Important User Information

Solid-state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation, and Maintenance of Solid State Controls (publication SGI-1.1 available from your local Rockwell Automation sales office or online at http://www.rockwellautomation.com/literature/) describes some important differences between solid-state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid-state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

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. Attention helps you identify a hazard, avoid a hazard, and recognize the consequence.

**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.

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

---

Allen-Bradley, Rockwell Software, and Rockwell Automation ControlLogix, sDevice, FactoryTalk, RSBaseWare, RSBaseWare Batch Historian, RSBaseWare Historian, RSBaseWare MaterialTrack, and RSLogix are trademarks of Rockwell Automation, Inc.

Trademarks not belonging to Rockwell Automation are property of their respective companies.
Import Important User Information

Solid-state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation, and Maintenance of Solid State Controls (publication SGI-1.1 available from your local Rockwell Automation sales office or online at http://www.rockwellautomation.com/literature/) describes some important differences between solid-state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid-state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

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.

**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.

**Important:** Identifies information that is critical for successful application and understanding of the product.
# Table of contents

## Chapter 1

**FactoryTalk Batch components**
- Document organization ................................................................. 9
- FactoryTalk Batch regional settings .............................................. 10

## Chapter 2

**FactoryTalk Batch communications**
- Create a shared directory ............................................................... 11
- Use OLE for Process Control (OPC) ............................................. 12
  - OPC group support ................................................................. 12

## Chapter 3

**FactoryTalk Batch security**
- FactoryTalk Batch Server user account configuration requirements ... 13
- FactoryTalk Batch system configuration .......................................... 14
  - System-wide COM security limit settings .................................. 14
  - FactoryTalk Batch Server install configuration ......................... 15
  - FactoryTalk Batch Client installation configuration .................. 16
  - FactoryTalk Batch Windows Firewall exceptions ....................... 17
  - Add the Batch COM service .................................................... 17
  - Set security on folders or files .................................................. 18
  - Add users and user groups ....................................................... 19
  - Change permissions on the BATCHCTL share ......................... 20
  - FactoryTalk Security for the Batch Clients .............................. 21
  - Tighten security for FactoryTalk Batch Clients ....................... 22
- FactoryTalk Batch View security .................................................. 23
  - Configure security for the FactoryTalk Batch View windows ...... 23
  - Security for FactoryTalk Batch commands ............................... 24
    - Configure security for FactoryTalk Batch commands ............... 25
    - Enable the FactoryTalk Batch View confirm settings ................ 27
  - Set the automatic logoff time period ....................................... 28
  - Change the logged-on user with your HMI ............................... 28
  - External security devices ....................................................... 30
    - Enable an external security device ...................................... 31
  - FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor security ......................................................... 32
    - Configure access mode security .......................................... 33
    - Configure feature security .................................................... 34
    - Modify FactoryTalk Batch Equipment Editor configuration options .......................................................... 35
  - FactoryTalk Batch default policy settings ................................. 36
    - Product policies in the local and network directories .............. 36
# Table of contents

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Chapter 4</strong></td>
<td>FactoryTalk Batch electronic signatures</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Command verification policies</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td>Command signature requests</td>
<td>44</td>
</tr>
<tr>
<td></td>
<td>Set parameter signature request</td>
<td>46</td>
</tr>
<tr>
<td></td>
<td>FactoryTalk Batch Parameter Prompt Acknowledge Signature request</td>
<td>47</td>
</tr>
<tr>
<td></td>
<td>Unit Bind Prompt Acknowledge Signature request</td>
<td>48</td>
</tr>
<tr>
<td></td>
<td>Phase Bind Prompt Acknowledge Signature request</td>
<td>49</td>
</tr>
<tr>
<td></td>
<td>Unit Bind Signature request</td>
<td>50</td>
</tr>
<tr>
<td></td>
<td>Remove Batch Signature request</td>
<td>51</td>
</tr>
<tr>
<td></td>
<td>Add Event Signature request</td>
<td>53</td>
</tr>
<tr>
<td></td>
<td>Active Step Change Signature request</td>
<td>54</td>
</tr>
<tr>
<td></td>
<td>Parameter deviation verification policies</td>
<td>54</td>
</tr>
<tr>
<td></td>
<td>Parameter deviation signature requests</td>
<td>55</td>
</tr>
<tr>
<td></td>
<td>Report parameter deviation signature requests</td>
<td>56</td>
</tr>
<tr>
<td></td>
<td>General usage phase logic requests</td>
<td>57</td>
</tr>
<tr>
<td></td>
<td>General usage signature requests</td>
<td>57</td>
</tr>
<tr>
<td></td>
<td>Electronic signature security</td>
<td>58</td>
</tr>
<tr>
<td></td>
<td>Configure the ESignature network directory policy setting</td>
<td>58</td>
</tr>
<tr>
<td><strong>Chapter 5</strong></td>
<td>FactoryTalk Batch audit trail</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Enable auditing for FactoryTalk Batch Equipment Editor or FactoryTalk Batch Recipe Editor</td>
<td>62</td>
</tr>
<tr>
<td></td>
<td>Audited FactoryTalk Batch Equipment Editor events</td>
<td>62</td>
</tr>
<tr>
<td></td>
<td>Audited FactoryTalk Batch Recipe Editor events</td>
<td>65</td>
</tr>
<tr>
<td></td>
<td>Audit messages</td>
<td>67</td>
</tr>
<tr>
<td></td>
<td>FactoryTalk Diagnostics Viewer</td>
<td>68</td>
</tr>
<tr>
<td></td>
<td>Standard viewer message fields in FactoryTalk Diagnostics Viewer</td>
<td>68</td>
</tr>
<tr>
<td></td>
<td>Extended viewer message fields in FactoryTalk Diagnostics Viewer</td>
<td>69</td>
</tr>
<tr>
<td></td>
<td>FactoryTalk AssetCentre Client</td>
<td>71</td>
</tr>
<tr>
<td></td>
<td>Viewer message fields</td>
<td>72</td>
</tr>
<tr>
<td><strong>Chapter 6</strong></td>
<td>The FactoryTalk Batch Service Manager</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Windows services</td>
<td>75</td>
</tr>
<tr>
<td></td>
<td>FactoryTalk Batch Service Manager interface</td>
<td>76</td>
</tr>
<tr>
<td></td>
<td>Connection</td>
<td>76</td>
</tr>
<tr>
<td></td>
<td>Service state</td>
<td>77</td>
</tr>
<tr>
<td></td>
<td>Server</td>
<td>78</td>
</tr>
<tr>
<td></td>
<td>Batch Server Statistics dialog box</td>
<td>78</td>
</tr>
<tr>
<td></td>
<td>Batch Server Statistics dialog box - General tab</td>
<td>79</td>
</tr>
<tr>
<td></td>
<td>Batch Server Statistics dialog box - Messages tab</td>
<td>80</td>
</tr>
<tr>
<td></td>
<td>Batch Server Statistics dialog box - PCD Communications tab</td>
<td>81</td>
</tr>
<tr>
<td></td>
<td>Batch Server Statistics dialog box - View Communications tab</td>
<td>82</td>
</tr>
<tr>
<td></td>
<td>Start the FactoryTalk Batch Server service</td>
<td>82</td>
</tr>
</tbody>
</table>
# The FactoryTalk Batch Server

Grace periods in FactoryTalk Batch Server ............................................................... 84  
Enable a grace period .................................................................................................. 85  
Add a custom service ................................................................................................. 86

## Chapter 7

FactoryTalk Batch Server service configuration ....................................................... 87  
Specify the startup type .............................................................................................. 87  
FactoryTalk Batch Server option configuration ....................................................... 88  
Server Options dialog box - Project Settings tab .................................................. 89  
Set project directories .............................................................................................. 90  
Recipe storage ......................................................................................................... 90  
Recipe versioning ...................................................................................................... 91  
Server Options dialog box - Restart Control tab .................................................. 91  
Redundant servers .................................................................................................... 92  
Server Options dialog box - Batch Reporting tab .................................................. 93  
Report Editor ......................................................................................................... 94  
Perform end of batch archiving ............................................................................. 94  
Perform incremental archiving .............................................................................. 95  
Perform user-defined end of batch ...................................................................... 96  
None....................................................................................................................... 96  
Never..................................................................................................................... 96  
Archiver Event Filters tab ....................................................................................... 96  
Batch Server tab ...................................................................................................... 97  
Hold propagation .................................................................................................... 98  
Communications timeout ...................................................................................... 98  
Miscellaneous ....................................................................................................... 99  
Cross Invocation Descriptors and Defaults tab .................................................. 100  
Resource types .................................................................................................. 100  
Descriptors ......................................................................................................... 101  
Default values ..................................................................................................... 101  
Hyperlink Descriptors and Defaults tab .......................................................... 101  
Material Policies tab ............................................................................................ 102  
Modify the .NET registry key ............................................................................... 103  
Use automatic restart control to restart the FactoryTalk Batch Server .............. 103  
Restart the FactoryTalk Batch Server .................................................................. 104  
Perform a manual failover ...................................................................................... 104  
Configure the FactoryTalk Batch Server for manual failover ......................... 105  
Configure the material-enabled FactoryTalk Batch Server for manual failover .................................................................................................................. 106  
Command handshake timeout .............................................................................. 107

## Chapter 8

Windows event log

View event log entry properties .............................................................................. 109  
Modify event log settings ...................................................................................... 113
# Chapter 9

**Cross invocation**

- Cross invocation overview.............................................................................................. 115
- Cross invocation string configuration ......................................................................... 117
- Cross invocation string escape sequences .................................................................... 118
- Cross invocation string example ........................................................................... 119
- FactoryTalk Batch View and ActiveX Controls configuration for cross invocation ........... 120

# Chapter 10

**The FactoryTalk Batch Phase Simulator**

- Start the FactoryTalk Batch Phase Simulator ............................................................ 124
- FactoryTalk Batch Phase Simulator interface ........................................................ 124
- Phase Configuration dialog box .................................................................................... 125
  - Change the dwell time............................................................................................. 126
  - Change the sequence of requests........................................................................... 126
  - Indicate a continuous phase.............................................................................. 127
- Phase Status dialog box ................................................................................................... 128
  - View phase parameter tag values ........................................................................ 128
  - View and update report parameter tag values...................................................... 129
  - View request data tag values ............................................................................... 129
  - Change ownership of a phase ........................................................................... 130
  - Command a phase............................................................................................... 131
    - Phase commands.................................................................................................... 131
    - Acknowledge and clear a request ...................................................................... 132
    - Restore and clear requests ............................................................................ 133
- Unit Tag Status dialog box ............................................................................................. 134
  - Change the value of a unit tag............................................................................... 134

# Chapter 11

**Network Editor**

- Open the Network Editor.............................................................................................. 137
- Network Editor interface ............................................................................................... 138
  - Hierarchy pane.......................................................................................................... 138
  - Detail pane.................................................................................................................. 139
  - Results pane............................................................................................................... 140
  - Menu bar..................................................................................................................... 140
  - Network Editor toolbar........................................................................................... 141
  - Status bar.................................................................................................................. 141
- Configure your network................................................................................................. 141
  - Select a network model........................................................................................... 142
  - Add a server............................................................................................................ 142
  - Remove a server....................................................................................................... 143
  - Synchronize network models............................................................................... 143
- User Preferences dialog box.......................................................................................... 144
  - User Preferences dialog box - General tab......................................................... 144
Table of contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Edit colors</td>
<td>145</td>
</tr>
<tr>
<td>Edit the application font</td>
<td>145</td>
</tr>
<tr>
<td>Edit information tips</td>
<td>146</td>
</tr>
<tr>
<td>User Preferences dialog box - Views tab</td>
<td>147</td>
</tr>
<tr>
<td>List View properties</td>
<td>147</td>
</tr>
<tr>
<td>User Preferences dialog box - Reports tab</td>
<td>148</td>
</tr>
<tr>
<td>Set report preferences</td>
<td>149</td>
</tr>
<tr>
<td>Change margins</td>
<td>151</td>
</tr>
<tr>
<td>Set the report fonts and colors</td>
<td>151</td>
</tr>
<tr>
<td>Column and row preferences in reports</td>
<td>153</td>
</tr>
<tr>
<td>Report printing formats</td>
<td>154</td>
</tr>
<tr>
<td>Preview reports</td>
<td>155</td>
</tr>
<tr>
<td>Print reports</td>
<td>156</td>
</tr>
<tr>
<td>Print a range of reports</td>
<td>156</td>
</tr>
</tbody>
</table>

**Chapter 12**

Server

- Abnormal FactoryTalk Batch termination                                    159
- Copied log file filename                                                  159
- Abnormal termination batch record entries                                 160
- Tag verification log information                                          161
- Tag verification log file                                                 161
- Windows Event Log                                                        162
- FactoryTalk Batch performance chart                                       163
- Open the FactoryTalk Batch performance chart                             163
- Batch identification                                                      164
- Enable editing of batch IDs                                               164
- Disable editing of batch IDs                                              165
- Define a custom default batch ID                                          165
- Create a custom batchidcreation.dll                                      166

**Chapter 13**

- New user account requirements for the FactoryTalk Batch Server           169
- Configure your FactoryTalk Batch system                                   170
- Server account password change locations                                 171
- Change the server user account for the FactoryTalk Batch Server service  172
- Change the server user account in FactoryTalk Batch COM+ applications    172
- Change the server user account in My Computer properties                 173
Chapter 14

Run multiple instances of FactoryTalk Batch View

Enable multiple FactoryTalk Batch View instances ........................................ 176
   Custom configuration files ........................................................................ 177
      Copy and rename a default configuration file ...................................... 177
      Create configuration files when launching FactoryTalk Batch View ... 178
   Configuration file log location .................................................................. 179
Start FactoryTalk Batch View from the command line ................................. 180
Start FactoryTalk Batch View with a configuration file ............................... 180

Appendix A

General troubleshooting errors

Automatic repair of batch component installation fails ............................... 183
FactoryTalk Batch Server does not start ..................................................... 183
FactoryTalk Batch Server runs for a specific period of time and then stops, and consistently stops at the same time interval ............................... 183
FactoryTalk Batch Service Manager does not display the FactoryTalk Batch Server or FactoryTalk Event Archiver ................................................. 184
FactoryTalk Batch Service Manager displays Batch.Server Class instead of Batch Server ............................................................................. 184
The FactoryTalk Batch View/Client is properly configured, but cannot view the Event Journals ................................................................. 184
The text in FactoryTalk Batch dialog boxes does not display properly ....... 184
Unable to locate files ................................................................................... 184
An Application Log error is generated ......................................................... 184
A Licensing error message is generated in Visual Basic ......................... 185
Recipes not visible from FactoryTalk Batch View program ..................... 185
List of available SQL server databases not displayed in the Server Options dialog box .......................................................... 186
Event Journal files are not being created ................................................. 186

Legal Notices

Index
Chapter 1

FactoryTalk Batch components

This guide contains instructions for tasks specific to FactoryTalk® 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

Document organization on page 9

FactoryTalk Batch regional settings on page 10

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, the Server user account, and multiple instances of FactoryTalk Batch View
- General troubleshooting

See also

FactoryTalk Batch Communications on page 11

FactoryTalk Batch security on page 13

The FactoryTalk Batch Server on page 87

Batch identification on page 163
FactoryTalk Batch components are supported in U.S. English only. Install the English language version of Windows Server, Windows 7, or Windows 8.1 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 13

- FactoryTalk Batch communications on page 11
Chapter 2

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 11

Use OLE for Process Control (OPC) on page 12

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 OLE for Process Control on page 12
Use OLE for Process Control (OPC)

The FactoryTalk Batch Server supports the use of the OLE for Process Control (OPC) communications 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 to communicate with FactoryTalk Batch View, SignatureList ActiveX control and PC-Based phases. COM is used to communicate with ActiveX controls or other high-level applications such as FactoryTalk Batch Material Manager.

See also

OPC group support on page 12

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 OLE for Process Control (OPC) on page 12
Chapter 3

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, click Start, point to All Programs > Rockwell Software > FactoryTalk Tools, and then click FactoryTalk Help.

See also

FactoryTalk Batch Server user account configuration requirements on page 13

FactoryTalk Batch system configuration on page 14

FactoryTalk Security for the Batch clients on page 21

FactoryTalk Batch default policy settings on page 36

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 14
FactoryTalk Batch security on page 13

FactoryTalk Batch system configuration

The FactoryTalk Batch and FactoryTalk eProcedure installation programs configure 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 14
FactoryTalk Batch Server install configuration on page 15
FactoryTalk Batch Client installation configuration on page 16
FactoryTalk Batch Windows Firewall exceptions on page 16
Add the Batch COM service on page 17

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 15
- FactoryTalk Batch Client installation configuration on page 16
- FactoryTalk Batch Windows Firewall exceptions on page 16
- Add the Batch COM service on page 17
- FactoryTalk Batch system configuration on page 14

During the FactoryTalk Batch Server installation, the Windows user group batchsvr_group is created 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 programs for the FactoryTalk Batch Server and FactoryTalk eProcedure Server applications require you to enter the FactoryTalk Batch Server user account name and password to configure the server computer. The installation adds the FactoryTalk Batch Server user account to these groups and services:

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

The services for the FactoryTalk Batch Server, FactoryTalk Event Archiver, eProcedure Server, and the Batch COM+ Application require identical user accounts to support the FactoryTalk Batch Server functionality.
Important: If an invalid FactoryTalk Batch Server user account is entered during installation, the Batch COM+ Application fails to install.

If you change the FactoryTalk Batch Server user account once the installation is complete, you must configure your FactoryTalk Batch system to use the new server user name and password.

See also

FactoryTalk Batch system configuration on page 14

FactoryTalk Batch Client installation configuration on page 16

FactoryTalk Batch Windows Firewall exceptions on page 16

Add the Batch COM service on page 17

FactoryTalk Batch Client installation configuration

When you install the FactoryTalk Batch and FactoryTalk eProcedure Client applications, you are prompted for the FactoryTalk Batch Server user account name.

If you change the FactoryTalk Batch Server user account after the installation is complete, you must configure your FactoryTalk Batch system to use the new server user name and password.

If the FactoryTalk Batch Server user account is a workgroup account or if the server computer needs to access files on the FactoryTalk Batch Client computer, add the server user account to each client computer.

See also

FactoryTalk Batch system configuration on page 14

FactoryTalk Batch Server install configuration on page 15

FactoryTalk Batch Windows Firewall exceptions on page 16

Change the FactoryTalk Batch Server user account on page 169
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:

<table>
<thead>
<tr>
<th>Windows Firewall Exceptions</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>• batchsvr.exe (Batch server installs)</td>
<td>• Port 135 TCP</td>
</tr>
<tr>
<td>• batchvew.exe (Batch client installs)</td>
<td>• Port 80 TCP (eProcedure Client installs)</td>
</tr>
<tr>
<td>• DaClient.exe</td>
<td>• RnaDiagnosticsSrv.exe</td>
</tr>
<tr>
<td>• EventClientMultiplexer.exe</td>
<td>• RnaDiagReceiver.exe</td>
</tr>
<tr>
<td>• EventServer.exe</td>
<td>• RnaDirServer.exe</td>
</tr>
<tr>
<td>• iexplore.exe</td>
<td>• RsvcHost.exe</td>
</tr>
<tr>
<td>• OPCEnum.exe</td>
<td>• VStudio.exe</td>
</tr>
</tbody>
</table>

If the Windows Firewall is enabled after you install the applications, run the FactoryTalk Windows Firewall Configuration Utility on each computer in your FactoryTalk Batch system to add the exceptions.

To run the FactoryTalk Windows Firewall Configuration Utility, click Start > All Programs > Rockwell Software > FactoryTalk Tools > Windows Firewall Configuration Utility.

See also

- FactoryTalk Batch system configuration on page 14
- System-wide COM security limit settings on page 14
- FactoryTalk Batch Server install configuration on page 15
- FactoryTalk Batch Client installation configuration on page 16
- Add the Batch COM service on page 17

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 > Rockwell Software > Batch > Scripts.
2. Double-click `BatchIDCreationCOMPlusInstall.vbs` to add the FactoryTalk Batch COM service to the computer.


4. Right-click `Batch > Properties`.

5. Click the `Identity` tab, select `This user`, and enter the user name and password for the FactoryTalk Batch Server.

6. Click `OK` and then close Component Services.

7. Click `Start > Administrative Tools > Services`.


9. Click the `Log On` tab and then select `This Account`.

10. Enter the user account name and password for the server.

11. Click `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.


13. Restart the computer.

See also

- [FactoryTalk Batch Server install configuration](#) on page 15
- [FactoryTalk Batch Client installation configuration](#) on page 16
- [FactoryTalk Batch Windows Firewall exceptions](#) on page 16
- [FactoryTalk Batch system configuration](#) on page 14

Set security on folders or files

Limit log file and event journal (.evt) 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) 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.

**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**

FactoryTalk Batch Server user account configuration requirements on page 13

Add users and user groups

Set security on folders and files to limit log file and event journal (.evt) 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 18

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.

See also

Set security on folders or files on page 18

Security settings are shared among FactoryTalk enabled products across a network or on the same computer using FactoryTalk Security in FactoryTalk Batch View, ActiveX 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 > All Programs > Rockwell Software > FactoryTalk Tools > 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.
Chapter 3  FactoryTalk Batch security

- **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](page 36)
- [Tighten security for FactoryTalk Batch Clients](page 22)

**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 going down 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**. Secure the **FactoryTalk Batch Configuration Options** product policies in the Local Directory to avoid tampering or unauthorized changes.

- 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**.
FactoryTalk Batch security

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. To secure FactoryTalk Batch, remove or restrict the All Users group for FactoryTalk Batch product policies. For example, to restrict access to FactoryTalk Batch View windows, remove the All Users group from the associated product policies and then add the necessary user accounts.

- 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. For a complete list of FactoryTalk Batch product policies, see FactoryTalk Batch default policy settings.

See also

Security for FactoryTalk Batch commands on page 24

Configure FactoryTalk security to specify which FactoryTalk Batch View toolbar buttons and windows are available to each logged on user. Removing a user group from a security policy disables the corresponding toolbar button for all users who are members of that user group. When a button is disabled, it no longer appears on the toolbar.

Restrict access to resources based on where a user is physically located, such as a computer used to perform actions. (For more information, see FactoryTalk Help.)

To open FactoryTalk Help, click Start, point to All Programs > Rockwell Software > FactoryTalk Tools, and then click FactoryTalk Help.

See also

Security for FactoryTalk Batch commands on page 24

To tighten security for a FactoryTalk Batch View window, remove the All Users group from the corresponding product policy and then add the necessary user accounts.

To configure security for the FactoryTalk Batch View windows:

1. Open the FactoryTalk Administration Console and log on to the appropriate FactoryTalk Directory.
2. Expand **System > Policies > Product Policies > Batch > BatchView & ActiveX**.

3. Right-click **View** and then select **Properties**.

4. Select the policy setting you want to configure and then click the corresponding browse button.

5. In the **Configure Securable Action** dialog box:
   - Select a user or group and then click **Remove**.
   - Allow/deny a user or group access to the feature by selecting or clearing the corresponding check box and then clicking **OK**.
     **Tip:** Product policies do not inherit security settings. When specifying permissions for product policies, clearing both the **Allow** and **Deny** check boxes denies access to the product feature.
   - 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 and Computer** dialog box you can:
   - Select a user, user group or computer 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 **View Properties** dialog box.

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

**See also**

[FactoryTalk Batch View security on page 23](#)

**Security for FactoryTalk Batch commands**

Configure FactoryTalk Security to require user confirmation for batch commands and phase commands. A command that has the confirm feature enabled does not execute until the user account is validated in the **Log on to Confirm User** dialog box.

To enable the Confirm feature, configure the FactoryTalk Batch product policy for the command and then set the associated **<Command>** Confirm policy setting to **True**. These policy settings are defined in FactoryTalk Directory for each command button within FactoryTalk Batch View and every phase command issued from the **Phase Control** window.
Tip: When the Confirm feature is enabled for a command, only active user accounts assigned to the Command are allowed to execute the command. Users validated in the Log on to Confirm User dialog box are not logged on to FactoryTalk Batch View.

The <Command> Confirm policy settings are found in the following locations:

- FactoryTalk Directory\Local | Network\System\Policies\Product Policies\Batch\BatchView and ActiveX\Commands
- FactoryTalk Directory\Local | Network\System\Policies\Product Policies\Batch\BatchView and ActiveX\Phase Commands

See also

FactoryTalk Batch View security on page 23
Configure security for FactoryTalk Batch commands on page 25

To allow only specified users to issue commands against a batch or a phase, add the appropriate users or user groups to the Command or Phase Command policy setting. Then remove or restrict the All Users group.

To configure security for FactoryTalk Batch commands:

1. Open the FactoryTalk Administration Console and log on to the appropriate FactoryTalk Directory.
2. Expand System > Policies > Product Policies > Batch > BatchView & ActiveX.
3. Right-click **Commands** and then select **Properties**.

![Command Properties Dialog Box](image)

4. Select the policy setting to configure and then select the corresponding browse button.

5. (optional) In the **Configure Securable Action** dialog box:
   - Select a user or group and then select **Remove**.
   - Allow or deny a user or group access to the feature by selecting or clearing the corresponding check box and then selecting **OK**.
   - Add a user account or user group by selecting **Add**. This option displays the **Select User and Computer** dialog box.

6. (optional) In the **Select User or Group** dialog box:
   - Select a user, user group, computer, or computer group and then select **OK**.
   - Select **Create New** to create and then add a new user, user group, computer, or computer group.

7. Select **OK** to close the **View Properties** dialog box.

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

See also

- **FactoryTalk Batch View security** on page 23
- **Security for FactoryTalk Batch commands** on page 24
Enable the FactoryTalk Batch View confirm settings

If specific users and/or groups are assigned to a batch command or phase command and the corresponding Confirm policy setting is enabled, only the specified users are allowed to issue the command. All commands in FactoryTalk Batch View have a command confirmation option. If confirm is enabled for a command, that command is not issued until a user with the appropriate permissions is validated using the Log on to Confirm User dialog box. Users validated in the Log on to Confirm User dialog box are not logged in to the FactoryTalk Batch View.

To enable the FactoryTalk Batch View confirm settings:

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

2. Expand System > Policies > Product Policies > Batch > BatchView & ActiveX.

3. Right-click Phase Commands (or Commands) and then select Properties.

4. Scroll to the BatchView and ActiveX - Phase Commands (or Commands) - Confirmations policy settings. Select the policy setting and then click the corresponding drop-down list box.

5. Select True to require a user log on each time the command button is clicked. Select False if no log on is required.

6. Click OK.
Set the automatic logoff time period

Configure a specified period of inactivity before logging off a user using FactoryTalk Batch View security. FactoryTalk Batch View uses the Windows screen saver activation to send a log off message. If the Batch List window of FactoryTalk Batch View is the top window in your system when the screen saver activates, FactoryTalk Batch View logs off the current user, the status bar displays No authenticated User, and FactoryTalk Batch View continues to run.

To set the automatic logoff time period:

1. Right-click the Windows desktop.
2. From the shortcut menu, select Properties.
3. From the Display Properties dialog box, click the Screen Saver tab.
4. Select a screen saver from the list, and then enter the automatic log off time period in the Wait box.
5. Click OK to return to the desktop.

See also

- Security for FactoryTalk Batch commands on page 24
- Change the logged-on user with your HMI on page 28
- External security devices on page 30

Change the logged-on user with your HMI

Configure your human-machine interface (HMI) to notify FactoryTalk Batch View to change the logged on user. The HMI sends a user name and password so the FactoryTalk Batch View can attempt to log on the specified user.

A Component Object Model (COM) component was created to log on and off of FactoryTalk Batch either programmatically, or by using an HMI. The following example, created in Visual Basic, is one way to implement external log-ons using the COM component.

Tip: Restart all FactoryTalk Batch components to update changes made in the FactoryTalk Directory.
To change the logged-on user with your HMI:

1. In Visual Basic, create a new project and add a reference to the BatchViewUser 1.0 Type Library.

![Reference - View Login.vb](image1)

2. Create a Visual Basic form that allows a user to enter a user name and a password.

3. Add two Command buttons, one for user log-ons and another to log off. Here is an example of a Visual Basic form:

![View Login](image2)

4. Create a new COM object and add code to support the log on functionality.
The following code sample demonstrates how to create a COM object in Visual Basic that is used for logging on and off of FactoryTalk Batch View.

```vbnet
Dim objView As BatchView.CBVUser

Private Sub Command1_Click()
    objView.Login txtUserName.Text, txtPassword.Text
End Sub

Private Sub Command2_Click()
    objView.Logout
End Sub

Private Sub Form_Load()
    Set objView = New BatchView.CBVUser
End Sub
```

5. Save the project and make an executable file.

6. Run the executable to log users on and off of FactoryTalk Batch View.

**Tip:** DDE is not a supported communication protocol with the FactoryTalk Batch suite of products.

**See also**

- Security for FactoryTalk Batch commands on page 24
- Set the automatic logoff time period on page 28
- External security devices on page 30

### External security devices

Use an external security device, such as a retinal scanner, with the FactoryTalk Batch View. You must use a .dll with a function that collects the user name and password. When the FactoryTalk Batch View security prompts the user for user name and password, your .dll is called. Your function is expected to copy the user name to `<USERNAME>` and the password to `<PASSWORD>`. The size limit on each of these is 1023 bytes.

