 116](#)

[Update process cells and process cell classes in an area model](#) on [page 125](#)

[Update resources and resource classes in an area model](#) on [page 126](#)

Update operation sequences in area model

Within the area model, an operation sequence class is a definition from which instances of that sequence program are added to a unit.

An operation sequence instance can be added to a unit either by creating a new operation sequence instance or by sharing an existing operation sequence instance already contained within other units. Only one instance of a specific operation sequence can be added to each unit. When an operation sequence is required by more than one unit, it must be shared by all units that require it. Prior to sharing a phase or operation sequence, verify the phase or operation sequence is physically able to be shared.

An operation sequence instance can be removed from a unit either by deleting an operation sequence instance or by unsharing an operation sequence instance that was shared with other units.

Permitted changes

You can make the following changes without impacting the operation of the area model because these changes are not going to interfere with process operation:

- Changing the **Default Controller Program Name** on an operation sequence class.
- Adding a parameter tag (controlled by Parameter/Tag locking on class).
- Deleting a parameter tag (controlled by Parameter/Tag locking on class).
- Adding a report tag (controlled by Parameter/Tag locking on class).
- Deleting a report tag (controlled by Parameter/Tag locking on class).
- Containing Units (this property is generally changed by sharing/unsharing the operation sequence between units).

Permitted with conditions changes

Some changes to an area model have to be assessed before they can be permitted. If the object being modified is required for a running process it cannot be changed until that process is complete.

Review this list for an overview of the edits that require additional planning before deployment:



Tip: Text preceded by % are placeholders for variables. In your environment the message will include the appropriate information from your area model.

- Adding an operation sequence instance to a unit

If adding the operation sequence instance would result in zero legal bind choices for one or more unbound class-based unit requirements within batches, then the operation sequence instance cannot be added.

This error is recorded in the Compare/Update log:

```
"Area Model Changes not permitted. Batch '%s' (Create ID:%ld) has a Unit Requirement '%s' for which no legal bind targets would exist."
```

- Deleting an operation sequence instance from a unit

An operation sequence instance cannot be deleted from a unit in these situations:

- If the operation sequence instance is referenced by an operation sequence step bound to the unit inside an instance-based recipe or by a bound operation sequence step bound to the unit inside a class-based recipe. This error is recorded in Compare/Update log:


```
"Operation '%s' cannot be removed from Unit '%s'. Batch '%s' (Create ID:%ld) affected due to step '%s' being bound to removed Operation."
```
- If the operation sequence instance is being used for manual operation control within the unit. This error is recorded in Compare/Update log:


```
"Operation '%s' cannot be removed from Unit '%s'. Operation is in use by Manual Operation Control Batch '%s' (CreateID:%ld)."
```
- If deleting the operation sequence instance from the unit will make the operation class non-common to a unit class and class-based unit requirements exist in one or more batches that do not have a "Require Operation" bind requirement for the affected operation sequence class. This error is recorded in Compare/Update log:


```
"Operation '%s' cannot be removed from Unit '%s'. Batch '%s' (CreateID:%ld) affected by Operation Class '%s' becoming non-common to Unit Class '%s'."
```
- If deleting the operation sequence instance from the unit results in one or more unbound unit requirements having their set of legal bind targets reduced to zero. This error is recorded in Compare/Update log:


```
"Area Model Changes not permitted. Batch '%s' (Create ID:%ld) has a Unit Requirement '%s' for which no legal bind targets would exist."
```
- Modifying the controller program name of an operation sequence instance.

The controller program name of an operation sequence instance cannot be modified in these situations:

- If the instance is being used by any active batch steps inside of batches. This error is recorded in the Compare/Update log:


```
"Operation '%s' cannot have the controller program name property modified. Batch '%s' (CreateID:%ld) affected due to active step '%s' bound to modified Operation."
```
- If the instance is being used for manual phase control. This error is recorded in the Compare/Update log:


```
"Operation '%s' cannot have the controller program name property modified. Operation is in use by Manual Phase Control Batch '%s' (CreateID:%ld)."
```
- Unsharing a operating sequence from a unit

An operation sequence cannot be unshared from a unit in these situations:

- If the operation sequence instance is referenced by an operation sequence step bound to the unit inside an instance-based recipe or by a bound operation sequence step bound to the unit inside a class-based recipe. This error is recorded in Compare/Update log:

```
"Operation '%s' cannot be removed from Unit '%s'. Batch '%s' (Create ID:%ld) affected due to step '%s' being bound to removed Operation."
```
- If the operation sequence instance is being used for manual operation control within the unit. This error is recorded in Compare/Update log:

```
"Operation '%s' cannot be removed from Unit '%s'. Operation is in use by Manual Operation Control Batch '%s' (CreateID:%ld)."
```
- If unsharing the operation sequence instance from the unit will make the operation class non-common to a unit class and class-based unit requirements exist in one or more batches that do not have a "Require Operation" bind requirement for the affected operation sequence class. This error is recorded in Compare/Update log:

```
"Operation '%s' cannot be removed from Unit '%s'. Batch '%s' (CreateID:%ld) affected by Operation Class '%s' becoming non-common to Unit Class '%s'."
```
- If unsharing the operation sequence instance from the unit results in one or more unbound Unit Requirements having their set of legal bind targets reduced to zero. This error is recorded in Compare/Update log:

```
"Area Model Changes not permitted. Batch '%s' (Create ID:%ld) has a Unit Requirement '%s' for which no legal bind targets would exist."
```
- Creating a parameter within the operation sequence class.
 A parameter cannot be created in these situations:
 - If an active phase step within a batch references the operation sequence class to which the parameter would be added. This error is recorded in Compare/Update log:

```
"Operation Class '%s' cannot have parameter(s) added. Batch '%s' (Create ID:%ld) affected due to active step '%s' referencing Operation Class."
```
 - If manual operation control is being performed on the instance of the operation sequence class to which the parameter would be added. This error is recorded in Compare/Update log:

```
"Operation Class '%s' cannot have parameter(s) added. Manual Operation Control Batch '%s' (CreateID:%ld) is using an instance of the Operation Class."
```
 - If creation of "Operator Prompt" values on steps referencing the operation sequence class inside of scaled batches on the batch list

would cause invalid scaling. This error is recorded in

Compare/Update log:

```
"Scaling issue for Parameter addition in Batch '%s'
(Create ID:%ld):%s"
```

- Enabling or disabling **Automatically Download Parameters** on **START**.

If any active steps are found referencing the modified operation sequence instance then the automatic download of parameters upon START cannot be enabled or disabled. This error is recorded in

Compare/Update log:

```
Operation '%s' cannot have automatic upload/download
properties modified. Batch '%s' (Create ID:%ld) affected
due to active step '%s' bound to modified Operation.
```

If the modified operation sequence instance is being used for Manual Operation Control then the automatic download of parameters upon START cannot be enabled or disabled. This error is recorded in

Compare/Update log:

```
Operation '%s' cannot have automatic upload/download
properties modified. Operation is in use by Manual
Operation Control Batch '%s' (CreateID:%ld).
```

- Enabling or disabling **Automatically Upload Parameters** upon **COMPLETE**.

If any active steps are found referencing the modified operation sequence instance then the automatic upload of parameters upon COMPLETE cannot be enabled or disabled. This error is recorded in

Compare/Update log:

```
Operation '%s' cannot have automatic upload/download
properties modified. Batch '%s' (Create ID:%ld) affected
due to active step '%s' bound to modified Operation.
```

If the modified operation sequence instance is being used for Manual Operation Control then the automatic upload of parameters upon COMPLETE cannot be enabled or disabled. This error is recorded in

Compare/Update log:

```
Operation '%s' cannot have automatic upload/download
properties modified. Operation is in use by Manual
Operation Control Batch '%s' (CreateID:%ld).
```

- Enabling or disabling **Automatically Upload Parameters** upon **STOPPED**.

If any active steps are found referencing the modified operation sequence instance then the automatic upload of parameters upon STOPPED cannot be enabled or disabled. This error is recorded in

Compare/Update log:

```
Operation '%s' cannot have automatic upload/download
properties modified. Batch '%s' (Create ID:%ld) affected
due to active step '%s' bound to modified Operation.
```

If the modified operation sequence instance is being used for Manual Operation Control then the automatic upload of parameters upon

STOPPED cannot be enabled or disabled. This error is recorded in

Compare/Update log:

Operation '%s' cannot have automatic upload/download properties modified. Operation is in use by Manual Operation Control Batch '%s' (CreateID:%ld).

- Enabling or disabling **Automatically Upload Parameters** upon **ABORTED**.

If any active steps are found referencing the modified operation sequence instance then the automatic upload of parameters upon ABORTED cannot be enabled or disabled. This error is recorded in

Compare/Update log:

Operation '%s' cannot have automatic upload/download properties modified. Batch '%s' (Create ID:%ld) affected due to active step '%s' bound to modified Operation.

If the modified operation sequence instance is being used for Manual Operation Control then then the automatic upload of parameters upon ABORTED cannot be enabled or disabled. This error is recorded in

Compare/Update log:

Operation '%s' cannot have automatic upload/download properties modified. Operation is in use by Manual Operation Control Batch '%s' (CreateID:%ld).

- Deleting a parameter within the operation sequence class.

A parameter cannot be deleted in these situations:

- If the operation sequence class is referenced by one or more operation steps within batches. This error is recorded in Compare/Update log:

"Operation Class '%s' cannot have parameter(s) deleted. Batch '%s' (Create ID:%ld) affected due to step '%s' referencing Operation Class."

- If manual operation control is being performed on the instance of the operation sequence class from which the parameter would be deleted. This error is recorded in Compare/Update log:

"Operation Class '%s' cannot have parameter(s) deleted. Manual Operation Control Batch '%s' (CreateID:%ld) is using an instance of the Operation Class."

- Creating an operation sequence class report.

An operation sequence class report cannot be created in these situations:

- If an active operation step within a batch references the operation sequence class to which the report would be added. This error is recorded in Compare/Update log:

"Operation Class '%s' cannot have report(s) added. Batch '%s' (Create ID:%ld) affected due to active step '%s' referencing Operation Class."

- If manual operation control is being performed on an instance of the operation sequence class to which the report would be added. This error is recorded in Compare/Update log:

```
"Operation Class '%s' cannot have report(s) added. Manual Operation Control Batch '%s' (CreateID:%ld) is using an instance of the Operation Class."
```
- Deleting an operation sequence class report.
An operation sequence class report cannot be deleted in these situations:
 - If an operation step within a batch references the operation sequence class from which the report would be removed. This error is recorded in Compare/Update log:

```
"Operation Class '%s' cannot have report(s) deleted. Batch '%s' (Create ID:%ld) affected due to active step '%s' referencing Operation Class."
```
 - If manual operation control is being performed on an instance of the operation sequence class from which the report would be removed. This error is recorded in Compare/Update log:

```
"Operation Class '%s' cannot have report(s) deleted. Manual Operation Control Batch '%s' (CreateID:%ld) is using an instance of the Operation Class."
```
- Modifying the **Equipment Allocation By Resource** (Resources Needed) property of an operation sequence.
If the operation sequence is owned or requested by any entity, then the equipment allocation cannot be changed in the area model. This error is recorded in Compare/Update log:

```
''Resources Needed' property of Operation '%s' cannot be modified. %s is currently owned and/or requested."
```

Not permitted changes

These edits are not permitted when the area model is running:

- Changing the bitmap (icon) assigned to the operation sequence class.
- Modifying these operation sequence properties:
 - Changing the **Equipment ID**.
 - Changing the **Operation Class**.
 - Changing the **Data Server**.

See also

[Update phases and phase classes in an area model](#) on [page 116](#)

[Update process cells and process cell classes in an area model](#) on [page 125](#)

[Update resources and resource classes in an area model](#) on [page 126](#)

Phases and classes in area model

A phase class is a reusable, process-oriented function. A phase class is usually a general representation of a group of equipment that performs a minor processing activity or a sub-function for a unit but does not describe the details of how that activity is accomplished. Examples of phase classes include Heat, Agitate, Add, and Transfer.

Phase classes may have associated parameters and reports. Each phase class contains attributes common to all phases of that specific phase class type. Phase classes are used to create recipes in FactoryTalk Batch Recipe Editor.

An instance of a phase class is a phase. Every phase inherits attributes of the phase class from which it is created. Phase classes and phases link the engineered logic in the process-connected device (PCD) to the recipe. A phase class describes the recipe phase, while the phase maps to the equipment operation or equipment phase in the engineered logic. Material-enabled phases include a list of containers and storage locations to which the phase connects.

Phase instances can be added to a unit either by creating a phase instance or by sharing an existing phase instance already contained within other units.

Phase instances can be removed from a unit either by deleting a phase instance or by unsharing a phase instance that was shared with other units.

Permitted changes

You can make the following changes without impacting the operation of the area model because these changes are not going to interfere with process operation:

- Creating a phase class
- Changing the **Default Controller Program Name** property for the phase class.
- Changing the **Number of Report Tags** for the phase class.
- Changing the **Number of Parameter Tags** for the phase class.
- Changing the default **Control Strategy** for the phase class.
- Creating a **Control Strategy** for the phase class.

Changing these items will be detected when you verify and deploy the area model and the difference is recorded in the log file.

Permitted with conditions changes

Some changes to an area model have to be assessed before they can be permitted. If the object being modified is required for a running process it cannot be changed until that process is complete.

Review this list for an overview of the edits that require additional planning before deployment:



Tip: Text preceded by % is a placeholder for variables. In your environment, the message will include the appropriate information from your area model.

- **Deleting a phase class**

A phase class cannot be deleted in these situations:

- If there is a manual phase control batch using an instance of the phase class to be deleted. This error is recorded in the Compare/Update log:

```
"Phase Class '%s' cannot be deleted. Manual Phase Control Batch '%s' (CreateID:%ld) is using an instance of the Phase Class."
```
- If the phase class to be deleted is referenced by any phase step in a batch. This error is recorded in the Compare/Update log:

```
"Phase Class '%s' cannot be deleted. Batch '%s' (Create ID:%ld) affected due to step '%s' referencing Phase Class."
```
- If the phase class to be deleted is referenced by a unit attribute binding requirement within a batch. This error is recorded in the Compare/Update log:

```
"Phase Class '%s' cannot be deleted. Batch '%s' (Create ID:%ld) contains a Unit Requirement '%s' with a Binding Requirement or Preference referencing Phase Class."
```

- **Creating or sharing a phase**

A phase cannot be created or shared if it would result in zero legal bind choices for one or more unbound class-based unit requirements within batches. This error is recorded in the Compare/Update log:

```
"Area Model Changes not permitted. Batch '%s' (Create ID:%ld) has a Unit Requirement '%s' for which no legal bind targets would exist."
```

- **Deleting a phase**

A phase cannot be deleted in these situations:

- If deletion of the phase would result in zero legal bind choices for one or more unbound class-based unit requirements within batches. This error is recorded in the Compare/Update log:

```
"Area Model Changes not permitted. Batch '%s' (Create ID:%ld) has a Unit Requirement '%s' for which no legal bind targets would exist."
```
- If the phase is owned or requested by any entity other than the operator. This error is recorded in the Compare/Update log:

```
"Phase '%s' cannot be deleted. %s is owned or requested by one or more non-Operator entities."
```
- If the phase is bound to a phase step referencing a unit from which it was removed within a batch. This error is recorded in the Compare/Update log:

```
"Phase '%s' cannot be removed from Unit '%s'. Batch '%s' (Create ID:%ld) affected due to step '%s' being bound to removed phase."
```

- If the phase is in use by a manual phase control batch within a unit from which the phase was removed. This error is recorded in the Compare/Update log:

```
"Phase '%s' cannot be removed from Unit '%s'. Phase is in use by Manual Phase Control Batch '%s' (CreateID:%ld)."
```

- If deletion of the phase will make the phase's phase class non-common to a unit class and that unit is referenced by a class-based unit requirement without a "Require Phase" bind requirement specifying the affected phase class, then it is not permitted to delete the phase. This error is recorded in the Compare/Update log:

```
"Phase '%s' cannot be removed from Unit '%s'. Batch '%s' (CreateID:%ld) affected by Phase Class '%s' becoming non-common to Unit Class '%s'."
```

- Unsharing a phase

A phase cannot be unshared in these situations:

- If unsharing of the phase would result in zero legal bind choices for one or more unbound class-based unit requirements within batches. This error is recorded in the Compare/Update log:

```
"Area Model Changes not permitted. Batch '%s' (Create ID:%ld) has a Unit Requirement '%s' for which no legal bind targets would exist."
```

- If the phase is bound to a phase step referencing a unit from which it was removed within a batch. This error is recorded in the Compare/Update log:

```
"Phase '%s' cannot be removed from Unit '%s'. Batch '%s' (Create ID:%ld) affected due to step '%s' being bound to removed phase."
```

- If the phase is in use by a manual phase control batch within a unit from which the phase was removed. This error is recorded in the Compare/Update log:

```
"Phase '%s' cannot be removed from Unit '%s'. Phase is in use by Manual Phase Control Batch '%s' (CreateID:%ld)."
```

- If unsharing of the phase will make the phase's phase class non-common to a unit class and that unit is referenced by a class-based unit requirement without a "Require Phase" bind requirement specifying the affected phase class, then it is not permitted to delete the phase. This error is recorded in the Compare/Update log:

```
"Phase '%s' cannot be removed from Unit '%s'. Batch '%s' (CreateID:%ld) affected by Phase Class '%s' becoming non-common to Unit Class '%s'."
```

- Creating a parameter for the phase class.

A parameter cannot be created in these situations:

- If the phase class is not control strategy enabled and an active phase step within a batch references the phase class. This error is recorded in the Compare/Update log:

```
"Phase Class '%s' cannot have parameter(s) added. Batch '%s' (Create ID:%ld) affected due to active step '%s' referencing Phase Class."
```
- If the phase class is not control strategy enabled and manual phase control is being performed on an instance of the phase class. This error is recorded in the Compare/Update log:

```
"Phase Class '%s' cannot have parameter(s) added. Manual Phase Control Batch '%s' (CreateID:%ld) is using an instance of the Phase Class."
```
- If the phase class is not control strategy enabled and creation of “Operator Prompt” values on steps referencing the phase class inside of scaled batches on the batch list would cause scaling issues, then creation of the parameter is not permitted. This error is recorded in the Compare/Update log:

```
"Scaling issue for Parameter addition in Batch '%s' (Create ID:%ld):%s"
```
- Deleting a parameter from the phase class.
A parameter cannot be deleted in these situations:
 - If the phase class is not control strategy enabled and is referenced by one or more phase steps within batches. This error is recorded in the Compare/Update log:

```
"Phase Class '%s' cannot have parameter(s) deleted. Batch '%s' (Create ID:%ld) affected due to step '%s' referencing Phase Class."
```
 - If the phase class is not control strategy enabled and manual phase control is being performed on an instance of the phase class. This error is recorded in the Compare/Update log:

```
"Phase Class '%s' cannot have parameter(s) deleted. Manual Phase Control Batch '%s' (CreateID:%ld) is using an instance of the Phase Class."
```
- Creating a report for the phase class.
A report cannot be created in these situations:
 - If the phase class is not control strategy enabled and an active phase step within a batch references the phase class. This error is recorded in the Compare/Update log:

```
"Phase Class '%s' cannot have report(s) added. Batch '%s' (Create ID:%ld) affected due to active step '%s' referencing Phase Class."
```
 - If the phase class is not control strategy enabled and manual phase control is being performed on an instance of the phase class. This error is recorded in the Compare/Update log:

```
"Phase Class '%s' cannot have report(s) added. Manual Phase Control Batch '%s' (CreateID:%ld) is using an instance of the Phase Class."
```

- Deleting a report from the phase class.

A report cannot be deleted in these situations:

- If the phase class is not control strategy enabled and a phase step within a batch references the phase class. This error is recorded in the Compare/Update log:

```
"%s Class '%s' cannot have report(s) deleted. Batch '%s' (Create ID:%ld) affected due to step '%s' referencing %s Class."
```
- If the phase class is not control strategy enabled and manual phase control is being performed on an instance of the phase class. This error is recorded in the Compare/Update log:

```
"Phase Class '%s' cannot have report(s) deleted. Manual Phase Control Batch '%s' (CreateID:%ld) is using an instance of the Phase Class."
```
- Modifying the **Equipment Allocation By Resource** (Resources Needed) property of a phase.

If the phase is owned or requested by any entity, then the equipment allocation cannot be changed in the area model. This error is recorded in the Compare/Update log:

```
"'Resources Needed' property of Phase '%s' cannot be modified. %s is currently owned and/or requested."
```

- Modifying the **Controller Program Name** property of a phase.

The property cannot be modified in these situations:

- If the phase is referenced by any active phase step in any batch. This error is recorded in the Compare/Update log:

```
"Phase '%s' cannot have the controller program name property modified. Batch '%s' (Create ID:%ld) affected due to active step '%s' bound to modified phase."
```
- If the phase is being used for Manual Phase Control. This error is recorded in the Compare/Update log:

```
Phase '%s' cannot have the controller program name property modified. Phase is in use by Manual Phase Control Batch '%s' (CreateID:%ld)."
```
- Changing the **Instruction File** property of eProcedure instruction-based phases.

The **Instruction File** property cannot be changed in these situations:

- If the submitted change results in setting the **Instruction File** property to nothing (empty string). This error is recorded in the Compare/Update log:

```
"eProcedure Phase '%s' cannot have instruction file property set empty."
```
- If the phase is referenced by any active phase step in any batch, then the change is not permitted.

```
"eProcedure Phase '%s' cannot have instruction file property modified. Batch '%s' (Create ID:%ld) affected due to active step '%s' bound to modified phase."
```

- If the phase is being used for Manual Phase Control. This error is recorded in the Compare/Update log:

```
"eProcedure Phase '%s' cannot have instruction file property modified. Phase is in use by Manual Phase Control Batch '%s' (CreateID:%ld)."
```

Not permitted changes

These edits are not permitted when the area model is online:

- Modifying these phase class properties:
 - Changing the bitmap (icon) assigned to the phase class.
 - Changing the **Number of Request Tags** for a phase class.
 - Changing the **Number of Message Partners** for a phase class.
 - Modifying the **Messages** configuration for a phase class.
 - Modifying the **Type** property for a phase class.
 - Enabling or disabling **Control Strategy** for a phase class.
 - Enabling or disabling **Parameter/Tag Locking** for a phase class.
- Modifying these phase properties:
 - Changing the **Equipment ID**.
 - Changing the **ERP Alias**.
 - Changing any **Cross Invocation Label**.
 - Changing the **Phase Class**.
 - Enabling or disabling **Automatically Download Parameters** on **START**.
 - Enabling or disabling **Automatically Upload Parameters** upon **COMPLETE**.
 - Enabling or disabling **Automatically Upload Parameters** upon **STOPPED**.
 - Enabling or disabling **Automatically Upload Parameters** upon **ABORTED**.
 - Deleting the **Container and Storage Location Associations** for a material-enabled phase.
 - Changing the feed type defined for a material-enabled phase.
 - Changing the **On Control Scan Rate** for the OPC or FactoryTalk Linx FactoryTalk Live Data server associated with the phase
 - Changing the **Off Control Scan Rate** for the OPC or FactoryTalk Linx FactoryTalk Live Data server associated with the phase
 - Changing the **Data Server**.

See also

[Update messages in an area model](#) on [page 122](#)

[Update control strategy in an area model](#) on [page 122](#)

Update messages in an area model

[Update parameters and reports in an area model](#) on [page 124](#)

Phase class messages are typically used for journal entry purposes and for troubleshooting phase logic problems. A phase class message has two pieces of data associated with it: the name and the message ID.

Not permitted changes

These edits are not permitted when the area model is running:

- Changing the **Name** of the message.
- Changing the **ID** of the message.

See also

[Update control strategy in an area model](#) on [page 122](#)

[Update parameters and reports in an area model](#) on [page 124](#)

[Update process cells and process cell classes in an area model](#) on [page 125](#)

[Update resources and resource classes in an area model](#) on [page 126](#)

[Phases and classes in area model](#) on [page 116](#)

Control strategy in an area model

Control strategies are user-defined groupings of phase class parameters and reports that can be associated with a single phase class. Organizing parameters and reports into separate control strategies allows phase classes to assume multiple roles without generating unnecessary phase I/O.

Permitted with conditions changes

Some changes to an area model have to be assessed before they can be permitted. If the object being modified is required for a running process it cannot be changed until that process is complete.

Review this list for an overview of the edits that require additional planning before deployment:



Tip: Text preceded by % are placeholders for variables. In your environment the message will include the appropriate information from your area model.

- Assigning a new parameter to the control strategy

A new parameter cannot be assigned to the control strategy in these situations:

- If an active phase step within a batch references the phase class and uses the control strategy. This error is recorded in the Compare/Update log:

"Control Strategy '%s' on Phase Class '%s' cannot have parameter(s) assigned. Batch '%s' (Create ID:%ld) affected due to active step '%s' referencing the Phase Class Control Strategy."

- If manual phase control is being performed on an instance of the phase class using the control strategy. This error is recorded in the Compare/Update log:

"Control Strategy '%s' on Phase Class '%s' cannot have parameter(s) assigned. Manual Phase Control Batch '%s' (CreateID:%ld) is using the Control Strategy on an instance of the Phase Class."

- If parameter includes "Operator Prompt" values on steps referencing the phase class using the control strategy inside of scaled batches on the batch list would cause scaling issue(s), then assignment of the Parameter is not permitted. This error is recorded in Compare/Update log:

"Scaling issue for Parameter assignment to Control Strategy '%s' addition in Batch '%s' (Create ID:%ld):%s"

- Removing a parameter from the control strategy

A parameter cannot be removed from the control strategy in these situations:

- If the phase class is referenced by one or more phase steps within batches using the control strategy. This error is recorded in Compare/Update log:

"Control Strategy '%s' on Phase Class '%s' cannot have parameter(s) deassigned. Batch '%s' (Create ID:%ld) affected due to step '%s' referencing the Phase Class Control Strategy."

- If manual phase control is being performed on an instance of the phase class. This error is recorded in Compare/Update log:

"Control Strategy '%s' on Phase Class '%s' cannot have parameter(s) deassigned. Manual Phase Control Batch '%s' (CreateID:%ld) is using the Control Strategy on an instance of the Phase Class."

- Adding a new report assignment

A new report cannot be assigned in these situations:

- If an active phase step within a batch references the phase class using the control strategy. This error is recorded in Compare/Update log:

"Control Strategy '%s' on Phase Class '%s' cannot have report(s) assigned. Batch '%s' (Create ID:%ld) affected due to active step '%s' referencing the Phase Class Control Strategy."

- If manual phase control is being performed on an instance of the phase class using the control strategy. This error is recorded in Compare/Update log:

"Control Strategy '%s' on Phase Class '%s' cannot have report(s) assigned. Manual Phase Control Batch '%s' (CreateID:%ld) is using the Control Strategy on an instance of the Phase Class."

- Removing a report assignment

A report cannot be removed in these situations:

- If a phase step within a batch references the phase class using the control strategy. This error is recorded in Compare/Update log:

"Control Strategy '%s' on Phase Class '%s' cannot have report(s) deassigned. Batch '%s' (Create ID:%ld) affected due to step '%s' referencing the Phase Class Control Strategy."
- If manual phase control is being performed on an instance of the phase class. This error is recorded in Compare/Update log:

"Control Strategy '%s' on Phase Class '%s' cannot have report(s) deassigned. Manual Phase Control Batch '%s' (CreateID:%ld) is using the Control Strategy on an instance of the Phase Class."

See also

[Update parameters and reports in an area model](#) on [page 124](#)

[Update process cells and process cell classes in an area model](#) on [page 125](#)

[Update resources and resource classes in an area model](#) on [page 126](#)

[Parameters and reports in area model](#) on [page 124](#)

Parameters and reports in area model

Parameters pass values from the FactoryTalk Batch Server to the process-connected device (PCD). The PCD uses reports for returning values to the server.

There are parameters for both phase classes and operation sequence classes.

Report values contain data that indicates how the instance of the phase or operation sequence class functioned. They are in the event journal file (.evt or .evtj) for the running batch and can be in FactoryTalk Batch reports.

Not permitted changes

These edits are not permitted when the area model is running:

- Modifying these parameter properties for an operation sequence or phase class:
 - Changing the **Type** (Datatype).
 - Changing the **Enum/E.U.** (Engineering Units).
 - Enabling or disabling scaling.
 - Changing the verification method.

- Changing the verification policy.
- Changing the defined **Reporting Contexts**.
- Changing the download parameter subsets - **Download (DL) on Start, Download (DL) on Transfer of Control (TOC)**.
- Modifying these report parameter properties for an operation sequence or phase class:
 - Changing the **Type** (Datatype)
 - Changing the **Enum/E.U.** (Engineering Units)
 - Changing the ERP Alias.
 - Changing the verification method.
 - Changing the verification policy.
 - Changing the defined **Reporting Contexts**.
 - Changing the upload parameter subsets - **Upload (UL) on Terminal State, Upload (UL) on Transfer of Control (TOC)**
 - Changing the **Accumulate** setting.

See also

[Update process cells and process cell classes in an area model](#) on [page 125](#)

[Update resources and resource classes in an area model](#) on [page 126](#)

[Update signature templates in an area model](#) on [page 128](#)

[Update units and unit classes in an area model](#) on [page 128](#)

Process cells and classes in area model

A process cell is a logical grouping of equipment, including units, required for the production of one or more batches. It can contain more than one grouping of equipment if more than one grouping is needed to make a batch. Process cells are grouped into classes. For each type of process cell in an area, a process cell class must first be created.

Permitted changes

You can make the following changes without impacting the operation of the area model because these changes are not going to interfere with process operation:

- Creating a process cell class.
- Deleting a process cell class.
- Adding a unit to a process cell.

Permitted with conditions changes

Some changes to an area model have to be assessed before they can be permitted. If the object being modified is required for a running process it cannot be changed until that process is complete.

Review this list for an overview of the edits that require additional planning before deployment:



Tip: Text preceded by % are placeholders for variables. In your environment the message will include the appropriate information from your area model.

- Modifying the **Equipment Allocation By Resource** (Resources Needed) property of a process cell.

If the process cell is owned or requested by any entity, then the equipment allocation cannot be changed in the area model. This error is recorded in the Compare/Update log:

```
"'Resources Needed' property of Process Cell '%s' cannot be modified. %s is currently owned and\\or requested."
```

- Deleting a process cell.

Before attempting to delete a process cell verify that it is not owned or requested by any entity other than the Operator. If it is, it cannot be deleted and this error is recorded in the Compare/Update log:

```
"Process cell '%s' cannot be deleted. %s is owned or requested by one or more non-Operator entities."
```

Not permitted changes

These edits are not permitted when the area model is online:

- Modifying these process cell properties:
 - Changing the **Equipment ID**.
 - Changing the number of **Max Owners**.
 - Changing the **ERP Alias**.
 - Changing any **Cross Invocation Label**.
 - Changing the **Class**.

See also

[Update resources and resource classes in an area model](#) on [page 126](#)

[Update signature templates in an area model](#) on [page 128](#)

[Update units and unit classes in an area model](#) on [page 128](#)

[Update unit attributes and tags in an area model](#) on [page 131](#)

Resources and classes in area model

Resource is a general term for equipment including process cells, units, phases and shared control modules. Resource classes (such as PUMPS) group together resource instances (such as PMP102) so that they can be shared

among equipment. After resource instances are defined in the area model, they can be assigned to process cells, units, phases, and other resources as necessary equipment that are allocated by the FactoryTalk Batch Server.

Permitted changes

You can make the following changes without impacting the operation of the area model because these changes are not going to interfere with process operation:

- Creating a resource class.
- Creating a resource.

Permitted with conditions changes

Some changes to an area model have to be assessed before they can be permitted. If the object being modified is required for a running process it cannot be changed until that process is complete.

Review this list for an overview of the edits that require additional planning before deployment:



Tip: Text preceded by % are placeholders for variables. In your environment the message will include the appropriate information from your area model.

- Modifying the **Equipment Allocation By Resource** (Resources Needed) property of a resource.

If the resource is owned or requested by any entity, then the equipment allocation cannot be changed in the area model. This error is recorded in the Compare/Update log:

```
"'Resources Needed' property of Resource '%s' cannot be modified. %s is currently owned and/or requested."
```

- Deleting a resource.

A resource cannot be deleted if the resource is owned or requested by any entity other than the Operator. This error is recorded in the Compare/Update log:

```
"Resource '%s' cannot be deleted. %s is owned or requested by one or more non-Operator entities."
```

Not permitted changes

- Modifying these resource properties:
 - Changing the **Equipment ID**.
 - Changing the number of **Max Owners**.
 - Changing the **Class**.
 - Changing any **Cross Invocation Label**.

See also

[Update signature templates in an area model](#) on [page 128](#)

[Update units and unit classes in an area model](#) on [page 128](#)

[Update unit attributes and tags in an area model](#) on [page 131](#)

Signature templates in area model

Signature template defines the number of signoffs required, the meaning of those signoffs, whether comments are included with the signoff, and which users or user groups have the required security permissions to complete the signoff for recipes and phases that require electronic signatures.

Permitted changes

You can make the following changes without impacting the operation of the area model because these changes are not going to interfere with process operation:

- Creating a signature template.
- Modifying these signature template properties:
 - Last Signoff.
 - Signoffs.
 - Adding or removing a signoff.
 - Adding or modifying the signoff meaning.
 - Adding or modifying signoff comments.
 - Adding or modifying security permissions.
- Deleting a signature template.

Not permitted changes

- Changing the template index.

See also

[Update signature templates in an area model](#) on [page 128](#)

[Update units and unit classes in an area model](#) on [page 128](#)

[Update unit attributes and tags in an area model](#) on [page 131](#)

[FactoryTalk Batch electronic signatures](#) on [page 37](#)

Units and classes in area model

A unit consists of phases, control modules, or both types of modules centered around a material-containing vessel that performs one or more major processing activities. Often process facilities contain multiple instances of

similar equipment that consist of the same or similar components and functionality. These groupings of similar equipment can be described as a unit class.

Permitted changes

You can make the following changes without impacting the operation of the area model because these changes are not going to interfere with process operation:

- Creating a unit class.
- Assigning a required unit attribute to a unit class.
- De-assigning a required unit attribute to a unit class
- Adding a phase to the unit.
- Adding an operation sequence to the unit.
- Adding process cells to the unit.
- Sharing or unsharing a unit between process cells.

Permitted with conditions changes

Some changes to an area model have to be assessed before they can be permitted. If the object being modified is required for a running process it cannot be changed until that process is complete.

Review this list for an overview of the edits that require additional planning before deployment:



Tip: Text preceded by % are placeholders for variables. In your environment the message will include the appropriate information from your area model.

- Deleting a unit class.

A unit class cannot be deleted in these situations:

- If the unit class to be deleted is referenced by a class-based unit requirement with a batch on the batch list. This error is recorded in the Compare/Update log:

```
"Unit Class '%s' cannot be deleted. Batch '%s' (Create ID:%ld) contains a class based Unit Requirement, '%s', that references the Unit Class."
```

- If the unit class to be deleted is referenced by an instance-based unit requirement within a batch on the batch list. This error is recorded in the Compare/Update log:

```
"Unit Class '%s' cannot be deleted. Batch '%s' (Create ID:%ld) contains an instance based Unit Requirement, '%s', that references a Unit of the Unit Class."
```

- Creating a unit.