The following is an example of a function that would return the user name and password to the FactoryTalk Batch View:

```c
int <func>(char username, char password)
return code of 0 == SUCCESS
```
Any value other than 0 is an error and an entry is recorded in the Batchview.log file. If the .dll fails to load, the function is not found in the .dll, or the function causes an exception, the log on mechanism is disabled. The standard logon dialog box must then be used.

The FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor also support an external security device. The product policies used to configure an external security device for the FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor are created in the following FactoryTalk Local Directory locations.

- System > Policies > Product Policies > Batch > Equipment Editor > Configuration > Options - External Login
- System > Policies > Product Policies > Batch > Recipe Editor > Configuration > Options - External Login

See also

Enable an external security device on page 31
Security for FactoryTalk Batch commands on page 24
Change the logged-on user with your HMI on page 28

Enable an external security device

Use the FactoryTalk Administration Console to modify the required security policy settings for enabling an external security device. These policy settings are located in the FactoryTalk Local Directory.

To enable an external security device:

1. Run the FactoryTalk Administration Console and log on to the Local Directory.

2. Expand System > Policies > Product Policies > Batch > BatchView & ActiveX > Configuration.

3. Right-click Options and then select Properties.

4. Edit the following policy settings:
   - Enabled = True
   - Name of DLL = <YourDLLNameWithPath>
   - Name of Function = <FunctionNameInDLL>
Chapter 3  FactoryTalk Batch security

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

See also

External security devices on page 30

Security for FactoryTalk Batch commands on page 24

Change the logged-on user with your HMI on page 28

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.
Configure access mode security

Configure access mode security

Configure feature security on page 34

Product policies in the local and network directories on page 36

Modify FactoryTalk Batch Equipment Editor configuration options on page 35

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.

See also

- FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor security on page 32
- Configure feature security on page 34
- Modify FactoryTalk Batch Equipment Editor configuration options on page 35

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).


4. Select Area Model: Secure and then click browse (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 32
- Configure access mode security on page 33
- Modify FactoryTalk Batch Equipment Editor configuration options on page 35

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 33
- Configure FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor security on page 32
- Configure feature security on page 34
- Product policies in the local and network directories on page 36
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 36

The following product policy configurations are created in the FactoryTalk Local and Network Directories for FactoryTalk Batch.

<table>
<thead>
<tr>
<th>Security Objects</th>
<th>Product Policy Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>View</strong></td>
<td>These policy settings are used to secure the FactoryTalk Batch View windows and the Exit, Configuration, and Help buttons. Users must have Allow permissions to access the View windows and buttons. Default = All Users have access. To tighten security, remove or restrict the All Users group and then add the necessary user accounts to the policy settings.</td>
</tr>
<tr>
<td><strong>Alarm Summary</strong>— Access the Alarm Summary window</td>
<td></td>
</tr>
<tr>
<td><strong>Arbitration</strong>— Access the Arbitration window</td>
<td></td>
</tr>
<tr>
<td><strong>Batch List</strong>— Access the Batch List window</td>
<td></td>
</tr>
<tr>
<td><strong>Configuration</strong>— Configure the Batch View</td>
<td></td>
</tr>
<tr>
<td><strong>Event Journals</strong>— Access the Event Journal window</td>
<td></td>
</tr>
<tr>
<td><strong>Exit</strong>— Exit the FactoryTalk Batch View</td>
<td></td>
</tr>
<tr>
<td><strong>Help</strong>— Activate help file</td>
<td></td>
</tr>
<tr>
<td><strong>Phase Control</strong>— Access the Phase Control window</td>
<td></td>
</tr>
<tr>
<td><strong>Phase Summary</strong>— Access the Phase Summary window</td>
<td></td>
</tr>
<tr>
<td><strong>Procedure as SFC</strong>— Access the SFC View window</td>
<td></td>
</tr>
<tr>
<td><strong>Procedure as Table</strong>— Access the Table View window</td>
<td></td>
</tr>
<tr>
<td><strong>Prompts</strong>— Access the Unacknowledged Prompts window</td>
<td></td>
</tr>
<tr>
<td><strong>Signatures</strong>— Access the Signature List window</td>
<td></td>
</tr>
</tbody>
</table>
### View — Confirmations

- **Configuration** — Confirm

This policy setting is used to secure access to the **System Configuration and Defaults** dialog box in the FactoryTalk Batch View. If the **Configuration-Confirm** policy is **True**, then only the users defined in the **Configuration** policy setting are allowed access to **System Configuration and Defaults** dialog box. Users are validated using the **Log On to FactoryTalk** dialog box. Default = **False**.

### Phase Commands

- **Abort** — Abort a phase
- **Ack** — Acknowledge a phase prompt
- **Acquire** — Acquire a phase
- **ClearFail** — Clear all failures
- **Disconnect** — Disconnect a phase
- **Hold** — Hold a phase
- **Pause** — Pause a phase
- **Phase Auto Mode** — Place phase in auto mode
- **Phase Semi-Auto Mode** — Place phase in semi-auto
- **Release** — Release phase
- **Reset** — Reset phase
- **Restart** — Restart phase
- **Resume** — Resume phase
- **Start** — Start phase
- **Stop** — Stop phase

Phase commands are secured objects that can be configured to require user validation prior to execution. To require user validation for a phase command, set the associated **Confirm** policy setting to **True**. To tighten security, remove or restrict the **All Users** group and then add the necessary user accounts to the policy setting. Default = **All Users** have access.
<table>
<thead>
<tr>
<th>Security Objects</th>
<th>Product Policy Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase Commands — Confirmations</td>
<td>If a <strong>Phase Command — Confirm</strong> policy is <strong>True</strong>, then only the users defined in the associated <strong>Phase Command</strong> policy setting are allowed to execute the phase command. Users are validated using the <strong>Log On to Confirm User</strong> dialog box. Phase commands are issued from the <strong>Manual Phase Control</strong> window in FactoryTalk Batch View. Confirm settings are configured separately for each phase command. Users validated in the <strong>Log on to Confirm User</strong> dialog box are not logged on to FactoryTalk Batch View. <strong>Default = False.</strong></td>
</tr>
<tr>
<td>• Abort — Confirm</td>
<td></td>
</tr>
<tr>
<td>• Ack — Confirm</td>
<td></td>
</tr>
<tr>
<td>• Acquire — Confirm</td>
<td></td>
</tr>
<tr>
<td>• ClearFail — Confirm</td>
<td></td>
</tr>
<tr>
<td>• Disconnect — Confirm</td>
<td></td>
</tr>
<tr>
<td>• Hold — Confirm</td>
<td></td>
</tr>
<tr>
<td>• Pause — Confirm</td>
<td></td>
</tr>
<tr>
<td>• PhaseAuto Mode — Confirm</td>
<td></td>
</tr>
<tr>
<td>• PhaseSemi-AutoMode — Confirm</td>
<td></td>
</tr>
<tr>
<td>• Release — Confirm</td>
<td></td>
</tr>
<tr>
<td>• Reset — Confirm</td>
<td></td>
</tr>
<tr>
<td>• Restart — Confirm</td>
<td></td>
</tr>
<tr>
<td>• Resume — Confirm</td>
<td></td>
</tr>
<tr>
<td>• Start — Confirm</td>
<td></td>
</tr>
<tr>
<td>• Stop — Confirm</td>
<td></td>
</tr>
</tbody>
</table>
Batch commands are secured objects that can be configured to require user validation prior to execution. To require user validation for a batch command, set the associated Confirm policy setting to True. To tighten security, remove or restrict the All Users group and then add the necessary user accounts to the policy setting. Default = All Users have access.

<table>
<thead>
<tr>
<th>Security Objects</th>
<th>Product Policy Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Commands</strong></td>
<td>Batch commands are secured objects that can be configured to require user validation prior to execution. To require user validation for a batch command, set the associated Confirm policy setting to True. To tighten security, remove or restrict the All Users group and then add the necessary user accounts to the policy setting. Default = All Users have access.</td>
</tr>
<tr>
<td>• Abort — Abort batch</td>
<td></td>
</tr>
<tr>
<td>• Ack — Acknowledge a prompt</td>
<td></td>
</tr>
<tr>
<td>• Acquire — Acquire a resource</td>
<td></td>
</tr>
<tr>
<td>• Add Batch — Add a Batch</td>
<td></td>
</tr>
<tr>
<td>• Auto — Place batch in auto mode</td>
<td></td>
</tr>
<tr>
<td>• Bind — Bind step to unit</td>
<td></td>
</tr>
<tr>
<td>• BindAck — Acknowledge a bind prompt</td>
<td></td>
</tr>
<tr>
<td>• ClearFail — Clear all failures</td>
<td></td>
</tr>
<tr>
<td>• Comment — Operator comment to Event Journal</td>
<td></td>
</tr>
<tr>
<td>• Disconnect — Disconnect a phase</td>
<td></td>
</tr>
<tr>
<td>• Hold — Place batch on hold</td>
<td></td>
</tr>
<tr>
<td>• Manual — Place batch in manual mode</td>
<td></td>
</tr>
<tr>
<td>• MatSvrControl — Control the Material Server</td>
<td></td>
</tr>
<tr>
<td>• ParamChange — Change parameter value</td>
<td></td>
</tr>
<tr>
<td>• ReactivateStep — Change the active eProcedure step</td>
<td></td>
</tr>
<tr>
<td>• Release — Release a resource</td>
<td></td>
</tr>
<tr>
<td>• Remove — Remove a batch</td>
<td></td>
</tr>
<tr>
<td>• Reorder — Reprioritize needed resources list</td>
<td></td>
</tr>
<tr>
<td>• Restart — Restart batch</td>
<td></td>
</tr>
<tr>
<td>• Resume — Resume batch</td>
<td></td>
</tr>
<tr>
<td>• Semi-Auto — Place batch in semi-auto mode</td>
<td></td>
</tr>
<tr>
<td>• Start — Start batch</td>
<td></td>
</tr>
<tr>
<td>• Step — Change the active step</td>
<td></td>
</tr>
<tr>
<td>• Stop — Stop the batch</td>
<td></td>
</tr>
</tbody>
</table>
### Security Objects

<table>
<thead>
<tr>
<th>Commands — Confirmations</th>
<th>Product Policy Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Abort — Confirm</td>
<td>If a Command — Confirm policy is True, then only the users defined in the associated Command policy setting are allowed to execute the batch command. Users are validated using the Log On to Confirm User dialog box. Users validated in the Log on to Confirm User dialog box are not logged on to FactoryTalk Batch View. Default = False.</td>
</tr>
<tr>
<td>Ack — Confirm</td>
<td></td>
</tr>
<tr>
<td>Acquire — Confirm</td>
<td></td>
</tr>
<tr>
<td>Add Batch — Confirm</td>
<td></td>
</tr>
<tr>
<td>Auto — Confirm</td>
<td></td>
</tr>
<tr>
<td>Bind — Confirm</td>
<td></td>
</tr>
<tr>
<td>BindAck — Confirm</td>
<td></td>
</tr>
<tr>
<td>ClearFail — Confirm</td>
<td></td>
</tr>
<tr>
<td>Comment — Confirm</td>
<td></td>
</tr>
<tr>
<td>Disconnect — Confirm</td>
<td></td>
</tr>
<tr>
<td>Hold — Confirm</td>
<td></td>
</tr>
<tr>
<td>Manual — Confirm</td>
<td></td>
</tr>
<tr>
<td>MatSvrControl — Confirm</td>
<td></td>
</tr>
<tr>
<td>ParamChange — Confirm</td>
<td></td>
</tr>
<tr>
<td>ReactivateStep — Confirm</td>
<td></td>
</tr>
<tr>
<td>Release — Confirm</td>
<td></td>
</tr>
<tr>
<td>Remove — Confirm</td>
<td></td>
</tr>
<tr>
<td>Reorder — Confirm</td>
<td></td>
</tr>
<tr>
<td>Restart — Confirm</td>
<td></td>
</tr>
<tr>
<td>Resume — Confirm</td>
<td></td>
</tr>
<tr>
<td>Semi-Auto — Confirm</td>
<td></td>
</tr>
<tr>
<td>Start — Confirm</td>
<td></td>
</tr>
<tr>
<td>Step — Confirm</td>
<td></td>
</tr>
<tr>
<td>Stop — Confirm</td>
<td></td>
</tr>
<tr>
<td>Security Objects</td>
<td>Product Policy Descriptions</td>
</tr>
<tr>
<td>--------------------------</td>
<td>--------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Equipment Editor — Access Modes</td>
<td></td>
</tr>
<tr>
<td>• Full Edit</td>
<td>Once logged on, the user can edit and save changes to the area model. To tighten security,</td>
</tr>
<tr>
<td></td>
<td>remove or restrict the All Users group and then add the necessary user accounts.</td>
</tr>
<tr>
<td></td>
<td>Default = All Users have access.</td>
</tr>
<tr>
<td>• ViewOnly</td>
<td>Once logged on, the user can view and edit the Area Model but cannot save changes.</td>
</tr>
<tr>
<td></td>
<td>Default = All Users have access.</td>
</tr>
</tbody>
</table>

| Equipment Editor — Feature Security |                                                                                             |
|• Area Model: Secure              | 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. |
### Recipe Editor — Access Modes

<table>
<thead>
<tr>
<th>Security Objects</th>
<th>Product Policy Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>• Full Edit</strong></td>
<td>Once logged on, the user can edit and save changes to recipes. To tighten security, remove or restrict the <em>All Users</em> group and then add the necessary user accounts. Default = <em>All Users</em> have access.</td>
</tr>
<tr>
<td><strong>• ViewOnly</strong></td>
<td>Once logged on, the user can view recipes, but cannot make changes or save the recipe with a new name using the <em>Save As</em> function. Default = <em>All Users</em> have access.</td>
</tr>
</tbody>
</table>

### Recipe Editor -- Feature Security

<table>
<thead>
<tr>
<th>Security Objects</th>
<th>Product Policy Descriptions</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>• Recipe: Secure</strong></td>
<td>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.</td>
</tr>
</tbody>
</table>

**See also**

[FactoryTalk Batch default policy settings](page 36) on page 36
Chapter 4

FactoryTalk Batch electronic signatures

FactoryTalk Batch electronic signatures are electronically recorded representations of a signature and their associated data. Signature data includes timestamps, 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, ActiveX 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 44
- Parameter deviation verification policies on page 54
- General usage phase logic requests on page 57
- Electronic signature security on page 58

Command verification policies

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

Actions initiated in FactoryTalk Batch View, ActiveX 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 44
- FactoryTalk Batch Parameter Prompt Acknowledge Signature request on page 47
- Unit Bind Signature request on page 50
- Active Step Change Signature request on page 54

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. |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Response</td>
<td>An ActionID is returned by the execute used to locate the generated signature.</td>
</tr>
</tbody>
</table>
| 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. |
| 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  
SEMINA-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

<table>
<thead>
<tr>
<th>Batch Server</th>
<th>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.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Exceptions</td>
<td>• Phase Reset and Phase Disconnect.</td>
</tr>
<tr>
<td></td>
<td>• Batch is created regardless of whether a Signature is required, or if the Signature is canceled.</td>
</tr>
<tr>
<td></td>
<td>• ABORT Command is not queued.</td>
</tr>
<tr>
<td>Response</td>
<td>An ActionID is returned by the Execute used to locate the generated signature.</td>
</tr>
<tr>
<td>Completion</td>
<td>When all Signature Signoffs generated by the queued command are complete, then the Command is issued to the target procedure.</td>
</tr>
<tr>
<td>Cancellation</td>
<td>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.</td>
</tr>
<tr>
<td>Overlap</td>
<td>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.</td>
</tr>
<tr>
<td>Automatic Cancellation</td>
<td>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.</td>
</tr>
<tr>
<td>Warm Reboot</td>
<td>Signature Requests generated in response to a Phase Execute Action are System Canceled in the event of a FactoryTalk Batch Server warm boot.</td>
</tr>
</tbody>
</table>

### 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.
- ABORT Command cannot be canceled by the user.

### See also
- [Command verification policies](#) on page 44
- [Set parameter signature request](#) on page 46

### 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:

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>SetParm Execute</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>Response</strong></td>
<td>An ActionID is returned by the execute that is used to locate the generated signature(s).</td>
</tr>
<tr>
<td><strong>Completion</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>Cancellation</strong></td>
<td>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 Cancelled.</td>
</tr>
<tr>
<td><strong>Overlap</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>Automatic Cancellation</strong></td>
<td>If any incomplete Signature Requests associated with a SetParm Execute is user cancelled, then the Execute Action is terminated and it’s remaining incomplete Signature Requests are System Cancelled.</td>
</tr>
<tr>
<td><strong>Warm Reboot</strong></td>
<td>Signature Requests generated in response to a SetParm Execute Action are restored in the event of a FactoryTalk Batch Server warm boot.</td>
</tr>
</tbody>
</table>

**See also**

- Command verification policies on page 44

**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

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch Server</td>
<td>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.</td>
</tr>
<tr>
<td>Response</td>
<td>An ActionID is returned by the Execute used to locate the generated signature(s).</td>
</tr>
<tr>
<td>Completion</td>
<td>When all Signature Signoffs generated by the Execute are completed, then the parameter value is stored.</td>
</tr>
<tr>
<td>Cancellation</td>
<td>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.</td>
</tr>
<tr>
<td>Overlap</td>
<td>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.</td>
</tr>
<tr>
<td>Automatic Cancellation</td>
<td>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.</td>
</tr>
<tr>
<td>Warm Reboot</td>
<td>Signature Requests generated in response to a Parameter Ack Execute Action are System Cancelled in the event of a FactoryTalk Batch Server warm boot.</td>
</tr>
</tbody>
</table>

See also

[Command verification policies](#) on page 44

### 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:

<table>
<thead>
<tr>
<th><strong>Unit Bind Execute</strong></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Batch Server</strong></td>
</tr>
<tr>
<td><strong>Response</strong></td>
</tr>
<tr>
<td><strong>Completion</strong></td>
</tr>
<tr>
<td><strong>Cancellation</strong></td>
</tr>
<tr>
<td><strong>Overlap</strong></td>
</tr>
<tr>
<td><strong>Automatic Cancellation</strong></td>
</tr>
<tr>
<td><strong>Warm Reboot</strong></td>
</tr>
</tbody>
</table>

**See also**

*Command verification policies* on page 44

**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

<table>
<thead>
<tr>
<th>Batch Server</th>
<th>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.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Response</td>
<td>An ActionID is returned by the Execute used to locate the generated signature.</td>
</tr>
<tr>
<td>Completion</td>
<td>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.</td>
</tr>
<tr>
<td>Cancellation</td>
<td>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.</td>
</tr>
<tr>
<td>Overlap</td>
<td>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.</td>
</tr>
<tr>
<td>Automatic Cancellation</td>
<td>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.</td>
</tr>
<tr>
<td>Warm Reboot</td>
<td>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.</td>
</tr>
</tbody>
</table>

### See also

[Command verification policies](#) on page 44

### 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. |
### Bind Execute

<table>
<thead>
<tr>
<th><strong>Response</strong></th>
<th>An ActionID is returned by the Execute used to locate the generated signature.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Completion</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>Cancellation</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>Overlap</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>Automatic Cancellation</strong></td>
<td>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.</td>
</tr>
<tr>
<td><strong>Warm Reboot</strong></td>
<td>Signature Requests generated in response to a Bind Execute Action are restored in the event of a FactoryTalk Batch Server warm boot.</td>
</tr>
</tbody>
</table>

### See also

[Command verification policies on page 44](#)

#### 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:

<table>
<thead>
<tr>
<th>Remove Execute</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Batch Server</strong></td>
</tr>
<tr>
<td><strong>Response</strong></td>
</tr>
<tr>
<td><strong>Completion</strong></td>
</tr>
<tr>
<td><strong>Overlap</strong></td>
</tr>
<tr>
<td><strong>Cancellation</strong></td>
</tr>
</tbody>
</table>
Remove Execute

<table>
<thead>
<tr>
<th>Automatic Cancellation</th>
</tr>
</thead>
<tbody>
<tr>
<td>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).</td>
</tr>
<tr>
<td>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).</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Warm Reboot</th>
</tr>
</thead>
<tbody>
<tr>
<td>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.</td>
</tr>
</tbody>
</table>

See also

Command verification policies on page 44

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

<table>
<thead>
<tr>
<th>Batch Server</th>
</tr>
</thead>
<tbody>
<tr>
<td>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.</td>
</tr>
<tr>
<td>An AddEvent is not generated when all batches are selected.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Response</th>
</tr>
</thead>
<tbody>
<tr>
<td>An ActionID is returned by the execute used to locate the generated signature.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Completion</th>
</tr>
</thead>
<tbody>
<tr>
<td>When all Signature Signoffs associated with an AddEvent execute are completed, then the AddEvent action is handled if it is still legal.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Cancellation</th>
</tr>
</thead>
<tbody>
<tr>
<td>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.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Overlap</th>
</tr>
</thead>
<tbody>
<tr>
<td>AddEvent executes targeted at the same batch are permitted to overlap. If the batch has a Remove Action pending, then AddEvent execute fails.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Warm Reboot</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signature Requests generated in response to an AddEvent execute action are restored in the event of a FactoryTalk Batch Server warm boot.</td>
</tr>
</tbody>
</table>
Active Step Change Signalure 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:

<table>
<thead>
<tr>
<th>Step Execute</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch Server</td>
</tr>
<tr>
<td>Response</td>
</tr>
<tr>
<td>Completion</td>
</tr>
<tr>
<td>Cancellation</td>
</tr>
<tr>
<td>Overlap</td>
</tr>
<tr>
<td>Warm Reboot</td>
</tr>
</tbody>
</table>

See also

Command verification policies on page 44

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 55
- Report parameter deviation signature requests on page 56

### Parameter deviation signature requests

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

<table>
<thead>
<tr>
<th>Parameter Deviations</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Batch Server</strong></td>
</tr>
<tr>
<td>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, SETPARM and ACK executes. After checking for deviations, the server carries out the parameter’s configured verification policy. A verification policy of &quot;None&quot; has no action, meaning that no electronic signatures or failures are generated even if a deviation exists. When the verification policy is &quot;Signature Template&quot;, the server initiates an action as defined by the signature template. When the verification policy is &quot;Not Allowed&quot;, the server fails the parameter change. The SETPARM or ACK execute response will be: &quot;FAIL: Value is outside the permitted limits.&quot; 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. 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.</td>
</tr>
</tbody>
</table>

| **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 SETPARM 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. |
### Parameter Deviations

<table>
<thead>
<tr>
<th>Section</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overlap</td>
<td>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. 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.</td>
</tr>
<tr>
<td>Automatic Cancellation</td>
<td>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 SETPARM if any of the required signatures are cancelled.</td>
</tr>
<tr>
<td>Warm Reboot</td>
<td>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.</td>
</tr>
</tbody>
</table>

**See also**

- [Parameter deviation verification policies](#) on page 54
- [Report parameter deviation signature requests](#) on page 56

**Report parameter deviation signature requests**

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

<table>
<thead>
<tr>
<th>Report Parameter Deviations</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch Server</td>
<td>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 &quot;Not Allowed&quot;. In this case, the request fails and the value is not uploaded. When the verification policy is &quot;None&quot;, no electronic signatures or failures are generated, even if a deviation exists. When the verification policy is &quot;Signature Template&quot;, 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. 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.</td>
</tr>
<tr>
<td>Response</td>
<td>An ActionID is returned by the execute used to locate the generated signature.</td>
</tr>
<tr>
<td>Completion</td>
<td>Completion of a signature generated for report deviation acknowledgment does not result in any additional action.</td>
</tr>
<tr>
<td>Cancellation</td>
<td>Report deviation signature requests cannot be cancelled.</td>
</tr>
<tr>
<td>Automatic Cancellation</td>
<td>There is no scenario under which a report deviation signature request is system cancelled.</td>
</tr>
<tr>
<td>Warm Reboot</td>
<td>A FactoryTalk Batch Server warm boot restores report parameter deviation signature requests.</td>
</tr>
</tbody>
</table>
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

- [*FactoryTalk Batch electronic signatures*](#) on page 43
- [Parameter deviation verification policies](#) on page 54
- [Parameter deviation signature requests](#) on page 55

General usage signature requests

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

<table>
<thead>
<tr>
<th>General Usage Signatures</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch Server</td>
<td>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.</td>
</tr>
<tr>
<td>Response</td>
<td>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.</td>
</tr>
<tr>
<td>Completion</td>
<td>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.</td>
</tr>
<tr>
<td>Cancellation</td>
<td>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.</td>
</tr>
<tr>
<td>Overlap</td>
<td>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.</td>
</tr>
<tr>
<td>Automatic Cancellation</td>
<td>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.</td>
</tr>
<tr>
<td>Warm Reboot</td>
<td>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.</td>
</tr>
</tbody>
</table>
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

General usage phase logic requests on page 57

Configure the ESignature network directory policy setting on page 58

Configure the ESignature network directory policy setting

The **ESignature Network Directory** policy setting must be set to **True** to enable signature signoff permission validations across a network. For stand alone computers, set the **ESignature Network Directory** policy setting to **False**.

**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 FactoryTalk Administration Console.

2. Log on to the FactoryTalk Local Directory.
3. Point to System > Policies > Product Policies > Batch > Equipment Editor > Configuration > Options.

4. Right-click Options, and then click Properties. The Option Properties dialog box opens.

5. Select the ESignature Network Directory policy setting. If you are using Network security select True, to use Electronic Signatures on a stand alone computer, select False.