If adding the unit causes a previously common phase class, unit class, or unit attribute of the parent unit class to no longer be common to all

instances of the class and there are one or more class-based UPC recipes in existing batches that reference the phase class, unit class, or unit attribute, then it is not permitted to add the unit class. This error is recorded in the Compare/Update log:

```
"Unit '%s' cannot be created. Associated Unit Class is referenced by one or more class based UPC steps inside of a BPC batch and new Unit would reduce the set of common phase classes and\\or common operation classes and\\or common unit attributes."
```

- Deleting a unit.

A unit cannot be deleted in these situations:

- If the unit is owned or requested by any entity other than the Operator. This error is recorded in the Compare/Update log:

```
"Unit '%s' cannot be deleted. %s is owned or requested by one or more non-Operator entities."
```

- If the Unit to be deleted is referenced by an instance-based Unit Requirement within a batch. This error is recorded in the Compare/Update log:

```
"Unit '%s' cannot be deleted. Batch '%s' (Create ID:%ld) contains a instance based Unit Requirement '%s' referencing the Unit to be deleted."
```

- If the Unit to be deleted is bound to by a class-based Unit Requirement within a batch. This error is recorded in the Compare/Update log:

```
"Unit '%s' cannot be deleted. Batch '%s' (Create ID:%ld) contains a class based Unit Requirement '%s' bound to the Unit to be deleted."
```

- If there is a Manual Phase Control Batch or Manual Operation Sequence Control Batch executing against the Unit to be deleted. This error is recorded in the Compare/Update log:

```
"Unit '%s' cannot be deleted. Manual Phase/Operation Control Batch '%s' (Create ID:%ld) is running within the Unit to be deleted."
```

- Modifying the **Equipment Allocation By Resource** (Resources Needed) property of a unit.

If the unit is owned or requested by any entity, then the equipment allocation cannot be changed in the area model. This error is recorded in the Compare/Update log:

```
"'Resources Needed' property of Unit '%s' cannot be modified. %s is currently owned and\\or requested."
```

- Modifying a downstream unit

If modification of the unit affects the flow path defined by the FactoryTalk Batch server during batch execution in such a way that there are no legal bind choices for one or more unbound class-based Unit Requirements within batches, then the unit cannot be changed in the area model. This error is recorded in the Compare/Update log:

```
"Area Model Changes not permitted. Batch '%s' (Create
ID:%ld) has a Unit Requirement '%s' for which no legal
bind targets would exist."
```

Not permitted changes

These edits are not permitted when the area model is online:

- Changing the bitmap (icon) assigned to the unit class.
- Removing a required unit attribute from a unit class.
- Modifying these unit properties:
 - Changing the **Equipment ID**.
 - Changing the **ERP Alias**.
 - Changing any **Cross Invocation Label**.
 - Changing the **Class**.
 - Changing the **Data Server**.

See also

[Update unit attributes and tags in an area model](#) on [page 131](#)

Update unit attributes and tags in an area model

The area model has two categories of tags:

- Tags associated with a unit.
- Tags associated with a phase.

Unit attribute tags are items in the area model database that represent a single piece of PCD information. Unit attribute tags link to a process tag such as a level, temperature, or flow process variable. Unit attribute tags allow actual process values in recipe transition conditions.

Phases that use either OPC or FactoryTalk LiveData for communication with the batch server use tag-based phases.

Permitted changes

You can make the following changes without impacting the operation of the area model because these changes are not going to interfere with process operation:

- Changing the data value of a static attribute tag.
- Changing the item name of a tag.
- Changing the access path of a tag.
- Changing the write item name of a tag.
- Changing the write access path of a tag.

Permitted with conditions changes

- Deleting a unit attribute.

A unit attribute cannot be deleted in these situations:

- If the Unit Attribute is referenced by a transition expression within a batch. This error is recorded in the Compare/Update log:


```
"Unit Attribute Tag Class '%s' cannot be deleted. Batch '%s' (Create ID:%ld) affected due to transition '%s' referencing Phase Class."
```
- If the Unit Attribute is referenced by a Binding Requirement or Binding Preference of a Unit Requirement within a batch. This error is recorded in the Compare/Update log:


```
"Unit Attribute Tag Class '%s' cannot be deleted. Batch '%s' (Create ID:%ld) contains a Unit Requirement '%s' with a Binding Requirement or Preference referencing Unit Attribute Tag Class."
```

- Creating a unit attribute tag

If creating the unit attribute tag, possibly in combination with other requested edits, would result in no legal bind choices for one or more unbound class-based Unit Requirements within batches, then the unit attribute cannot be created in the area model. This error is recorded in the Compare/Update log:

```
"Area Model Changes not permitted. Batch '%s' (Create ID:%ld) has a Unit Requirement '%s' for which no legal bind targets would exist."
```

- Deleting a unit attribute tag

A unit attribute tag cannot be deleted in these situations:

- If deleting the unit attribute tag, possibly in combination with other requested edits, would result in no legal bind choices for one or more unbound class-based Unit Requirements within batches. This error is recorded in the Compare/Update log:


```
"Area Model Changes not permitted. Batch '%s' (Create ID:%ld) has a Unit Requirement '%s' for which no legal bind targets would exist."
```
- If the unit attribute tag is referenced by a transition expression from inside of a bound recipe. This error is recorded in the Compare/Update log:


```
"Unit Attribute Tag '%s' cannot be deleted. Batch '%s' (Create ID:%ld) affected due to transition reference '%s'."
```
- If deleting the unit attribute tag would make the unit attribute no longer common to the unit's parent class and one or more unit requirements within batches reference the unit class and do not have a "Require Attribute" bind requirement.


```
"Unit Attribute Tag '%s' cannot be deleted. Batch '%s' (Create ID:%ld) affected by Tag Class '%s' becoming non-common to Unit Class '%s'."
```


Not permitted changes

These edits are not permitted when the area model is running:

- Modifying these unit attribute properties:
 - Changing the data type.
 - Changing the enumeration set.
 - Changing the **Description**.
 - Changing the engineering units.
- Modifying these tag properties:
 - Changing the tag class.
 - Changing the data type.
 - Changing the enumeration set.
 - Changing the engineering units.
 - Changing the data server.
 - Changing the server type (protocol).

See also

[Units and classes in area model](#) on [page 128](#)

Windows event log

The Windows Event Log service records application, system, and security information. View this information using the Event Viewer.



Tip: Supported Windows versions for FactoryTalk Batch are listed in the *FactoryTalk Batch Components Upgrade and Installation Guide*.

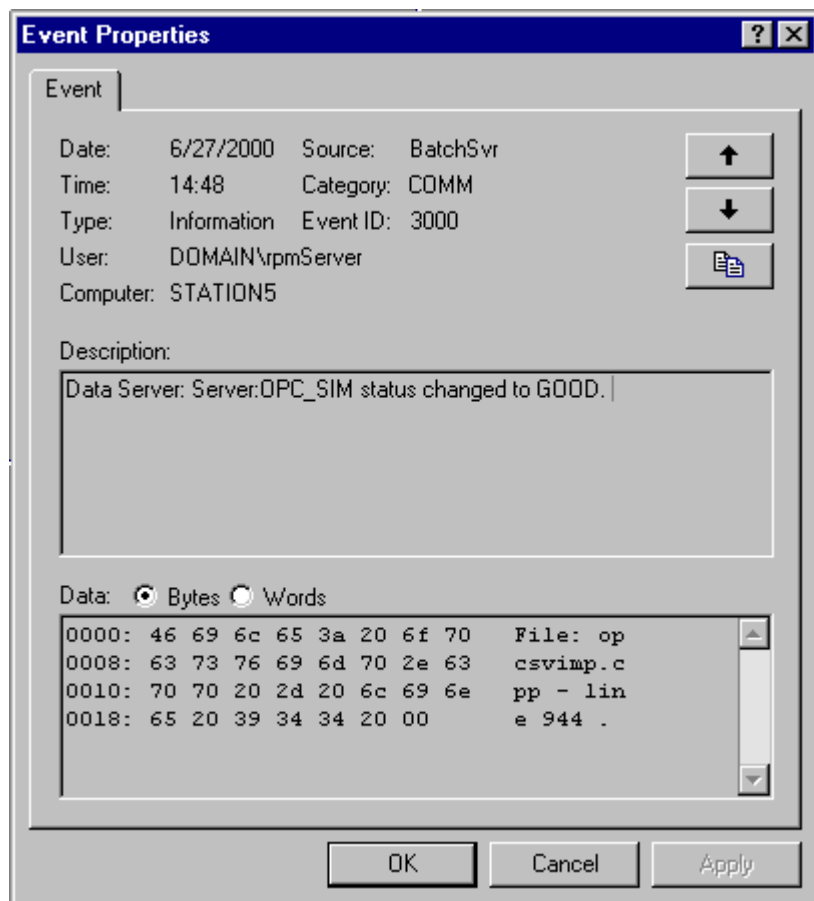
See also

[View event log entry properties](#) on [page 135](#)

[Modify event log settings](#) on [page 138](#)

View event log entry properties

From the Windows Event Viewer, double-click any log entry to display its properties.



There are 10 items for a Windows event log entry. Six of these are identical for all FactoryTalk Batch Server events. These items, along with a description, are

listed below. The Category, Event ID, and Description are described in the table below the list.

Item	Description
Date	Date that the event occurred.
Time	Time that the event occurred.
Type	Type of the event (that is, Information, Error, and so on).
User	Configured domain user.
Computer	Computer on which the FactoryTalk Batch Server is running.
Source	BatchSvr
Category	Category of the event.
Event ID	ID of the event.
Description	Description of the event.
Data	File and line number within the FactoryTalk Batch Server source code that generated the entry.

Category	Event ID	Description
Start	1000	Server demand started successfully in demo mode.
Start	1001	Server demand started successfully in production mode.
Start	1002	Server auto started successfully.
Start	1003	Server failed to auto start. See the <i>Batchsvr.log</i> file for more information.
Start	1004	Server failed to demand start. See the <i>Batchsvr.log</i> file for more information.

Important: Do not edit the *Batchsvr.log* files. The *Batchsvr.log* contains critical information Technical Support must determine the cause of a FactoryTalk Batch Server failure.

Start	1005	Command line arguments are inconsistent.
Start	1006	Unexpected command line argument [Descriptive String].
Start	1007	Service handler not installed.
Start	1008	Not running as a service or <i>StartServiceCtrlDispatcher</i> failed.
Start	1009	Log directory does not exist. Validate ErrorLogDirectory key value in <i>Batchsvr.ini</i> .

Important: Do not edit the *Batchsvr.ini* file unless specifically instructed by Technical Support.

Start	1010	Unable to start due to access permissions on log directory specified by ErrorLogDirectory key value in <i>Batchsvr.ini</i> .
Start	1011	<i>Batchsvr.log</i> file could not be created/opened. Validate pathname in <i>Batchsvr.ini</i> .
Start	1012	Unable to start. Log file (<i>Batchsvr.log</i> or <i>Verify.log</i>) does not have write access permission.
Start	1013	Unable to start. Unable to access log file (<i>Batchsvr.log</i> or <i>Verify.log</i>).
Start	1014	Unable to start. <i>Verify.log</i> file does not have write access permission.
Start	1015	Unable to start due to missing <i>Batchsvr.ini</i> file.
Start	1016	Unable to start due to access permissions on <i>Batchsvr.ini</i> file.
Start	1017	Unable to start due to access problem with <i>Batchsvr.ini</i> file.
Start	1018	Access error on <i>Batchsvr.ini</i> detected before logging on as user specified in <i>Batchsvr.ini</i> .
Start	1019	Access error on <i>Batchsvr.ini</i> detected while running as user specified in <i>Batchsvr.ini</i> .
Start	1020	Access error on log file detected while running as user specified in <i>Batchsvr.ini</i> .
Start	1021	Access error on event files detected while running as user specified in <i>Batchsvr.ini</i> .
Start	1022	Unable to start because primary event file directory specified by EventDirectoryPath key in <i>Batchsvr.ini</i> is missing.
Start	1023	Unable to start because secondary event file directory specified by EventDirectoryPathSecondary key in <i>Batchsvr.ini</i> is missing.

Category	Event ID	Description
Start	1024	Unable to start because of access permissions on primary event file directory specified by EventDirectoryPath key in <i>Batchsvr.ini</i> .
Start	1025	Unable to start because of access permissions on secondary event file directory specified by EventDirectoryPathSecondary key in <i>Batchsvr.ini</i> .
Start	1026	Unable to start because of access permissions on <i>Archque.txt</i> file in primary event file directory.
Start	1027	Unable to start because of access permissions on <i>Archque.txt</i> file in secondary event file directory.
Start	1028	Unable to start because of access permissions on <i>Eventdir.txt</i> file in primary event file directory.
Start	1029	Unable to start because of access permissions on <i>Eventdir.txt</i> file in secondary event file directory.
Start	1030	Unable to start because RestartDirectoryPath key not specified in <i>Batchsvr.ini</i> .
Start	1031	Unable to start because RestartDirectoryPath specified in <i>Batchsvr.ini</i> missing.
Start	1032	Unable to start because of access permissions on RestartDirectoryPath specified in <i>Batchsvr.ini</i> .
Start	1033	Unable to start because of access permissions (security?) on RestartDirectoryPath specified in <i>Batchsvr.ini</i> .
Start	1034	Unable to start because RestartDirectoryPath2 specified in <i>Batchsvr.ini</i> missing.
Start	1035	Unable to start because of access permissions on RestartDirectoryPath2 specified in <i>Batchsvr.ini</i> .
Start	1036	Unable to start because of access permissions (security) on RestartDirectoryPath2 specified in <i>Batchsvr.ini</i> .
Start	1037	Server started by an HMI Controls client.
Start	1038	Domain user name is blank and must be initialized in the SECURITY section of the <i>Batchsvr.ini</i> . The Server will not have any network credentials, which means it cannot access anything off node (that is, data servers, event journal files, or recipes).
Start	1039	Internal Error. Call to ColnitalizeSecurity failed.
Start	1040	Internal Error. Call to RegisterClassObjects failed.
Start	1041	The SQL Server database cannot be opened.
Boot	2000	Server warm booting.
Boot	2001	Server cold booting.
Boot	2002	Server started.
Boot	2003	Server warm all booting.
Boot	3000	Data Server: [Descriptive String] status changed to GOOD.
Boot	3001	Data Server: [Descriptive String] status changed to SUSPECT.
Boot	3002	Data Server: [Descriptive String] status changed to LOST.
Boot	3003	Data Server: [Descriptive String] status changed to BAD.
Boot	3004	Data Server: [Descriptive String] status changed to an unknown conversation status.
Boot	3005	Data Server: [Descriptive String].
Shutdown	4000	Server shutting down.
Shutdown	4001	Server exception occurred. See the <i>Batchsvr.log</i> file for information.
Shutdown	4002	Server automatically shutting down from demo mode.
Tag_Verification	5000	Server tag verification process Started.
Tag_Verification	5001	Server tag verification process Aborted.
Tag_Verification	5002	Server tag verification completed, [Descriptive String1] of [Descriptive String2] tags verified successfully.

Category	Event ID	Description
Registration	6000	Server registered.
Registration	6001	Server unregistered.
Registration	6002	Server registered as a service.
Control	7000	Notify SCM: The service is stopping.
Control	7001	Notify SCM: Service stopped.
Control	7002	Notify SCM: The service is starting.
Control	7003	Notify SCM: The service is running.
Control	7004	SERVICE_CONTROL_STOP received from Service Control Manager.
Control	7005	Unexpected SERVICE_CONTROL_PAUSE received from Service Control Manager.
Control	7006	Unexpected SERVICE_CONTROL_CONTINUE received from Service Control Manager.
Control	7007	SERVICE_CONTROL_INTERROGATE received from Service Control Manager.
Control	7008	SERVICE_CONTROL_SHUTDOWN received from Service Control Manager.
Control	7009	Bad Service Request.
Category_Debug	8000	Command Line: [Descriptive String].
Category_Debug	8001	Argument received from SCM or BatchSCM: [Descriptive String].
Category_Debug	8002	HRESULT = [Descriptive String].
Category_Debug	8003	Thread [Descriptive String] started.
Category_Debug	8004	Exception in object [Descriptive String1] in method [Descriptive String2].
Category_Debug	8005	Service could not be marked for deletion by the SCM.
Category_Debug	8006	Debug: [Descriptive String].

See also

[Windows event log](#) on [page 135](#)

[Modify event log settings](#) on [page 138](#)

Modify event log settings

To prevent the system event logs from exceeding the maximum allowable size, configure the application and system log to overwrite events as needed.

To modify event log settings:

1. Navigate to **Control Panel > Administrative Tools > Event Viewer**.
2. Right-click **Application**, and then select **Properties**. The **Application Properties** dialog box opens.
3. In the **Log size** area, select **Overwrite events as needed**, and then select **OK**.
4. Repeat steps 2 and 3 for the **System**.
5. Exit the **Event Viewer**.

See also

[Windows event log](#) on [page 135](#)

[View event log entry properties](#) on [page 135](#)

Cross invocation

Cross Invocation is the means by which the HMI Controls pass data to, and activate, an automation server such as an HMI or Visual Basic program. The data passed is specific to an equipment resource selected from the FactoryTalk Batch Client applications.

FactoryTalk Batch provides a procedure-centric view of the batches running in the plant. The HMI provides an equipment-centric view of these batches. If the HMI is an automation server that supports the Cross Invocation interface, cross invocation can send equipment-specific information about a batch to your HMI. The HMI can then act based on the data received. For example, it may invoke a graphic specific to the selected equipment resource or information specific to an Equipment Module's running logic.

See also

[Cross invocation overview](#) on [page 141](#)

[Cross invocation string configuration](#) on [page 142](#)

[Cross invocation string escape sequences](#) on [page 143](#)

Cross invocation overview

Cross invocation is accomplished by way of user-defined and selectable cross invocation menu items. If configured for a selected equipment resource, the cross invocation menu items are visible and selectable by right-clicking on the running object in the FactoryTalk Batch client .