6. Click OK.

7. Restart the server and client applications.

See also

FactoryTalk Batch electronic signatures on page 43

Electronic signature security on page 58

FactoryTalk Batch default policy settings on page 36
Chapter 5

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 62

Audited FactoryTalk Batch Equipment Editor events on page 62

Audited FactoryTalk Batch Recipe Editor events on page 65

Audit messages on page 67
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.
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 61
Audited Equipment Editor events on page 62
Audited Recipe Editor events on page 65
Audit messages on page 67

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.
<table>
<thead>
<tr>
<th>Message Type</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>Area Model</td>
<td>- Invoke FactoryTalk Batch Equipment Editor</td>
</tr>
<tr>
<td></td>
<td>- Open an existing area model</td>
</tr>
<tr>
<td></td>
<td>- Create a new area model</td>
</tr>
<tr>
<td></td>
<td>- Save an area model</td>
</tr>
<tr>
<td></td>
<td>- Edit/update an area model property</td>
</tr>
<tr>
<td></td>
<td>- Delete an area model</td>
</tr>
<tr>
<td></td>
<td>- Add a signature template</td>
</tr>
<tr>
<td></td>
<td>- Delete a signature template</td>
</tr>
<tr>
<td></td>
<td>- Update a signature template</td>
</tr>
<tr>
<td></td>
<td>- Enable/disable recipe approvals</td>
</tr>
<tr>
<td></td>
<td>- Enable/disable recipe versioning</td>
</tr>
<tr>
<td></td>
<td>- Update a command verification policy</td>
</tr>
<tr>
<td>Area Model Security</td>
<td>- Security authority is configured for an area model</td>
</tr>
<tr>
<td></td>
<td>- Security authority identifier (SAI) is set or removed</td>
</tr>
<tr>
<td></td>
<td>- Security authority configuration change is accepted by user</td>
</tr>
<tr>
<td></td>
<td>- Security authority changes are saved to the area model</td>
</tr>
<tr>
<td>Data Server</td>
<td>- Add an OPC server</td>
</tr>
<tr>
<td></td>
<td>- Delete an OPC server</td>
</tr>
<tr>
<td></td>
<td>- Edit/update an OPC server property</td>
</tr>
<tr>
<td></td>
<td>- Add a Logix5000 CIP server</td>
</tr>
<tr>
<td></td>
<td>- Delete a Logix5000 CIP server</td>
</tr>
<tr>
<td></td>
<td>- Edit a Logix5000 CIP server</td>
</tr>
<tr>
<td></td>
<td>- Add an FTD server</td>
</tr>
<tr>
<td></td>
<td>- Delete an FTD server</td>
</tr>
<tr>
<td></td>
<td>- Edit an FTD server</td>
</tr>
<tr>
<td></td>
<td>- Edit/update an FTD server property</td>
</tr>
<tr>
<td>Enumeration Set</td>
<td>- Create an enumeration set</td>
</tr>
<tr>
<td></td>
<td>- Delete an enumeration set</td>
</tr>
<tr>
<td></td>
<td>- Edit/update an enumeration set property</td>
</tr>
<tr>
<td>Enumeration</td>
<td>- Add an enumeration</td>
</tr>
<tr>
<td></td>
<td>- Delete an enumeration</td>
</tr>
<tr>
<td></td>
<td>- Edit/update an enumeration property</td>
</tr>
<tr>
<td>Resource Class</td>
<td>- Add a resource class</td>
</tr>
<tr>
<td></td>
<td>- Delete a resource class</td>
</tr>
<tr>
<td></td>
<td>- Edit/update a resource class property</td>
</tr>
<tr>
<td>Resource</td>
<td>- Add a resource</td>
</tr>
<tr>
<td></td>
<td>- Delete a resource</td>
</tr>
<tr>
<td></td>
<td>- Edit/update a resource property</td>
</tr>
<tr>
<td>Process Cell Class</td>
<td>- Add a process cell class</td>
</tr>
<tr>
<td></td>
<td>- Copy a process cell class</td>
</tr>
<tr>
<td></td>
<td>- Delete a process cell</td>
</tr>
<tr>
<td></td>
<td>- Edit/update a process cell class property</td>
</tr>
<tr>
<td>Process Cell</td>
<td>- Add a process cell</td>
</tr>
<tr>
<td></td>
<td>- Copy a process cell</td>
</tr>
<tr>
<td></td>
<td>- Delete a process cell</td>
</tr>
<tr>
<td></td>
<td>- Edit/update a process cell property</td>
</tr>
</tbody>
</table>

**Message Type**

- **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

- **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 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
<table>
<thead>
<tr>
<th>Message Type</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>Unit Attribute</td>
<td>Add a unit attribute&lt;br&gt;Delete a unit attribute&lt;br&gt;Edit/update a unit attribute property</td>
</tr>
<tr>
<td>Unit</td>
<td>Add a unit&lt;br&gt;Copy a unit&lt;br&gt;Edit a unit&lt;br&gt;Delete a unit&lt;br&gt;Edit/update a unit property</td>
</tr>
<tr>
<td>Global Unit Binding Requirements</td>
<td>Add global unit attribute&lt;br&gt;Edit global unit attribute&lt;br&gt;Delete global unit attribute</td>
</tr>
<tr>
<td>Unit Flow Path</td>
<td>Link units with a flow path&lt;br&gt;Unlink units with a flow path</td>
</tr>
<tr>
<td>Equipment Phase Class (Recipe Phase)</td>
<td>Add an equipment phase class&lt;br&gt;Copy an equipment phase class&lt;br&gt;Delete an equipment phase class&lt;br&gt;Edit/update an equipment phase class property&lt;br&gt;Edit/update an equipment phase class control strategy set of parameters&lt;br&gt;Edit/update an equipment phase class optional material parameters property&lt;br&gt;Edit/update an equipment phase class control strategy set of reports&lt;br&gt;Edit/update an equipment phase class control strategy</td>
</tr>
<tr>
<td>Equipment Phase</td>
<td>Add an equipment phase&lt;br&gt;Copy an equipment phase&lt;br&gt;Delete an equipment phase&lt;br&gt;Edit/update an equipment phase property&lt;br&gt;Edit/update an equipment phase’s server property&lt;br&gt;Reset an equipment phase’s tags to default values&lt;br&gt;Share an equipment phase between units</td>
</tr>
<tr>
<td>Recipe Parameter</td>
<td>Add a recipe parameter&lt;br&gt;Delete a recipe parameter&lt;br&gt;Edit/update a recipe parameter property</td>
</tr>
<tr>
<td>Recipe Approval Step</td>
<td>Add a recipe approval step&lt;br&gt;Delete a recipe approval step&lt;br&gt;Edit/update a recipe approval step</td>
</tr>
<tr>
<td>Report Parameter</td>
<td>Add a report parameter&lt;br&gt;Delete a recipe parameter&lt;br&gt;Edit/update a recipe parameter property</td>
</tr>
<tr>
<td>Phase Message</td>
<td>Add a phase message&lt;br&gt;Delete a phase message&lt;br&gt;Edit/update a phase message property</td>
</tr>
<tr>
<td>Synchronize</td>
<td>Synchronize a phase</td>
</tr>
</tbody>
</table>
## 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.

<table>
<thead>
<tr>
<th>Message Type</th>
<th>Event</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recipe Object</td>
<td>Open an existing recipe</td>
</tr>
<tr>
<td></td>
<td>Create a new recipe</td>
</tr>
<tr>
<td></td>
<td>Save a recipe</td>
</tr>
<tr>
<td></td>
<td>Save all recipes</td>
</tr>
<tr>
<td></td>
<td>Remove a recipe (User-supplied comment when deleting)</td>
</tr>
<tr>
<td></td>
<td>Import a recipe (User-supplied comment when importing)</td>
</tr>
<tr>
<td></td>
<td>Export a recipe (User-supplied comment when exporting)</td>
</tr>
<tr>
<td></td>
<td>Rebuild the recipe directory (User-supplied comment when rebuilding</td>
</tr>
<tr>
<td></td>
<td>Delete an existing recipe</td>
</tr>
<tr>
<td>Recipe Print</td>
<td>Print a Procedure, Unit Procedure, or Operation</td>
</tr>
<tr>
<td>Recipe Header</td>
<td>Edit/update recipe header properties (except Recipe Name)</td>
</tr>
<tr>
<td>Recipe Step</td>
<td>Add a recipe step, insert step, or parallel recipe step</td>
</tr>
<tr>
<td></td>
<td>Delete a recipe step</td>
</tr>
<tr>
<td></td>
<td>Edit/update a recipe step</td>
</tr>
<tr>
<td>Recipe Approval Step</td>
<td>Signoff on an approval step</td>
</tr>
<tr>
<td></td>
<td>Revert an approval step</td>
</tr>
<tr>
<td>Message Type</td>
<td>Event</td>
</tr>
<tr>
<td>------------------</td>
<td>----------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Recipe Versioning</td>
<td>Check in a recipe (create a new version of a recipe)</td>
</tr>
<tr>
<td></td>
<td>Check out a recipe (create a work-in-progress (WIP) recipe)</td>
</tr>
<tr>
<td></td>
<td>A recipe is obsoleted</td>
</tr>
<tr>
<td></td>
<td>An obsolete recipe is saved</td>
</tr>
<tr>
<td></td>
<td>An obsolete recipe is reinstated</td>
</tr>
<tr>
<td></td>
<td>A reinstated recipe is saved</td>
</tr>
<tr>
<td></td>
<td>Version conflict on recipe import</td>
</tr>
<tr>
<td></td>
<td>Version conflict on recipe directory rebuild (including auto-rebuild of the directory when FactoryTalk Batch Recipe Editor opens)</td>
</tr>
<tr>
<td></td>
<td>A recipe is deleted due to version conflict resolution (during import or recipe directory rebuild)</td>
</tr>
<tr>
<td></td>
<td>Signoff(s) removed due to verification failure (for versioned and non-versioned recipes)</td>
</tr>
<tr>
<td>Recipe Security</td>
<td>Change a recipe security state</td>
</tr>
<tr>
<td></td>
<td>Edit/modify a security authority identifier (SAI)</td>
</tr>
<tr>
<td></td>
<td>Change a FactoryTalk Services Platform SAI host computer</td>
</tr>
<tr>
<td>Recipe Text Box</td>
<td>Add a text box</td>
</tr>
<tr>
<td></td>
<td>Edit a text box (only if comment actually changes)</td>
</tr>
<tr>
<td></td>
<td>Delete a text box</td>
</tr>
<tr>
<td></td>
<td>Associate a text box</td>
</tr>
<tr>
<td></td>
<td>Disassociate a text box</td>
</tr>
<tr>
<td>Recipe Formula Parameter</td>
<td>Add a recipe formula parameter to a step</td>
</tr>
<tr>
<td></td>
<td>Delete a recipe formula parameter from a step</td>
</tr>
<tr>
<td></td>
<td>Edit/update a recipe formula parameter of a step</td>
</tr>
<tr>
<td>Formula Value</td>
<td>Edit/update a formula value’s non-origin properties</td>
</tr>
<tr>
<td></td>
<td>Edit/update a formula value’s origin property</td>
</tr>
<tr>
<td></td>
<td>Edit/update a formula value’s value property for deferred parameters</td>
</tr>
<tr>
<td>Report Limit</td>
<td>Add a report limit</td>
</tr>
<tr>
<td></td>
<td>Delete a report limit</td>
</tr>
<tr>
<td></td>
<td>Edit/update a report limit</td>
</tr>
<tr>
<td>Bind Requirement</td>
<td>Add/update a bind requirement</td>
</tr>
<tr>
<td></td>
<td>Delete a bind requirement</td>
</tr>
<tr>
<td>Bind Preference</td>
<td>Add/update bind preference</td>
</tr>
<tr>
<td></td>
<td>Delete bind preference</td>
</tr>
<tr>
<td>Recipe Transition</td>
<td>Add a transition to a recipe</td>
</tr>
<tr>
<td></td>
<td>Delete a transition to a recipe</td>
</tr>
<tr>
<td></td>
<td>Edit/update a recipe transition</td>
</tr>
<tr>
<td>Link Group</td>
<td>Add a recipe link group</td>
</tr>
<tr>
<td></td>
<td>Delete a recipe link group</td>
</tr>
<tr>
<td></td>
<td>Edit/update a recipe link group</td>
</tr>
<tr>
<td>Message Type</td>
<td>Event</td>
</tr>
<tr>
<td>-------------------</td>
<td>----------------------------------------------------------------------</td>
</tr>
<tr>
<td>Equipment Requirement</td>
<td>Add an equipment requirement to a recipe</td>
</tr>
<tr>
<td></td>
<td>Delete an equipment requirement of a recipe</td>
</tr>
<tr>
<td></td>
<td>Edit/update procedure-only equipment requirement properties</td>
</tr>
<tr>
<td></td>
<td>(except downstream units)</td>
</tr>
<tr>
<td></td>
<td>Edit/update procedure-only equipment requirement's downstream units property</td>
</tr>
<tr>
<td></td>
<td>Edit/update operation and unit procedure equipment requirements</td>
</tr>
<tr>
<td>Recipe Verification</td>
<td>Verify a single recipe</td>
</tr>
<tr>
<td></td>
<td>Verify all recipes</td>
</tr>
<tr>
<td>Material Loop</td>
<td>Add a material loop to a recipe</td>
</tr>
<tr>
<td></td>
<td>Undo a material loop</td>
</tr>
<tr>
<td>Recipe Link</td>
<td>Add a recipe link</td>
</tr>
<tr>
<td></td>
<td>Delete a recipe link</td>
</tr>
</tbody>
</table>

See also

Enable auditing for FactoryTalk Batch Equipment Editor or FactoryTalk Batch Recipe Editor on page 62

Audit messages on page 67

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 Diagnostic Viewer on page 67

FactoryTalk AssetCentre Client on page 71
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 61
- Standard viewer message fields in FactoryTalk Diagnostics Viewer on page 68
- Extended viewer message fields in FactoryTalk Diagnostics Viewer on page 69

### Standard viewer message fields in FactoryTalk Diagnostics Viewer

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

<table>
<thead>
<tr>
<th>Field</th>
<th>Data Type</th>
<th>Description</th>
<th>Used In</th>
</tr>
</thead>
<tbody>
<tr>
<td>Message</td>
<td>Variant</td>
<td>Application and event-specific message associated with the event being reported.</td>
<td>If provided by the user, applies to all FactoryTalk Batch Recipe Editor and FactoryTalk Batch Equipment Editor audit messages.</td>
</tr>
</tbody>
</table>
### Field Data Type Description Used In

<table>
<thead>
<tr>
<th>Field</th>
<th>Data Type</th>
<th>Description</th>
<th>Used In</th>
</tr>
</thead>
<tbody>
<tr>
<td>Severity</td>
<td>Enum</td>
<td>This field specifies the severity of the audit message. This value, along with the Audience parameter, defines where this message is routed.</td>
<td>This is RNADiagSeverity_Info. This is not configurable.</td>
</tr>
<tr>
<td>Audience</td>
<td>Enum</td>
<td>This field specifies the audience for the audit message. This value, in conjunction with the Severity parameter, defines where this message is routed.</td>
<td>This is RNADiagAudience_Op. This is not configurable.</td>
</tr>
<tr>
<td>Time</td>
<td>Date</td>
<td>Date and time, specified as (GMT/UTC) Represented in the TimeStamp field of the audit message.</td>
<td></td>
</tr>
<tr>
<td>Location</td>
<td>BSTR</td>
<td>Name of the host computer. Represented in the MachineName field in the audit message.</td>
<td></td>
</tr>
<tr>
<td>Provider</td>
<td>BSTR</td>
<td>Name of the client application or component. Not used.</td>
<td></td>
</tr>
<tr>
<td>User</td>
<td>BSTR</td>
<td>Name of the person logged into the editor. Represented in the UserName field of the audit message.</td>
<td></td>
</tr>
<tr>
<td>User Description</td>
<td>BSTR</td>
<td>Fully resolved user name. Represented in the UserDescription field of the audit message.</td>
<td></td>
</tr>
<tr>
<td>Args</td>
<td>Variant</td>
<td>Used for Visual Basic Compatibility. Not used.</td>
<td></td>
</tr>
</tbody>
</table>

**See also**

- [FactoryTalk Diagnostics Viewer](#) on page 67
- [Extended viewer message fields in FactoryTalk Diagnostics Viewer](#) on page 69

**Extended viewer message fields in FactoryTalk Diagnostics Viewer**

Extended fields are provided in the FactoryTalk Diagnostics Viewer.

<table>
<thead>
<tr>
<th>Field</th>
<th>Data Type</th>
<th>Description</th>
<th>Used In</th>
</tr>
</thead>
<tbody>
<tr>
<td>Verbosity</td>
<td>Long</td>
<td>This value field is arbitrary; values from 1 through 100 are valid. Use to further define or filter messages.</td>
<td>Not used.</td>
</tr>
<tr>
<td>Action</td>
<td>BSTR</td>
<td>The updating/editing action made by the user.</td>
<td>All audit messages.</td>
</tr>
<tr>
<td>Area</td>
<td>BSTR</td>
<td>For Equipment objects, the area name the object belongs to. This field is used when needed.</td>
<td>Some Equipment audit messages.</td>
</tr>
<tr>
<td>ContainingObject</td>
<td>BSTR</td>
<td>The object type that contains the EditedObject. This is used to help define exactly what object is being edited/updated.</td>
<td>Some Equipment audit messages.</td>
</tr>
<tr>
<td>Field</td>
<td>Data Type</td>
<td>Description</td>
<td>Used In</td>
</tr>
<tr>
<td>--------------------</td>
<td>-----------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------------------------------------------</td>
</tr>
<tr>
<td>ContainingObjectID</td>
<td>BSTR</td>
<td>The sub-property or object property field of the equipment or recipe object affected by the action.</td>
<td>Some Equipment and Recipe audit messages.</td>
</tr>
<tr>
<td>EditedObject</td>
<td>BSTR</td>
<td>The Equipment or Recipe object affected by the action.</td>
<td>All audit messages.</td>
</tr>
<tr>
<td>EditedObjectID</td>
<td>BSTR</td>
<td>The most common way to identify the EditedObject.</td>
<td>All audit messages.</td>
</tr>
<tr>
<td>EditedProperty</td>
<td>BSTR</td>
<td>The property affected by the action.</td>
<td>Some Recipe, Equipment and Area audit messages.</td>
</tr>
<tr>
<td>EditedProperty1</td>
<td>BSTR</td>
<td></td>
<td></td>
</tr>
<tr>
<td>EditedProperty2</td>
<td>BSTR</td>
<td>Preference priority</td>
<td></td>
</tr>
<tr>
<td>EquipmentPhase</td>
<td>BSTR</td>
<td>For Equipment objects, the equipment phase name the object belongs to. This field is used when needed.</td>
<td>Some Equipment audit messages.</td>
</tr>
<tr>
<td>NewValue</td>
<td>BSTR</td>
<td>The value of a property after it was changed.</td>
<td>Updating audit events only.</td>
</tr>
<tr>
<td>NewValue1</td>
<td>BSTR</td>
<td>The value of a property after it was changed:</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Phase Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Attribute Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Expression text</td>
<td></td>
</tr>
<tr>
<td>NewValue2</td>
<td>BSTR</td>
<td>Priority number</td>
<td></td>
</tr>
<tr>
<td>OldValue</td>
<td>BSTR</td>
<td>The value of a property before it was changed.</td>
<td>Updating audit events only.</td>
</tr>
<tr>
<td>OldValue1</td>
<td>BSTR</td>
<td>The value of a property before it was changed:</td>
<td>Updating audit events only.</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Phase Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Attribute Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Expression text</td>
<td></td>
</tr>
<tr>
<td>OldValue2</td>
<td>BSTR</td>
<td>Priority number</td>
<td>Updating audit events only.</td>
</tr>
<tr>
<td>Operation</td>
<td>BSTR</td>
<td>For Recipe objects, the operation name the object belongs to. This field is used when needed.</td>
<td>Some Recipe audit messages.</td>
</tr>
<tr>
<td>Procedure</td>
<td>BSTR</td>
<td>For Recipe objects, the procedure name the object belongs to. This field is used when needed.</td>
<td>Some Recipe audit messages.</td>
</tr>
<tr>
<td>ProcessCell</td>
<td>BSTR</td>
<td>For Equipment objects, the process cell name the object belongs to. This field is used when needed.</td>
<td>Some Equipment audit messages.</td>
</tr>
<tr>
<td>SubAction</td>
<td>BSTR</td>
<td>An updating/editing action to a list that is a property of the EditedObject. This can be Add or Delete.</td>
<td>Updating audit events for properties that are lists only.</td>
</tr>
<tr>
<td>Field</td>
<td>Data Type</td>
<td>Description</td>
<td>Used In</td>
</tr>
<tr>
<td>---------------</td>
<td>-----------</td>
<td>--------------------------------------------------------------------------------</td>
<td>----------------------------------------------</td>
</tr>
<tr>
<td>SubObject</td>
<td>BSTR</td>
<td>The object type referenced by the EditedObject.</td>
<td>Updating audit events for properties that are lists only.</td>
</tr>
<tr>
<td>SubObjectID</td>
<td>BSTR</td>
<td>The value of a property after it was changed.</td>
<td>Updating audit events for properties that are lists only.</td>
</tr>
<tr>
<td>Unit</td>
<td>BSTR</td>
<td>For Equipment objects, the unit name the object belongs to. This field is used when needed.</td>
<td>Some Equipment audit messages.</td>
</tr>
<tr>
<td>UnitProcedure</td>
<td>BSTR</td>
<td>For Recipe objects, the unit procedure name the object belongs to. This field is used when needed.</td>
<td>Some Recipe audit messages.</td>
</tr>
<tr>
<td>Working Set</td>
<td>BSTR</td>
<td>For Recipe objects, the computer name and database where the recipes are stored.</td>
<td>Some Recipe audit messages.</td>
</tr>
</tbody>
</table>

See also

FactoryTalk Diagnostics Viewer on page 67

Standard viewer message fields in FactoryTalk Diagnostics Viewer on page 68

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 61

Viewer message fields on page 72

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 71
- Enable auditing for FactoryTalk Batch Equipment Editor or FactoryTalk Batch Recipe Editor on page 62
Chapter 6

The FactoryTalk Batch Service Manager

The FactoryTalk Batch Service Manager is used with the FactoryTalk Batch Server, eProcedure 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 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 75
FactoryTalk Batch Service Manager interface on page 76
Batch Server Statistics dialog box on page 78
Start the batch server service on page 82
Add a custom service on page 86

Windows services

The FactoryTalk Batch 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 75

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

- Connection
- Service state
- Server

**See also**

*Connection* on page 76

*Service state* on page 76

*Server* on page 78

### Connection

The *Connection* area contains:

<table>
<thead>
<tr>
<th>Item</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer</td>
<td>Displays the selected computer.</td>
</tr>
<tr>
<td>Select Computer</td>
<td>Selects the computer where the service is located.</td>
</tr>
<tr>
<td>Service</td>
<td>Allows you to select a service from a list of batch services for the selected computer.</td>
</tr>
</tbody>
</table>

**See also**

*The FactoryTalk Batch Service Manager* on page 75

*Service state* on page 76

*Server* on page 78
Service state

The Service State area contains buttons, text and a graphic traffic light service state representation.

The following possible states and corresponding colors are displayed:

<table>
<thead>
<tr>
<th>State</th>
<th>Color</th>
</tr>
</thead>
<tbody>
<tr>
<td>STARTING</td>
<td>None</td>
</tr>
<tr>
<td>RUNNING</td>
<td>Green</td>
</tr>
<tr>
<td>STOPPED</td>
<td>Red</td>
</tr>
<tr>
<td>PAUSED</td>
<td>Yellow</td>
</tr>
<tr>
<td>NOT CONNECTED</td>
<td>None</td>
</tr>
<tr>
<td>START PENDING</td>
<td>None</td>
</tr>
</tbody>
</table>

The Service State area contains the following buttons:

<table>
<thead>
<tr>
<th>Item</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Stop</td>
<td>Stops the selected service.</td>
</tr>
<tr>
<td>Pause</td>
<td>Pauses the selected service. This button is not enabled for the batch server.</td>
</tr>
<tr>
<td>Start/Continue</td>
<td>Starts the selected service or continues a service that has been paused.</td>
</tr>
</tbody>
</table>

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.

See also

The FactoryTalk Batch Service Manager on page 75
Chapter 6  The FactoryTalk Batch Service Manager

Connection on page 76

Server on page 78

Server

The Server area contains:

<table>
<thead>
<tr>
<th>Item</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Allow Demo Mode</td>
<td>The FactoryTalk Batch Server runs with full functionality without a license for a two-hour period. After two hours, the server stops running.</td>
</tr>
<tr>
<td>Allow Grace Period</td>
<td>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.</td>
</tr>
<tr>
<td>Cold Boot</td>
<td>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.</td>
</tr>
<tr>
<td>Warm Boot</td>
<td>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.</td>
</tr>
<tr>
<td>Warm All Boot</td>
<td>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.</td>
</tr>
<tr>
<td>Server Statistics</td>
<td>This button opens the Batch Server Statistics dialog box.</td>
</tr>
</tbody>
</table>

See also

The FactoryTalk Batch Service Manager on page 75

Service state on page 76

Connection on page 76

Batch Server Statistics dialog box

The Batch Server Statistics dialog box displays an overview of the current system's status. The data is view-only and cannot be modified. The COM server and area model file referenced are specified in the Server Options dialog box accessed from FactoryTalk Batch Equipment Editor. The Batch Server Statistics dialog box is organized into four tabs.

See also

Batch Server Statistics dialog box - General tab on page 79

Batch Server Statistics dialog box - Messages tab on page 79

Batch Server Statistics dialog box - PCD Communications tab on page 80
Batch Server Statistics dialog box - General tab

The **General** tab lists various 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 was started, and the name and creation date of the current area model.

The **General** tab contains the following:

<table>
<thead>
<tr>
<th>Field</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch Count</td>
<td>The number of batches loaded in the FactoryTalk Batch Server.</td>
</tr>
<tr>
<td>Prompts Count</td>
<td>The number of prompts currently existing within all batches.</td>
</tr>
<tr>
<td>Version</td>
<td>The version number of the FactoryTalk Batch Server.</td>
</tr>
<tr>
<td>Time</td>
<td>The time as specified by Windows on the FactoryTalk Batch Server.</td>
</tr>
<tr>
<td>Started at</td>
<td>The time the FactoryTalk Batch Server was started.</td>
</tr>
<tr>
<td>Area Model File Name</td>
<td>The name of the currently active area model.</td>
</tr>
<tr>
<td>Area Model Date</td>
<td>The creation date of the currently active area model.</td>
</tr>
</tbody>
</table>

See also

- The FactoryTalk Batch Service Manager on page 75
- Batch Server Statistics dialog box on page 78
Batch Server Statistics 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 consists of the following:

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

See also

The FactoryTalk Batch Service Manager on page 75

Batch Server Statistics dialog box on page 78
Batch Server Statistics 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 consists of the following components:

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

See also

The FactoryTalk Batch Service Manager on page 75

Batch Server Statistics dialog box on page 78
Batch Server Statistics 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 Ole for Process Control (OPC) conversations taking place between the FactoryTalk Batch Server and Client software, such as FactoryTalk Batch View and ActiveX Controls.

The View Communications tab consists of the following:

<table>
<thead>
<tr>
<th>Item</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>DDE Connections</td>
<td>The number of DDE communications attached to the FactoryTalk Batch Server.</td>
</tr>
<tr>
<td>COM Connections</td>
<td>The number of COM communications attached to the server.</td>
</tr>
<tr>
<td>OPC Connections</td>
<td>The number of OPC communications attached to the server.</td>
</tr>
<tr>
<td>Items</td>
<td>The number of communication items the server is currently supporting.</td>
</tr>
</tbody>
</table>

See also

The FactoryTalk Batch Service Manager on page 75

Batch Server Statistics dialog box on page 78

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.
To start the FactoryTalk Batch Server service:

1. Select the **Start** button, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Batch Service Manager**. The **Batch Service Manager** opens.

2. Select **Select Computer**. The **Select Computer** dialog box opens.

   **Tip:** If running on Windows 7 or Windows Server 2008, run the FactoryTalk Batch Service Manager as an administrator.

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 two 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, SAI(s) 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.

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.


   **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. For more information, see Start the FactoryTalk Batch phase simulator.

See also

Grace periods in FactoryTalk Batch Server on page 84

Enable a grace period on page 85

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

Enable a grace period on page 85

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 > All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select 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. Click 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. Click the Start/Continue button 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. Click Close. The Batch Service Manager dialog box closes.

See also

Grace periods in FactoryTalk Batch Server on page 84

Start the batch server service on page 82
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 75
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 87
FactoryTalk Batch Server option configuration on page 88
Modify the .NET registry key on page 103

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 2008 user. Changing the user account in which the server runs is done through Services.

See also

The FactoryTalk Batch Server on page 87
Specify the startup type on page 87

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:


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 87

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 View and the data server(s). Configure FactoryTalk Event Archiver to specify if and how electronic batch record files (.evt) 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 89
- [Server Options dialog box - Restart Control tab](#) on page 91
- [Server Options dialog box - Batch Reporting tab](#) on page 93
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.

See also

- Set project directories on page 89
- Recipe storage on page 90
- Recipe versioning on page 91
Set project directories

Set directories for the following: Primary Journal, Secondary Journal, Error Logging, Instructions (if you are using FactoryTalk eProcedure), 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 89
- FactoryTalk Batch Server option configuration on page 88

Recipe storage

In the FactoryTalk Batch Equipment Editor, Server Options dialog box, under Store Recipes Using select one of the following 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 Batch Client, perform a custom install 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.
The FactoryTalk Batch Server

Chapter 7

XML Files: Stores recipes in separate .xml files:
- Stores procedures as .pxml files
- Stores unit procedures as .uxml files
- Stores operations as .oxml files

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
- Recipe storage on page 90
- Set project directories on page 89

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 92
- Define redundant servers on page 92
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 104

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, click on a selection, or accept the default, Warm Restart.
4. Click the Primary Path browse button. 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 click OK.
6. Click the **Secondary Path** browse button. 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 click **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 92
- Server Options dialog box - Restart Control tab on page 91
- Set project directories on page 89

**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.

To use the **Server Options dialog box - Batch Reporting tab**:

1. Select the **Batch Reporting** tab.

2. Select the appropriate option from the **Reporting Application** area. The following sections define the available reporting application options.

**See also**

- Report Editor on page 93
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 the standard reporting that is available using the Rockwell Automation Knowledgebase - Answer ID 62367.

See also

Server Options dialog box - Batch Reporting tab on page 93

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 for the deletion of the .evt 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.

   The Database Connect String box should contain, at a minimum, ODBC; It updates automatically when the data source is selected the first time FactoryTalk Event Archiver is started after installation. However, if you know the name of the data source, enter ODBC;DSN=DataSourceName; and skip the FactoryTalk Event Archiver setup function.

   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 93

**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 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 93
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 93

None

Select this option if no batch reporting or archiving application is to be used.

See also

Server Options dialog box - Batch Reporting tab on page 93

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 93

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 88

**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 98

[Communications timeout](#) on page 98

[Miscellaneous](#) on page 99
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 \( \text{PHASE}_F > 0 \). A Hold command associated with a failure propagates up through the recipe hierarchy as high as the mode and selected option allows.

<table>
<thead>
<tr>
<th>Hold Propagation Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Phase</td>
<td>Only the affected phase is held, allowing the balance of the phase’s unit and all other units to continue running.</td>
</tr>
<tr>
<td>Operation</td>
<td>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.</td>
</tr>
<tr>
<td>Unit</td>
<td>The phase and the related unit are held and all other units continue running.</td>
</tr>
<tr>
<td>Batch</td>
<td>(Default) Holds the entire batch when a failure is detected by the FactoryTalk Batch Server.</td>
</tr>
</tbody>
</table>

The following 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.