If a recipe step is not bound to an equipment resource the FactoryTalk Batch client does not display the **Cross Invocation** menu when you right-click on the step.

Within the FactoryTalk Batch client applications, there are a number of selectable procedural elements. Each of these procedural elements relates to a specific equipment resource. Each equipment resource may be configured to have cross invocation strings (defined on the **Cross Invocation** tab of any configurable equipment resource in the area model).

Procedural Element > Equipment Resource > Cross Invocation Strings

Each cross invocation string defines:

- The text for displaying the **Cross invocation** menu item.
- The data for passing to the automation server.

This table shows the various equipment resource for which a selected procedural element is linked. For cross invocation to be implemented, the cross invocation strings must be configured for the equipment resource.

Procedural elements	Corresponding equipment resource
Batch	Process Cell (Procedure)
	Unit (Unit Proc or Operation)
Batch	Process Cell (Procedure)
	Unit (Unit Proc or Operation)
Unit Procedure	Unit
Operation	Unit
Recipe Phase	Phase
Batch	Process Cell (Procedure)
	Unit (Unit Proc or Operation)
Unacknowledged Prompts	Phase
Process Cell	Process Cell
Unit	Unit
Phase	Phase
Recipe	Process Cell (Procedure)
	Unit (Unit Proc or Operation)
Process Cell	Process Cell
Unit	Unit
Phase	Phase
Resource	Resource
N/A	N/A
Phase	Phase

See also

[Cross invocation](#) on [page 141](#)

[Cross invocation string configuration](#) on [page 142](#)

Cross invocation string configuration

Use FactoryTalk Batch Equipment Editor to define up to five cross invocation strings for each resource defined within the area model. Each time a procedural element is selected from the HMI Controls, the cross invocation strings associated with the equipment resource corresponding to the selected element are retrieved from the FactoryTalk Batch Server and evaluated. The contents of these strings become the caption(s) for the cross invocation menu item(s) and optional submenu items as well as the context data that is passed to the automation server. The context data is encapsulated in escape sequences.

Cross invocation strings are defined according to the following syntax:

- A cross invocation string must be enclosed entirely in square brackets []. Anything outside of the square brackets is ignored by the Server.
- Within the square brackets, the initial text (to the first semicolon) indicates the menu item(s) caption(s).

- All text within the square brackets after the first semicolon defines the context data passed to the automation server when the menu item is selected.
- A single string can specify one cross invocation item with any number of subitems. The first comma-delimited portion of the cross invocation string identifies the menu item for the shortcut menu, and each subsequent substring identifies a subitem.
- The data passed in response to all subitems is identical. Subitems are optional.

Example:

[Menu Item, Subitem 1, Subitem 2; %P, %S, %R]

The cross invocation string portion that specifies the data to pass to the automation server when selecting a menu item is defined by escape sequences. The escape sequences are replaced by the context data they represent before the string is passed. The valid escape sequences are identified in this table along with their meaning.

Escape sequence	Replace by
%P	The path to the selected step. In some instances, this value can be empty. (See Context Data Responses for more information.) Examples: CreateID CreateID \t Procedure \t Unit Procedure \t Operation \t Phase
%N	Equipment resource name to which the selected control recipe corresponds.
%C	Class name from which the equipment resource corresponding to the step was created.
%T	Equipment resource type to which the selected control recipe corresponds.
%S	Computer on which the FactoryTalk Batch Server is executing.
%B	Batch ID of the selected control recipe.
%M	Menu item caption that was selected. If submenu items exist, they appear in the following order: the caption of the menu item appended with a colon, then the index number of the submenu item appended with a colon, and then the caption of the submenu item selected.
%R	Resource ID of the equipment resource to which the selected control recipe corresponds.

See also

[Cross invocation overview](#) on [page 141](#)

Cross invocation string escape sequences

The escape sequences used in the cross invocation strings correspond to the current selection the HMI Controls. The context data passed to the automation server is relative to the current selection.

The table describes each escape sequence relative to the type of procedural element selected:

Escape Sequence	Procedure	Unit Procedure	Operation	Phase
%P	Path	Path	Path	Path
%N	Corresponding Process Cell Name	Corresponding Unit name	Corresponding Unit name	Corresponding Phase name

%C	Corresponding Process Cell class name	Corresponding Unit Class name	Corresponding Unit Class name	Corresponding Phase Class name
%T	1 (Process Cell)	2 (Unit)	2 (Unit)	3 (Phase)
%S	Node	Node	Node	Node
%B	Batch ID	Batch ID	Batch ID	Batch ID
%M	Menu caption	Menu caption	Menu caption	Menu caption
%R	Corresponding Process Cell resource ID	Corresponding Unit resource ID	Corresponding Unit resource ID	Corresponding Phase resource ID

If the control recipe runs across multiple process cells, the context data replaced by %N and %C is the first process cell in the process cells list of associated with the recipe.

See also

[Cross invocation](#) on [page 141](#)

[Cross invocation string configuration](#) on [page 142](#)

The FactoryTalk Batch Phase Simulator

FactoryTalk Batch comes with a phase logic simulation program that allows you to run the FactoryTalk Batch software and simulate your batch process without being connected to a PCD. The FactoryTalk Batch Phase Simulator imitates the functionality of a data server and communicates with the FactoryTalk Batch Server using the OPC communication protocol. This is a powerful tool for testing, experimentation and demonstration purposes.



Tip: The Simulator is automatically started by the FactoryTalk Batch Server if required by the area model.

The Simulator allows you to save report parameter values.



Tip: When running a material-enabled phase, RUNNING logic should not have any configured requests. You must set the FEED_COMPLETE report parameter to 1 and set a value for the ACTUAL_AMOUNT report parameter. Save the simulation file after making these changes.

See also

[Start the FactoryTalk Batch Phase Simulator](#) on [page 145](#)

[Phase Configuration dialog box](#) on [page 146](#)

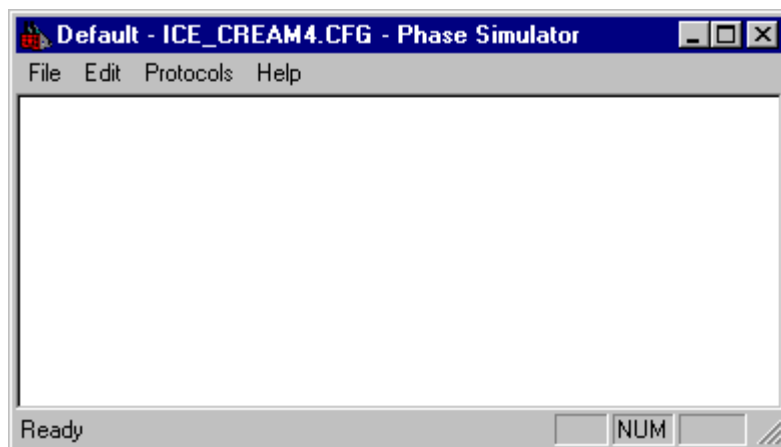
[Phase Status dialog box](#) on [page 149](#)

Start the FactoryTalk Batch Phase Simulator

Start the FactoryTalk Batch Phase Simulator for the FactoryTalk Batch Server and FactoryTalk Batch View to run in a simulated environment.

To start the FactoryTalk Batch Phase Simulator:

- Select **Start > Rockwell Software > Simulator**. The Phase Simulator opens.



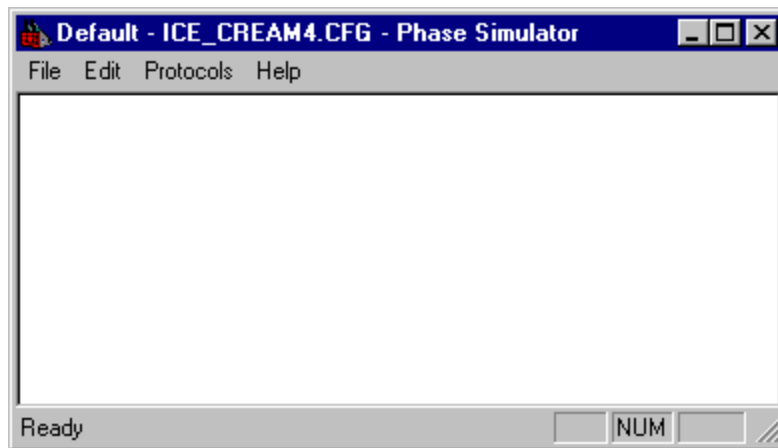
See also

[The FactoryTalk Batch Phase Simulator](#) on [page 145](#)

[FactoryTalk Batch Phase Simulator interface](#) on [page 146](#)

FactoryTalk Batch Phase Simulator interface

Upon startup, the FactoryTalk Batch Phase Simulator reads the area model that is specified in the Batchsvr.ini file. The area model file name is displayed on the title bar. By default, when you first install the software it is set up to read ice_cream1.cfg, one of the sample area models shipped with FactoryTalk Batch. This file contains the equipment required to run the Sampledemo1 recipe. The Simulator simulates the RUNNING, STOPPING, HOLDING, ABORTING, and RESTARTING states for all phases configured in the recipe.



Tip: The Simulator simulates only phase states for area models that have no more than 50 parameters and 50 report parameters.

The Simulator menu bar consists of these options:

Option	Definition
File	Open a simulator (.sim) file, save a file, or exit the Simulator.
Edit	Edit phase configuration, phase status, or unit tag status.
Protocols	Enable or disable the Watchdog and Command Handshake functions.
Help	Open the Simulator About box.

See also

[The FactoryTalk Batch Phase Simulator](#) on [page 145](#)

[Start the FactoryTalk Batch Phase Simulator](#) on [page 145](#)

Phase Configuration dialog box

The FactoryTalk Batch Phase Simulator allows you to edit a phase's configuration. You can set these items for each sequencing state of a phase:

- Dwell Time
Indicates the length of time, in seconds, that a specific state is active for the selected phase.
- Time of Request
Indicates the time, within the dwell period, at which a request is made.

- **Type of Request**
Indicates the type of request to make. (Refer to the FactoryTalk Batch PCD Programmer Technical Reference Guide for more information about requests.)
- **Request Parameters**
Each request can have up to three parameters that further describe the request.
- **Continuous Phase Indication**
A continuous phase is a phase whose phase logic does not transition to the COMPLETE state. You can simulate continuous phase logic by selecting Continuous Phase. When a transition that follows a step running continuous phase logic evaluates to TRUE, and the step is not involved in Transfer of Control, then the FactoryTalk Batch Server sends a STOP command to the phase. If the phase with continuous phase logic is involved in Transfer of Control, and the transition evaluates to TRUE, the Transfer of Control procedure is executed, transferring ownership of the phase from the step prior to the transition to the step following the transition. A NEW_PARAMETERS command is issued to the phase's phase logic when the Transfer of Control occurs, notifying the phase logic that new recipe parameter values associated with the new recipe step are available for download if the phase logic requests a download.



Tip: To use a continuous phase in an actual batch, write appropriate phase logic to define non-terminating behavior.

Refer to the NEW_PARAMETERS Command section in the FactoryTalk Batch PCD Programmer Technical Reference Guide for more information.

See also

[Change the dwell time](#) on [page 147](#)

[Change the sequence of requests](#) on [page 148](#)

[Indicate a continuous phase](#) on [page 148](#)

Change the dwell time

Change the length of time that a specific state is active for the selected phase.

To change the dwell time:

1. Start the **FactoryTalk Batch Phase Simulator**.
2. Select **Edit > Phase Configuration**. The **Phase Configuration** dialog box opens.
3. From the **Select Phase** list, select the appropriate phase.
4. From the **Select State** list, select the state to change the dwell time.
5. In **Dwell Time**, type the desired dwell time for the selected state.
6. Select **Update Phase**.

See also

[Phase Configuration dialog box](#) on [page 146](#)

[Change the sequence of requests](#) on [page 148](#)

[Indicate a continuous phase](#) on [page 148](#)

Change the sequence of requests

Change the time, type, and parameters of the request in a phase.

To change the sequence of requests:

1. Start the **FactoryTalk Batch Phase Simulator**.
2. Select **Edit > Phase Configuration**.
3. In the **Phase Configuration** dialog box, from the **Select Phase** list, select the appropriate phase.
4. Type an appropriate value in **@Time** (time is in seconds). This value determines when the request is processed. This value must not be greater than the **Dwell Time**.

For example, if the **Dwell Time** is 30 seconds, and the **@Time** value is 20 seconds, then this request will be processed after the selected state has been active for 20 seconds.
5. Type an appropriate request number in **_RQ**. (Refer to the *FactoryTalk Batch PCD Programmer Technical Reference Guide* for information on request numbers.)
6. Type any required request parameter data in **Qo1**, **Qo2**, and **Qo3**. These parameters are used to further define the request.
7. Select the **Update Phase** button.
8. Select **Close**.

See also

[Phase Configuration dialog box](#) on [page 146](#)

[Change the dwell time](#) on [page 147](#)

[Indicate a continuous phase](#) on [page 148](#)

Indicate a continuous phase Simulate continuous phase logic by selecting **Continuous Phase**.**To indicate a continuous phase:**

1. Start the **FactoryTalk Batch Phase Simulator**.
2. Select **Edit > Phase Configuration**. The **Phase Configuration** dialog box opens.
3. From the **Select Phase** list, select the appropriate phase.
4. If this is a continuous phase, select **Continuous Phase**.
5. Select **Update Phase**.

6. Select **Close**.

See also

[Phase Configuration dialog box](#) on [page 146](#)

[Change the dwell time](#) on [page 147](#)

[Change the sequence of requests](#) on [page 148](#)

Phase Status dialog box

The status of a phase can be changed during run time to test phase logic prior to implementation.

- View parameter tag values and associated parameter names
- View request tag values
- Set owner of phase to External
- Change the phase's state for phases under External Control
- Restore active requests for HELD/RESTARTED phases
- Change report parameter tag values
- Change value of the Failure bit
- Set phase to COMPLETE
- Acknowledge/Clear requests of phases under External control

See also

[View phase parameter tag values](#) on [page 149](#)

[View and update report parameter tag values](#) on [page 150](#)

[Change ownership of a phase](#) on [page 151](#)

[Command a phase](#) on [page 151](#)

[Acknowledge and clear a request](#) on [page 153](#)

View phase parameter tag values

The phase parameter tag values are read-only and cannot be modified in the Simulator. The FactoryTalk Batch Server writes these tags when a download request is processed.

To view phase parameter tag values:

1. Start the **FactoryTalk Batch Phase Simulator**.
2. Select **Edit > Phase Status**.
3. Select the appropriate phase from the **Select Phase** list.
4. In the **Arrays** list, select **Phase Parameter Tags**. Each parameter tag associated with this phase is displayed, along with its current value and type.

See also

[View and update report parameter tag values](#) on [page 150](#)

[Change the value of a unit tag](#) on [page 155](#)

[View request data tag values](#) on [page 150](#)

View and update report parameter tag values

The report parameter tag values can be modified and uploaded to the FactoryTalk Batch Server. The uploaded value(s) are reflected in the electronic batch record (.evt or .evtj files).

To view and update report parameter tag values:

1. Start the **FactoryTalk Batch Phase Simulator**.
2. Select **Edit > Phase Status**.
3. In the **Select Phase** list, select the appropriate phase.
4. In the **Arrays** list, select **Report Parameter Tags**. Each report parameter tag associated with this phase displays, along with its current value and type.
5. Enter a new value for the appropriate Report Parameter Tag.
6. Select **Update Report Data** to update the tag value. The report tag values are uploaded to the Batch Server when an upload request is processed.

See also

[View request data tag values](#) on [page 150](#)

[Change ownership of a phase](#) on [page 151](#)

[Command a phase](#) on [page 151](#)

[Acknowledge and clear a request](#) on [page 153](#)

[Restore and clear requests](#) on [page 153](#)

View request data tag values

The request data tag values are read-only. These may be set by the FactoryTalk Batch Phase Simulator when making a configured request or may be written to the FactoryTalk Batch Server when processing a phase logic Receive Message request.

To view request data tag values:

1. Start the **FactoryTalk Batch Phase Simulator**.
2. Select **Edit >Phase Status**.
3. Select the appropriate phase from the **Select Phase** list.

4. From the **Arrays** list, select **Request Data Tags**. Each request data tag associated with this phase is listed, along with its current value and type.

See also

[View and update report parameter tag values](#) on [page 150](#)

[Change ownership of a phase](#) on [page 151](#)

[Command a phase](#) on [page 151](#)

[Acknowledge and clear a request](#) on [page 153](#)

[Restore and clear requests](#) on [page 153](#)

Change ownership of a phase

The **Owner** button acts as a toggle, switching the value in the **Owner** register between zero and one. A zero represents Batch as the owner and a one represents an external owner.

To change ownership of a phase:

1. Start the **FactoryTalk Batch Phase Simulator**.
2. Select **Edit > Phase Status**.
3. From the **Select Phase** list, select the appropriate phase.
4. To set ownership to **External**, select **Owner** when the value in the **Owner** register is **0**.

To set ownership to **Batch**, select **Owner** when the value in the **Owner** register is **1**.

See also

[View and update report parameter tag values](#) on [page 150](#)

[View request data tag values](#) on [page 150](#)

[Command a phase](#) on [page 151](#)

[Acknowledge and clear a request](#) on [page 153](#)

[Restore and clear requests](#) on [page 153](#)

Command a phase

The FactoryTalk Batch Phase Simulator allows you to change the state of a phase.

To command a phase

1. Start the **FactoryTalk Batch Phase Simulator**.
2. In the **Edit** menu, select **Phase Status**.

3. In the **Select Phase** list, select the phase to command.
4. To set the phase to COMPLETE, select **Set Complete**.
5. For all other state change commands, set the **Owner** of the phase to External.
6. Select the appropriate state change command (**Start/Stop**, **Abort/Reset**, or **Hold/Restart**).