---

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**.

---

See also

**Batch Server tab** on page 97
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):

\[
0.95 \times (5 \times 10000) = 47500
\]

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 97

**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 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 97
- **Batch identification** on page 163

Cross Invocation Descriptors and Defaults tab

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 ActiveX 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 100
- **Cross invocation** on page 115

Resource types

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

See also

- **Cross Invocation Descriptors and Defaults tab** on page 100
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 100

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 101

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 88

**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

![Server Options](image)

**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**

Choose the **Yes** option to hold batches or the **No** option to continue to run batches in the event of a split feed.
The **Yes** option is the default option and the batch is HELD when a split feed occurs.

For the **No** option, 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 88

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 87

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 104

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 the 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 click the Auto button to place a batch in AUTOMATIC mode.

3. Click the Clear All Failures button to clear the failures.

4. Select the batch, and then click the Restart Batch button. 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 103

The FactoryTalk Batch Service Manager on page 75

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 87

Define redundant servers on page 92
Configure the FactoryTalk Batch Server for manual failover

When the FactoryTalk Batch Server fails, complete the following 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 104
- Modify the .NET registry key on page 103
- Configure the material-enabled Batch server for manual failover on page 106
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 the following 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, click **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. Click the **Computer Name** box, and then select the computer to be used as the secondary server.
9. From the **File** menu, click **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 103](#)

---

### 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 87
Chapter 8

Windows event log

The Windows Event Log service records application, system, and security information. This information can be viewed 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 109

Modify event log settings on page 113

View event log entry properties

From the Windows Event Viewer, you can 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 following the list.

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date</td>
<td>Date that the event occurred.</td>
</tr>
</tbody>
</table>

![Event Properties Window](image)
### Windows event log

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td>Time that the event occurred.</td>
</tr>
<tr>
<td>Type</td>
<td>Type of the event (i.e., Information, Error, etc.).</td>
</tr>
<tr>
<td>User</td>
<td>Configured domain user.</td>
</tr>
<tr>
<td>Computer</td>
<td>Computer on which the FactoryTalk Batch Server is running.</td>
</tr>
<tr>
<td>Source</td>
<td>BatchSvr</td>
</tr>
<tr>
<td>Category</td>
<td>Category of the event.</td>
</tr>
<tr>
<td>Event ID</td>
<td>ID of the event.</td>
</tr>
<tr>
<td>Description</td>
<td>Description of the event.</td>
</tr>
<tr>
<td>Data</td>
<td>File and line number within the FactoryTalk Batch Server source code that generated the entry.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Category</th>
<th>Event ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>1000</td>
<td>Server demand started successfully in demo mode.</td>
</tr>
<tr>
<td>Start</td>
<td>1001</td>
<td>Server demand started successfully in production mode.</td>
</tr>
<tr>
<td>Start</td>
<td>1002</td>
<td>Server auto started successfully.</td>
</tr>
<tr>
<td>Start</td>
<td>1003</td>
<td>Server failed to start. See the Batchsvr.log file for more information.</td>
</tr>
<tr>
<td>Start</td>
<td>1004</td>
<td>Server failed to demand start. See the Batchsvr.log file for more information.</td>
</tr>
</tbody>
</table>

**Important:** Do not edit the Batchsvr.log files. The Batchsvr.log contains critical information Technical Support needs to 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 StartServiceCtrlDispatcher failed.              |
| Start    | 1009     | Log directory does not exist. Validate ErrorLogDirectory key value in Batchsvr.ini. |

**Important:** Do not edit the Batchsvr.ini file unless specifically instructed by Technical Support.

<p>| Start    | 1010     | Unable to start due to access permissions on log directory specified by ErrorLogDirectory key value in Batchsvr.ini. |
| Start    | 1011     | Batchsvr.log file could not be created/opened. Validate pathname in Batchsvr.ini. |
| Start    | 1012     | Unable to start. Log file (Batchsvr.log or Verify.log) does not have write access permission. |
| Start    | 1013     | Unable to start. Unable to access log file (Batchsvr.log or Verify.log).     |
| Start    | 1014     | Unable to start. Verify.log file does not have write access permission.     |
| Start    | 1015     | Unable to start due to missing Batchsvr.ini file.                            |
| Start    | 1016     | Unable to start due to access permissions on Batchsvr.ini file.              |
| Start    | 1017     | Unable to start due to access problem with Batchsvr.ini file.               |</p>
<table>
<thead>
<tr>
<th>Category</th>
<th>Event ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>1018</td>
<td>Access error on <code>Batchsvr.ini</code> detected prior to logging on as user specified in <code>Batchsvr.ini</code>.</td>
</tr>
<tr>
<td>Start</td>
<td>1019</td>
<td>Access error on <code>Batchsvr.ini</code> detected while running as user specified in <code>Batchsvr.ini</code>.</td>
</tr>
<tr>
<td>Start</td>
<td>1020</td>
<td>Access error on log file detected while running as user specified in <code>Batchsvr.ini</code>.</td>
</tr>
<tr>
<td>Start</td>
<td>1021</td>
<td>Access error on event files detected while running as user specified in <code>Batchsvr.ini</code>.</td>
</tr>
<tr>
<td>Start</td>
<td>1022</td>
<td>Unable to start because primary event file directory specified by EventDirectoryPath key in <code>Batchsvr.ini</code> is missing.</td>
</tr>
<tr>
<td>Start</td>
<td>1023</td>
<td>Unable to start because secondary event file directory specified by EventDirectoryPathSecondary key in <code>Batchsvr.ini</code> is missing.</td>
</tr>
<tr>
<td>Start</td>
<td>1024</td>
<td>Unable to start because of access permissions on primary event file directory specified by EventDirectoryPath key in <code>Batchsvr.ini</code>.</td>
</tr>
<tr>
<td>Start</td>
<td>1025</td>
<td>Unable to start because of access permissions on secondary event file directory specified by EventDirectoryPathSecondary key in <code>Batchsvr.ini</code>.</td>
</tr>
<tr>
<td>Start</td>
<td>1026</td>
<td>Unable to start because of access permissions on <code>Archque.txt</code> file in primary event file directory.</td>
</tr>
<tr>
<td>Start</td>
<td>1027</td>
<td>Unable to start because of access permissions on <code>Archque.txt</code> file in secondary event file directory.</td>
</tr>
<tr>
<td>Start</td>
<td>1028</td>
<td>Unable to start because of access permissions on <code>Eventdir.txt</code> file in primary event file directory.</td>
</tr>
<tr>
<td>Start</td>
<td>1029</td>
<td>Unable to start because of access permissions on <code>Eventdir.txt</code> file in secondary event file directory.</td>
</tr>
<tr>
<td>Start</td>
<td>1030</td>
<td>Unable to start because RestartDirectoryPath key not specified in <code>Batchsvr.ini</code>.</td>
</tr>
<tr>
<td>Start</td>
<td>1031</td>
<td>Unable to start because RestartDirectoryPath specified in <code>Batchsvr.ini</code> is missing.</td>
</tr>
<tr>
<td>Start</td>
<td>1032</td>
<td>Unable to start because of access permissions on RestartDirectoryPath specified in <code>Batchsvr.ini</code>.</td>
</tr>
<tr>
<td>Start</td>
<td>1033</td>
<td>Unable to start because of access permissions (security?) on RestartDirectoryPath specified in <code>Batchsvr.ini</code>.</td>
</tr>
<tr>
<td>Start</td>
<td>1034</td>
<td>Unable to start because RestartDirectoryPath2 specified in <code>Batchsvr.ini</code> is missing.</td>
</tr>
<tr>
<td>Start</td>
<td>1035</td>
<td>Unable to start because of access permissions on RestartDirectoryPath2 specified in <code>Batchsvr.ini</code>.</td>
</tr>
<tr>
<td>Start</td>
<td>1036</td>
<td>Unable to start because of access permissions (security) on RestartDirectoryPath2 specified in <code>Batchsvr.ini</code>.</td>
</tr>
<tr>
<td>Start</td>
<td>1037</td>
<td>Server started by an ActiveX client.</td>
</tr>
<tr>
<td>Start</td>
<td>1038</td>
<td>Domain user name is blank and must be initialized in the SECURITY section of the <code>Batchsvr.ini</code>. The Server will not have any network credentials, which means it cannot access anything off node (i.e., data servers, event journal files, or recipes).</td>
</tr>
<tr>
<td>Start</td>
<td>1039</td>
<td>Internal Error. Call to CoInitializeSecurity failed.</td>
</tr>
<tr>
<td>Start</td>
<td>1040</td>
<td>Internal Error. Call to RegisterClassObjects failed.</td>
</tr>
<tr>
<td>Start</td>
<td>1041</td>
<td>The SQL Server database cannot be opened.</td>
</tr>
<tr>
<td>Category</td>
<td>Event ID</td>
<td>Description</td>
</tr>
<tr>
<td>---------------</td>
<td>----------</td>
<td>-------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Boot</td>
<td>2000</td>
<td>Server warm booting.</td>
</tr>
<tr>
<td>Boot</td>
<td>2001</td>
<td>Server cold booting.</td>
</tr>
<tr>
<td>Boot</td>
<td>2002</td>
<td>Server started.</td>
</tr>
<tr>
<td>Boot</td>
<td>2003</td>
<td>Server warm all booting.</td>
</tr>
<tr>
<td>Boot</td>
<td>3000</td>
<td>Data Server: [Descriptive String] status changed to GOOD.</td>
</tr>
<tr>
<td>Boot</td>
<td>3001</td>
<td>Data Server: [Descriptive String] status changed to SUSPECT.</td>
</tr>
<tr>
<td>Boot</td>
<td>3002</td>
<td>Data Server: [Descriptive String] status changed to LOST.</td>
</tr>
<tr>
<td>Boot</td>
<td>3003</td>
<td>Data Server: [Descriptive String] status changed to BAD.</td>
</tr>
<tr>
<td>Boot</td>
<td>3004</td>
<td>Data Server: [Descriptive String] status changed to an unknown conversation status.</td>
</tr>
<tr>
<td>Boot</td>
<td>3005</td>
<td>Data Server: [Descriptive String].</td>
</tr>
<tr>
<td>Shutdown</td>
<td>4000</td>
<td>Server shutting down.</td>
</tr>
<tr>
<td>Shutdown</td>
<td>4001</td>
<td>Server exception occurred. See the Batchsvr.log file for information.</td>
</tr>
<tr>
<td>Shutdown</td>
<td>4002</td>
<td>Server automatically shutting down from demo mode.</td>
</tr>
<tr>
<td>Tag_Verification</td>
<td>5000</td>
<td>Server tag verification process Started.</td>
</tr>
<tr>
<td>Tag_Verification</td>
<td>5001</td>
<td>Server tag verification process Aborted.</td>
</tr>
<tr>
<td>Tag_Verification</td>
<td>5002</td>
<td>Server tag verification completed, [Descriptive String1] of [Descriptive String2] tags verified successfully.</td>
</tr>
<tr>
<td>Registration</td>
<td>6000</td>
<td>Server registered.</td>
</tr>
<tr>
<td>Registration</td>
<td>6001</td>
<td>Server unregistered.</td>
</tr>
<tr>
<td>Registration</td>
<td>6002</td>
<td>Server registered as a service.</td>
</tr>
<tr>
<td>Control</td>
<td>7000</td>
<td>Notify SCM: The service is stopping.</td>
</tr>
<tr>
<td>Control</td>
<td>7001</td>
<td>Notify SCM: Service stopped.</td>
</tr>
<tr>
<td>Control</td>
<td>7002</td>
<td>Notify SCM: The service is starting.</td>
</tr>
<tr>
<td>Control</td>
<td>7003</td>
<td>Notify SCM: The service is running.</td>
</tr>
<tr>
<td>Control</td>
<td>7004</td>
<td>SERVICE_CONTROL_STOP received from Service Control Manager.</td>
</tr>
<tr>
<td>Control</td>
<td>7005</td>
<td>Unexpected SERVICE_CONTROL_PAUSE received from Service Control Manager.</td>
</tr>
<tr>
<td>Control</td>
<td>7006</td>
<td>Unexpected SERVICE_CONTROL_CONTINUE received from Service Control Manager.</td>
</tr>
<tr>
<td>Control</td>
<td>7007</td>
<td>SERVICE_CONTROL_INTERROGATE received from Service Control Manager.</td>
</tr>
<tr>
<td>Control</td>
<td>7008</td>
<td>SERVICE_CONTROL_SHUTDOWN received from Service Control Manager.</td>
</tr>
<tr>
<td>Control</td>
<td>7009</td>
<td>Bad Service Request.</td>
</tr>
<tr>
<td>Category_Debug</td>
<td>8000</td>
<td>Command Line: [Descriptive String].</td>
</tr>
<tr>
<td>Category_Debug</td>
<td>8001</td>
<td>Argument received from SCM or BatchSCM: [Descriptive String].</td>
</tr>
<tr>
<td>Category_Debug</td>
<td>8002</td>
<td>HRESULT = [Descriptive String].</td>
</tr>
<tr>
<td>Category_Debug</td>
<td>8003</td>
<td>Thread [Descriptive String] started.</td>
</tr>
<tr>
<td>Category_Debug</td>
<td>8004</td>
<td>Exception in object [Descriptive String1] in method [Descriptive String2].</td>
</tr>
<tr>
<td>Category_Debug</td>
<td>8005</td>
<td>Service could not be marked for deletion by the SCM.</td>
</tr>
</tbody>
</table>
### Windows event log

#### Category Event ID Description

<table>
<thead>
<tr>
<th>Category</th>
<th>Event ID</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Category_Debug</td>
<td>8006</td>
<td>Debug: [Descriptive String]</td>
</tr>
</tbody>
</table>

See also

- [Windows event log](#) on page 109
- [Modify event log settings](#) on page 113

---

**Modify event log settings**

In order 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:**

2. Right-click Application, and then click Properties. The Application Properties dialog box opens.
3. In the Log size area, select Overwrite events as needed, and then click OK.
4. Repeat steps 2 and 3 for the System.
5. Exit the Event Viewer.

See also

- [Windows event log](#) on page 109
- [View event log entry properties](#) on page 109
Chapter 9

Cross invocation

Cross Invocation is the means by which the FactoryTalk Batch View and ActiveX 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 115
- Cross invocation string configuration on page 117
- Cross invocation string escape sequences on page 118
- Cross invocation string example on page 119
- FactoryTalk Batch View and ActiveX Controls configuration for cross invocation on page 120

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 from:

- FactoryTalk Batch View when the Goto HMI button is selected.
- The ActiveX Controls when the ActiveX control shortcut menus are invoked by right-clicking on the running object.

If a recipe step is not bound to an equipment resource:

- The FactoryTalk Batch ProcedureView ActiveX control does not display the Cross Invocation menu when you right-click on the step.
• FactoryTalk Batch View displays a message indicating the step is not bound when you select the Goto HMI button.

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 following:

• The text for displaying the Cross invocation menu item.
• The data for passing to the automation server.

The following table shows the various FactoryTalk Batch View windows and the equipment resource to which a selected procedural element is linked. For cross invocation to be implemented, the cross invocation strings must be configured for the equipment resource.

<table>
<thead>
<tr>
<th>FactoryTalk Batch View window</th>
<th>Selectable elements</th>
<th>Corresponding equipment resource</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch List</td>
<td>Batch</td>
<td>Process Cell (Procedure)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Unit (Unit Proc or Operation)</td>
</tr>
<tr>
<td>Procedure as SFC/Table</td>
<td>Batch</td>
<td>Process Cell (Procedure)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Unit (Unit Proc or Operation)</td>
</tr>
<tr>
<td></td>
<td>Unit Procedure</td>
<td>Unit</td>
</tr>
<tr>
<td></td>
<td>Operation</td>
<td>Unit</td>
</tr>
<tr>
<td></td>
<td>Recipe Phase</td>
<td>Phase</td>
</tr>
<tr>
<td>Event Journal</td>
<td>Batch</td>
<td>Process Cell (Procedure)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Unit (Unit Proc or Operation)</td>
</tr>
<tr>
<td>Unacknowledged Prompts</td>
<td>Unacknowledged Prompts</td>
<td>Phase</td>
</tr>
<tr>
<td>Phase Control</td>
<td>Process Cell</td>
<td>Process Cell</td>
</tr>
<tr>
<td></td>
<td>Unit</td>
<td>Unit</td>
</tr>
<tr>
<td></td>
<td>Phase</td>
<td>Phase</td>
</tr>
<tr>
<td>Arbitration</td>
<td>Recipe</td>
<td>Process Cell (Procedure)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Unit (Unit Proc or Operation)</td>
</tr>
<tr>
<td></td>
<td>Process Cell</td>
<td>Process Cell</td>
</tr>
<tr>
<td></td>
<td>Unit</td>
<td>Unit</td>
</tr>
<tr>
<td></td>
<td>Phase</td>
<td>Phase</td>
</tr>
<tr>
<td></td>
<td>Resource</td>
<td>Resource</td>
</tr>
<tr>
<td>Alarm Summary</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>Phase Summary</td>
<td>Phase</td>
<td>Phase</td>
</tr>
</tbody>
</table>
Cross invocation on page 115

Cross invocation string configuration on page 117

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 in FactoryTalk Batch View or from the ActiveX 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 sub-menu 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 sub-string 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 the following table along with their meaning.
### Cross invocation string escape sequences

The escape sequences used in the cross invocation strings correspond to the current selection in FactoryTalk Batch View or ActiveX Controls applications. The context data passed to the automation server is relative to the current selection. (See the FactoryTalk Batch View User Guide for information regarding selectable elements.)

The following describes each escape sequence relative to the type of procedural element selected:

<table>
<thead>
<tr>
<th>Escape Sequence</th>
<th>Procedure</th>
<th>Unit Procedure</th>
<th>Operation</th>
<th>Phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>%P</td>
<td>Path</td>
<td>Path</td>
<td>Path</td>
<td>Path</td>
</tr>
<tr>
<td>%N</td>
<td>Corresponding Process Cell Name</td>
<td>Corresponding Unit name</td>
<td>Corresponding Unit name</td>
<td>Corresponding Phase name</td>
</tr>
<tr>
<td>%C</td>
<td>Corresponding Process Cell class name</td>
<td>Corresponding Unit Class name</td>
<td>Corresponding Unit Class name</td>
<td>Corresponding Phase Class name</td>
</tr>
<tr>
<td>%T</td>
<td>1 (Process Cell)</td>
<td>2 (Unit)</td>
<td>2 (Unit)</td>
<td>3 (Phase)</td>
</tr>
<tr>
<td>%S</td>
<td>Node</td>
<td>Node</td>
<td>Node</td>
<td>Node</td>
</tr>
<tr>
<td>%B</td>
<td>Batch ID</td>
<td>Batch ID</td>
<td>Batch ID</td>
<td>Batch ID</td>
</tr>
<tr>
<td>%M</td>
<td>Menu caption</td>
<td>Menu caption</td>
<td>Menu caption</td>
<td>Menu caption</td>
</tr>
</tbody>
</table>

See also

[Cross invocation overview on page 115](#)
%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.

The following describes each escape sequence relative to the element selected within the FactoryTalk Batch View **Phase Control** window. (See the *FactoryTalk Batch View User Guide* for information regarding selectable elements.)

<table>
<thead>
<tr>
<th>Escape Sequence</th>
<th>Process Cell</th>
<th>Unit</th>
<th>Phases</th>
</tr>
</thead>
<tbody>
<tr>
<td>%P</td>
<td>Empty</td>
<td>Path (if exists). The context data that replaces %P (when selecting a unit) is the unit procedure path or operation that owns that unit, and blank otherwise.</td>
<td>Create ID (if exists). The context data that replaces %P when selecting an equipment phase is the Create ID of the batch within which the phase is executing. This only exists if that equipment phase is currently executing within the context of a batch.</td>
</tr>
<tr>
<td>%N</td>
<td>Name of process cell</td>
<td>Name of unit</td>
<td>Name of phase</td>
</tr>
<tr>
<td>%C</td>
<td>Process Cell Class name</td>
<td>Unit Class name</td>
<td>Phase Class name</td>
</tr>
<tr>
<td>%T</td>
<td>1 (process cell)</td>
<td>2 (unit)</td>
<td>3 (phase)</td>
</tr>
<tr>
<td>%S</td>
<td>Node</td>
<td>Node</td>
<td>Node</td>
</tr>
<tr>
<td>%B</td>
<td>Empty</td>
<td>Batch ID (if exists)</td>
<td>Batch ID (if exists)</td>
</tr>
<tr>
<td>%M</td>
<td>Menu caption</td>
<td>Menu caption</td>
<td>Menu caption</td>
</tr>
<tr>
<td>%R</td>
<td>Process cell resource ID</td>
<td>Unit resource ID</td>
<td>Phase resource ID</td>
</tr>
</tbody>
</table>

See also

*Cross invocation* on page 115

*Cross invocation string configuration* on page 117

**Cross invocation string example**

This is an example of a cross invocation string configured in FactoryTalk Batch Equipment Editor and the resulting cross invocation menu as seen from FactoryTalk Batch View.

```
[DETAIL DISPLAY, VIEW MODE, EDIT MODE; %P, %N, %T, %C, %S, %B, %R, %M]
```
This illustrates the type of data that could be passed to the automation server when the Batch ID is selected in FactoryTalk Batch View and the batch is running.

Tip: The automation server must be configured for responding appropriately to the data.

See also

Cross invocation overview on page 115

Cross invocation string configuration on page 117

FactoryTalk Batch View and ActiveX Controls configuration for cross invocation

Once the cross invocation strings are defined, FactoryTalk Batch View and ActiveX Controls must be configured to support cross invocation.

FactoryTalk Batch View is configured to support cross invocation under the General tab of the System Configuration and Defaults window. On this tab, cross invocation must be selected and the program identifier of the automation server must be specified.

Once configured in FactoryTalk Batch View, the cross invocation menu items become visible when a procedural element is selected (that corresponds to an Equipment Resource with configured cross invocation strings) followed by selecting the Goto HMI button.

The ActiveX Controls are configured to support cross invocation when the following two properties are configured for the ActiveX object:

- InvocationProgID
• **ShowCustomMenu**

The InvocationProgID property must contain the program identifier of the automation server. The ShowCustomMenu property must be set to True so that the cross invocation strings are visible on the ActiveX object’s shortcut menu during runtime as shown:

See also

- **Cross invocation string configuration** on page 117
- **Cross invocation overview** on page 115
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 123
- [Phase Configuration dialog box](#) on page 125
- [Phase Status dialog box](#) on page 128
Chapter 10  The FactoryTalk Batch Phase Simulator

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 > All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch > Simulator. The Phase Simulator opens.

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 the following options:

<table>
<thead>
<tr>
<th>Option</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>File</td>
<td>Open a simulator (.sim) file, save a file, or exit the Simulator.</td>
</tr>
<tr>
<td>Edit</td>
<td>Edit phase configuration, phase status, or unit tag status.</td>
</tr>
<tr>
<td>Protocols</td>
<td>Enable or disable the Watchdog and Command Handshake functions.</td>
</tr>
<tr>
<td>Help</td>
<td>Open the Simulator About box.</td>
</tr>
</tbody>
</table>

See also

The FactoryTalk Batch Phase Simulator on page 123

Start the FactoryTalk Batch Phase Simulator on page 123

The FactoryTalk Batch Phase Simulator allows you to edit a phase’s configuration. You can set the following 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 126
- [Change the sequence of requests](#) on page 126
- [Indicate a continuous phase](#) on page 127

## 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 125
- [Change the sequence of requests](#) on page 126
- [Indicate a continuous phase](#) on page 127

## 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 \texttt{@Time} (time is in seconds). This value determines when the request is processed. This value must not be greater than the \texttt{Dwell Time}.

For example, if the \texttt{Dwell Time} is 30 seconds, and the \texttt{@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 \texttt{_RQ}. (Refer to the \textit{FactoryTalk Batch PCD Programmer Technical Reference Guide} for information on request numbers.)

6. Type any required request parameter data in \texttt{Q01}, \texttt{Q02}, and \texttt{Q03}. These parameters are used to further define the request.

7. Select the \textbf{Update Phase} button.

8. Select \textbf{Close}.

\textbf{See also}

- \textit{Phase Configuration dialog box} on page \pageref{Phase Configuration dialog box}
- \textit{Change the dwell time} on page \pageref{Change the dwell time}
- \textit{Indicate a continuous phase} on page \pageref{Indicate a continuous phase}

\textbf{Indicate a continuous phase}

Simulate continuous phase logic by selecting \textbf{Continuous Phase}.

\textbf{To indicate a continuous phase:}

1. Start the \textit{FactoryTalk Batch Phase Simulator}.

2. Select \textit{Edit > Phase Configuration}. The \textit{Phase Configuration} dialog box opens.

3. From the \textit{Select Phase} list, select the appropriate phase.

4. If this is a continuous phase, select \textbf{Continuous Phase}.

5. Select \textbf{Update Phase}.

6. Select \textbf{Close}.

\textbf{See also}

- \textit{Phase Configuration dialog box} on page \pageref{Phase Configuration 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

**Phase Status dialog box**

**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 129

Change the value of a unit tag on page 134

View request data tag values on page 129

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 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 129

Change ownership of a phase on page 130

Command a phase on page 131

Acknowledge and clear a request on page 132

Restore and clear requests on page 133

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 129
- [Change ownership of a phase](#) on page 130
- [Command a phase](#) on page 131
- [Acknowledge and clear a request](#) on page 132
- [Restore and clear requests](#) on page 133

### 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 129
- [View request data tag values](#) on page 129
- [Command a phase](#) on page 131
- [Acknowledge and clear a request](#) on page 132
- [Restore and clear requests](#) on page 133
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, click Phase Status.
3. In the Select Phase list, select the phase to command.
4. To set the phase to COMPLETE, click the Set Complete button.
5. For all other state change commands, set the Owner of the phase to External.
6. Click the appropriate state change command button (Start/Stop, Abort/Reset, or Hold/RRestart).

If the requested state change is legal, the server sends the command to FactoryTalk Batch View and the phase transitions to the appropriate state. The new state value is displayed in the Status register in the Phase Status dialog box.

a. If the phase is running and the Abort/Reset button is selected, the phase goes to ABORTING. To transition the phase to the ABORTED state, select the Set Complete button.

b. If a phase is running and the Hold/RRestart button is selected, the phase goes to HOLDING. To transition the phase to the HELD state, select the Set Complete button.

See also

View and update report parameter tag values on page 129
View request data tag values on page 129
Phase commands on page 131
Acknowledge and clear a request on page 132
Restore and clear requests on page 133

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/RRestart 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.)

The following table shows the phase’s transition when the state change command is received, along with the associated state value (##).

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>INITIAL STATE</th>
<th>FINAL STATE</th>
</tr>
</thead>
<tbody>
<tr>
<td>COMPLETE</td>
<td>COMPLETE (70)</td>
<td></td>
</tr>
<tr>
<td>START</td>
<td>RUNNING (50)</td>
<td></td>
</tr>
<tr>
<td>STOP</td>
<td>STOPPING (30)</td>
<td>STOPPED (80)</td>
</tr>
<tr>
<td>ABORT</td>
<td>ABORTING (10)</td>
<td>ABORTED (90)</td>
</tr>
<tr>
<td>RESET</td>
<td>IDLE (100)</td>
<td></td>
</tr>
<tr>
<td>HOLD</td>
<td>HOLDING (20)</td>
<td>HELD (60)</td>
</tr>
<tr>
<td>RESTART</td>
<td>RESTARTING (40)</td>
<td>RUNNING (50)</td>
</tr>
</tbody>
</table>

See also

Phase Status dialog box on page 128

The FactoryTalk Batch Phase Simulator on page 123

Command a phase on page 131

**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 be commanded.
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.
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 be commanded.
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 129

View request data tag values on page 129

Command a phase on page 131

Acknowledge and clear a request on page 132
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 consists of the following:

<table>
<thead>
<tr>
<th>Item</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tagname</td>
<td>A list of all configured unit tags for the active area model.</td>
</tr>
<tr>
<td>Class</td>
<td>The class associated with the unit tag.</td>
</tr>
<tr>
<td>Type</td>
<td>The type value associated with the unit tag.</td>
</tr>
<tr>
<td>Item</td>
<td>The item (addressable location) associated with the unit tag.</td>
</tr>
<tr>
<td>Value</td>
<td>The current value for the unit tag. This value can be modified and sent to</td>
</tr>
<tr>
<td></td>
<td>the FactoryTalk Batch Server.</td>
</tr>
<tr>
<td>Update Values</td>
<td>Updates the tag values in the Simulator.</td>
</tr>
</tbody>
</table>

**See also**

- Change the value of a unit tag on page 134
- The FactoryTalk Batch Phase Simulator on page 123

**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 128
- View request data tag values on page 129
- Acknowledge and clear a request on page 132
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.

See also

- Network Editor interface on page 138
- Configure your network on page 141
- User Preferences dialog box on page 144
- Report printing formats on page 154

Open the Network Editor

Open the Network Editor to configure your network.

Tip: If running on Windows Server 2008 or Windows 7, you must run the Network Editor as administrator. Right-click Network Editor and select Run as administrator.

To open the Network Editor:

- Click the Start button, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Network Editor. The Network Editor window opens.

See also

- Network Editor on page 137
Network Editor interface

The Network Editor interface has the following areas:

- Hierarchy pane
- Detail pane
- Results pane
- Toolbar
- Status bar.

See also

Hierarchy pane on page 138

Detail pane on page 139

Results pane on page 140

Network Editor toolbar on page 141

Status bar on page 141

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 following application servers listed as Network Model members: Material Server, BatchERP Server, and FactoryTalk Batch Server 1 through Batch Server 10. The following indicators appear next to each server listed:

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>✓</td>
<td>Indicates the server is recognized on the network</td>
</tr>
<tr>
<td>✗</td>
<td>Indicates the server is not defined or recognized</td>
</tr>
<tr>
<td>❓</td>
<td>Indicates there is a problem with the server or the server’s configuration</td>
</tr>
</tbody>
</table>

See also

Network Editor interface on page 138
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.

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>System Name</td>
<td>Allows you to name your FactoryTalk Batch system. The default is Network Model.</td>
</tr>
<tr>
<td>Network Model from</td>
<td>Displays the name of the computer from which the FactoryTalk Batch system was last synchronized.</td>
</tr>
<tr>
<td>Time Stamp</td>
<td>Displays the time of the last synchronization.</td>
</tr>
<tr>
<td>User Name</td>
<td>Displays the Windows user ID of the person who performed the last synchronization.</td>
</tr>
<tr>
<td>Apply Changes</td>
<td>Saves changes made to the FactoryTalk Batch system information.</td>
</tr>
<tr>
<td>Undo Changes</td>
<td>Undoes changes made to the FactoryTalk Batch system information that are not applied.</td>
</tr>
<tr>
<td>Synchronize All Application Servers</td>
<td>Initiates the synchronization of the application servers.</td>
</tr>
</tbody>
</table>

**Application server information**

The following items are available when a server or **Undefined** is selected in the Hierarchy pane.

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Alias</td>
<td>Displays a user-defined alias for the selected application server.</td>
</tr>
<tr>
<td>Computer Name</td>
<td>Displays the computer name for the selected server.</td>
</tr>
<tr>
<td>Configure</td>
<td>Allows you to specify the name of the material database when using FactoryTalk Batch Material Manager. The default database is MaterialBasedRecipe.</td>
</tr>
</tbody>
</table>
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 138

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.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Information</th>
<th>Error</th>
<th>Warning</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>Represents an information message.</td>
<td>Indicates an error message.</td>
<td>Indicates a warning message.</td>
</tr>
</tbody>
</table>

See also

Network Editor interface on page 138

Menu bar

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.

See also

Network Editor interface on page 138
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 138

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.

See also

Network Editor interface on page 138

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 each of these functions.

See also

Select a network model on page 142
Add a server on page 142
Remove a server on page 143
Synchronize network models on page 143

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.
5. Click the Synchronize button.

See also

Configure your network on page 141
Remove a server on page 143
Add a server on page 142

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 **OK**.

**See also**

- [Remove a server](#) on page 143
- [Configure your network](#) on page 141
- [Network Editor](#) on page 137
- [Select a network model](#) on page 142

### Remove a server

Use the Network Editor to remove a server.

**To remove a server:**

1. Open the **Network Editor**.

2. In the **Hierarchy** pane, select the server to remove.

3. Click the **Remove Server** button.

**See also**

- [Add a server](#) on page 142
- [Select a network model](#) on page 142
- [Configure your network](#) on page 141

### 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.

See also

- Select a network model on page 142
- Add a server on page 142
- Remove a server on page 143

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.

See also

- Edit colors on page 145
- Edit the application font on page 145
- Edit information tips on page 146
- Set report preferences on page 149

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.

**See also**

[User Preferences dialog box on page 144](#)

**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**.

**See also**

[User Preferences dialog box on page 144](#)

[Network Editor interface on page 138](#)

**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.

See also

User Preferences dialog box on page 144
Network Editor interface on page 138

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**.

See also

   - User Preferences dialog box on page 144
   - Network Editor interface on page 138

**User Preferences dialog box - Views tab**

If you want 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.

![User Preferences dialog box - Views tab](image)

See also

   - User Preferences dialog box on page 144
   - Network Editor interface on page 138

**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:

![Scroll Bars Example]

• **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 144

Use the User Preferences dialog box - Reports tab to configure the layout and appearance of reports.

**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:

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Report</td>
<td>Lists the report templates available for the item or view that is currently active in the Network Editor.</td>
</tr>
<tr>
<td>List</td>
<td>Enables the list report style, if available for the associated report.</td>
</tr>
</tbody>
</table>
### Item Description

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Formatted</td>
<td>Enables the formatted report style, which is the default format for most reports.</td>
</tr>
<tr>
<td>Include Related Data</td>
<td>Adds any additional data associated with the selected item to the printed report.</td>
</tr>
<tr>
<td>Footer Text</td>
<td>Prints the word or phrase entered at the bottom of each report page.</td>
</tr>
<tr>
<td>List Reports Format</td>
<td>Opens the Report Page Configuration dialog box where you can set the margins, font, and text color for a specific report template.</td>
</tr>
<tr>
<td>Formatted Reports Graphical Reports</td>
<td></td>
</tr>
</tbody>
</table>

### See also

- [Set report preferences](#) on page 149
- [Change margins](#) on page 150
- [Set the report fonts and colors](#) on page 151
- [Column and row preferences in reports](#) on page 153

### 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.
See also

User Preferences dialog box on page 144

Network Editor interface on page 138
### 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 144

### 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.

![Image of Fonts dialog box]

2. Click the button for the report section you want to edit. The **Font** dialog box opens.

![Image of Font dialog box]

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 154*

*Set report preferences on page 149*

**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:

<table>
<thead>
<tr>
<th>Action</th>
<th>How to Perform</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sorting columns</td>
<td>Click a column header to sort the items in the column alphabetically or numerically.</td>
</tr>
<tr>
<td>Sizing columns</td>
<td><strong>Automatically</strong>: To make the column width fit the contents, double-click the boundary to the right of the column heading. <strong>Manually</strong>: Drag the boundary on the right side of the column heading until the column is the desired width.</td>
</tr>
<tr>
<td>Action</td>
<td>How to Perform</td>
</tr>
<tr>
<td>------------------------</td>
<td>-----------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Editing the item name</td>
<td>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.</td>
</tr>
<tr>
<td></td>
<td>Double-click on a row to open the Properties dialog box, allowing you to edit the item name.</td>
</tr>
<tr>
<td></td>
<td>Right-click on a row to display the shortcut menu; select Rename.</td>
</tr>
<tr>
<td>Editing a row</td>
<td>Double-click on a row to open the Properties dialog box to edit the row’s information.</td>
</tr>
<tr>
<td></td>
<td>Right-click on a row to display the shortcut menu; select Properties or other editing operations.</td>
</tr>
<tr>
<td>Copy, delete, or move rows</td>
<td>Right-click on a row to display the shortcut menu; select Cut, Copy, Paste, or Delete.</td>
</tr>
</tbody>
</table>

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

<table>
<thead>
<tr>
<th>Result</th>
<th>00000000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer</td>
<td>STATION_3</td>
</tr>
<tr>
<td>Message</td>
<td>Begin synchronization on all computers.</td>
</tr>
<tr>
<td>Type</td>
<td>Information</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Result</th>
<th>00000000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer</td>
<td>STATION_3</td>
</tr>
<tr>
<td>Message</td>
<td>Begin writing network model to computer 'STATION_3'.</td>
</tr>
<tr>
<td>Type</td>
<td>Information</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Result</th>
<th>00000000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer</td>
<td>STATION_3</td>
</tr>
<tr>
<td>Message</td>
<td>The network model was successfully written on computer 'STATION_3'.</td>
</tr>
<tr>
<td>Type</td>
<td>Information</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Result</th>
<th>00000000</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer</td>
<td>STATION_1</td>
</tr>
<tr>
<td>Message</td>
<td>Begin writing network model to computer 'STATION_1'.</td>
</tr>
<tr>
<td>Type</td>
<td>Information</td>
</tr>
</tbody>
</table>
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.

<table>
<thead>
<tr>
<th>Result</th>
<th>Computer</th>
<th>Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>08000830</td>
<td>STATION_3</td>
<td>Begin synchronization on all computers.</td>
</tr>
<tr>
<td>08000830</td>
<td>STATION_3</td>
<td>Begin writing network model to computer STATION_3.</td>
</tr>
<tr>
<td>08000830</td>
<td>STATION_3</td>
<td>The network model was successfully written on computer STATION_3.</td>
</tr>
<tr>
<td>08000830</td>
<td>STATION_1</td>
<td>Begin writing network model to computer STATION_1.</td>
</tr>
<tr>
<td>08000830</td>
<td>STATION_1</td>
<td>The network model was successfully written on computer STATION_1.</td>
</tr>
<tr>
<td>08000830</td>
<td>STATION_3</td>
<td>The network model was successfully synchronized on all computers.</td>
</tr>
</tbody>
</table>

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" for more information.)

See also

Set report preferences on page 149

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.

See also

Report printing formats on page 154

Print reports on page 156
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 for instructions.

See also

Set report preferences on page 149

Preview reports on page 155

Report printing formats on page 154

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 154

Preview reports on page 155
Chapter 12

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 159

Tag verification log information on page 161

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 159

Abnormal termination batch record entries on page 160

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 159

Abnormal termination batch record entries

The electronic batch record (.evt file) will contain 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

<table>
<thead>
<tr>
<th>Description</th>
<th>Event</th>
<th>PValue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event File Name</td>
<td>MachineName\Program Files (x86)\Rockwell Software\Batch\YourProject\JOURNALS\4123.evt</td>
<td></td>
</tr>
<tr>
<td>YourBatchDescription</td>
<td>System Message</td>
<td>Batch terminated by COLD boot of FactoryTalk Batch server</td>
</tr>
<tr>
<td>YourBatchDescription</td>
<td>System Message</td>
<td>End Of BATCH</td>
</tr>
</tbody>
</table>

Example: Termination of FactoryTalk Batch During Warm Restart

<table>
<thead>
<tr>
<th>Description</th>
<th>Event</th>
<th>PValue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event File Name</td>
<td>MachineName\Program Files (x86)\Rockwell Software\Batch\YourProject\JOURNALS\4125.evt</td>
<td></td>
</tr>
<tr>
<td>YourBatchDescription</td>
<td>System Message</td>
<td>Batch terminated due to error attempting to restore batch during WARM restart</td>
</tr>
</tbody>
</table>
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.


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:
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 161](#)

### Windows Event Log

The following events regarding tag verification are recorded in the Windows Event Log.

<table>
<thead>
<tr>
<th>Event</th>
<th>Generate Windows Event Log</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>&quot;Batch Server Tag Verification Process Started&quot;</td>
</tr>
<tr>
<td>Abort</td>
<td>&quot;Batch Server Tag Verification Process Aborted&quot;</td>
</tr>
<tr>
<td>Completion</td>
<td>&quot;Batch Server Tag Verification Completed, &lt;xxx&gt; of &lt;yyy&gt; tags verified successfully&quot; where &lt;xxx&gt; is the number of tags that were successfully verified, and &lt;yyy&gt; is the total number of tags on which verification was attempted.</td>
</tr>
</tbody>
</table>

**See also**

[Tag verification log information on page 161](#)
The FactoryTalk Batch performance chart is a pre-defined 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

[General troubleshooting errors on page 183](#)

The performance chart automatically opens the Windows System Monitor with the pre-defined charts loaded.

**To open the FactoryTalk Batch performance chart:**

1. Click the **Start** button.
2. Point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**.
3. Click **Performance Chart**. The **Performance** dialog box opens with the pre-defined charts listed.

See also

[Using the performance chart on page 162](#)
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. In order to implement this option, you must perform steps in FactoryTalk Batch View and FactoryTalk Batch Equipment Editor, edit some Visual Basic code and compile it into a new DLL. If you are using the FactoryTalk Batch ActiveX controls, you can use the BatchID editable property to enable or disable editing of batch IDs. For more information, see the FactoryTalk Batch ActiveX Controls Library Reference.

See also

**Enable editing of batch IDs**

The batch ID is shown on the **Batch Creation** dialog box when a batch is created, and by default, is editable. However, if editing of batch IDs has been disabled, follow these steps to enable this option.

To enable editing of batch IDs:

1. Open FactoryTalk Batch View.
2. Select **Configuration and Defaults**.
3. Enter your system configuration password, and then select **OK**.
4. Select the **General** tab, and in the **BatchID Editing** area, select the **Allow Editing of Batch ID** check box.

See also

**Batch identification** on page 163

**Disable editing of batch IDs** on page 164
Disable editing of batch IDs

If you create a custom batch ID, we recommend that you prevent editing of batch IDs. This ensures that the custom batch ID you specified appears when you add a batch to the batch list, and that it cannot be changed by the operator.

The following steps disable editing of the batch ID. In this case, the box on the Batch Creation dialog, which displays when a batch is added to the batch list, is grayed out.

**Tip:** If a failure occurs in the generation of a custom batch ID, the batch ID is editable regardless of the setting.

**To disable editing of batch IDs:**

1. Open FactoryTalk Batch View.
2. Select Configuration and Defaults.
3. Enter your system configuration password, and then select OK.
4. Click the General tab, and then in the Batch ID Editing section, deselect the Allow Editing of Batch ID check box.

See also

- Batch identification on page 163
- Enable editing of batch IDs on page 164

Define a custom default batch ID

If you define a custom default batch ID, when a new batch is added to the batch list, the box on the Batch Creation dialog is automatically populated with the custom default batch ID. If you want to create a custom default batch ID, follow these steps.

**To define a custom default batch ID:**

1. Open FactoryTalk Batch Equipment Editor.
2. From the Options menu, select Server Options.
3. Select the Batch Server tab.
4. In the **Miscellaneous** section, type a default batch ID in the **Default Batch ID** box.

![Server Configuration](image)

5. Click **OK**.

**See also**

- **Batch identification** on page 163
- **Disable editing of batch IDs** on page 164
- **Enable editing of batch IDs** on page 164

### Create a custom batchidcreation.dll

Automatically generate custom batch IDs in any values or increments. If this option is chosen, modify some Visual Basic code, compile it into a new Dynamic Link Library (DLL) file, and then place it in the `BATCHCTL\Bin` folder.

We recommend that you make a backup copy of the original `BatchIDCreation.dll` before making any changes. Supported versions of Visual Basic are shown in the **System Requirements** of the **FactoryTalk Batch Getting Results Guide**.

**Tip:** The FactoryTalk Batch Server and the Visual Basic development environment must be installed on the same computer.

**Tip:** If the AutoBatchID execute fails or the generated batch ID contains invalid characters, the **Batch ID** box in FactoryTalk Batch View becomes editable, even if the **Allow Editing of Batch ID** check box in the **View/Configuration** or **Defaults/General** is disabled. If you created a default batch ID, it populates the **Batch ID** box and is editable. You can also enter the batch ID manually in the event of a failure.

**To customize batchidcreation.dll:**

1. Copy the **Custom** folder from the FactoryTalk Batch CD, including all subdirectories, to the local hard drive, leaving the structure intact.

2. Double-click **BatchIDCreation.vbp**. The Visual Basic project opens. If you see error messages about references that are not yet in the Visual Basic
project, click **Yes** until you see the message that Visual Basic is unable to set the version compatible component. Click **OK**.

3. From the **Project** menu, select **References**. The **References** dialog box opens.

   There are several files that are required for references used in the Visual Basic project. These references must be set manually within the Visual Basic IDE.

4. From the **Available References** list, select the following references:

   - **COM+ Services Type Library**
     The COM+ Runtime library, **COMSVCS.dll**, that is needed for COM+ functionality. If this file is not on your system, use **MTXAS.dll**.
   - **OLE Automation**
     The OLE Automation support DLL, **STDOLE32.tlb**, which is usually a standard Visual Basic reference.
   - **batchidcreationinsertion 1.0 Type Library**
     The AutoBatchID type library, **batchidcreationinsertion.tlb**, required to create your customized version of the **BatchIDCreation.dll**.

5. Click **OK**.

6. From the **Project** menu, select **BatchIDCreation Properties**. The **Project Properties** dialog box opens.

7. Select the **Component** tab, and then select **Binary Compatibility**.

8. Click the browse button, locate **C:\Program Files (x86)\Rockwell Software\Bin\BatchID\-Creation.dll**, and then click **Open**.

9. Click **OK**.

10. Modify the source code located between the **Begin - Code Modifications** heading and the **End - Code Modifications** heading within 
    **[GenerateBatch]** to define the BatchID value.
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).

11. From the File menu, click Make BatchIDCreation.dll. The Make Project dialog box opens. Compile the project into a .dll named BatchIDCreation.dll.

12. Click OK.

13. Replace the default BatchIDCreation.dll

14. Restart your computer.

See also

Batch identification on page 163
Define a custom default batch ID on page 165
Chapter 13

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 14
- [System-wide COM security limit settings](#) on page 14
- [FactoryTalk Batch Server install configuration](#) on page 15
- [FactoryTalk Batch Client installation configuration](#) on page 16
- [Add the Batch COM service](#) on page 17

When creating a new user account for the FactoryTalk Batch Server, the following 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.
- 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.
- The user account/user group must exist on all workgroup computers—if you want a local 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.

New user account requirements for the FactoryTalk Batch Server
Configure your FactoryTalk Batch system

After you create a new user account for the FactoryTalk Batch Server, configure your FactoryTalk Batch system.

To configure your FactoryTalk Batch system:

1. Configure the FactoryTalk Batch Server computer:
   a. 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.
   b. If FactoryTalk Batch Material Manager is part of your system, add the new FactoryTalk Batch Server user account to the MTBatchServer group on the Material Server computer.
   c. Add the new FactoryTalk Batch Server user account to the FactoryTalk Batch Server service.
   d. If you are running FactoryTalk Event Archiver in incremental mode, add the new FactoryTalk Batch Server user account in the Batch Archiver service.
   e. If FactoryTalk eProcedure is part of your system, add the new FactoryTalk Batch Server user account to the eProcedure Server service.
   f. Add the new FactoryTalk Batch Server user account to the Batch COM object in Component Services > My Computer > COM+ Applications.

2. Configure the FactoryTalk Batch Client computer(s):
   a. If you are using workgroup security, create the new FactoryTalk Batch Server user account on all FactoryTalk Batch Client computers in the workgroup.
c. If eProcedure is part of your system, add the new FactoryTalk Batch Server user account to **Component Services > My Computer > Properties > COM Security** on all remote eProcedure Client computers. Configure the account with **Access Permissions: Local Access and Remote Access**.

**See also**

- [Server account password change locations](#) on page 171
- [Change the server user account for the FactoryTalk Batch Server service](#) on page 172
- [Change the server user account in FactoryTalk Batch COM+ applications](#) on page 172
- [Change the server user account in My Computer properties](#) on page 173

**Server account password change locations**

If you change the password for the existing FactoryTalk Batch Server user account, change the password in these locations:

- If you are using Workgroup security, change the password for the FactoryTalk Batch Server user account on all FactoryTalk Batch computers in the workgroup.

- Change the password for the FactoryTalk Batch Server user account in the FactoryTalk Batch server service.

  - If you are running FactoryTalk Event Archiver in incremental mode, change the password for the FactoryTalk Batch Server user account to the **Batch Archiver** service.

  - If eProcedure is part of your system, change the password for the FactoryTalk Batch Server user account in the **eProcedure Server** service.

  - Change the password for the FactoryTalk Batch Server account in the **Batch COM object in Component Services > My Computer > COM+ Applications**.

**See also**

- [Configure your FactoryTalk Batch system](#) on page 170
To change the server 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. Point to **Administrative Tools > Services**, and then click **Services**. The **Services** dialog box opens.

2. Right-click **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. Click **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**.

**See also**

Configure your FactoryTalk Batch system on page 170

---

To change the FactoryTalk Batch Server user account in Batch COM+ applications, complete these steps on the FactoryTalk Batch Server computer.

**To change the server user account in FactoryTalk Batch COM+ applications:**

1. Point to **Administrative Tools > Component Services**.

2. Expand **Component Services, Computers, My Computer** and **COM+ Applications**.

3. Right-click **Batch**, and then select **Properties**. The **Batch Properties** dialog box opens.

4. On the **Identity** tab, select **This user** and type the new FactoryTalk Batch Server user name and/or password.

5. Click **OK**.
Tip: The new permissions are applied the next time the FactoryTalk Batch Server is started on the computer.

See also

Server account password change locations on page 171

Change the server user account in My Computer properties

To change the FactoryTalk Batch 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. Point to Administrative Tools > Component Services.
2. Expand Component Services and Computers.
5. Click the Add button to open the Select Users, Computers, or Groups dialog box.
6. Type the new FactoryTalk Batch Server user account name and then click OK.
7. In the Access Permission dialog box, select the new FactoryTalk Batch Server account. In the Permissions list, select Allow for Local Access and Remote Access.
8. Click OK twice.
10. Restart the computer.

See also

Server account password change locations on page 171
Run multiple instances of FactoryTalk Batch View

Batch allows multiple instances of the Batch View, running locally on one computer, to view batches running on one FactoryTalk Batch Server. One or more people can open an instance of FactoryTalk Batch View on the same computer running the FactoryTalk Batch Client software. Each instance of the FactoryTalk Batch View is fully functional and can be used to initiate and control the batch process and view running batches.

See also

Enable multiple FactoryTalk Batch View instances on page 176

Custom configuration files on page 177

Configuration file log location on page 179
Enable multiple FactoryTalk Batch View instances

When FactoryTalk Batch is installed, the `SupportMultipleDisplays` function is disabled by default. Before running multiple instances of FactoryTalk Batch View, enable it in the Windows Registry and create individual configuration files for each instance. When the registry variable `SupportMultipleDisplays` is absent or its value is FALSE in the FactoryTalk Batch Server computer, only one instance of FactoryTalk Batch View is allowed on each computer in the FactoryTalk Batch system. When `SupportMultipleDisplays` is set to TRUE in the FactoryTalk Batch Server computer, multiple instances of FactoryTalk Batch View are allowed on each computer in the FactoryTalk Batch system.

**Important:** The `SupportMultipleDisplays` functionality must be enabled on the computer where the FactoryTalk Batch Server is installed and running.

To enable multiple FactoryTalk Batch View instances:

1. On the computer where the FactoryTalk Batch Server is installed, open the Registry Editor.
   - From the Start menu, select Run, type `regedit` or `regedt32` in the Open box, and then select OK.

2. Navigate to `HKEY_LOCAL_MACHINE > SOFTWARE > BATCHCTL > Setup`.

3. In the right pane, double-click `SupportMultipleDisplays`. The Edit String dialog box opens.

4. Enter **TRUE** in the Value data box, and then click OK.

5. Close the Registry Editor.
See also

Run multiple instances of FactoryTalk Batch View on page 175

Custom configuration files

Each running instance of the Batch View requires a unique configuration file. The configuration file contains the application settings and the user preference settings. The default configuration file is BatchVEW.ini. More configuration files can be created using the file name of BatchVEW_description.ini. All configuration file names must start with BatchVEW and end with .ini.

Example: A configuration file for a user named John is labelled BatchVEW_John.ini.

All configuration files are located in the bin folder where FactoryTalk Batch View is installed. The default location for FactoryTalk Batch Clients is C:\Program Path\Rockwell Software\Batch; the location of the configuration files would be C:\Program Path\Rockwell Software\Batch\bin.

There are two ways to create configuration files: Copy and rename BatchVEW.ini or create a configuration file when Batch View opens.

See also

Copy and rename default configuration file on page 177

Create configuration files when launching Batch View on page 178

Copy and rename a default configuration file

Copy the default BatchVEW.ini configuration file and rename it for the user. Configuration files are created using the file name of BatchVEW_description.ini. All configuration file names must start with BatchVEW and end with .ini.

To copy and rename a default configuration file:

1. On the Batch client computer, using Windows Explorer, navigate to the folder containing BatchVEW.ini.
2. Select BatchVEW.ini and copy it by pressing Ctrl+C.
3. Paste (Ctrl+V) the copy of the file.
4. Right-click the copied file (labelled Copy of BatchVEW.ini), and then select Rename from the popup menu.
5. Type the desired name for the new configuration file in the format of BatchVEW_description.ini, where _description is the user name.
6. From the Start menu, select All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch > View. The Display Configuration dialog box opens.

7. Click Start to launch the Batch View with the default BatchVEW.ini file, or select a configuration file from the Available Configurations list, and then click Start. The Batch View opens. The title bar displays the name of the configuration file used to start the program.

See also

Custom configuration files on page 177
Create configuration files when launching Batch View on page 178

Create configuration files when launching FactoryTalk Batch View

Create unique configuration files for each instance of the Batch View when the Display Configuration dialog box opens.

To create configuration files when launching FactoryTalk Batch View:

1. From the Start menu, select All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch > View. The Display Configuration dialog box opens.

2. Select the Start View with a New Configuration option.

3. Select a configuration on which to base the new configuration from the Base Configuration list, for example, BatchVEW_John.ini.
4. Enter a new user name in the **New Configuration Description** box.

   The new configuration file name displays in the **New Configuration File** box.

5. Click **Start**. The Batch View opens with the name of the configuration file used to start the program displayed in the title bar.

![Batch View Configuration](image)

See also

- Custom configuration files on page 177
- Copy and rename default configuration file on page 177

**Configuration file log location**

Each running instance of the Batch View generates its own log file. The Batch View automatically generates the log files based on the configuration files. The log file has the same file name as the configuration file name with the extension of .log. For example, the configuration file `BatchVEW_John.ini` would be associated with a log file called `BatchVEW_john.log`.

All log files are located in the `\bin` folder where FactoryTalk Batch is installed. The default location for FactoryTalk Batch, for example, is `C:\Program Files (x86)\Rockwell Software\Batch`; the location of the log files would be `C:\Program Files (x86)\Rockwell Software\Batch\Bin`.

![Log File Location](image)

See also

- Run multiple instances of FactoryTalk Batch View on page 175
Instances of the FactoryTalk Batch View can be launched from the command line, with or without a configuration file. If a custom configuration file is specified from the command line, it must already exist or you will get an error message.

Run a customized instance of the Batch View from the command line regardless of whether or not the `SupportMultipleDisplays` value is set to `TRUE` or `FALSE`, or the value exists in the registry.

**To start FactoryTalk Batch View from the command line:**

1. From the **Start** menu, select **Run**, then enter `cmd`.
2. Navigate to `C:\Program Files (x86)\Rockwell Software\Batch\Bin`.
3. Enter `BatchVEW.exe`.

   **Tip:** If the `BatchVEW.ini` file does not exist, the Batch View displays an error message and then terminates.

**See also**

- [Enable multiple FactoryTalk Batch View instances](#) on page 176
- [Run multiple instances of FactoryTalk Batch View](#) on page 175
- [Start FactoryTalk Batch View with a configuration file](#) on page 180

You can specify the configuration file name in the command prompt when starting FactoryTalk Batch View from the command line.

If the `SupportMultipleDisplays` option is set to `TRUE` on the FactoryTalk Batch Server computer and no configuration file is specified in the command prompt, the **Display Configuration** dialog box opens. You can select an existing configuration file or create a new one.

If the `SupportMultipleDisplays` is set to `TRUE` on the FactoryTalk Batch Server computer and the specified configuration file does not exist, FactoryTalk Batch View displays an error message and then terminates.

**To start FactoryTalk Batch View with a configuration file:**

1. From the **Start** menu, select **Run**, then enter `cmd`.
2. Navigate to `C:\Program Files (x86)\Rockwell Software\Batch\Bin`.
3. Enter `BatchVEW.exe BatchVEW_description.ini`
Tip: If the `BatchVEW_description.ini` file does not exist, FactoryTalk Batch View displays an error message and then terminates.

See also

- Enable multiple FactoryTalk Batch View instances on page 176
- Run multiple instances of FactoryTalk Batch View on page 175
Appendix A

General troubleshooting errors

This section outlines errors that may occur while installing and running FactoryTalk Batch and possible solutions.

Automatic repair of batch component installation fails.

This can happen if one or more folders were moved in Windows explorer. For example, if the Schema folder gets moved away from its original directory, the installation package starts up automatically and tries to repair the problem. This may clear out the Model and Server keys in the registry, which will prevent the batch client editors from finding the Network Model.

The solution is to completely uninstall the application and then reinstall it.