If the requested state change is legal, the server sends the command and the phase transitions to the appropriate state. The new state value is displayed in the **Status** register in the **Phase Status** dialog box.

- If the phase is running and **Abort/Reset** is selected, the phase goes to ABORTING. To transition the phase to the ABORTED state, select **Set Complete**.
- If a phase is running and **Hold/Restart** is selected, the phase goes to HOLDING. To transition the phase to the HELD state, select **Set Complete**.

See also

[View and update report parameter tag values](#) on [page 150](#)

[View request data tag values](#) on [page 150](#)

[Phase commands](#) on [page 152](#)

[Acknowledge and clear a request](#) on [page 153](#)

[Restore and clear requests](#) on [page 153](#)

Phase commands

The FactoryTalk Batch Phase Simulator allows you to change the state of a phase. An active phase can be commanded to COMPLETE without gaining phase ownership. All other state change commands require the phase be under external control. The **Start/Stop**, **Abort/Reset** and **Hold/Restart** buttons are modal, in that the command issued is determined based on the phase's active state at the time the button is clicked. Only legal state change commands are executed. For example, if a phase is RUNNING and the **Start/Stop** button is clicked, the STOP command is issued. START is not a legal command to a phase in the RUNNING state. (Refer to the *FactoryTalk Batch PCD Programmer Technical Reference Guide* for additional information on legal state transitions.)

This table shows the phase's transition when the state change command is received, along with the associated state value (##).

COMMAND	INITIAL STATE	FINAL STATE
COMPLETE		COMPLETE (70)
START		RUNNING (50)
STOP	STOPPING (30)	STOPPED (80)
ABORT	ABORTING (10)	ABORTED (90)
RESET		IDLE (100)

COMMAND	INITIAL STATE	FINAL STATE
HOLD	HOLDING (20)	HELD (60)
RESTART	RESTARTING (40)	RUNNING (50)

See also

[Phase Status dialog box](#) on [page 149](#)

[The FactoryTalk Batch Phase Simulator](#) on [page 145](#)

[Command a phase](#) on [page 151](#)

Acknowledge and clear a request

Requests must be acknowledged and then cleared to allow a phase to continue processing when running under External control. The **Ack/Clear Request** button must be clicked twice, first to acknowledge the request, then to clear the request.

To acknowledge and clear a request:

1. Start the **FactoryTalk Batch Phase Simulator**.
2. Select **Edit > Phase Status**.
3. In the **Select Phase** list, select the phase to command.
4. If the **Request** register contains a request, select **Ack/Clear Request** to acknowledge the request. The value in the **Request** register changes to **10**, indicating that the request confirmation was received from the Batch Server.
5. Select **Ack/Clear Request** to clear the request. Upon clearing the request, the value in the **Request** register changes to **0**, indicating that the request was cleared. The phase continues to process.

See also

[View and update report parameter tag values](#) on [page 150](#)

[View request data tag values](#) on [page 150](#)

[Command a phase](#) on [page 151](#)

[Restore and clear requests](#) on [page 153](#)

Restore and clear requests

Requests generated by a phase can be stored in the request buffer registers should the phase receive a Hold command. The requests can then be copied back to the request registers when the phase receives a Restart command. The **Restore Req** and **Clear Req** registers must both contain a **1** (one) to allow for the saving and restoring of requests for a HELD/RESTARTED phase.

To restore and clear requests:

1. Start the **FactoryTalk Batch Phase Simulator**.
2. Select **Edit > Phase Status**.
3. In the **Select Phase** list, select the phase to command.
4. In the **Restore Req** register, type a **1**.
5. In the **Clear Req** register, type a **1**.

See also

[View and update report parameter tag values](#) on [page 150](#)

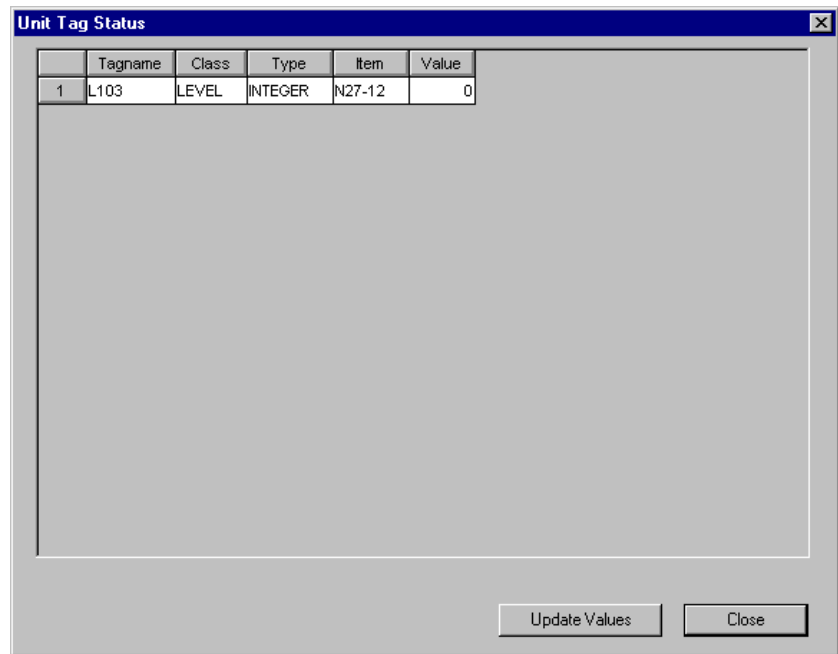
[View request data tag values](#) on [page 150](#)

[Command a phase](#) on [page 151](#)

[Acknowledge and clear a request](#) on [page 153](#)

Unit Tag Status dialog box

Values of unit tags configured in the area model can be changed during run time in the **Unit Tag Status** dialog box.



The **Unit Tag Status** dialog box shows:

Item	Definition
Tagname	A list of all configured unit tags for the active area model.
Class	The class associated with the unit tag.
Type	The type value associated with the unit tag.
Item	The item (addressable location) associated with the unit tag.
Value	The current value for the unit tag. This value can be modified and sent to the FactoryTalk Batch Server.
Update Values	Updates the tag values in the Simulator.

See also

[Change the value of a unit tag](#) on [page 155](#)

[The FactoryTalk Batch Phase Simulator](#) on [page 145](#)

Change the value of a unit tag

Change the value of a unit tag using FactoryTalk Batch Phase Simulator.

To change the value of a unit tag:

1. Start the **FactoryTalk Batch Phase Simulator**.
2. Select **Edit > Unit Tag Status**.
3. Make any required modifications to the Unit Tag Values, and select **Update Values**. The new values are written into the FactoryTalk Batch Phase Simulator unit tags.
4. Select **OK** to return to the FactoryTalk Batch Phase Simulator window.

See also

[Phase Status dialog box](#) on [page 149](#)

[View request data tag values](#) on [page 150](#)

[Acknowledge and clear a request](#) on [page 153](#)

Network Editor

The Network Editor is a utility that allows you to indicate where other FactoryTalk® Batch and/or FactoryTalk Batch Material Manager Servers are located on the network. This network configuration information simplifies integration with other FactoryTalk products and the process of reconfiguring a multi-computer system.

During installation, the network configuration information is completed based on your responses. If you have multiple servers in your network, use the Network Editor to update the location of all your servers.

The Network Editor is installed automatically with the FactoryTalk Batch and Material Servers.

Open the Network Editor

Open the Network Editor to configure your network.

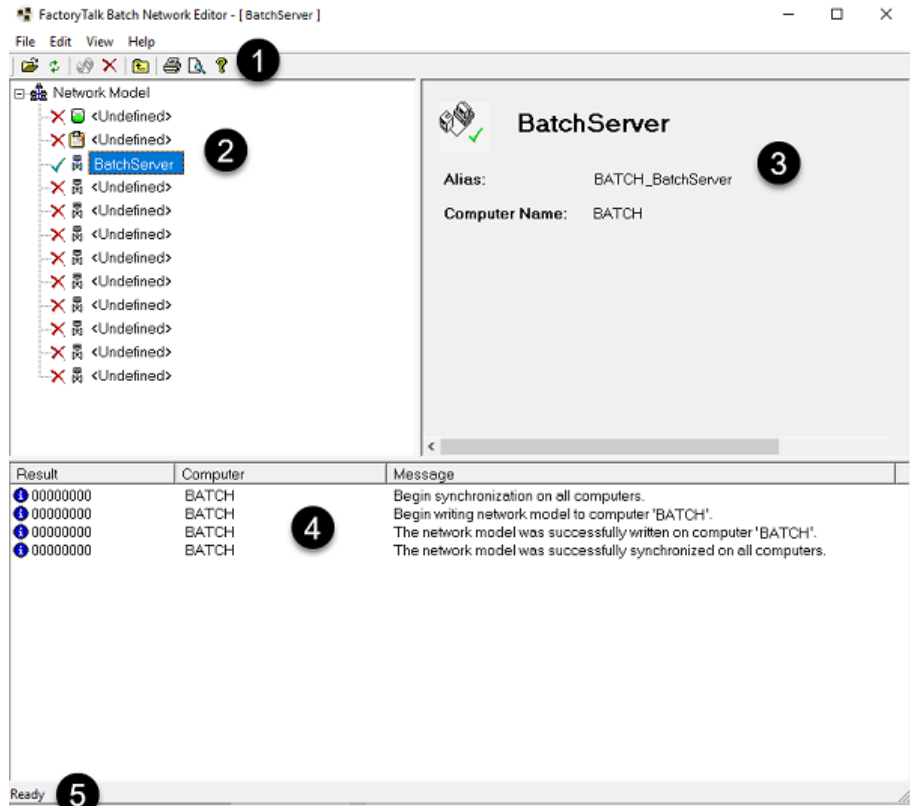
IMPORTANT Administrator privileges are required to perform this procedure.

To open the Network Editor:

- Select **Start > Rockwell Software > Network Editor**. The **Network Editor** window opens.

Network Editor interface

This illustration identifies the different areas of the Network Editor interface:



1. [Menu bar and toolbar](#)
2. [Hierarchy pane](#)
3. [Detail pane](#)
4. [Results pane](#)
5. [Status bar](#)

Hierarchy pane

The **Hierarchy** pane provides a hierarchical view of the application servers in your FactoryTalk Batch system. The Network Model is the parent item in the hierarchy list, with the Material Server and FactoryTalk Batch Servers listed as Network Model members.

The following indicators appear next to each server listed:

✓	Indicates the server is recognized on the network
✗	Indicates the server is not defined or recognized
?	Indicates there is a problem with the server or the server's configuration

Detail pane

The **Detail** pane provides information specific to the item selected in the Hierarchy pane.

Network model information

The following items are available when **Network Model** is selected in the Hierarchy pane.

Item	Description
System Name	Allows you to name your FactoryTalk Batch system. The default is Network Model.
Network Model from	Displays the name of the computer from which the FactoryTalk Batch system was last synchronized.
Time Stamp	Displays the time of the last synchronization.
User Name	Displays the Windows user ID of the person who performed the last synchronization.
Apply Changes	Saves changes made to the FactoryTalk Batch system information.
Undo Changes	Undoes changes made to the FactoryTalk Batch system information that are not applied.
Synchronize All Application Servers	Initiates the synchronization of the application servers.

Application server information

The following items are available when a server or **Undefined** is selected in the Hierarchy pane.

Item	Description
Alias	Displays a user-defined alias for the selected application server.
Computer Name	Displays the computer name for the selected server.
Configure	Allows you to specify the name of the material database when using FactoryTalk Batch Material Manager. The default database is MaterialBasedRecipe.

IMPORTANT The Network Editor does not verify database names. Changes made to the database name are immediate for the clients and servers. If the database name is invalid, the Material Editor will not open. Before changing the database name, check the spelling and synchronize the Network Editor. After changing the database name, confirm that the Material Editor opens and can read the database.




See also

[Network Editor interface](#) on [page 157](#)

Results pane

The **Results** pane provides information messages regarding the results of the last synchronization. Included in the list is an icon indicating the type of message listed, the name of the computer that performed the synchronization, and the message generated by the synchronization process.

The table below shows icons used in the messages and explains what types of messages they represent.

	Information	Represents an information message.
	Error	Indicates an error message.
	Warning	Indicates a warning message.

Menu bar and toolbar

The menu bar contains the menus for **File**, **Edit**, **View**, and **Help**.

- **File menu**

From the **File** menu, you can select a Network Model, synchronize all servers, print, and exit the Network Editor application.

- **Edit menu**









From the **Edit** menu, you can add or remove a server from the Network Editor.

- **View menu**

From the **View** menu, you can show or hide the toolbar/status bar, and refresh the Network Editor. You can also specify how data is viewed in the panes as well as setting other user preferences.

The Network Editor includes a toolbar which allows you to perform various functions. The buttons are enabled based on the item currently selected in the Network Editor. As you move your cursor over a button, a tool tip displays the button name and a detailed functional description displays in the status bar.







The Network Editor Toolbar buttons are:

	Select Network Model: Allows you to select a network model from another computer.
	Synchronize: Allows you to synchronize your application servers.
	Add Server: Allows you to add a previously undefined server.
	Remove Server: Allows you to remove a server.
	Up One Level: Moves you up one level in the Hierarchy pane.
	Print: Prints a report for the selected item.
	Print Preview: Allows you to view a report on the selected item.
	Help: Opens the Network Editor help.

Network Editor toolbar

The Network Editor includes a toolbar which allows you to perform various functions. The buttons are enabled based on the item currently selected in the Network Editor. As you move your cursor over a button, a tool tip displays the button name and a detailed functional description displays in the status bar.

The Network Editor Toolbar buttons are:

	Select Network Model: Allows you to select a network model from another computer.
	Synchronize: Allows you to synchronize your application servers.
	Add Server: Allows you to add a previously undefined server.
	Remove Server: Allows you to remove a server.
	Up One Level: Moves you up one level in the Hierarchy pane.
	Print: Prints a report for the selected item.



Print Preview: Allows you to view a report on the selected item.



Help: Opens the Network Editor help.

See also

[Network Editor interface](#) on [page 157](#)

Status bar

The status bar at the bottom of the window displays text messages on the left side. In addition, the status bar displays one or more icons on the right side, representing the status of the application and data.

The list of states that can be displayed in the status bar:

	Database: The Network Editor is accessing data.
	Server: The Network Editor is accessing other servers.
	Printing: A report is being printed or previewed.
	Processing: The Network Editor is processing data.
	Waiting: The Network Editor is busy.

Configure your network

The Network Editor allows you to configure your FactoryTalk Batch system. You can add or remove application servers, print reports, and synchronize your network model with another computer.

This section provides instructions for performing these functions.

- [Select a network model](#) on [page 161](#)
- [Add a server](#) on [page 162](#)
- [Remove a server](#) on [page 162](#)
- [Synchronize network models](#) on [page 162](#)

Select a network model

When you select a network model, you are retrieving an existing network model from another computer and using it as the network model for the local computer. You must do this if you add, remove, or rename a server in your system and need to update your network model with your existing system, or if you want to use the network model from another computer.

To select a network model

1. Open the **Network Editor**.
2. In the **Network Editor** toolbar, click the **Select Network Model** button.
3. Type the computer name in the "Enter the object name to select" area, or click the **Advanced** button to search for a computer.
4. Click **OK**.

Add a server

Use the Network Editor to add a previously undefined server.

To add a server

1. Open the **Network Editor**.
2. In the **Hierarchy** pane, select the type of server to add.
3. Click **Add Server**.
4. In the **Alias** box, type an alias name. This is the name that displays in the Network Editor.
5. In the **Computer Name** box, type the computer name, or use the browse button to locate and select the computer from the list of computers on your domain.
6. Click the **Synchronize** button.
7. Click **OK**.

Remove a server

Use the Network Editor to remove a server from your network model.

To remove a server

1. On the main FactoryTalk Batch server open the **Network Editor** and navigate to the root network model.
2. In the **Hierarchy** pane, select the server to remove.
3. Click the **Remove Server** button.
4. Click the **Synchronize** button. The servers listed in the model are updated with the new member server list.
The server you removed is not updated.
5. Log in to the server you removed and open Network Editor.
6. Repeat steps 2 through 4 to remove the server manually.

Synchronize network models

The synchronization process copies the network model data from your computer to each of the computers included in your FactoryTalk Batch system.

Click the **Synchronize** button. The Network Editor refreshes and the results of the synchronization process display in the Results pane.



Tip: It is a good idea to synchronize after adding or removing a server from the network model.

User Preferences dialog box

The **User Preferences** function allows you to customize the behavior and appearance of the various panes and views within the Network Editor.

You can turn on information tips, change the application text/background colors, configure the various views, and set up reports used for printing.

This section provides instructions for performing these functions:

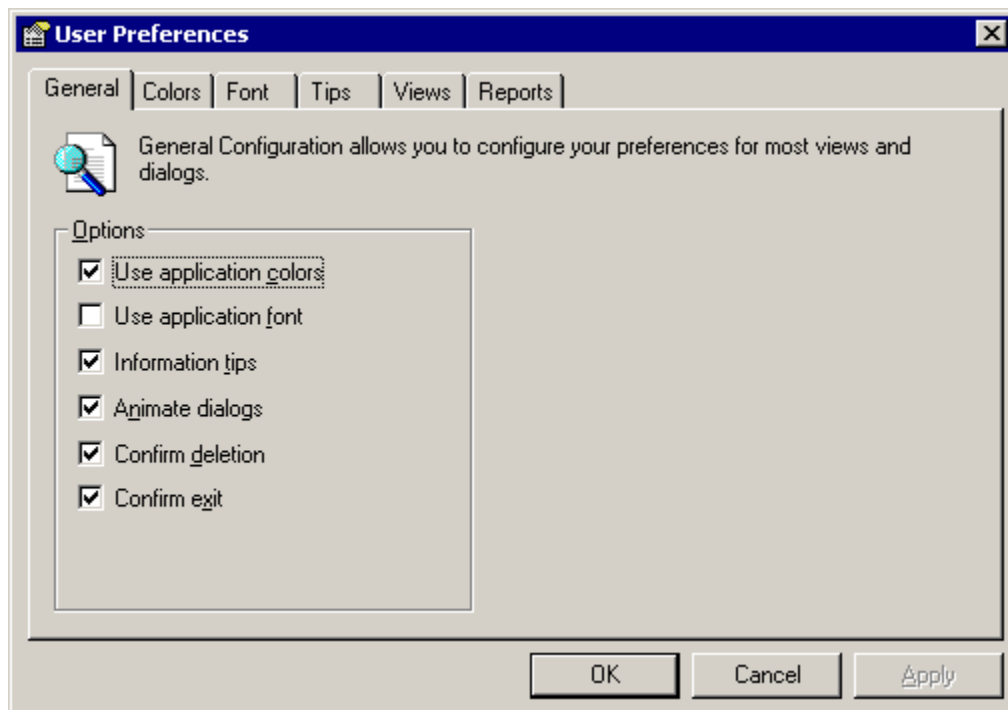
- [Edit colors](#) on [page 164](#)
- [Edit the application font](#) on [page 164](#)
- [Edit information tips](#) on [page 164](#)
- [Set report preferences](#) on [page 167](#)

See these topics for references to the **User Preferences** settings:

- [User Preferences dialog box - General tab](#) on [page 163](#)
- [User Preferences dialog box - Views tab](#) on [page 165](#)

User Preferences dialog box - General tab

The **General** tab is used to configure the basic behavior and appearance of the Network Editor views and dialog boxes. The selections on the **General** tab are used in conjunction with other sections of the **User Preferences** dialog box.



In the **Options** area, you have the following selections:

- **Use application colors:** Select to assign colors, other than the current Windows colors, to the background and text in the views.
- **Use application font:** Select to change the font, font style, and size from the default Windows style.
- **Information tips:** Select to enable single line tips that appear when you move the cursor over a button or field.
- **Animate dialogs:** Select to have dialog boxes fly out from the location from which they are opened.
- **Confirm deletion:** Select to display an "Are you sure?" message when you delete an item. Leave disabled to delete an item immediately without confirmation.
- **Confirm exit:** Select to have the application ask if you want to exit. Leave disabled to exit without a confirmation message.

Edit colors

The Network Editor defaults to the current Microsoft Windows color scheme. You can change the color of the text and background.

To edit colors

1. Open the **Network Editor**.
2. From the **View** menu, select **User Preferences**.
3. On the **General** tab in the **User Preferences** dialog box, select **Use application colors**.
4. Select the **Colors** tab.
5. Disable **Use Windows colors** to enable the **Background color** and **Text color** lists.
6. Select the desired colors from the appropriate list, and then click **Apply**.

Edit the application font

The Network Editor defaults to the current Windows fonts. You can change the display font.

To edit the application font

1. Open the **Network Editor**.
2. From the **View** menu, select **User Preferences**.
3. On the **General** tab in the **User Preferences** dialog box, select **Use application font**.
4. Select the **Font** tab.
5. Select the desired font, style, and size from the appropriate lists, and then click **OK**.
6. Click **Apply**.
7. Click **OK**.

Edit information tips

Information tips appear when you move the cursor over a button or field. You can change the font and colors of tips, and set tip timing, which determines how soon a tip displays.

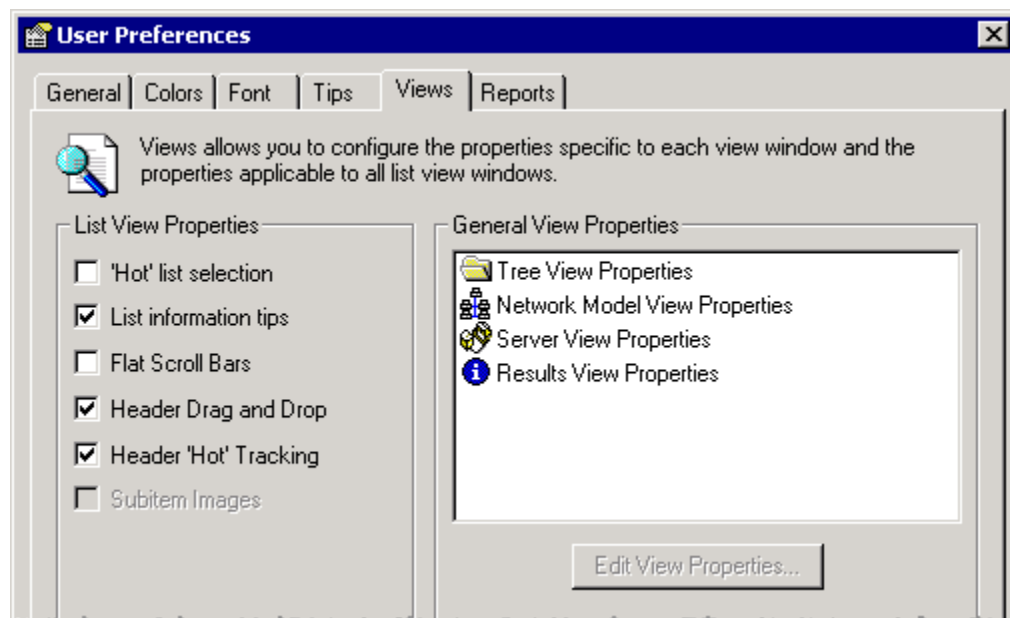
To edit information tips

1. Open the **Network Editor**.
2. From the **View** menu, select **User Preferences**.
3. On the **General** tab in the **User Preferences** dialog box, select **Information tips**.
4. Select the **Tips** tab.

5. To enable the **Background color** and **Text color** lists, disable **Use Windows colors**.
6. To change the tip color, select the text and background colors from the appropriate lists, and then click **OK**.
7. To change the tip font, click the **Change Font** button. The **Font** dialog box opens. Select the desired font, style, and size from the appropriate list, and then click **OK**.
8. To change the timing of information tips, use the **Up** and **Down** arrows to change the settings in the Timing area.
 - **Delay** determines how soon a tip displays when the cursor stops on an area with a tip.
 - **Visible duration** determines how long the tip is visible before closing.
9. Click **Apply**.
10. Click **OK**.

User Preferences dialog box - Views tab

To configure properties that are specific to each view and list in the Network Editor, use the **Views** tab. For all list views, you can enable information tips, alter the behavior of drag and drop, change the appearance of scroll bars, and set selection options. On selected views, you can change the font style and color, as well as the background color.



List View properties

The **List View Properties** area contains the following options to change the appearance and behavior of items seen in a list view.

- **"Hot" list selection** – If selected, an item is automatically highlighted and selected when the cursor remains over the item for a certain period of time.

- **List information tips** – If selected, single-line tips appear when you move the cursor over a button or field.
- **Flat scroll bars** – If selected, scroll bars appear 2-dimensional. If disabled, scroll bars have a 3-dimensional appearance (Windows default). The following figure illustrates both the default (on the left) and flat (on the right) scroll bars:



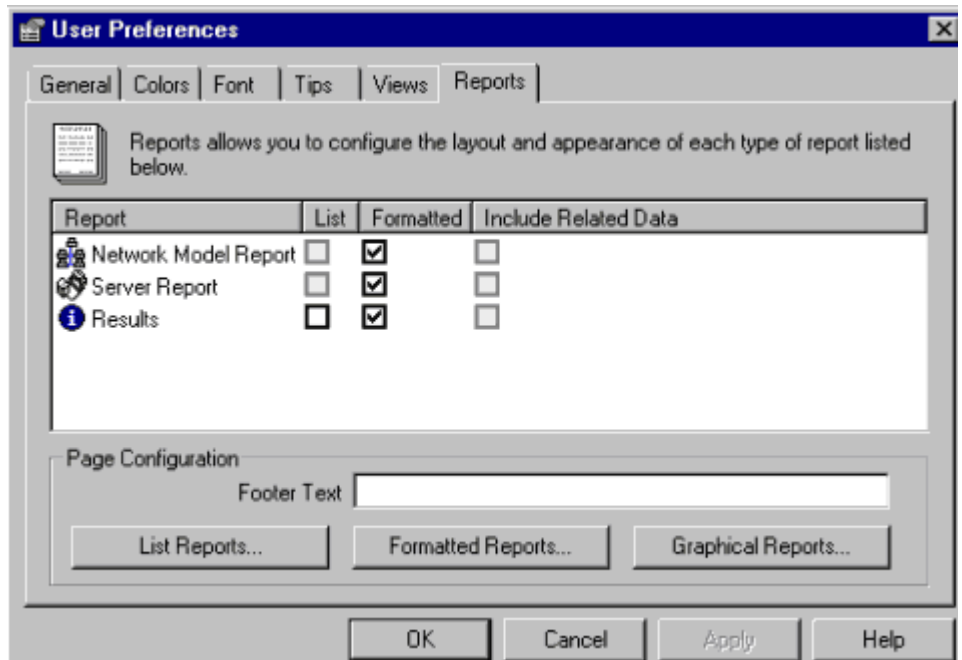
- **Header drag and drop** – If selected, users can click and drag a header from one location to another in order to rearrange the columns.
- **Header "hot" tracking** – If selected, an item is automatically highlighted but not selected when the cursor moves over a header item. You must still click the item at least once to activate it.
- **Subitem images** – Not currently enabled.

See also

[User Preferences dialog box](#) on [page 162](#)

User Preferences dialog box - Reports tab

Use the User Preferences dialog box - Reports tab to configure the layout and appearance of reports.



The **Reports** tab contains the following items:

Item	Description
Report	Lists the report templates available for the item or view that is currently active in the Network Editor.
List	Enables the list report style, if available for the associated report.

Item	Description
Formatted	Enables the formatted report style, which is the default format for most reports.
Include Related Data	Adds any additional data associated with the selected item to the printed report.
Footer Text	Prints the word or phrase entered at the bottom of each report page.
List Reports Formatted Reports Graphical Reports	Opens the Report Page Configuration dialog box where you can set the margins, font, and text color for a specific report template.

See also

[Set report preferences](#) on [page 167](#)

[Change margins](#) on [page 167](#)

[Set the report fonts and colors](#) on [page 168](#)

[Column and row preferences in reports](#) on [page 170](#)

Set report preferences

You can define a footer to print on each report page and select the font and color (if your printer supports color printing) for the various report sections.

To set report preferences

1. From the **View** menu, select **User Preferences**. The **User Preferences** dialog box opens to the **General** tab.
2. Select the **Reports** tab. The default report style for most views is **Formatted**.
3. To change the type of printed report, select either **List** or **Formatted** to the right of the appropriate report.

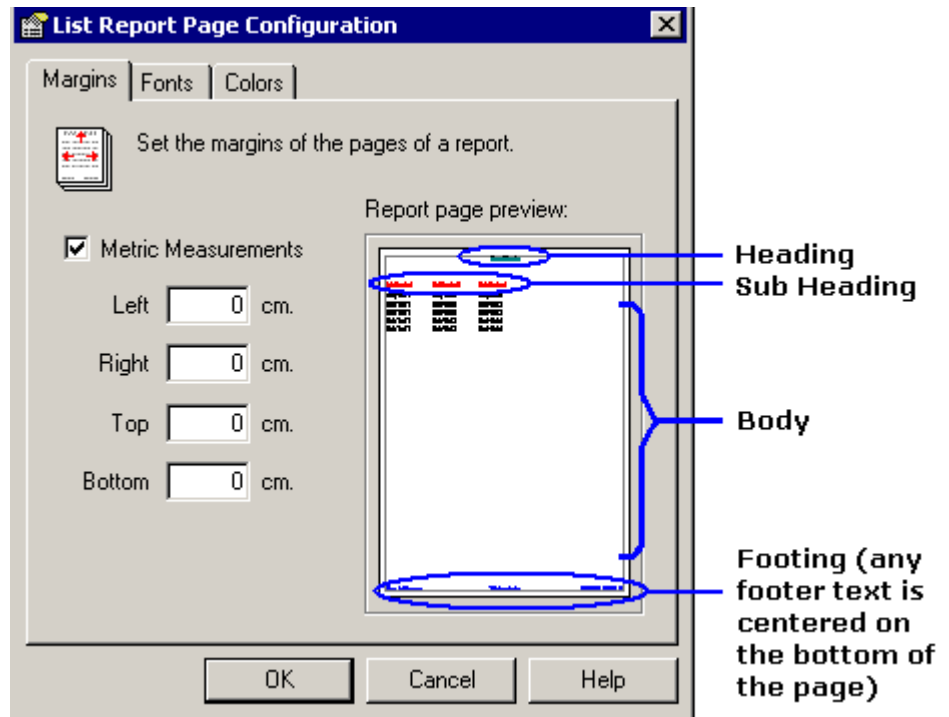
Tip: If you select the **List** format, set the page orientation to landscape. From the **File** menu, select **Print Setup**, select **Landscape** in the Orientation area, and then click **OK**.
4. Where available, you can select **Include Related Data**, and any additional data associated with the selected item (single items only – not lists or groups) also prints on the report.
5. In the **Footer Text** box, type a word or phrase to appear centered at the bottom of each page.
6. Click **Apply**.

Change margins

Use the **Report Page Configuration** dialog box to change margins.

To change margins:

1. To change the margins, fonts, and colors of a printed report, click the desired **Report** button (List, Formatted, or Graphical). The **Report Page Configuration** dialog box opens.



2. Change the margin widths of the printed report if desired. Type the desired amounts in each box for the **Left**, **Right**, **Top**, and **Bottom** margins. A visual representation of each margin change displays in the Report page preview area. Note that this is not to scale.
3. If you want to use inches instead of metric measurements, disable **Metric Measurements**.
4. Click **OK** to close the **Report Page Configuration** dialog box.
5. Click **Apply**.

See also

[User Preferences dialog box](#) on [page 162](#)

Set the report fonts and colors

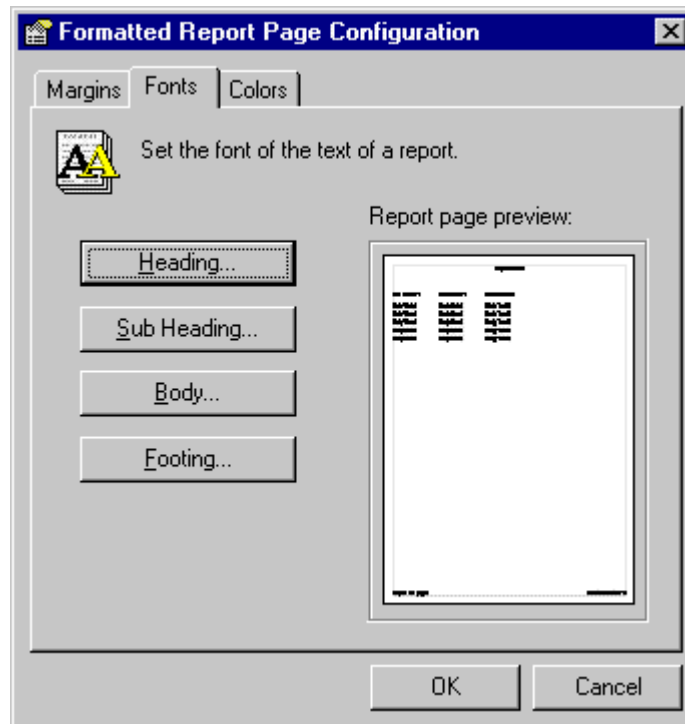
There are four areas in a report for which you can edit the fonts and colors:

- **Heading** – Prints across the top of each page, generally in a larger, bolder font than the rest of the report.
- **Sub Heading** – Refers to the field names on the report, such as Server. In formatted reports, subheadings print to the left of the actual report data.
- **Body** – Refers to the actual report data. Body text is usually smaller and lighter than the headings.

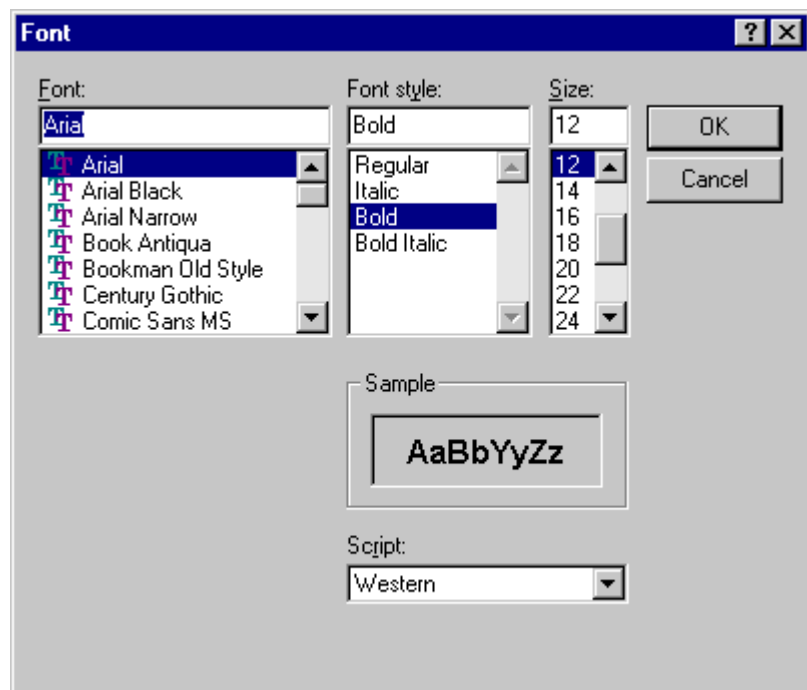
- **Footing** – Prints the page number, date and time that the report is printed across the bottom of each page, generally in a smaller font than the rest of the report.