Error Message: (none)

FactoryTalk Batch Server does not start.

This can happen if:

- Journal path has not been defined in FactoryTalk Batch Server options. Modify the FactoryTalk Batch Server options to include a valid Primary Journal path. Verify that the Secondary Journal path, if defined, is valid.
- Restart path has not been defined in FactoryTalk Batch Server options. Modify the FactoryTalk Batch Server options to include a valid Primary Restart path. Verify that the Secondary Restart path, if defined, is valid.
- Incorrect .Net registry key value.

The FactoryTalk Batch Server is always installed with the default user and password.

The FactoryTalk Batch Server’s logon and password settings are not migrated from the server .ini file during an upgrade. After a new installation, to run the server as a different user, you must change the logon and password settings, using the Control Panel Services dialog box.

Error Message: (none)

Check the FactoryTalk Batch Service Manager. If the Allow Demo Mode check box is selected, clear the check box.

FactoryTalk Batch Server runs for a specific period of time and then stops, and consistently stops at the same time interval.
Appendix A  General troubleshooting errors

FactoryTalk Batch Service Manager does not display the FactoryTalk Batch Server or FactoryTalk Event Archiver.

Error Message: (none)

- The computer where the services are located is not selected.
  Click the Select Computer button and select the appropriate computer.

- The FactoryTalk Batch Server or FactoryTalk Event Archiver services are not registered with the Windows Registry.
  To register a service, click the Start button, point to All Programs, and then select Command Prompt. Navigate to the Program Files (x86)\Rockwell Software\Batch\bin directory. To register the FactoryTalk Batch Server type batchsrv /service at the command prompt, and press Enter. To register FactoryTalk Event Archiver, type batcharc /service at the command prompt, and press Enter.

FactoryTalk Batch Service Manager displays Batch.Server Class instead of Batch Server.

Error Message: (none)

The service displays the class name instead of the display name.

This is caused by DCOMCNFG configuring 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.

The FactoryTalk Batch View/Client is properly configured, but cannot view the Event Journals. The text in FactoryTalk Batch dialog boxes does not display properly.

Error Message: (none)

There may be a version conflict in one or more of the required files.

Error Message: (none)

FactoryTalk Batch does not support the use of large system fonts.
Use the Display Properties dialog box to change the default system font size to Small Fonts.

Unable to locate files.

Error Message: The Recipe directory file could not be found.

The server options for the FactoryTalk Batch Server are not configured to point to the folder that contains the Recipe.dir file. (See Edit the FactoryTalk Batch server options for more information.)

An Application Log error is generated.

Error Message: The Application log file is full.

The Application log is not set for wrapping.
Use the **Event Viewer** to change the Application Event Log Wrapping setting to **Overwrite Events as Needed**:

1. Click the **Start** button, point to **All Programs > Administrative Tools**, and then select **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**.
5. Exit the **Event Viewer**.

**A Licensing error message is generated in Visual Basic.**

**Error Message:** *License information for this component not found. You do not have an appropriate license to use this functionality in the design environment.*

The ActiveX controls did not install correctly.

Visual Basic should be installed *before* FactoryTalk Batch is installed. If you have already installed FactoryTalk Batch:

1. Uninstall the FactoryTalk Batch Client to remove the ActiveX controls. (Refer to the *FactoryTalk Batch Components Upgrade and Installation Guide* for information on uninstalling the client).
2. Re-install Visual Basic.
3. Re-install the FactoryTalk Batch Client using the FactoryTalk Batch Setup program.

**Recipes not visible from FactoryTalk Batch View program.**

**Error Message:** *(none)*

When you click **Create a Batch**, no recipes display in the **Select a Recipe** dialog box.

The **Release Recipe to Production** check box is not selected in the recipe header data. Or, if recipe approvals are enabled, the **Release Recipe to Production** step is not approved.

**Recipe approvals disabled:** 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.
Recipe approvals enabled: 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 Message: (none)

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.

Error Message: (none)

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.

See also

FactoryTalk Batch performance chart on page 162
Copyright notice

Copyright © 2018 Rockwell Automation Technologies, Inc. All Rights Reserved. Printed in USA.

This document and any accompanying Rockwell Software products are copyrighted by Rockwell Automation Technologies, Inc. Any reproduction and/or distribution without prior written consent from Rockwell Automation Technologies, Inc. is strictly prohibited. Please refer to the license agreement for details.

End User License Agreement (EULA)

You can view the Rockwell Automation End-User License Agreement ("EULA") by opening the License.rtf file located in your product's install folder on your hard drive.

Trademark Notices

Allen-Bradley, CompactLogix, ControlLogix, Connected Components Workbench, eProcedure, FactoryTalk, GuardLogix, GuardPLC, Micro800, MicroLogix, MobileView, MobileView Guard, PanelBuilder, PanelView, PhaseManager, PlantLink, PlantPAx, PLC-2, PLC-3, PLC-5, PowerMonitor, RSBizWare, RSBizWare BatchHistorian, RSBizWare Historian, RSBizWare MaterialTrack, RSLinx, Rockwell Automation, Rockwell Software, SLC, SoftLogix, Studio 5000, and XMare trademarks of Rockwell Automation, Inc.

Any Rockwell Automation software or hardware not mentioned here is also a trademark, registered or otherwise, of Rockwell Automation, Inc.

For a complete list of products and their respective trademarks, go to: http://www.rockwellautomation.com/rockwellautomation/legal-notices/overview.page?tab4

Warranty

This product is warranted in accordance with the product license. The product's performance may be affected by system configuration, the application being performed, operator control, maintenance, and other related factors. Rockwell Automation is not responsible for these intervening factors. The instructions in this document do not cover all the details or variations in the equipment, procedure, or process described, nor do they provide directions for meeting every possible contingency during installation, operation, or maintenance. This product's implementation may vary among users.
This document is current as of the time of release of the product; however, the accompanying software may have changed since the release. Rockwell Automation, Inc. reserves the right to change any information contained in this document or the software at any time without prior notice. It is your responsibility to obtain the most current information available from Rockwell when installing or using this product.

**Environmental compliance**


**Contact Rockwell Automation**

Customer Support Telephone — 1.440.646.3434

Index

A

ActiveX controls  
  troubleshoot  175
administrative Tools  73
allowable failures  95
Archiver  
  (Incremental)  91
  event filters  94
  service  164
audit
  Equipment Editor  60
  message generation  59
  Recipe Editor  60
AutoBatchID  
  execute failure  159
  automatic
    restart control  101

B

Batch  
  audit  59
Batch COM+ Application  15
Batch IDs  
  customize  157
Batch Server  
  cold boot  76, 154
  computer name  101
  demo mode  73, 76
  does not start  175
  end of batch  154
  failure  153
  grace period  76
  options  86
  performance chart  156
  restart  153
  user group  15
  warm all boot  76
  warm boot  76
  warm restart  154
Batch Server Service

...
names 134
Network Editor 134
SQL Server 175
default
cross invocation values 97
hyperlink values 99
descriptors
cross invocation 97
hyperlink label 99
directory share 11
disable
Archiver service 91
DSDF 12
dwell time 124

E
electronic batch record 154
Electronic Signatures
policy setting 55
eProcedure
hyperlinks 99
Equipment Editor
access modes 32
security 31
error message
application log is full 175
license for component not found 175
recipe directory file not found 175
ErrorLogDirectory 155
errors
troubleshooting 175
event journal signatures
electronic batch records 95
event journals 18
Event Viewer 107
external security device 29

F
FactoryTalk AssetCentre
message fields 70
FactoryTalk Batch server install configuration 15
FactoryTalk Directory 20
FactoryTalk Help 20
FactoryTalk Security 20
FactoryTalk Services Platform 20
FactoryTalk Windows Firewall Configuration Utility 16
failure
handshake timeout 95
phase 95
watchdog timeout 95
formatted report 149

G
General Usage Phase Logic Requests 54

H
HMI
change logged-on user 27
hold propagation
setting 95
hyperlink
default values 99

I
Incremental Archiver 91

L
list report 149
list view properties
change 142
log files 18

M
manual failover 102
Material Manager 99
Maximum Log File Size box 95
Minimum Disk Space box 95

N
Network Editor
database 134
network model
information 134
synchronize 137
Object Database Connectivity (ODBC)  86, 91
OLE for process control (OPC)  11
OPC groups  12

Period (msec) box  95
permissions  11
phase
  material-enabled  121
phase logic simulation  121
policy settings
  ESignature Network Directory  55
print preview  150
print reports
  print a range of pages  151
product policies  21

Recipe Editor
  access modes  32
  configuration options  33
recipes
  release to production  175
report preferences
  set  145
reports
  change margins  145
  column and row preferences  145
  configure footings  145
  configure headings  145
resource types
  cross invocation  97
  hyperlinks  99
restart Batch Server  153
restore a batch  154

security
  automatic logoff time period  27
  Batch View  22
  event journal  18
  external device  29
  file  18
  folder  18

log file  18
Windows  13
Server Options
  Archiver (End of Batch)  91
  Archiver (Incremental)  91
  Archiver Event Filters  94
  Batch Reporting tab  91
  Batch Server tab  95
  Cross Invocation tab  97
  edit  86
  redundant servers  89
  Restart Control tab  89
Server Options dialog box  76
set
  report preferences  145
  user preferences  139
shared directory  11
Signature Requests
  General Usage  54
Signature Templates  41
simulator  121
ack/clear requests  130
change dwell time  124
command a phase  128
indicate a continuous phase  125
interface  122
phase configuration  123
phase ownership  128
phase parameter tag  125
phase status  125
report parameter tag  126
request data tags  126
request sequence  124
restore/clear requests  130
unit tag status  132
synchronize  137
System Log file  175
system problems  153

Tag verification  155
troubleshoot
  ActiveX controls  175
  application event log  175
  Batch Server failure  153
  Batch Service Manager  175
  batch terminated  154
error messages 153
event journal files 175
recipes not visible 175
SQL Server selection 175
system problems 153
tag verification 155
text in Batch dialog boxes 175

U
user account
  create 13
User Preference dialog box
  change colors 140, 142
  change fonts 142
  change view properties 142
  column and row preferences 145
  edit information tips 141
  list view properties 142
  set 139
User-Defined (End of Batch) 91

V
view properties 142

W
Windows Desktop 27
Windows Event Log
  modify settings 111
  tag verification 156
Windows registry
  .NET key 101
Windows services
  Archiver 73
  Batch Server 73
workgroup
  configuring client 21
Rockwell Automation support

Rockwell Automation provides technical information on the web to assist you in using its products. At http://www.rockwellautomation.com/support you can find technical and application notes, sample code, and links to software service packs. You can also visit our Support Center at https://rockwellautomation.custhelp.com for software updates, support chats and forums, technical information, FAQs, and to sign up for product notification updates.

In addition, we offer multiple support programs for installation, configuration, and troubleshooting. For more information, contact your local distributor or Rockwell Automation representative, or visit http://www.rockwellautomation.com/services/online-phone.

Installation assistance

If you experience a problem within the first 24 hours of installation, review the information that is contained in this manual. You can contact Customer Support for initial help in getting your product up and running.

<table>
<thead>
<tr>
<th>United States or Canada</th>
<th>1.440.646.3434</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outside United States or Canada</td>
<td>Use the Worldwide Locator available at <a href="http://www.rockwellautomation.com/locations">http://www.rockwellautomation.com/locations</a>, or contact your local Rockwell Automation representative.</td>
</tr>
</tbody>
</table>

New product satisfaction return

Rockwell Automation tests all of its products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning and needs to be returned, follow these procedures.

<table>
<thead>
<tr>
<th>United States</th>
<th>Contact your distributor. You must provide a Customer Support case number (call the phone number above to obtain one) to your distributor to complete the return process.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outside United States</td>
<td>Please contact your local Rockwell Automation representative for the return procedure.</td>
</tr>
</tbody>
</table>

Documentation feedback

Your comments will help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete the feedback form, publication RA-DU002.

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

www.rockwellautomation.com

Power, Control and Information Solutions Headquarters
America: Rockwell Automation, 1201 South Second Street, Milwaukee, WI 53204-2496 USA. Tél: (1) 414.382.2000, Fax: (1) 414.382.4444
Europe/Middle East/Africa: Rockwell Automation NV, Pegasus Park, De Kleerlaan 12a, 1831 Diegem, Belgium, Tél: (32) 2 663 0600, Fax: (32) 2 663 0640
Asia Pacific: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tél: (852) 2887 4788, Fax: (852) 2508 1846

Rockwell Automation Publication FactoryTalk Batch Administrator Guide - November 2018

Copyright © 2018 Rockwell Automation Technologies, Inc. All Rights Reserved. Printed in the U.S.A.
FactoryTalk® Batch Equipment Editor
Important User Information

Solid-state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation, and Maintenance of Solid State Controls (publication SGI-1.1 available from your local Rockwell Automation sales office or online at http://www.rockwellautomation.com/literature/) describes some important differences between solid-state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid-state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

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

Allen-Bradley, Rockwell Software, and Rockwell Automation ControlLogix, iProcedure, FactoryTalk, RSbaWare, RSbaWare BatchHistorian, RSbaWare Historian, RSbaWare MaterialTrack, and RSlinx are trademarks of Rockwell Automation, Inc.

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

## Chapter 1

**FactoryTalk Batch Equipment Editor introduction**

- Additional resources .......................................................................................................... 12
- FactoryTalk Batch Equipment Editor security ............................................................ 13
  - Open FactoryTalk Batch Equipment Editor .......................................................... 14
- FactoryTalk Batch Equipment Editor interface .......................................................... 15
  - Menu bar ...................................................................................................................... 16
  - Toolbar ......................................................................................................................... 17

## Chapter 2

**Area models**

- Create a new area model ................................................................................................... 20
- Open an existing area model ............................................................................................ 21
  - Area models migration .............................................................................................. 22
  - Area models and FactoryTalk Batch Material Manager ..................................... 22
    - Material database changes..................................................................................... 23
    - Material database is unavailable ....................................................................... 23

## Chapter 3

**Import and export area models overview**

- XML area models ............................................................................................................... 25
  - Import an XML area model ..................................................................................... 25
    - Import enumerations .......................................................................................... 27
    - Import system phase classes .............................................................................. 28
  - Export an area model to XML ................................................................................. 29
    - Convert XML to binary from the command line overview.................................. 29

## Chapter 4

**Enumerations overview**

- System enumeration sets ............................................................................................... 32
- Create enumeration sets .............................................................................................. 33
  - Add enumeration members ..................................................................................... 34
- Edit enumerations and enumeration sets .................................................................... 35
  - Delete enumerations and enumeration sets ............................................................ 36

## Chapter 5

**Parameters and reports with FactoryTalk Batch**

- Parameters and reports in recipes .................................................................................... 39
  - Parameters example ................................................................................................... 40
  - How parameters and reports work with phase tags .............................................. 40
    - Additional request, parameter, and report tags for phases .............................. 40
    - Phase tags ................................................................................................................. 42
  - Batch parameters and reports ............................................................................... 43
Table of contents

Phase class parameters ................................................................. 43
Phase class reports ................................................................. 43
Material parameters ................................................................. 44
Types of containers ................................................................. 44
Default material parameters .................................................. 45
Scale parameters ........................................................................ 46
Verification method for parameter deviations ......................... 46
Control strategies overview ...................................................... 46
Control strategies example ....................................................... 47
Implement control strategies .................................................. 48
Parameter and report subsets overview ..................................... 48
Phase class with different subsets example ............................... 50
Download parameter requests ............................................... 50
Upload report requests .......................................................... 51
Automatic Upload/Download ................................................ 52
Control strategies with upload/download parameter subsets ........ 53

Chapter 6

Electronic signatures

Types of electronic signatures ................................................. 55
How electronic signatures work ............................................... 56
Signature template ........................................................................ 57
  Edit signature templates dialog box ....................................... 58
  Add a signature template ..................................................... 59
  Configure signature template signoffs .................................. 61
    Assign security permissions for a signoff .............................. 63
    Remove signoff permissions for a user or group .................. 65
  Delete a signature template ............................................... 66
  Modify a signature template ................................................. 67
Command verification policies .................................................. 68
  Add a signature requirement to commands ......................... 68
    Edit command verification policies dialog box .................... 70
  Change the verification policy for a command ...................... 70
  Remove the signature requirement for a command ............... 71

Recipe approval process overview

Recipe approval tasks ................................................................. 73
Configure recipe approval steps .............................................. 74
  Recipe Approvals Configuration dialog box ............................. 76
Edit a recipe approval ............................................................ 77
Disable recipe approval .......................................................... 77
## Table of contents

<table>
<thead>
<tr>
<th>Chapter 8</th>
<th>Security authority overview</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Security authority configuration ...................................................... 80</td>
</tr>
<tr>
<td></td>
<td>Secure an area model .............................................................................. 81</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 9</th>
<th>Data servers overview</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Data server definition files ................................................................. 83</td>
</tr>
<tr>
<td></td>
<td>Area model data servers .......................................................................... 84</td>
</tr>
<tr>
<td></td>
<td>Data Servers dialog box .......................................................................... 85</td>
</tr>
<tr>
<td></td>
<td>OPC data servers ...................................................................................... 85</td>
</tr>
<tr>
<td></td>
<td>Add an OPC data server ............................................................................ 87</td>
</tr>
<tr>
<td></td>
<td>RSLinx Enterprise FactoryTalk Live Data server .................................... 88</td>
</tr>
<tr>
<td></td>
<td>Add a data server ..................................................................................... 89</td>
</tr>
<tr>
<td></td>
<td>RSLinx Enterprise server path .................................................................. 90</td>
</tr>
<tr>
<td></td>
<td>Configure the RSLinx Enterprise server path .......................................... 91</td>
</tr>
<tr>
<td></td>
<td>Edit a data server ................................................................................... 91</td>
</tr>
<tr>
<td></td>
<td>Delete a data server ................................................................................. 92</td>
</tr>
<tr>
<td></td>
<td>Assign data servers .................................................................................. 93</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 10</th>
<th>Resources overview</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Arbitration of resources ......................................................................... 95</td>
</tr>
<tr>
<td></td>
<td>Define shared resources ........................................................................... 97</td>
</tr>
<tr>
<td></td>
<td>Configure general information for resources .......................................... 98</td>
</tr>
<tr>
<td></td>
<td>Configure arbitration information for resources ...................................... 98</td>
</tr>
<tr>
<td></td>
<td>Configure cross invocation information for resources ............................ 100</td>
</tr>
<tr>
<td></td>
<td>Configure hyperlink information for resources ....................................... 101</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 11</th>
<th>Global binding requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Unit binding expression builder ................................................................ 103</td>
</tr>
<tr>
<td></td>
<td>Expression operators .................................................................................. 104</td>
</tr>
<tr>
<td></td>
<td>Supported data types ................................................................................ 105</td>
</tr>
<tr>
<td></td>
<td>Operands .................................................................................................... 105</td>
</tr>
<tr>
<td></td>
<td>Expression validation ............................................................................... 105</td>
</tr>
<tr>
<td></td>
<td>Create global binding requirements .......................................................... 106</td>
</tr>
<tr>
<td></td>
<td>Edit binding requirements ........................................................................ 108</td>
</tr>
<tr>
<td></td>
<td>Delete binding requirements ..................................................................... 108</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Chapter 12</th>
<th>Process cell overview</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Create a process cell class ...................................................................... 111</td>
</tr>
<tr>
<td></td>
<td>Edit a process cell class ......................................................................... 112</td>
</tr>
<tr>
<td></td>
<td>Create and configure process cell instances ............................................. 112</td>
</tr>
</tbody>
</table>
Configure a process cell instance ................................................................. 113
Configure the general data for a process cell.............................................. 113
  Edit process cell dialog box - General tab .............................................. 114
Configure the arbitration data for a process cell...................................... 114
  Edit Process Cell dialog box - Arbitration tab .................................... 116
Configure cross invocation data for a process cell.................................. 116
  Edit process cell dialog box - Cross Invocation tab............................ 117
Configure hyperlink data for a process cell............................................. 118
  Edit process cell dialog box - Hyperlink tab ....................................... 119
Remove a process cell instance................................................................. 119
Verify icon paths......................................................................................... 120
Unique icons............................................................................................... 121

Chapter 13

Unit overview

Create a unit class ...................................................................................... 124
Edit a unit class ......................................................................................... 125
Create a unit instance ................................................................................. 125
Unit instance configuration ...................................................................... 126
  Open the Edit Unit dialog box ............................................................. 126
  Configure the general data for a unit................................................... 127
    Edit Unit dialog box - General tab ................................................. 128
  Configure unit attribute tags data....................................................... 128
    Edit Unit dialog box - Attribute Tags tab ....................................... 129
  Configure arbitration data for a unit.................................................. 129
    Edit Unit dialog box - Arbitration tab ........................................... 131
  Configure cross invocation data for a unit....................................... 131
    Edit Unit dialog box - Cross Invocation tab .................................. 132
  Configure hyperlink data for a unit.................................................... 132
    Edit Unit dialog box - Hyperlink tab ............................................ 133
Remove a unit instance ........................................................................... 134
Link unit instances .................................................................................. 134
Share units ............................................................................................... 134

Chapter 14

Tags overview

Unit attribute tags .................................................................................... 138
  Add a unit attribute .............................................................................. 138
  Edit a unit attribute ............................................................................ 140
  Delete a unit attribute ......................................................................... 140
Create a unit attribute tag .................................................................... 141
  Unit Attribute Tags dialog box ......................................................... 142
Phase tags ............................................................................................... 143
  Edit tags from the menu .................................................................... 144
Default equipment names ...................................................................... 145
  Set all tag items to default name and access path ................................. 145
# Table of contents

- Global unit attribute: 146
- Create a global unit attribute: 147
- Edit a global unit attribute: 148
- Delete a global unit attribute: 149
- Remove a global unit attribute: 149

## Chapter 15

### Phase class overview

- Tags: 152
- Create and configure phase classes: 152
  - Configure phase class general attributes: 153
    - Create Phase Class dialog box - General tab: 154
  - Configure phase class parameters: 155
    - Create Phase Class dialog box - Parameters tab: 157
  - Configure phase class reports: 160
    - Create Phase Class dialog box - Reports tab: 161
  - Select context IDs: 164
  - Configure phase class messages: 165
    - Create Phase Class dialog box - Messages tab: 166
  - Configure phase class control strategies: 166
    - Create Phase Class dialog box - Control Strategy tab: 168
  - Verification policies: 169
    - How verification policies work: 169
    - Configure the verification policy for parameters and reports: 171
      - Verification Policies dialog box: 173
    - How verification policies work with control strategies: 175
    - How editing target parameters affects report verification policies: 176
  - Material-enabled phases: 176
    - Standard phase class with matching parameter and report names: 177
    - Standard phase class with control strategies configured: 178
  - Disable material on a material-enabled phase class: 178
  - Duplicate a phase class: 178
  - Copy a phase class: 178
  - Cut and paste a phase class: 179

## Chapter 16

### Configure phases

- Create a phase: 182
  - Configure the general data for a phase: 183
    - Edit Phase dialog box - General tab: 184
  - Configure the tag data for a phase: 185
    - Edit Phase dialog box - Tag tab: 186
    - Phase tag: 187
  - Set tag items to the default addresses: 190
  - Configure the arbitration data for a phase: 191
    - Edit Phase dialog box - Arbitration tab: 192
## Table of contents

Configure cross invocation data for a phase ................................................................. 192  
  Edit Phase dialog box - Cross Invocation tab ........................................................... 193  
Configure hyperlink data for a phase ........................................................................ 193  
  Edit Phase dialog box - Hyperlink tab ................................................................. 194  
Configure container data ....................................................................................... 194  
  Edit Phase dialog box - Containers tab .............................................................. 195  
Parameter limit tags .............................................................................................. 197  
  Enable parameter limit tags ................................................................................ 197  
  Edit Phase dialog box - Parameter Limit Tags tab ........................................... 197  
  Map parameter tags to parameter limit tags .................................................... 198  
Report limit tags .................................................................................................. 199  
  Edit Phase dialog box - Report Limit Tags tab ............................................... 199  
  Enable report limit tags .................................................................................... 200  
  Map report tags to report limit tags ................................................................. 201  
Configure parameter and report limit tags ............................................................... 202  
  How edits to parameters, reports, and phase classes affect limit tags .......... 203  
Overview of sharing phases and operation sequences between units ................... 203  
  Share a phase or operation sequence ................................................................. 204

## Chapter 17

Create an operation sequence ............................................................................... 205  
  Edit Operation Sequence dialog box - General tab .......................................... 206  
  Configure the general data for an operation sequence ..................................... 206  
  Edit Operation Sequence dialog box - Arbitration tab .................................... 207  
  Configure the arbitration data for an operation sequence ............................ 207  
  Edit Operation Sequence dialog box - Tags tab .............................................. 208  
  Configure operation sequence class parameters ............................................ 209  
  Configure operation sequence class reports .................................................... 210  
Overview of sharing phases and operation sequences between units ................... 211  
  Share a phase or operation sequence ................................................................. 212

## Appendix A

Area model object warm restart support ................................................................. 213  
Resource classes warm restart support ............................................................... 213  
Resources warm restart support ......................................................................... 214  
Unit attributes warm restart support ................................................................. 214  
Unit attribute tags warm restart support ........................................................... 215  
Equipment phase tags warm restart support ..................................................... 215  
Enumeration sets warm restart support .............................................................. 215  
Data servers warm restart support ..................................................................... 216  
Process cell classes warm restart support ........................................................ 216  
Process cells warm restart support .................................................................... 217  
Unit classes warm restart support ..................................................................... 217  
Units warm restart support ................................................................................ 218
## Table of contents

Parameters warm restart support ................................................................. 218  
  Added parameters warm restart support ................................................. 219  
Report warm restart support ........................................................................... 220  
  Added reports warm restart support ......................................................... 220  
Configured messages warm restart support .............................................. 221  
Recipe phases warm restart support ................................................................. 221  
Equipment phases warm restart support ......................................................... 222

## Appendix B

### Troubleshoot issues

- Unable to access instruction file ................................................................. 223  
- Illegal characters encountered in writing instruction file .......................... 224  
- Unable to access Material Server for material support per container ........ 224  
  
### COM-related exception errors

- XML parser error messages ........................................................................ 225  
- Material-specific errors ............................................................................. 226

### Invalid signoff

- Resolve invalid signoff caused by deleted group or user ......................... 228  
- Resolve invalid signoff when required information is omitted ............... 229

### No Signature templates configured within the Area Model

- Equipment Editor: Area model is secured to a different FactoryTalk Network Directory ................................................................. 231
  
Recipe Editor: Area model is secured to a different FactoryTalk Network Directory ................................................................. 231

## Legal Notices

- Legal Notices ............................................................................................... 233
FactoryTalk Batch Equipment Editor introduction

Use FactoryTalk Batch Equipment Editor to create and maintain an equipment database. The area model is the physical component of a batch facility. A database that consists of all equipment in the facility and all of the tasks that it is capable of performing. The area model is stored in a file with a .cfg file extension and is available to all other FactoryTalk Batch programs, including FactoryTalk Batch Recipe Editor, FactoryTalk Batch View, and Phase Simulator.

For example, during recipe configuration, the area model provides a list of available units and phases. Later during recipe verification, it verifies that the designated equipment is capable of executing the procedures. During recipe execution, resource arbitration functions use the area model to allocate equipment based on recipe and operator requests.

Arbitration is a form of coordination control that determines how a resource allocate when there are more requests for the resource than can be accommodated at one time.

**Important:** Verify any recipes that run against a new or modified area model, and restart the FactoryTalk Batch Server service.

As outlined in the ISA S88.01 Batch Control Standard, the facility area model organizes into model components:

- Process Cell
- Unit
- Phase
- Control Module

In addition to the creation of an area model, FactoryTalk Batch Equipment Editor:

- Defines signature templates for electronic signatures and recipe approval signoffs.
- Configures steps and approval signoffs for recipe approval.
Chapter 1  
FactoryTalk Batch Equipment Editor introduction

- Enables recipe version control.
- Secures an area model using security authority.
- Specifies server communication functions.
- Specifies FactoryTalk Event Archiver filters and reporting options.

If using FactoryTalk Batch with FactoryTalk Batch Material Manager, set security for FactoryTalk Batch to access the material database.