To set the report fonts and colors:

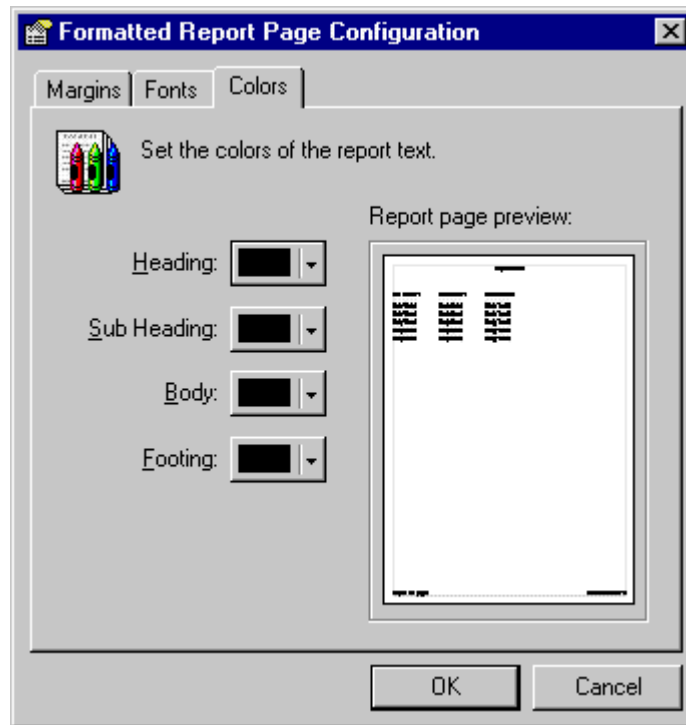
1. To change the font type, style, and size of each section in the report template (heading, subheading, body, or footer), select the **Fonts** tab.



2. Click the button for the report section you want to edit. The **Font** dialog box opens.



3. Select the desired font, font style/size, and then click **OK**.
4. If your printer supports color printing, you can add color to the report. Select the **Colors** tab.



5. Click the color selection button corresponding to the report section to which you want to add color. The **Color** dialog box opens.
6. Select a color from the **Color** dialog box, and then click **OK**. You will see a visual representation of each color change in the Report page preview area of the dialog box.
7. Repeat for the other sections of the report, as desired. When you are finished changing fonts and colors, click **OK** to close the **Report Page Configuration** dialog box.
8. Click **Apply**.

See also

[Report printing formats](#) on [page 171](#)

[Set report preferences](#) on [page 167](#)

Column and row preferences in reports

The Results pane contains data items in columnar format. The data items cannot be edited directly.

If you select List in the **User Preferences** dialog box, you can manipulate the columns in the Results pane as follows:

Action	How to Perform
Sorting columns	Click a column header to sort the items in the column alphabetically or numerically.

Action	How to Perform
Sizing columns	<p>Automatically: To make the column width fit the contents, double-click the boundary to the right of the column heading.</p> <p>Manually: Drag the boundary on the right side of the column heading until the column is the desired width.</p>
Editing the item name	Click once on a row to highlight it, and then click again. The item name is encased in an editing box so that you can rename the item.
	Double-click on a row to open the Properties dialog box, allowing you to edit the item name.
	Right-click on a row to display the shortcut menu; select Rename .
Editing a row	Double-click on a row to open the Properties dialog box to edit the row's information.
	Right-click on a row to display the shortcut menu; select Properties or other editing operations.
Copy, delete, or move rows	Right-click on a row to display the shortcut menu; select Cut , Copy , Paste , or Delete .

Report printing formats

The Network Editor report printing feature allows you to print network, server, or result pane information, in a report formatted with headers, footers and column headings (in list reports). There are two report formats available: formatted and list. The header (at the top of the page) consists of the report name, while the footer (at the bottom of the page) contains the printing date/time and the page count.

A formatted report presents the information down the page, with bold field names as shown in this figure.

Results

```

Result: 00000000
Computer: STATION_3
Message: Begin synchronization on all computers.
Type: Information

Result: 00000000
Computer: STATION_3
Message: Begin writing network model to computer 'STATION_3'.
Type: Information

Result: 00000000
Computer: STATION_3
Message: The network model was successfully written on computer 'STATION_3'.
Type: Information

Result: 00000000
Computer: STATION_1
Message: Begin writing network model to computer 'STATION_1'.
Type: Information

```

A list report presents the data in a columnar, spreadsheet format, with one line per data item. The field names become column headings as shown in this figure.

Results		
Result	Computer	Message
00000000	STATION_3	Begin synchronization on all computers.
00000000	STATION_3	Begin writing network model to computer 'STATION_3'.
00000000	STATION_3	The network model was successfully written on computer 'STA...
00000000	STATION_1	Begin writing network model to computer 'STATION_1'.
00000000	STATION_1	The network model was successfully written on computer 'STA...
00000000	STATION_3	The network model was successfully synchronized on all comp...

You can customize the reports by setting margins, changing font styles, and selecting text colors. In addition, you can enter custom text that is also included in the footer. (See [Set report preferences](#) on [page 167](#) for more information.)

Preview reports

The Print Preview function of the Network Editor allows you to see how a printed report looks before you send it to the printer.

To preview reports

1. Click the **Print Preview** button. The Print Preview window opens, displaying the report as it will look when printed. The following options are available from the Print Preview window:
 - **Next Page** – If there is more than one page in the report, click **Next Page** to view the next page of the report.
 - **Prev Page** – Click to view the previous report pages (if there is more than one page in the report).
 - **Two Page** – Allows you to view two pages of a report in the same window.
 - **One Page** – Returns to the default preview setting of one page per window.
 - **Zoom In/Out** – Click **Zoom In** to magnify the preview page. Click **Zoom Out** to reduce the size of the page in the preview window.
 - **Close** – Closes the preview window.
2. Click **Close** to return to the Network Editor. You can either print the report or make additional report configuration changes.

Print reports

The Network Editor report printing feature allows you to print network, server, or result pane information, in a report formatted with headers, footers and column headings (in list reports).

To print reports

1. In the Network Editor's Hierarchy pane, select the list or item(s) that you want to print.
2. From the **File** menu, click **Print**. You could also click the **Print** button or press ctrl+p to display the **Print** dialog box.
3. Ensure that the correct printer is selected, and click **OK**.



Tip: To configure heading and footer fonts and add a custom footer to your reports, see **Set report preferences** on [page 167](#) for instructions.

Print a range of reports

You may want to print just certain pages of the report. You can do so by entering the starting and ending page numbers of the section you want printed.

To print a range of reports:

1. In the Network Editor's Hierarchy pane, select the list or item(s) that you want to print.
2. Click **Print**.
3. In the Page Range area, click **Pages**. Type the page numbers to print in the **From** and **To** boxes.
4. Ensure that the correct printer is selected and click **OK**.

See also

[Report printing formats](#) on [page 171](#)

[Preview reports](#) on [page 172](#)

Server

This section contains information on abnormal FactoryTalk Batch Server termination, error log and tag verification log generation, and how the logs are used to troubleshoot system problems.

See also

[Abnormal FactoryTalk Batch termination](#) on [page 175](#)

[Tag verification log information](#) on [page 177](#)

Abnormal FactoryTalk Batch termination

After a FactoryTalk Batch Server failure, FactoryTalk Batch restarts your batches where they were before the failure. If the reason for the FactoryTalk Batch Server failure is not obvious, like a power failure, the FactoryTalk Batch Server writes this critical information to a log file. The log file contains the information the Application Support group needs when determining the cause of a server failure. The log file is a circular file, and eventually overwrites itself. To eliminate the possibility of losing this critical information, the FactoryTalk Batch Server reads the value of **AbnormalTermination** within the **Batchsvr.ini** file upon startup. If the field value is NO, or the field is not present, then no action is taken. If the field value is YES, then the current **Batchsvr.log** file is copied to the path specified by the ErrorLogDirectory entry in the **Batchsvr.ini** file. The name of the copied log file is determined based on **CopyFileID**.

After determining whether or not to make a copy of the current log file, the FactoryTalk Batch Server writes a value of **YES** into the field. At completion of a normal shutdown, the server writes a value of NO into the field.

See also

[Copied log file filename](#) on [page 175](#)

[Abnormal termination batch record entries](#) on [page 176](#)

Copied log file filename

If the FactoryTalk Batch Server generates a name for a copied log file, and the file name already exists in the directory where the copy is to be stored, the server attempts to create a new name for the copied log file. The server generates new log file names by incrementing the CopyFileID value until an unused file name is found. The integer used to generate this file name is incremented one last time and this value stored into the CopyFileID field in

the **Batchsvr.ini** file. If no unused file names are found, then the server is forced to overwrite the file name generated with the CopyFileID field's original value.

The value of the CopyFileID field is an integer from which a file name of the following format is generated:

<nnn>.LOG

Where <nnn> is the value of the CopyFileID field obtained from the **Batchsvr.ini** file, with leading zeros. The leading zeros help sort the files when viewed from Windows Explorer or a similar program. Every time the FactoryTalk Batch Server uses this value to generate a name for a copied log file, it increments the integer value stored in the field. An attempt to increment the integer beyond a value of 999 results in a wrapping of the value back to one.

If this value is not present or is invalid, then a current value of one (1) is assumed. This results in the creation of a log file copy with the name 001.log and the updating of the field to a value of two (2).

See also

[Abnormal FactoryTalk Batch termination](#) on [page 175](#)

Abnormal termination batch record entries

The electronic batch record (.evt or .evtj file) contains three entries if the batch is terminated by a FactoryTalk Batch Server cold boot or the batch is terminated due to an error when attempting to restore the batch from a server warm restart.

The first entry repeats the Event File Name event field. The second entry is a System Message event field indicating the reason the batch was terminated. The last entry is a System Message event field indicating the end of the batch.

Example: Termination of FactoryTalk Batch Due to Cold Boot

Description	Event	PValue
	Event File Name	MachineName\Program Files (x86)\Rockwell Software\Batch\YourProject\JOURNALS\4123.evt
YourBatchDescription	System Message	Batch terminated by COLD boot of FactoryTalk Batch server
YourBatchDescription	System Message	End Of BATCH

Example: Termination of FactoryTalk Batch During Warm Restart

Description	Event	PValue
	Event File Name	MachineName\Program Files (x86)\Rockwell Software\Batch\YourProject\ JOURNALS\4125.evt
YourBatchDescription	System Message	Batch terminated due to error attempting to restore batch during WARM restart
YourBatchDescription	System Message	End Of BATCH

See also

[Abnormal FactoryTalk Batch termination](#) on [page 175](#)

[Copied log file filename](#) on [page 175](#)

Tag verification log information

Tag verification is an operator-initiated process that causes the FactoryTalk Batch Server to attempt to read the current value of most tags defined in the area model. The tags are read up to 30 at a time, based on the value specified in the **Batchsvr.ini** file, and as reads are completed, new reads are initiated.

A dedicated log file, **Verify.log**, records the most recent verification results. This file is replaced with a new copy each time a verification begins. The **Verify.log** file writes to the same directory as the **Batchsvr.log** file.

Verification start, completion, and abort records in the standard FactoryTalk Batch Server log file, **Batchsvr.log**. Individual tag results record in the verification log file, **Verify.log**.

See also

[Abnormal FactoryTalk Batch termination](#) on [page 175](#)

[Tag verification log file](#) on [page 177](#)

[Windows Event log tag verification events](#) on [page 178](#)

Tag verification log file

The record structure in the tag verification log file (**Verify.log**) is identical to the record structure used in the **Batchsvr.log** file. The tag verification log file structure reserves the first record for field labels. The installation program installs a default file into the directory specified by the **ErrorLogDirectory** value in the **Batchsvr.ini** file.

The default file contains the following records:

- The first record is the standard header record that labels the fields.
- The second record indicates whether the verify function was performed.

The format of this record is depicted below:

Field #	Name	Value
1	Time	This field is left blank.
2	Severity	INFO
3	File	This field is left blank.
4	Line	This field is left blank.
5	RC_ID	This field is left blank.
6	Batch ID	This field is left blank.
7	Path	This field is left blank.
8	Element ID	This field is left blank.
9	Element Name	This field is left blank.
10	Message	Tag Verify Has Not Been Run.

The **Verify.log** file is limited to 9,000,000 bytes and wraps if that limit is exceeded. This provides enough space to verify roughly 50,000 tags before the output wraps.

The FactoryTalk Batch Server only keeps the verification log file open during the verification process. When a program has a file open, the file cannot be deleted or renamed. The server overwrites any existing verification log file each time a verification begins. If the server encounters a verification log file that is read only, the server changes the protection on the file to allow it to be overwritten.

See also

[Tag verification log information](#) on [page 177](#)

Windows Event log tag verification events

The following events regarding tag verification are recorded in the Windows Event Log.

Event	Generate Windows Event Log
Start	"Batch Server Tag Verification Process Started"
Abort	"Batch Server Tag Verification Process Aborted"
Completion	"Batch Server Tag Verification Completed, <xxx> of <yyy> tags verified successfully" where <xxx> is the number of tags that were successfully verified, and <yyy> is the total number of tags on which verification was attempted.

See also

[Tag verification log information](#) on [page 177](#)

FactoryTalk Batch performance chart

The FactoryTalk Batch performance chart is a predefined set of charts for use with the Windows System Monitor.



Tip: The predefined performance monitor charts are in English only. Non-English customers must configure the charts manually. (Refer to your Windows documentation for more information on performance charts.)

The Windows System Monitor can examine the use of computer resources by FactoryTalk Batch components, as well as other processes and applications, and help determine if a process or application is using too much CPU time or memory. The performance chart charts specific processes that can affect the performance of FactoryTalk Batch, and is an important tool in identifying and troubleshooting system problems.



Tip: The performance chart is intended to be used primarily as a debugging and diagnostic tool by FactoryTalk Batch product support personnel.

The Windows System Monitor is used to view chart data, alerts, logs and reports. Additionally, charts can be saved for future review. To view the FactoryTalk Batch performance chart, make sure that the System Monitor is set to chart view mode.

See also

[Open the FactoryTalk Batch performance chart](#) on [page 179](#)

Open the FactoryTalk Batch performance chart

The performance chart automatically opens the Windows System Monitor with the pre-defined charts loaded.

To open the FactoryTalk Batch performance chart:

1. Select **Start > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**.
2. Select **Performance Chart**. The **Performance** dialog box opens with the pre-defined charts listed.

See also

[FactoryTalk Batch performance chart](#) on [page 178](#)

Batch identification

For purposes of batch identification, you can create custom default batch IDs, and allow or prevent editing of batch IDs. You can also automatically generate custom batch IDs. To implement this option, you must edit some C# code and compile it into a new DLL.

See also

[Create a custom BatchIDCreate.dll](#) on [page 181](#)

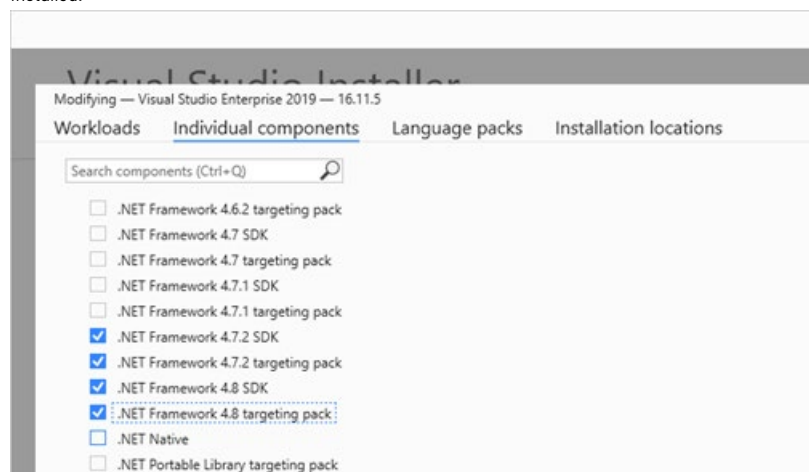
Create a custom BatchIDCreate.dll

You can use the BatchIDCreate solution to automatically generate custom batch IDs in any values or increments. If this option is chosen, you must modify some C# code in an example project, compile it into a new Dynamic Link Library (DLL) file, and then replace the existing DLL in the **BATCH\Bin** folder.

It is recommended that you make a backup copy of the original BatchIDCreate.dll before making any changes. The example solution project provides a starting point that you can modify as needed. You can build this solution using Microsoft Visual Studio 2019 (version 16.11.9 or later).



Tip: Before attempting to build the project confirm that the .NET Framework 4.8 targeting pack is installed:



To customize BatchIDCreate.dll

1. Copy the **C:\Program Files x86\Rockwell Software\Batch\Custom\BatchIDCreate\bin\Release** folder,

including all subdirectories, to the local hard drive, leaving the structure intact.



Tip: The **Release** folder can also be copied from the **Batchctl\Custom** shared network location.

2. Double-click **BatchIDCreate.sln** to open the Visual Studio solution.
3. Edit the **BatchIDCreateClass** to include your C# code that modifies the Custom ID generation algorithm then save the solution.



Tip: The batch ID can contain a maximum of 255 characters with no spaces or invalid characters. The batch ID can be comprised of any characters except the following: The defined list separator, single or double quotes, brackets, parentheses, the percent sign (%), the tab character (\t), the carriage return character (\r), or the new line character (\n).

4. In the Visual Studio toolbar set the build type to **Release**.
5. In the Solution Explorer right-click **BatchIDCreate** and then select **Build**.
6. Copy the new **BatchIDCreate.dll** from `Custom\BatchIDCreate\bin\Release`.
7. Paste the new **BatchIDCreate.dll** into the `C:\Program Files (x86)\Rockwell Software\Batch\Bin` folder to replace the default **BatchIDCreate.dll**.
8. Restart your computer.

See also

[Batch identification](#) on [page 181](#)

Change the BatchIDCreate.dll registration

After creating a custom BatchID, you might need to unregister the `BatchIDCreate.dll` and register it again for it to be updated on the Batch Server. You can use the provided `.bat` files to perform this task.