See also

FactoryTalk Batch Equipment Editor security on page 13

Additional resources

These documents contain additional information concerning related Rockwell Automation products.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FactoryTalk Batch ActiveX Controls User Guide</td>
<td>Quick and easy access to information and instructions required to use the FactoryTalk Batch ActiveX Custom Controls.</td>
</tr>
<tr>
<td>FactoryTalk Batch Administrator Guide</td>
<td>Instructions for configuring security and services, and implementation and use of components not normally accessed or used by batch operators, such as the FactoryTalk Batch Server.</td>
</tr>
<tr>
<td>FactoryTalk Batch Server API Communication Language Reference Guide</td>
<td>Provides quick and easy access to 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 those who want to develop custom interfaces.</td>
</tr>
<tr>
<td>FactoryTalk Event Archiver User Guide</td>
<td>The FactoryTalk Event Archiver collects data from the batch record files (.evt files) generated by the FactoryTalk Batch Server and stores the data in a Microsoft SQL Server database.</td>
</tr>
<tr>
<td>FactoryTalk eProcedure Client User Guide</td>
<td>Provides information and procedural instructions required to create and command batches using the FactoryTalk® eProcedure® Client, and can be used as a reference information by the operator.</td>
</tr>
<tr>
<td>FactoryTalk Batch PCD Programmer Technical Reference Guide</td>
<td>Provides information and instructions about the FactoryTalk® Batch-PCD interface design. It is intended to be used as a reference guide.</td>
</tr>
</tbody>
</table>
## Resource Description

<table>
<thead>
<tr>
<th>Resource</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FactoryTalk Batch PhaseManager User Guide</td>
<td>Describes the integration of the FactoryTalk Batch software with the Studio 5000 Logix Designer™ application and the Logix5000™ family of controllers. This integration simplifies the configuration and maintenance of the FactoryTalk Batch automation system, provides a superior means of communication between the FactoryTalk Batch Server and the Logix5000 controller, and significantly reduces the programming effort required to develop the phase logic code that resides in your Logix5000 controller.</td>
</tr>
<tr>
<td>FactoryTalk Batch Recipe Editor User Guide</td>
<td>Use 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 graphically organize recipes into procedures, unit procedures, operations, and phases. Build recipes using either the SFC format or a table-based format.</td>
</tr>
<tr>
<td>FactoryTalk Batch View User Guide</td>
<td>FactoryTalk Batch View is used to initiate and execute FactoryTalk Batch automation processing. FactoryTalk Batch View secured objects are located in the FactoryTalk Diagnostics and are modified using the FactoryTalk Administration Console. A system administrator can customize FactoryTalk Batch View security to meet the needs of the facility. FactoryTalk Batch View is used in conjunction with a Human-Machine Interface (HMI).</td>
</tr>
</tbody>
</table>

View or download publications at [http://www.rockwellautomation.com/literature](http://www.rockwellautomation.com/literature). To order paper copies of technical documentation, contact the local Rockwell Automation distributor or sales representative.

### FactoryTalk Batch Equipment Editor security


FactoryTalk Batch Equipment Editor **Single Sign-On** product policy setting created in the FactoryTalk Local Directory in conjunction with the FactoryTalk **Use single sign-on** system policy setting enables and disables the FactoryTalk single sign-on. Single sign-on allows users to log on just once, per directory, on a given computer. Once a user logs on to the FactoryTalk Directory, all participating FactoryTalk-enabled products that run in that directory, on that computer, automatically use the logged on user’s security credentials.
When opening FactoryTalk Batch Equipment Editor, the user logged on to the FactoryTalk Directory automatically logs on to FactoryTalk Batch Equipment Editor or the Log on to FactoryTalk dialog box opens requiring a user name and password.

If enabled, the FactoryTalk Use single sign-on system policy setting and FactoryTalk Batch Equipment Editor Single Sign-On product policy setting is True, these scenarios are possible when opening FactoryTalk Batch Equipment Editor:

- If a FactoryTalk Security user with FactoryTalk Batch Equipment Editor permissions opens FactoryTalk Batch Equipment Editor while currently logged on to the FactoryTalk Directory, the security permissions are the same.
- If a Windows-linked user with FactoryTalk Batch Equipment Editor permissions opens FactoryTalk Batch Equipment Editor, the user logs on to the FactoryTalk Directory.
- If a user is not found, the Log on to FactoryTalk dialog box opens.

**Tip:** FactoryTalk Windows-linked user: A user account created in Windows and then added to the FactoryTalk Directory.

If disabled the FactoryTalk Use single sign-on system policy setting or FactoryTalk Batch Equipment Editor Single Sign-On product policy setting is False, then the Log on to FactoryTalk dialog box displays and a FactoryTalk Security user with permissions to open FactoryTalk Batch Equipment Editor must be entered.

See also

Open FactoryTalk Batch Equipment Editor on page 14

Use these instructions to open FactoryTalk Batch Equipment Editor.

To open FactoryTalk Batch Equipment Editor:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.
2. (optional) If the Log On to FactoryTalk dialog box displays, type the user name and password and then select OK.

3. (optional) If a FactoryTalk user name and password is not found in the FactoryTalk Directory or if the user is not configured to run FactoryTalk Batch Equipment Editor, the Logon Message dialog box opens.

See also

FactoryTalk Batch Equipment Editor security on page 13

FactoryTalk Batch Equipment Editor interface

Use FactoryTalk Batch Equipment Editor to configure the facility area model. The components defined in the FactoryTalk Batch Equipment Editor interface with the facility process-connected devices.

The following illustration shows the FactoryTalk Batch Equipment Editor interface:
The sections of the FactoryTalk Batch Equipment Editor interface include:

<table>
<thead>
<tr>
<th>Item</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Menu bar</td>
<td>The menu bar shows the available menus.</td>
</tr>
<tr>
<td>2</td>
<td>Toolbar</td>
<td>The toolbar shows all the available operations.</td>
</tr>
<tr>
<td>3</td>
<td>Location bar</td>
<td>The location bar indicates the unit and process cell currently working. Select View &gt; Location Bar to show and hide.</td>
</tr>
<tr>
<td>4</td>
<td>Class View area</td>
<td>Shows the existing cell classes, unit classes, or phases in the active area model. The column heading reflects the active level. Double-click (or right-click) an icon to open the Edit dialog box for that item.</td>
</tr>
<tr>
<td>5</td>
<td>Design View area</td>
<td>Use this area to construct the area model and display the layout of the active level. In Select mode, double-click an item to display the layout of the next lower level. Right-click the item to open the Edit dialog box.</td>
</tr>
<tr>
<td>6</td>
<td>Status bar</td>
<td>The right side of the status bar displays the selected unit, process cell, and current user name.</td>
</tr>
</tbody>
</table>

See also

- Menu bar on page 16
- Toolbar on page 17
- Area models on page 19

**Menu bar**

The **Menu** bar commands include:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>File</td>
<td>Lists options for opening, creating, saving, securing, and importing and exporting area models.</td>
</tr>
<tr>
<td>Edit</td>
<td>Edit the various area model components, perform a tag import, and configure signoffs for recipe approval.</td>
</tr>
<tr>
<td>Class</td>
<td>Create a new class of items for the current level.</td>
</tr>
<tr>
<td>View</td>
<td>Specify the FactoryTalk Batch Equipment Editor components to display, and indicate the zoom percentage.</td>
</tr>
<tr>
<td>Options</td>
<td>Specify directories for area model icons and configure the FactoryTalk Batch Server options. Server options include project settings (directories, recipe storage file type, recipe versioning enable and disable), archiver event filters, batch reporting, and other settings.</td>
</tr>
<tr>
<td>Help</td>
<td>Provides on-line help and information regarding the software and the system.</td>
</tr>
</tbody>
</table>

See also

- FactoryTalk Batch Equipment Editor interface on page 15
Toolbar

The FactoryTalk Batch Equipment Editor toolbar contains a group of buttons used to perform commands in FactoryTalk Batch Equipment Editor. The buttons, from left to right, are:

<table>
<thead>
<tr>
<th>Graphic</th>
<th>Name</th>
<th>Description</th>
<th>Keyboard Shortcut</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="new.png" alt="icon" /></td>
<td>New</td>
<td>Creates an area model.</td>
<td>CTRL + N</td>
</tr>
<tr>
<td><img src="open.png" alt="icon" /></td>
<td>Open</td>
<td>Opens an existing area model.</td>
<td>CTRL + O</td>
</tr>
<tr>
<td><img src="save.png" alt="icon" /></td>
<td>Save</td>
<td>Saves the active area model.</td>
<td>CTRL + S</td>
</tr>
<tr>
<td><img src="cut.png" alt="icon" /></td>
<td>Cut</td>
<td>Deletes the highlighted item and saves it to the clipboard.</td>
<td>Delete</td>
</tr>
<tr>
<td><img src="copy.png" alt="icon" /></td>
<td>Copy</td>
<td>Copies the highlighted item and saves it to the clipboard.</td>
<td>CTRL + C</td>
</tr>
<tr>
<td><img src="paste.png" alt="icon" /></td>
<td>Paste</td>
<td>Pastes the cut or copied item from the clipboard.</td>
<td>CTRL + V</td>
</tr>
<tr>
<td><img src="share.png" alt="icon" /></td>
<td>Share</td>
<td>Shares the selected unit or phase class.</td>
<td>CTRL + H</td>
</tr>
<tr>
<td><img src="go_down.png" alt="icon" /></td>
<td>Go Down</td>
<td>Moves down through the levels and displays the next lower level.</td>
<td>None</td>
</tr>
<tr>
<td><img src="go_up.png" alt="icon" /></td>
<td>Go Up</td>
<td>Moves up through the levels and displays the next higher level.</td>
<td>None</td>
</tr>
<tr>
<td><img src="zoom.png" alt="icon" /></td>
<td>Zoom</td>
<td>Specifies the magnification of the current sequential function chart (SFC) display.</td>
<td>None</td>
</tr>
<tr>
<td><img src="add_new_object.png" alt="icon" /></td>
<td>Add New Object</td>
<td>Opens the appropriate Create dialog box for the active level.</td>
<td>None</td>
</tr>
<tr>
<td><img src="edit_area.png" alt="icon" /></td>
<td>Edit Area</td>
<td>Edit the area information.</td>
<td>CTRL + A</td>
</tr>
<tr>
<td><img src="edit_enumerations.png" alt="icon" /></td>
<td>Edit Enumerations</td>
<td>Opens the Create Enumeration Sets and Enumerations dialog box.</td>
<td>CTRL + M</td>
</tr>
<tr>
<td><img src="edit_servers.png" alt="icon" /></td>
<td>Edit Servers</td>
<td>Opens the Create Servers dialog box.</td>
<td>CTRL + D</td>
</tr>
<tr>
<td><img src="synchronize.png" alt="icon" /></td>
<td>Synchronize</td>
<td>Synchronizes the equipment area model with the Logix Designer project file.</td>
<td>CTRL + L</td>
</tr>
<tr>
<td><img src="edit_shared_resources.png" alt="icon" /></td>
<td>Edit Shared Resources</td>
<td>Opens the Edit Resources dialog box.</td>
<td>CTRL + R</td>
</tr>
<tr>
<td><img src="edit_properties.png" alt="icon" /></td>
<td>Edit Properties</td>
<td>Opens the appropriate Edit dialog box for the selected item.</td>
<td>CTRL + E</td>
</tr>
<tr>
<td><img src="edit_tags.png" alt="icon" /></td>
<td>Edit Tags</td>
<td>Opens the Edit Tags dialog box.</td>
<td>CTRL + T</td>
</tr>
<tr>
<td><img src="edit_unit_attributes.png" alt="icon" /></td>
<td>Edit Unit Attributes</td>
<td>Opens the Edit Unit Attributes dialog box.</td>
<td>CTRL + U</td>
</tr>
<tr>
<td><img src="edit_global_binding_requirements.png" alt="icon" /></td>
<td>Edit Global Binding Requirements</td>
<td>Opens the Edit Global Unit Binding Requirements dialog box.</td>
<td>CTRL + B</td>
</tr>
<tr>
<td><img src="edit_signature_templates.png" alt="icon" /></td>
<td>Edit Signature Templates</td>
<td>Opens the Edit Signature Templates dialog box.</td>
<td>None</td>
</tr>
<tr>
<td><img src="edit_command_verification_policies.png" alt="icon" /></td>
<td>Edit Command Verification Policies</td>
<td>Opens the Edit Command Verification Policies dialog box.</td>
<td>CTRL + P</td>
</tr>
</tbody>
</table>
### Graphic Name Description

<table>
<thead>
<tr>
<th>Graphic</th>
<th>Name</th>
<th>Description</th>
<th>Keyboard Shortcut</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="Select Instance" /></td>
<td>Select Instance</td>
<td>Changes the cursor to the selection tool.</td>
<td>None</td>
</tr>
<tr>
<td><img src="image" alt="Link Units" /></td>
<td>Link Units</td>
<td>Changes the cursor to the linking tool. Links define the upstream and downstream relationships between units within a process cell. The FactoryTalk Batch Server uses the links to determine which units are available for selection when defining equipment requirements for procedures.</td>
<td>None</td>
</tr>
<tr>
<td><img src="image" alt="Unlink Units" /></td>
<td>Unlink Units</td>
<td>Changes the cursor to the unlinking tool, to unlink the units within a process cell.</td>
<td>None</td>
</tr>
<tr>
<td><img src="image" alt="Invoke Recipe Editor" /></td>
<td>Invoke Recipe Editor</td>
<td>Opens FactoryTalk Batch Recipe Editor.</td>
<td>None</td>
</tr>
</tbody>
</table>

**See also**

[FactoryTalk Batch Equipment Editor interface](#) on page 15
Chapter 2

Area models

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

Create the area model in this order:

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

In some cases, re-configure upper-level equipment after defining the lower-level equipment.

See also

Enumerations overview on page 31
Data servers overview on page 83
Resources overview on page 95
Phase class overview on page 151
Create a new area model on page 19
Create a new area model

Certain FactoryTalk Batch modules use the area model name to identify the FactoryTalk Batch system data to associate with the area model.

**Important:** The area name must be unique for each FactoryTalk Batch Server running on a network.

To create a new area model:

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. Select **File > New**. A blank area model opens.

3. Select **Edit Area** (€). The **Edit Area** dialog box opens.

4. In **Name**, type a unique name for the area.

5. In **Version**, type a version number.

6. Select **OK** to return to the **Design View** area. The area name displays in the title bar of the **Design View** area.

7. Select **File > Save**.

8. Type an appropriate file name, and then select **Save** to save the area model.

The area model saves to the configuration file and can open at any time for modification.

**Important:** Restart the FactoryTalk Batch Server to initiate the changes and re-verify any recipes that run against any modified area model.

**Tip:** If running FactoryTalk Batch Material Manager and the FactoryTalk Batch Server loses communications with the Material Server while creating or editing the area model, some of the enumeration sets contain only the default values — NULL_MATERIAL, NULL_CLASS, and NULL_CONTAINER.

See also

[Open an existing area model on page 20](#)
Open an existing area model

Use these instructions to open an existing area model.

To open an existing area model:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select File > Open.

3. Locate and select the area model file (.cfg), and then select Open.

   If security authority is applied to the area model, and there is a mismatch between the security authority identifier (SAI) in the area model and the SAI in the FactoryTalk Network Directory, an error message opens:

4. Make note of the FactoryTalk Network Directory and host computer information, then select OK to close the error message.

5. To recover a backup of the FactoryTalk Network Directory SAI, or if the area model in unsecured form is needed.
   
   - Restore the FactoryTalk Network Directory SAI securing the area model. The dialog box provides the name of the computer that hosts that SAI. Use the FactoryTalk Administration Console to restore a saved backup of the SAI. Then use File > Open to open the secured area model.
   
   - Open a copy of the area model. If you have not previously saved the area model, save a copy of the area model. A copy of the area model that is not secured is needed to securing the area model. Once secured the area model, an option to recovery a secured area model becomes available.

See also

Area models migration on page 22

Area models and FactoryTalk Batch Material Manager on page 22
Area models migration

When opening area models created using FactoryTalk Batch version 11.00 or later, the option to have the area model automatically migrated to the latest FactoryTalk Batch version is available. The existing data migrates to the new schema. If the area model is not migrated, the current FactoryTalk Batch Equipment Editor version is unable to open.

Tip: If upgrading from a FactoryTalk Batch version older than the previous major release (such as from version 10.xx or earlier to version 12.xx), contact the Rockwell Customer Support Representative to have the area model upgraded to the new equipment database schema.

See also

Open an existing area model on page 20

Material database changes on page 22

Material database is unavailable on page 23

Area models and FactoryTalk Batch Material Manager

When opening area models containing enumeration sets of the same name as the system enumeration sets (YES_NO, and if FactoryTalk Batch Material Manager is supported, MATERIALS, MATERIAL_CLASSES, and CONTAINERS), rename these non-FactoryTalk Batch Material Manager enumeration sets. Renaming the enumeration sets updates all parameters, report parameters, and unit attribute tags to the renamed enumeration set and creates the new system enumeration set. If not renamed, the area model cannot open in the current FactoryTalk Batch Equipment Editor version.

Tip: When using FactoryTalk Batch Material Manager in conjunction with FactoryTalk Batch, there is a dependency between the area model and the material database. When an area model opens, the MATERIALS and CONTAINERS enumeration sets recreate, based on the materials and containers that exist in the material database. If the Material Server is unavailable, the enumeration sets contain only the default values — NULL_MATERIAL and NULL_CONTAINER.

See also

Open an existing area model on page 20

Material database changes on page 22

Material database is unavailable on page 23
Material database changes

FactoryTalk Batch Material Manager only.

If a member of the MATERIAL enumeration set previously assigned to a phase class is no longer in the material database, a warning message displays the error.

- Select **Continue** to substitute the NULL_MATERIAL enumeration for the unsupported enumeration.
- Select **Cancel** to open the area model unchanged. When affected phase class’s **Parameter** tab opens, the first enumeration in the alphabetically sorted list substitutes for the unsupported enumeration.

**Important:** The controller ID of a material and container is the ordinal value for the enumeration. If modified the item controller ID in the material database, the ordinal value of the associated enumeration changes. This change could affect the phase logic.

See also

- Material database is unavailable on page 23
- Area models and FactoryTalk Batch Material Manager on page 22

Material database is unavailable

The MATERIALS and CONTAINERS enumeration sets recreate each time an area model creates, opens, or imports. If the Material Server is unavailable, the enumeration sets only contain the default values — NULL_MATERIAL and NULL_CONTAINER.

See also

- Area models and FactoryTalk Batch Material Manager on page 22
Chapter 3

Import and export area models overview

FactoryTalk Batch gives the ability to import XML area models or export area models to XML format. XML provides a standard, non-proprietary format for distributing the area model configuration.

See also

Import an XML area model on page 25

Export an area model to XML on page 29

XML area models

XML schemas and XML instance documents have become the industry standard for providing implementation of neutral information. With FactoryTalk Batch, FactoryTalk Batch Equipment Editor can import and export an area model in XML format. This function allows for the exchange of the area model information with other companies in a standard, non-proprietary XML format. The area model information may be edited using text editors or XML editors.

Tip: When securing an area model to a Security Authority Identifier, the Export command disables, and the area model cannot export.

See also

Import an XML area model on page 25

Export an area model to XML on page 29

Convert XML to binary from the command line overview on page 29

Import an XML area model

When an XML area model imports, FactoryTalk Batch Equipment Editor constructs the area model in memory. Save the imported area model to the proprietary binary file or export it to another text file.

FactoryTalk Batch Equipment Editor validates the information provided by the XML file. Only well-formed, validated XML files import. The XML files must have the extension .axml to import. FactoryTalk Batch Equipment Editor attempts to read the files as XML files only if the files have the correct file extension.
Files with extensions other than .axml read as tab-delimited files. Although the .txt file extension is expected for the tab-delimited text files, FactoryTalk Batch Equipment Editor attempts to read all files with extensions other than .axml as tab-delimited text files.

**Tip:** The YES_NO, MATERIALS, MATERIAL_CLASSES, and CONTAINERS enumeration sets recreate each time a new area model creates, opens, or imports. Existing values are not imported. If the Material Server is not available, the MATERIALS, MATERIAL_CLASSES, and CONTAINERS enumeration sets create with the NULL_MATERIAL, NULL_CLASS, and NULL_CONTAINER default values only. (See Import enumerations for more information.)

### To import an XML area model:

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. Select **File > Import**.

3. From the **Files of type** list, select **XML Files (*.axml)**.

4. Locate and select the XML file to import.

5. Select **Open** to initiate the import process. A message box displays the name of each component imported and validated.

6. Select **OK** to complete the import process and return to FactoryTalk Batch Equipment Editor.

7. Select **File > Save As** to save the newly imported file.

   The **Save As** dialog box opens with **Untitled.cfg** highlighted in the **File name** box.
8. In **File name**, type the new file name. Give the file the same name as the currently selected area model to overwrite the current file.

9. Select **Save**. The new area model saves and the file name displays on the FactoryTalk Batch Equipment Editor title bar.

**Tip:** If the file name is anything other than the area model currently selected in the **Server Options** dialog box, change the **Server Options** settings before the new area model is used.

**See also**

- [Import enumerations](#)
- [Import system phases](#)

**Import enumerations**

Create and maintain these system enumeration sets in FactoryTalk Batch Equipment Editor:

- **PHASE_FAILURES**
- **YES_NO**
- **REPORTING_CONTEXTS**
- **MATERIALS** (with an installed Material Server)
- **MATERIAL_CLASSES** (with an installed Material Server)
- **CONTAINERS** (with an installed Material Server)

With the exception of **PHASE_FAILURES** and **REPORTING_CONTEXTS**, the data type or the members of these system-maintained enumeration sets cannot change. The XML file must include these system enumeration sets for validation against the schema. These system enumeration sets cannot import.

**Tip:** The **PHASE_FAILURES** and **REPORTINGCONTEXTS** system enumeration sets can import along with any user-defined enumerations in these enumeration sets.

**See also**

- [Import an XML area model](#)
- [Import system phases](#)
Import system phase classes

FactoryTalk Batch Equipment Editor creates and maintains these system phase class sets:

- $NULL
- $TIMER

On import of system phase classes, the entire phase class validates and only those that match completely import.

Tip: If any of the system phase classes ($TIMER or $NULL) contain errors, only the first error show.

$NULL phase

The $NULL phase class validates:

- The name of the phase class must be $NULL.
- There must be no configured recipe or report parameters.
- There must be no configured messages.
- There must be no configured control strategies.
- There must be no configured request parameters.
- There must be no configured message partners.
- All other configurable values are not set.

$TIMER phase

The $TIMER phase class validates:

- The name of the phase class must be $TIMER_{X} (where X is the type of Timer and the unit of measure; for example, $TIMER_DN_SECOND or $TIMER_UP_DAY).
- There must be no configured messages.
- There must be no configured control strategies.
- There must be no configured request parameters.
- There must be no configured message partners.

See also

XML area models on page 25
Export an area model to XML

FactoryTalk Batch allows export of an area model to a well-formed XML format. With the exception of PHASE_FAILURES and REPORTING_CONTEXTS, the data type or the members of the system-maintained enumeration sets cannot change. The XML file must include these system enumeration sets for validation against the schema. These system enumeration sets can export but cannot import.

All system phases ($NULL or $TIMER) export. All associated recipe and report parameters and associated data (for $TIMER_X phases) are part of the export (for example, where X is the type of Timer and the unit of measure; $TIMER_DN_SECOND or $TIMER_UP_DAY).

Tip: The PHASE_FAILURES AND REPORTING_CONTEXTS system enumeration sets export along with any user-defined enumerations in these enumeration sets.

Tip: When an area model secures to a FactoryTalk Network Directory Security Authority Identifier, the Export command disables.

To export an area model to XML:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Open the area model to export.


4. Locate and select the file to export the area model or, in File name, type a new file name.

5. In Save as type, select XML Files (*.axml).

6. Select Save to initiate the export process. If the file already exists, confirm to replace the file.

7. Select OK to complete the export process and return to FactoryTalk Batch Equipment Editor.

See also

Import and export area models overview on page 25

Convert XML to binary from the command line overview

Convert XML area models to the proprietary binary format from the command line using the supplied program AreaModelConvert.exe. Only well-formed, schema-conformed, and program-validated XML files convert.
When exporting from a command line, change to the directory containing `AreaModelConvert.exe` and the paths cannot contain any spaces. UNC names are also supported, as demonstrated in this syntax example:

The syntax for the command line is:

```
C:AreaModelConvert.EXE InputFile OutputFile LogFile
```

Where:

- **InputFile** specifies the path for the input XML filename.
- **OutputFile** specifies the path for output binary filename.
- **LogFile** specifies the path for log file.

Example:

```
C:AreaModelConvert.EXE
\Batchctl\SampleDemo1\recipes\ice_cream1.axml
\Batchctl\SampleDemo1\recipes\ice_cream1.cfg
\Batchctl\SampleDemo1\logs\convert.log
```

The command line validates the information in the XML file and overwrites the existing file specified by **OutputFile**. Error messages append to the log file.

---

**Important:** The command line program will not validate material-specific information, which includes:

- Running status of the Material Server
- Existence of the referenced materials
- Existence of the referenced material classes
- Existence of the referenced containers

If the command line arguments are valid, the program returns a **0** (zero), indicating a successful import. If the command line arguments are invalid, the command line program returns a **1** (one), indicating a failure. View the return code in the log file to check if the conversion succeeded or failed.

---

See also

[XML area models on page 25](#)
Chapter 4

Enumerations overview

Enumerations are a variable data type consisting of a numeric value (called an ordinal value) and an associated text string. The process-connected device (PCD) and the FactoryTalk Batch Server communicate using the numeric ordinal value. The server then displays the ordinal value associated text string to the operator. Use enumerations to display meaningful text to an operator, instead of a number.

Enumeration sets contain groupings of related enumerations. The FactoryTalk Batch and the PCD create and maintain these default system enumeration sets:

- PHASE_FAILURES
- YES_NO
- MATERIALS (for Material Server-installed system)
- MATERIAL_CLASSES (for Material Server-installed system)
- CONTAINERS (for Material Server-installed system)
- REPORTING_CONTEXTS

The data type and the members of the YES_NO, MATERIALS, MATERIAL_CLASSES, or CONTAINERS system-maintained enumeration sets cannot change.

Create custom enumerations and enumeration sets within an area model.

Use negative ordinal as shown in this example.

The agitator in Premixer B (Unit #2) might use:

<table>
<thead>
<tr>
<th>Enumeration Set Name</th>
<th>MOTORS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Enumeration Name</td>
<td>Ordinal</td>
</tr>
<tr>
<td>Enumeration_1</td>
<td>0</td>
</tr>
<tr>
<td>Enumeration_2</td>
<td>1</td>
</tr>
<tr>
<td>Enumeration_3</td>
<td>-1*</td>
</tr>
</tbody>
</table>

See also

System enumeration sets on page 32
System enumeration sets

There are five default system enumeration sets. The data type or the members of these system-maintained enumeration sets are not allowed to change, except for the Phase_Failures and Reporting_Contexts enumeration sets.

<table>
<thead>
<tr>
<th>System Enumeration Set</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>PHASE_FAILURES</td>
<td>By default, there are no members in this enumeration set. Add enumerations based on the particular implementation requirements.</td>
</tr>
<tr>
<td>Important: PHASE_FAILURES</td>
<td>enumerations cannot have ordinal values greater than 32767 or less than 0 (zero).</td>
</tr>
<tr>
<td>REPORTING_CONTEXTS</td>
<td>This enumeration set makes it possible to add reporting context strings to phase class parameters and reports. The events about the sets may be more easily queried and sorted. Add, remove, and edit enumeration set members to this enumeration set. The default enumeration is NULL, with an ordinal value of 0.</td>
</tr>
<tr>
<td>PHASE_FAILURES and REPORTING_CONTEXTS</td>
<td>are the only system enumeration sets to which enumerations can be added. System enumeration sets display in bold text to differentiate them from custom enumeration sets.</td>
</tr>
<tr>
<td>YES_NO</td>
<td>This is a simple enumeration set that contains two members: NO with an ordinal value of 0 and YES with an ordinal value of 1. This enumeration set cannot be modified.</td>
</tr>
<tr>
<td>MATERIALS (for Material Server-installed system)</td>
<td>Represents the set of materials configured in the FactoryTalk Batch Material Manager material database. The default enumeration is NULL_MATERIAL, with an ordinal value of 0. This value allows assignment of an enumeration to a parameter of type MATERIALS, which delays the actual material assignment until the recipe is built. All other enumeration set members are recreated each time an area model creates, opens, or imports, based on the material data in the material database. The controller ID range for Materials is from 1 through 9999999999.</td>
</tr>
<tr>
<td>CONTAINERS (for Material Server-installed system)</td>
<td>If the FactoryTalk Batch Material Manager material database is unavailable, the MATERIALS and CONTAINERS enumeration sets only contain the default enumerations.</td>
</tr>
<tr>
<td></td>
<td>The controller ID of a material and container is the ordinal value for the enumeration. If modified the controller ID in the material database, the ordinal value of the associated enumeration changes, which could affect the phase logic.</td>
</tr>
</tbody>
</table>
System Enumeration Set | Purpose
--- | ---
MATERIAL_CLASSES (for Material Server-installed system) | Represents the set of material classes configured in the FactoryTalk Batch Material Manager material database. The default enumeration is NULL_CLASS, with an ordinal value of 0. All other enumeration set members are recreated each time an area model is created, opened, or imported, based on the material class data in the material database. The MATERIAL_CLASSES enumeration set members created from the Material Server have a one-to-one relation with the material classes in the Material Server. The name of the enumeration member is the material class name and the ordinal of the enumeration member is the controller ID of that material class. The Material Class controller ID range is from 1 through 9999999999.