To re-register the BatchIDCreate.dll

1. Open the Windows command prompt as an administrator.
2. Navigate to the `C:\Program Files x86\Rockwell Software\Batch\Custom\BatchIDCreate\bin\Release` folder.
3. At the prompt type **UnregisterBatchIDcreate.bat**.
4. After it completes, at the prompt type **RegisterBatchIDcreate.bat**.

Change the FactoryTalk Batch Server user account

Depending on the facility's security requirements, there may be a requirement to periodically change the FactoryTalk Batch Server user name or password, or both. If you change the server's user account after installing the FactoryTalk Batch components, configure the FactoryTalk Batch system to use the new server user account.

IMPORTANT The user account for the **FactoryTalk Batch Server**, **Batch Archiver**, **eProcedure Server** services and the **Batch COM+** Application must be identical.

See also

[FactoryTalk Batch system configuration](#) on [page 20](#)

[System-wide COM security limit settings](#) on [page 20](#)

[FactoryTalk Batch Server install configuration](#) on [page 21](#)

[Add the Batch COM service](#) on [page 22](#)

New server user account requirements

When creating a new user account for the FactoryTalk Batch or Material Manager Server, these requirements must be met.

- The password must be configured never to expire--if the password ever expires, the service eventually fails to log on.
- The user account must never be disabled or deleted--if this account is ever disabled/deleted, the service eventually fails to log on.
- If using a domain, the domain user account must have a unique name--if the user account is a domain account, remove any local user accounts with the same name.



Tip: The Material Manager Server and FactoryTalk Batch Server can use the same server user account. If they use separate accounts, the Material Manager Server account is only used on the material server computer.

Configure your FactoryTalk Batch system with the new server user account

After you create a new server user account, configure your FactoryTalk Batch system.

To configure your FactoryTalk Batch system

1. Add the new server user account to the **batchsvr_group** and administrators group. By adding the new account to the **batchsvr_group** it inherits all the required user rights and access permissions.
2. If FactoryTalk Batch Material Manager is part of your system, add the new server user account to the **MTUsers** group on the Material Server computer.
3. Add the new server user account to the **FactoryTalk Batch Server** service.
4. If you are running FactoryTalk Event Archiver in incremental mode, add the new server user account in the **Batch Archiver** service.
5. If FactoryTalk eProcedure is part of your system, add the new server user account to the **eProcedure Server** service.
6. Add the new server user account to the **Batch COM** object in **Component Services > My Computer > COM+ Applications**.

Server account password change locations

If you change the password for the existing server user account, change the password in these locations:

- Change the password for the server user account in the FactoryTalk Batch server service.
 - If you are running FactoryTalk Event Archiver in incremental mode, change the password for the server user account to the **Batch Archiver** service.
 - If eProcedure is part of your system, change the password for the server user account in the **eProcedure Server** service.
- Change the password for the server user account in the **Batch COM** object in **Component Services > My Computer > COM+ Applications**.
- If FactoryTalk Batch Material Manager is part of your system, change the password for the server user account in the **MaterialTrack COM** object in **Component Services > My Computer > COM+ Applications**.



Tip: Make sure the server user account is a member of the MTUsers group on the Material Server computer.

See also

[Configure your FactoryTalk Batch system with the new server user account](#) on [page 183](#)

Change the server user account for the FactoryTalk Batch Server service

To change the user account for the FactoryTalk Batch Server service, complete these steps on the server computer.

To change the server user account for the FactoryTalk Batch Server service

1. Select **Start > Windows Administrative Tools > Services**. The **Services** dialog box opens.
2. Right-click **FactoryTalk Batch Server**, and then click **Properties**. The **FactoryTalk Batch Server Properties** dialog box opens.
3. On the **Log On** tab, select **This Account** and type the new user and/or password.
4. Select **OK**.
 - If you are using FactoryTalk Event Archiver in incremental mode, change the user account name and/or password for the **Batch Archiver** service.
 - If eProcedure is part of your system, change the user account name and/or password for the **eProcedure Server** service.
5. Close **Services**.

To change the FactoryTalk Batch Server or Material Manager Server user account in Batch COM+ applications, complete these steps on the FactoryTalk Batch Server computer.

Change the server user account in FactoryTalk Batch COM+ applications

To change the server user account in FactoryTalk Batch COM+ applications

1. Select **Start > Windows Administrative Tools > Component Services**.
2. Expand **Component Services, Computers, My Computer** and **COM+ Applications**.
3. For FactoryTalk Batch, right-click **Batch**, and then select **Properties**. The **Batch Properties** dialog box opens.
 - On the **Identity** tab, select **This user** and type the new FactoryTalk Batch Server user name and/or password.
 - Select **OK**.
4. For FactoryTalk Batch Material Manager, right-click **MaterialTrack**, and then select **Properties**. The **MaterialTrack Properties** dialog box opens.
 - On the **Identity** tab, select **This user** and type the new Material Manager Server user name and/or password.
 - Select **OK**.




Tip: The new permissions are applied the next time the FactoryTalk Batch Server and the Material Manager Server are started. The same server user account can be used for both servers.

Change the server user account in My Computer properties

To change the server user account in **My Computer** properties, complete these steps on the FactoryTalk Batch client computers.

To change the server user account in My Computer properties

1. Select **Start > Windows Administrative Tools > Component Services**.
2. Expand **Component Services** and **Computers**.
3. Right-click **My Computer**, and then select **Properties**. The **My Computer Properties** dialog box opens.
4. On the **COM Security** tab, select **Edit Default** in the **Access Permissions** area. The **Access Permission** dialog box opens.
5. Select **Add** to open the **Select Users, Computers, or Groups** dialog box.
6. Type the new server user account name and then select **OK**.
7. In the **Access Permission** dialog box, select the new server user account. In the **Permissions** list, select **Allow** for Local Access and Remote Access.
8. Select **OK** twice.
 **Tip:** If you are using separate user accounts for FactoryTalk Batch Server and Material Manager Server, repeat steps 5 - 8 to add the additional server user account.
9. Close **Component Services**.
10. Restart the computer.

General troubleshooting errors

This section outlines errors that may occur while installing and running FactoryTalk Batch and possible solutions.

Error: Automatic repair of batch component installation fails

Possible cause	Solution
<p>This may occur if one or more folders were moved in Windows explorer.</p> <p>For example, if the Schema folder gets moved from its original directory, the installation package starts up automatically and tries to repair the problem.</p> <p>This may clear out the Model and Server keys in the registry, which will prevent the batch client editors from finding the Network Model.</p>	<p>Completely uninstall the application and then reinstall it.</p>

Error: FactoryTalk Batch Server does not start

Possible cause	Solution
<p>Journal path has not been defined in FactoryTalk Batch Server options.</p>	<p>Modify the FactoryTalk Batch Server options to include a valid Primary Journal path. Verify that the Secondary Journal path, if defined, is valid.</p>
<p>Restart path has not been defined in FactoryTalk Batch Server options.</p>	<p>Modify the FactoryTalk Batch Server options to include a valid Primary Restart path. Verify that the Secondary Restart path, if defined, is valid.</p>
<p>The FactoryTalk Batch Server is always installed with the default user and password.</p>	<p>Verify that the FactoryTalk Batch Server service user account is correct.</p> <ol style="list-style-type: none"> 1. Navigate to the Windows services and the FactoryTalk Batch Server. 2. Open the service. 3. On the Log On tab, verify "This account" is selected and the batch server account and password are correct by re-entering the user name and password.

Error: FactoryTalk Batch Server runs for a specific period of time and then stops, and consistently stops at the same two-hour interval

Possible cause	Solution
<p>The FactoryTalk Batch Server did not read the product license successfully on startup, and demo mode was selected by the user to start the FactoryTalk Batch Server service.</p>	<p>Check the FactoryTalk Batch Activation Server and make sure there is a working FactoryTalk Batch server license. You can also check the batchsvr.log file to determine how the batch server is starting (demo mode or license mode) and change to license mode.</p>

Error: FactoryTalk Batch Service Manager does not display the FactoryTalk Batch Server or FactoryTalk Event Archiver

Possible cause	Solution
The computer where the services are located is not selected.	Click the Select Computer button and select the appropriate computer.
The FactoryTalk Batch Server, FactoryTalk Event Archiver, and FactoryTalk eProcedure services are installed but are not registered with the Windows Registry.	<p>To register a service:</p> <ol style="list-style-type: none"> 1. Select Start > Windows System > Command Prompt. 2. Navigate to the Program Files (x86)\Rockwell Software\Batch\bin directory. <ul style="list-style-type: none"> • To register the FactoryTalk Batch Server, type batchsrv /service at the command prompt, and then press Enter. • To register FactoryTalk Event Archiver, type batcharc /service at the command prompt, and then press Enter. • To register FactoryTalk eProcedure, type batchmbs /service at the command prompt, and then press Enter.

Error: FactoryTalk Batch Service Manager displays the class name (Batch.Server Class) instead of the display name (Batch Server)

Possible cause	Solution
DCOMCNFG configures the FactoryTalk Batch Server service to display the class name of the service instead of the display name.	When this occurs, the server no longer supports its COM interface and must be reinstalled.

Error: The text in FactoryTalk Batch dialog boxes does not display properly

Possible cause	Solution
FactoryTalk Batch does not support the use of large system fonts.	In the Display Properties dialog box, change the default system font size to Small Fonts .

Error: Unable to locate files. The Recipe directory file could not be found

Possible cause	Solution
The server options for the FactoryTalk Batch Server are not configured to point to the folder that contains the Recipe.dir file.	<p>Configure the server options to point to the folder containing the Recipe.dir file.</p> <p>See Edit the FactoryTalk Batch server options for more information.</p>

Error: Application Log error *The Application log file is full is generated*

Possible cause	Solution
The Windows Application log is not set for wrapping.	Use the Event Viewer to change the Application Event Log Wrapping setting to Overwrite Events as Needed : <ol style="list-style-type: none"> 1. Select Start > Windows Administrative Tools > Event Viewer. The Event Viewer opens. 2. Right-click Application Log, and select Properties. The Application Log Properties dialog box opens. 3. In the Log size section, select Overwrite events as needed. 4. Click OK.

Error: Recipes not visible from FactoryTalk Batch View Select a Recipe dialog box

Possible cause	Solution
The Release Recipe to Production check box is not selected in the recipe header data.	For each recipe that should appear in the Select a Recipe dialog box, ensure that the Release Recipe to Production check box is selected. This can be done by selecting the Header Data menu option from the Recipe Editor Recipe menu for the recipe and selecting the Release Recipe to Production check box.
If recipe approvals are enabled, the Release Recipe to Production step is not approved.	For each recipe that should appear in the Select a Recipe dialog box, ensure that the Release Recipe to Production step is approved. This can be done by signing off on the Release Recipe to Production approval step as part of the approval process defined in the area model for the recipe.

Error: List of available SQL server databases not displayed in the Server Options or Database Selection dialog box

Possible cause	Solution
When the browse button for SQL server databases is clicked, no databases are displayed in the Database Selection dialog box.	Install the SQL Server Client Tools Connectivity option. Do a custom install of SQL server to install the SQL Server Client Tools Connectivity option. Or just type in the database name.

Error: Event Journal files are not being created

Possible cause	Solution
If the Primary Journals directory is located on a computer other than the server computer, or not within the BATCHCTL share, and that computer becomes inaccessible, the server continues executing batches but is unable to create event journals for those batches. You will not receive a notification of the problem.	The primary journal directory must be located on the FactoryTalk Batch Server computer in the BATCHCTL share. To mitigate risk of an event journal loss in your system, it is recommended that a secondary event journal path be established to a local and separate physical drive on the FactoryTalk Batch Server node, or to a location for networked storage.

See also

[FactoryTalk Batch performance chart](#) on [page 178](#)

Index

.
.evt file 176
.NET registry key 90

A

administrative Tools 63
allowable failures 83
Archiver
 (Incremental) 79
 event filters 82
 service 183
audit
 Equipment Editor 52
 message generation 51
 Recipe Editor 52
AutoBatchID
 execute failure 181
automatic
 restart control 90

B

Batch
 audit 51
Batch COM+ Application 21
Batch IDs
 customize 181
Batch Server
 cold boot 176
 computer name 90
 demo mode 63
 does not start 187
 end of batch 176
 failure 175
 options 74
 performance chart 178
 restart 175
 user group 21
 warm restart 176
Batch Server Service
 start 68
 startup type 73
Batch Server Statistics
 COM information 63
 dialog box 66
Batch Service Manager
 interface 64

BATCHHIS table
 select 79
BatchIDCreation.dll 181
Batchsvr.ini 177
batchsvr_group 21

C

change
 colors 164, 165
 fonts 165
 list view properties 165
 report margins 167
 view properties 165
colors
 change 164, 165
command handshake timeout 93
Command Signature 38
Command Verification policy 38
communications timeout 83
configure
 formatted reports 171
 list reports 171
 report footings 167
 report headings 167
 user preferences 162
continuous phase 146

D

data server definition file (DSDF) 18
database
 names 158
 Network Editor 158
 SQL Server 187
default
 hyperlink values 88
descriptors
 hyperlink label 88
directory share 17
disable
 Archiver service 79
DSDF 18
dwelt time 147

E

electronic batch record 176
Electronic Signatures
 policy setting 49
eProcedure
 hyperlinks 88
Equipment Editor
 access modes 28
 security 27
error message

- application log is full 187
- license for component not found 187
- recipe directory file not found 187

ErrorLogDirectory 177**errors**

- troubleshooting 187

event journal signatures

- electronic batch records 83

event journals 23**Event Viewer 135****F****FactoryTalk AssetCentre**

- message fields 61

FactoryTalk Batch server install configuration 21**FactoryTalk Directory 25****FactoryTalk Help 25****FactoryTalk Security 25****FactoryTalk Services Platform 25****FactoryTalk Windows Firewall Configuration Utility 21****failure**

- handshake timeout 83
- phase 83
- watchdog timeout 83

formatted report 171**G****General Usage Phase Logic Requests 48****H****hold propagation**

- setting 83

hyperlink

- default values 88

I**Incremental Archiver 79****L****list report 171****list view properties**

- change 165

log files 23**M****manual failover 91****Material Manager 88****Maximum Log File Size box 83****Minimum Disk Space box 83****N****Network Editor**

- database 158

network model

- information 158
- synchronize 161

O**OLE for process control (OPC) 17****OPC groups 18****P****Period (msec) box 83****permissions 17****phase**

- material-enabled 145

phase logic simulation 145**policy settings**

- ESignature Network Directory 49

print preview 172**print reports**

- print a range of pages 173

product policies 26**R****Recipe Editor**

- access modes 28
- configuration options 29

recipes

- release to production 187

report preferences

- set 167

reports

- change margins 167
- column and row preferences 167
- configure footings 167
- configure headings 167

resource types

- cross invocation 87
- hyperlinks 88

restart Batch Server 175**restore a batch 176****S****security**

- event journal 23
- file 23
- folder 23
- log file 23
- Windows 19

Server Options

- Archiver (End of Batch) 79

- Archiver (Incremental) 79
- Archiver Event Filters 82
- Batch Reporting tab 79
- Batch Server tab 83
- Cross Invocation tab 87
- edit 74
- redundant servers 77
- Restart Control tab 77
- Server Options dialog box 66**
- set**
 - report preferences 167
 - user preferences 162
- shared directory 17**
- Signature Requests**
 - General Usage 48
- Signature Templates 37**
- simulator 145**
 - ack/clear requests 153
 - change dwell time 147
 - command a phase 151
 - indicate a continuous phase 148
 - interface 146
 - phase configuration 146
 - phase ownership 151
 - phase parameter tag 149
 - phase status 149
 - report parameter tag 150
 - request data tags 150
 - request sequence 148
 - restore/clear requests 153
 - unit tag status 155
- synchronize 161**
- System Log file 187**
- system problems 175**

T

- tag verification 177**
- troubleshoot**
 - ActiveX controls 187
 - application event log 187
 - Batch Server failure 175
 - Batch Service Manager 187
 - batch terminated 176
 - error messages 175
 - event journal files 187
 - recipes not visible 187
 - SQL Server selection 187
 - system problems 175
 - tag verification 177
 - text in Batch dialog boxes 187

U

- user account**
 - create 19
- User Preference dialog box**
 - change colors 164, 165

- change fonts 165
- change view properties 165
- column and row preferences 167
- edit information tips 164
- list view properties 165
- set 162

User-Defined (End of Batch) 79

V

view properties 165

W

Windows Event Log

- modify settings 138
- tag verification 178

Windows registry

- .NET key 90

Windows services

- Archiver 63
- Batch Server 63

workgroup

- configuring client 26

Rockwell Automation support

Use these resources to access support information.

Technical Support Center	Find help with how-to videos, FAQs, chat, user forums, and product notification updates.	rok.auto/support
Knowledgebase	Access Knowledgebase articles.	rok.auto/knowledgebase
Local Technical Support Phone Numbers	Locate the telephone number for your country.	rok.auto/phonesupport
Literature Library	Find installation instructions, manuals, brochures, and technical data publications.	rok.auto/literature
Product Compatibility and Download Center (PCDC)	Get help determining how products interact, check features and capabilities, and find associated firmware.	rok.auto/pcdc

Documentation feedback

Your comments help us serve your documentation needs better. If you have any suggestions on how to improve our content, complete the form at rok.auto/docfeedback.

Waste Electrical and Electronic Equipment (WEEE)



At the end of life, this equipment should be collected separately from any unsorted municipal waste.





Rockwell Automation maintains current product environmental information on its website at rok.auto/pec.

Allen-Bradley, expanding human possibility, Logix, Rockwell Automation, and Rockwell Software are trademarks of Rockwell Automation, Inc.

EtherNet/IP is a trademark of ODVA, 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 EEE Yönetmeliğine Uygundur

Connect with us.    

rockwellautomation.com ————— expanding **human possibility**[™]

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