See also

Enumerations overview on page 31

Create enumeration sets

Use these instructions to create enumeration sets.

To create enumeration sets:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select Edit > Enumeration Sets.


4. Type the new enumeration set name. Select OK to return to the Create Enumeration Sets and Enumerations dialog box.
The new enumeration set name displays in the **Enumeration Sets** column.

**Tip:** If members are not added to a new enumeration set, the set will not display as an option in the **Enum/E.U.** list when updating phase class parameters and reports.

See also

[Add enumeration members](#) on page 34

### Add enumeration members

Use these instructions to add enumeration members.

**To add enumeration members:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. Select **Edit > Enumeration Sets**.

3. Select the enumeration set.

4. Under **Enumerations**, select **New**.

   **Tip:** Members to the system enumeration sets cannot be added: YES_NO, MATERIALS, MATERIAL_CLASSES, and CONTAINERS.

   **Important:** PHASE_FAILURES enumerations only support positive ordinals from 0 through 32767.

5. Type the enumeration name and the ordinal value, or accept the default ordinal value.

6. Select **OK** to return to the **Create Enumeration Sets and Enumerations** dialog box. The new enumeration name displays under **Enumerations**.

See also

[Create enumeration sets](#) on page 33

[Edit enumerations and enumeration sets](#) on page 34
Edit enumerations and enumeration sets

Use these instructions to edit enumerations and enumeration sets.

To edit enumerations and enumeration sets:

Tip: The YES_NO, MATERIALS, MATERIAL_CLASSES, or CONTAINERS system enumeration sets cannot be renamed or edited. PHASE_FAILURES and REPORTING_CONTEXTS are the only system enumeration sets to which enumerations can be added. System enumeration sets display in bold text to differentiate them from custom enumeration sets.

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select Edit > Enumeration Sets to open the Create Enumeration Sets and Enumerations dialog box.

3. Select the enumeration set or enumeration to edit.

4. Select the corresponding Edit button (Sets or Enumerations).

5. If editing an enumeration, type the new enumeration name.

   ![Edit Enumeration in [MDIM]](image)

   Important: PHASE_FAILURES enumerations support positive ordinals from 0 to 32767 only.

6. If editing an unassigned enumeration set, the Edit Enumeration Set Name dialog box displays. Type the new enumeration set name and select OK to return to the Create Enumeration Sets and Enumerations dialog box.
If editing an assigned enumeration set, a warning appears with a list of the items with the enumeration assigned.

<table>
<thead>
<tr>
<th>Enumeration Set Modification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modifying enumeration set FLAVORING.</td>
</tr>
<tr>
<td>The listed parameters in the corresponding phases will be changed:</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>ADD_FLAVOR</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>The listed report parameters in the corresponding phases will be changed:</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>FLAVOR, SEED</td>
</tr>
<tr>
<td></td>
</tr>
<tr>
<td>Do you want to continue?</td>
</tr>
<tr>
<td>Yes</td>
</tr>
</tbody>
</table>

7. Select Yes to edit the enumeration set name. The items listed in this dialog box modify to include the new enumeration set name. Or, select No to cancel the modification.

8. Select OK to return to the Create Enumeration Sets and Enumerations dialog box.

See also

Delete enumerations and enumeration sets on page 36

Delete enumerations and enumeration sets

Use these instructions to delete enumerations and enumeration sets.

Tip: The system enumeration sets or members (enumerations) of the system enumeration sets cannot be deleted: YES_NO, MATERIALS, MATERIAL_CLASSES, and CONTAINERS.

To delete enumerations and enumeration sets:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select Edit > Enumeration Sets to open the Create Enumeration Sets and Enumerations dialog box.

3. Select the enumeration set or enumeration to delete.

4. Select Delete.
• If deleting an enumeration, continue to step 5.
• If deleting an enumeration set, skip to step 6.

5. To delete an enumeration:

• If the enumeration is not the default for a phase class parameter, a warning displays verifying deletion of the enumeration. Select Yes to delete the enumeration. Select No to cancel the deletion.
• If the enumeration is the default for a phase class parameter and Yes is selected to delete the enumeration, a message displays stating that the enumeration cannot be deleted and gives the phase class parameter name with this enumeration as the default value. First, modify the phase class parameter to remove the enumeration as the default value. Select OK to close the warning dialog box and return to the Create Enumeration Sets and Enumerations dialog box.

6. To delete an enumeration set:

   **Tip:** Items that were using the deleted enumeration set are defaulted to Type = Integer and Eng.Units = Eng.Units.

• A warning displays verifying deletion of the enumeration set. Select Yes to delete the enumeration. Select No to cancel the deletion.
• A warning displays with a list of assigned items.
  a. Select Yes to delete the enumeration set. A message displays the number of tags, parameters, and reports that change from deletion. Select OK to close the message dialog box and return to the Create Enumeration Sets and Enumerations dialog box.
  b. Select No to cancel the deletion.
  c. Select Close to return to the Design View area.

See also

Edit enumerations and enumeration sets on page 34
Chapter 5

Parameters and reports with FactoryTalk Batch

The FactoryTalk Batch Server uses parameters to send values to the process-connected device (PCD). Reports return values from the PCD to the FactoryTalk Batch Server. Use FactoryTalk Batch Equipment Editor to configure parameters and reports for the phase. The phase instance inherits the parameters and reports of the phase class from which it was created.

See also

Parameters and reports in recipes on page 39

How parameters and reports work with phase tags on page 40

Batch parameters and reports on page 42

Material parameters on page 44

Scale parameters on page 46

Parameters and reports in recipes

When adding a phase class to an operation-level recipe using FactoryTalk Batch Recipe Editor, the phase class parameters are formula values, and the phase class reports are report values. When defining the formula values, the recipe author has these options:

- Accept the default formula value (this is the corresponding phase class parameters default value).
- Assign a new value.
- Defer the formula value assignment to the operator or to a recipe formula parameter - a parameter associated solely with the recipe.

Using parameters can minimize the number of phase classes or recipes needed.

See also

Parameters warm restart support on page 218

Parameters example on page 40
If a plant has a unit that mixes several different products, define a phase class to indicate the agitator’s speed in the mixer unit. Either define a different phase class for each of the required speeds, or define one phase class and configure a parameter for the agitator speed. When using the phase class in a recipe, assign a value immediately or later. By deferring the value assignment, one operation is created to use in multiple recipes.

See also

Parameters and reports in recipes on page 39

A phase, which is an instance of a phase class in an area model, uses tags to communicate with the engineered logic in the process-connected device (PCD). Each tag in the phase maps to a specific memory address or tag in the PCD where data associated with the phase is stored during the recipe’s execution. The FactoryTalk Batch Server and the PCD use the tags to exchange information regarding the phase.

Tip: The configuration for RSLinx Enterprise FactoryTalk Live Data tags is identical to that of OPC tags.

Every OPC phase has a minimum of ten standard tags. When configuring the phase class from which the OPC phase is created, define the number of tags required for parameter and report communications, as well as additional request tags.

Tip: For PhaseManager phases assigned to a Logix5000 CIP data server, it is not necessary to define the number of parameter, report, and request tags. The correct number of tags automatically configure when the area model synchronizes with the Logix Designer project.

See also

Additional request, parameter, and report tags for phases on page 40

Phase tags on page 42

A Request tag is one of the ten standard tags associated with every OPC phase. The process-connected device (PCD) uses the Request tag to request service from the FactoryTalk Batch Server. Sometimes, the PCD needs to communicate more information to the FactoryTalk Batch Server than can be accommodated by the single standard Request tag. For this purpose, specify on the phase class, the number of additional request tags needed for an OPC phase.
Tip: For PhaseManager phases assigned to a Logix5000 CIP data server, do not configure request tags. The tags automatically create for Logix5000 CIP phases when the area model synchronizes with the Logix Designer project.

It is possible to configure a different number of phase class parameters and reports than parameter and report tags for an OPC phase. Some OPC phase logic programs may not want the number of parameters to match the number of parameter tags, or the number of reports to match the number of report tags.

To have the number of parameters and reports on the phase class always equal the number of parameter tags and report tags on the associated phases, enable Parameter/Tag Locking on the phase class. If Parameter/Tag Locking is disabled, specify on the phase class the number of parameter and report tags that exist on the associated OPC phases. This number may be different from the number of defined phase class parameters and reports for OPC phases.

Tip: For Logix5000 CIP phases, Parameter/Tag Locking always enables, regardless of the setting selected. When the area model synchronizes with the Logix Designer project file, FactoryTalk Batch Equipment Editor creates the same number of parameter tags as parameters that exist on the phase class, and the same number of report tags as reports that exist on the phase class.

In summary, when creating the phase, specify the following:

- The number of parameters and reports associated with the phase class.
- The number of tags in the phase associated with those parameters, reports, and additional requests. Either enable Parameter/Tag Locking, or specify the actual number of tags.

Tip: The configuration for RSLinx Enterprise FactoryTalk Live Data tags is identical to OPC tags.

When creating the OPC phase instance (phase assigned an OPC data server), specify:

- The memory addresses or tags in the PCD associated with each of the ten standard tags.
- The memory addresses or tags in the PCD associated with any configured parameters, reports, and additional request tags.
Tip: If the number of phase tags does not equal the number of parameters for the specified phase class (the number of phase tags should be equal to the highest parameter ID number), the Upload/Download functionality does not operate correctly. Additionally, to correctly pass parameter values, the phase logic must use request data parameters for specifying how FactoryTalk Batch uploads the parameters.

See also

Phase class overview on page 151
Phase tags on page 42

Phase tags

Review the configuration of the process-connected device (PCD) to verify that the number of tags configured for the OPC phase corresponds to an equal number of tags in the PCD. The parameter and report data types defined in FactoryTalk Batch Equipment Editor must be compatible with the corresponding tag data types in the PCD.

Tip: The configuration for RSLinx Enterprise FactoryTalk Live Data tags is identical to that of OPC tags.

This table shows each parameter and report data type that can be defined in FactoryTalk Batch Equipment Editor and its corresponding data types that can be stored in PCD memory:

<table>
<thead>
<tr>
<th>FactoryTalk Batch Equipment Editor Data Type</th>
<th>Compatible PCD Data Types</th>
</tr>
</thead>
<tbody>
<tr>
<td>Real</td>
<td>Real</td>
</tr>
<tr>
<td>Integer</td>
<td>Real, Integer</td>
</tr>
<tr>
<td>Enumeration</td>
<td>Real, Integer</td>
</tr>
<tr>
<td>String</td>
<td>String</td>
</tr>
</tbody>
</table>

See also

Additional request, parameter, and report tags for phases on page 40
Batch parameters and reports

Parameters pass values from the FactoryTalk Batch Server to the process-connected device (PCD). The PCD uses reports for returning values to the server.

**Important:** Do not use these words or phrases as parameter names or report names:
- all
- length
- recipepath
- outerhtml
- finalinstructions
- stepindex
- submitstring
- submit

Using these phrases as parameter or report names can cause errors during manual phase control.

See also
- Phase parameters on page 43
- Phase reports on page 43

Phase class parameters

Phase class parameters pass from the FactoryTalk Batch Server to the phase logic during the recipe phase’s execution. Phase class parameters usually correspond to phase tags, which are addressed to actual, configured parameters in the process-connected device (PCD).

When configuring the phase class, define phase class parameters. When configuring the OPC phase, define the corresponding parameter tags, including the address or tag in the PCD to which it is addressed.

**Tip:** For PhaseManager phases assigned to a Logix5000 CIP data server, the parameter tags automatically create when the area model synchronizes with the Logix Designer project. Parameter tags for Logix5000 CIP phases do not have to be defined.

See also
- Phase class reports on page 43

Phase class reports

Phase class reports values pass from the process-connected device (PCD) to the FactoryTalk Batch Server. Report values contain data that indicates how the phase functioned. They are in the event journal file (.evt) for the running batch and can be in FactoryTalk Batch reports. Phase class reports usually correspond to phase tags addressed to configured parameters in the PCD.
Tip: The configuration for RSLinx Enterprise FactoryTalk Live Data report tags is identical to that of OPC tags.

When configuring the phase class, define phase class reports for the phase. When configuring the OPC phase, define the corresponding report tags, including the address or tag in the PCD to which it is addressed.

Tip: For PhaseManager phases assigned to a Logix5000 CIP data server, the report tags automatically create when the area model synchronizes with the Logix Designer project. There is no need to define report tags for Logix5000 CIP phases.

See also

Phase class parameters on page 43

Material parameters

If using FactoryTalk Batch Material Manager with FactoryTalk Batch, create material-enabled phase classes. Material-enabled phase classes use system parameters that specify the material and amount of material to use within a recipe.

The default parameters associated with a material-enabled phase class are MATERIAL and AMOUNT. The default report parameters associated with each material-enabled phase class are ACTUAL_AMOUNT and FEED_COMPLETE. Each material-enabled phase class can be assigned optional material parameters: CONTAINER, LOT, LABEL, and MATERIAL_CLASS. The values of the MATERIAL, LOT, and LABEL parameters are combined to create the material specification.

The material specification determines the container from which a material is drawn or into which a material distributes when a batch runs. The material specification always contains a material name. The specification may include a lot name, a label, or both based on the container type.

See also

Types of containers on page 44

Default material parameters on page 45

Types of containers

FactoryTalk Batch Material Manager uses three types of containers: Composite, Plug-Flow and Pallet. All three containers require the material specification to contain at least a material name. Composite and plug-flow containers support a material specification that includes a lot name. Pallet containers support a material specification that includes a lot name, or a label, or both.

If a material specification is only a material name, then any three of the container types are valid. However, if a material specification includes a material name, lot, and label, then only the pallet container type is valid.
Default material parameters

See also

Default material parameters on page 45

Default material parameters:

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>MATERIAL</td>
<td>Generally configured with the default value of NULL_MATERIAL; the actual material is assigned in the recipe. Assign a material to the parameter, but then the phase class can only be used to feed that specific material. The combination of the MATERIAL parameter, the unit for the phase in which the phase class is running, and the high and low values for the AMOUNT parameter determines if the phase class is going to be an addition of material, distribution, or both.</td>
</tr>
<tr>
<td>FEED_COMPLETE</td>
<td>Reports that a material addition or distribution is complete. The ACTUAL_AMOUNT report parameter records the actual quantity of material produced or consumed.</td>
</tr>
<tr>
<td>CONTAINER</td>
<td>Used to hold the name of the container to which a step is bound when a batch runs. It is created with a default value of NULL_CONTAINER. The CONTAINER parameter cannot be assigned a value in the area model or in a recipe. Containers are considered to be equipment outside of the area model. Each the FactoryTalk Batch Server or the operator determines the appropriate container to which the step is bound. At that time, the value of the bound container name is assigned to the CONTAINER parameter. Other than the default value, the parameter only contains a value while the step is bound during runtime.</td>
</tr>
<tr>
<td>MATERIAL_CLASS</td>
<td>Represents a set of material classes configured in the material database. It is created with a default value of NULL_CLASS. The only attributes of MATERIAL_CLASS that can be modified are Default and Download on Start, depending on whether or not control strategies are enabled.</td>
</tr>
<tr>
<td>LOT</td>
<td>Used to refine the list of materials to which a phase class can be bound. The lot specifies a discrete quantity of a given material. When this parameter is blank at run time, the FactoryTalk Batch Server binds to a container without regard to the lot.</td>
</tr>
<tr>
<td>LABEL</td>
<td>Represents a sublot of a material, which is a discrete quantity of a given material's lot distributed to a specific container. In material addition phase classes, the label is used to refine the material list to which a phase class can be bound. In material distribution phase classes, the label is applied to the sublot of material distributed into inventory.</td>
</tr>
</tbody>
</table>
Tip: The values of the optional material parameters (except for the MATERIAL_CLASS parameter) within FactoryTalk Batch Equipment Editor cannot be edited.

See also

Types of containers on page 44

Scale parameters

Select the Scale check box to set a parameter to automatically adjust proportionally when a batch containing a phase based on this phase class scaled at runtime.

For example, if a batch containing a phase based on this phase class runs at a scale of 200%, the ADD_AMOUNT value automatically increments by 200%. The ADD_AMOUNT adjusts to 100, since the default is 50. Likewise, if the batch runs at a scale of 50%, the ADD_AMOUNT adjusts to 25.

See also

Parameters and reports with FactoryTalk Batch on page 39

Verification method for parameter deviations

A parameter deviation occurs when the parameter or report value is outside the acceptable range defined by the designer of the area model. A verification policy defines the set of acceptable limits for each parameter and the corresponding action to take when the parameter or report value exceeds or falls short of the limits. The Verification Method is the policy portion that defines the set of acceptable limits for a parameter or report. When adding a parameter or report to a phase class, either accept the default Verification Method, which is No Limits, or select a different Verification Method.

Tip: To require electronic signatures as part of a Verification Policy, set up Signature Templates before configuring the Verification Policy.

See also

Verification policies on page 169

Signature template on page 57

Control strategies overview

Create parameters and reports applicable to a specific phase class in FactoryTalk Batch Equipment Editor. The parameters and reports are directly associated with the phase class. All the parameters and reports assume to be applicable to each recipe that includes the phase class. However, some phase classes may assume
different roles within a single recipe or different roles across different recipes. In this situation, only a subset of the parameters and reports associated with the phase class may be required, resulting in unnecessary phase I/O when the recipe executes. This is where control strategies are helpful.

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.

Create multiple control strategies for a phase class, each with its own parameters and reports. Within a phase class associate parameters and reports with a single or multiple control strategies. If a parameter or report is associated with more than one control strategy, the value range, engineering units, and the engineering units of the report can be defined independently for each control strategy with which the parameter or report is associated.

Tip: In FactoryTalk Batch Recipe Editor, when viewing recipe phase class enabled control strategies, only the parameter values with specified control strategy display. Associating parameters with specific control strategies eliminates the need for the recipe author or operator to determine which phase class parameters, and report parameters are applicable when creating, viewing, and executing a specific recipe.

See also

- Control strategies example on page 47
- Implement control strategies on page 48

Control strategies example

This table represents the configured parameters, reports, and control strategies for a single phase class. The table illustrates the relationship between parameters, reports, and control strategies. An X placed below a control strategy indicates that the parameter or report is associated with that control strategy.

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>CONTROL_STRATEGY_1</th>
<th>CONTROL_STRATEGY_2</th>
<th>Strategy 1 Value</th>
<th>Strategy 2 Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>PARAMETER_A</td>
<td>X</td>
<td></td>
<td>150</td>
<td></td>
</tr>
<tr>
<td>PARAMETER_B</td>
<td>X</td>
<td>X</td>
<td>200</td>
<td>275</td>
</tr>
<tr>
<td>PARAMETER_C</td>
<td></td>
<td>X</td>
<td></td>
<td>70</td>
</tr>
<tr>
<td>REPORT_A</td>
<td>X</td>
<td>X</td>
<td>125</td>
<td>125</td>
</tr>
</tbody>
</table>

For example, a phase class contains three phase class parameters, PARAMETER_A, PARAMETER_B, and PARAMETER_C, and a report
parameter named REPORT_A. As indicated above, a parameter can associate with only one control strategy as is the case with PARAMETER_A and PARAMETER_C. Or, a parameter associate with more than one control strategy as is the case with PARAMETER_B and REPORT_A. Parameters associated with more than one control strategy can have separately defined values for each control strategy as indicated by PARAMETER_B.

See also

Implement control strategies on page 48

Implement control strategies

Configure the phase class in an area model to enable or disable the control strategy functionality. For phase classes that have the control strategy functionality enabled, configure one or more strategies for that phase class. Assign phase class parameters and reports to each control strategy, as well as configure their ranges, values, and engineering units.

When creating master recipes using FactoryTalk Batch Recipe Editor, specify which control strategy to use for each instance of the phase class within the recipe. When the recipe executes, the phase class uses only the parameters associated with the control strategy specified during recipe creation. Likewise, when the executed recipe is viewed in the FactoryTalk Batch View or other client program, a control strategy-enabled phase class only displays the specified control strategy and any parameters associated with that control strategy. However, the default values of parameters associated with control strategies can be changed. Additionally, if a phase class is executed using manual phase control, select which control strategy to use before running the phase class.

Tip: Configure the tags in the phase and the phase logic in the process-connected device(s) (PCD) to support control strategy use. When a PCD requests to download phase class parameters or upload phase class reports, only the appropriate phase class parameters and reports are loaded, based on the control strategy. Configure and map the phase class and report parameters to the tags in the PCD(s).

See also

Configure control strategies on page 166

Parameter and report subsets overview

Use the download parameter subsets to define subsets of phase class parameters for Download (DL) on Start and Download (DL) on Transfer of Control (TOC). These subsets provide for a more efficient batch execution. Also, the upload report subsets allows definition of a subset of the phase class reports for Upload (UL) on Terminal State or Upload (UL) on Transfer of Control (TOC). These subsets provide for a more efficient process.
Tip: Transfer of control means that two instances of the same phase exist in a recipe and are separated by a recipe transition condition containing something other than `<Phase Name>.State = Complete`. In this case, the running logic does not terminate when the transition condition becomes true.

Assign parameters to none, one, or both of the upload/download sets. By assigning a parameter or report to a subset, that parameter or report, along with the other members of the subset, are downloaded (parameters) or uploaded (reports) when the appropriate phase logic request is issued.

Tip: The 145 (Request Failed) command value can be issued by the FactoryTalk Batch Server when processing a phase logic request experiences an error. This enhanced error handling phase logic protocol is supported by making a request where the value is 10,000 greater than the current standard requests.

For example, to upload a report parameter using the enhanced phase logic request, add 10,000 to the standard request code of 2200 to get the enhanced request code of 12200.

Important: When using Auto Upload with the FactoryTalk Batch Server configured for Warm or Warm All Restart, the Server may upload addition material information twice when a phase is transitioned to External during the upload. Check the journal file. If the upload has occurred twice, manually adjust the amount in the material database. The setpoint of the amount parameter is recalculated with each warm restart as well. This number may need to be manually updated.

See also

- Phase class with different subsets example on page 49
- Download parameter requests on page 50
- Upload report requests on page 51
- Configure phase class parameters on page 155
- Configure phase class reports on page 160
This table is an example of a phase class and how its parameters and reports can be assigned to the different subsets:

<table>
<thead>
<tr>
<th>AGITATE_1</th>
<th>DL on Start</th>
<th>DL on TOC</th>
<th>Reports</th>
<th>UL on Terminal State</th>
<th>UL on TOC</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPEED</td>
<td>X</td>
<td>X</td>
<td>AGITATE_TIME</td>
<td>X</td>
<td>X</td>
</tr>
<tr>
<td>TIME</td>
<td>X</td>
<td>X</td>
<td>AVG_TEMP</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>LEVEL</td>
<td>X</td>
<td>X</td>
<td>AVG_SPEED</td>
<td>X</td>
<td></td>
</tr>
<tr>
<td>TEMP</td>
<td>X</td>
<td></td>
<td>AMT_DUMPED</td>
<td>X</td>
<td></td>
</tr>
</tbody>
</table>

See also

Parameter and report subsets overview on page 48

Download parameter requests on page 50

Upload report requests on page 51

Download parameter subsets allows definition of subsets of parameters downloaded to the process-connected device (PCD) when the appropriate phase logic requests are sent to the FactoryTalk Batch Server.

- If DL on Start is selected, the FactoryTalk Batch Server downloads these parameters to the PCD before the FactoryTalk Batch Server issues a START command to the phase. Once the download is verified, the FactoryTalk Batch Server starts the phase. The phase logic can explicitly request this or the phase can be configured for the FactoryTalk Batch Server to automatically download these parameters before it issues a START command.

- If DL on TOC is selected, the parameters may be downloaded to the PCD when a transfer of control situation occurs. Transfer of control means that two instances of the same phase exist in a recipe and are separated by a recipe transition condition that contains something other than <Phase Name>.State = Complete. In this case, the phase’s running logic does not terminate when the transition condition becomes true. When the transition condition becomes true, the phase logic must explicitly request the Server to download the transfer of control subset of parameters to the PCD.

To download only the parameters assigned to the DL on Start subset, use these download requests:
Parameters and reports with FactoryTalk Batch

Chapter 5

Parameters and reports with FactoryTalk Batch Equipment Editor User Guide - November 2018

51

To download only the parameters assigned to the DL on TOC subset, use these download requests:

<table>
<thead>
<tr>
<th>(OPC or FTD) phases</th>
<th>Logix5000 CIP phases</th>
</tr>
</thead>
<tbody>
<tr>
<td>1501 or 11501 request</td>
<td>PXRQ instruction, External request: Download Input Parameters Subset DINT[0] - 1</td>
</tr>
</tbody>
</table>

When one of these requests is issued, each parameter is downloaded to the memory address or tag in the PCD that corresponds to the parameter tag in the phase.

See also

Parameter and report subsets overview on page 48
Upload report requests on page 51
Configure phase class parameters on page 155
Configure phase class reports on page 160

Upload report requests

If UL on Terminal State is selected, the FactoryTalk Batch Server uploads this subset of reports from the process-connected device (PCD) after the phase logic transitions to a terminal state of COMPLETE, STOPPED, or ABORTED. The phase logic can explicitly request the server to upload these reports or the phase can be configured for the server to automatically upload these reports after the phase logic transitions to a terminal state.

If UL on TOC is selected, the reports are uploaded to the PCD when a transfer of control situation occurs. In this case, the phase logic must explicitly request the FactoryTalk Batch Server to upload the transfer of control report subset from the PCD.

To upload only the reports assigned to the UL on Terminal State subsets:

<table>
<thead>
<tr>
<th>(OPC or FTD) phases</th>
<th>Logix5000 CIP phases</th>
</tr>
</thead>
<tbody>
<tr>
<td>2501 or 12501 request</td>
<td>PXRQ instruction, External request: Upload Output Parameters Subset DINT[0] - 1</td>
</tr>
</tbody>
</table>

To upload only the reports assigned to the UL on TOC subset:
Automatic Upload/Download decreases batch duration time as well as the amount of phase logic programming required. Automatic Upload/Download works with Download Parameter Subsets and Upload Report Subsets. Automatic Upload/Download allows the FactoryTalk Batch Server to automatically download parameters grouped into the download on START subset or to upload reports grouped into the upload on terminal state subset. Phase logic can be developed to make these requests; however, it is not required.

Use Automatic Upload/Download to configure the FactoryTalk Batch Server to:

- Automatically download the parameters grouped into the download on start subset when all the conditions required to start the phase are true.
- Automatically upload the appropriate report values grouped into the upload on terminal state subset when the phase transitions to a terminal state of COMPLETE, STOPPED, or ABORTED.

Automatic Upload/Download eliminates the need to send the download on start or upload on Terminal State request from the phase logic to the FactoryTalk Batch Server.

Tip: Downloading parameters on Transfer of Control and uploading reports on transfer of control is not automatic and still requires a phase logic request.

The Automatic Download Parameter Upon Start and UL On Terminal State are configured for each individual phase (as long as the parameters and reports of the phase class from which the phase was created are grouped into the appropriate subsets). The Automatic Download parameter Upon Start and UL on Terminal State may also be configured independently of each other on a phase. For UL on Terminal State, specify any or all terminal states for which the automatic function is applicable: COMPLETE, STOPPED, or ABORTED.

### Table: Parameters and reports with FactoryTalk Batch

<table>
<thead>
<tr>
<th>(OPC or FTD) phases</th>
<th>Logix5000 CIP phases</th>
</tr>
</thead>
<tbody>
<tr>
<td>2501 or 12501 request</td>
<td>PXRQ instruction, External request: Upload Output Parameters Subset DINT[0] - 2</td>
</tr>
</tbody>
</table>

See also

Parameter and report subsets overview on page 48
Tip: The 145 (Request Failed) command value can be issued by the FactoryTalk Batch Server when processing a phase logic request experiences an error. This enhanced error handling phase logic protocol is supported by making a request where the value is 10,000 greater than the current standard requests. For example, to upload a report parameter using the enhanced phase logic request, add 10,000 to the standard request code of 2200 to get the enhanced request code of 12200.

See also

- Parameter and report subsets overview on page 48
- Control strategies with upload/download parameter subsets on page 53
- Configure the general data for a phase on page 183

Control strategies with upload/download parameter subsets

If assigned parameters or reports to upload/download parameter subsets, and enabled control strategies, only those parameters common to all the involved subsets and selected control strategy upload or download.

In this example, the phase class has a number of reports and parameters. The control strategies are enabled and the CS1 control strategy is the default. When a phase logic request:

- Is issued for Download on Start, only the SPEED, TIME, LEVEL, and TEMP parameters are included.
- Is issued for Download on TOC, only the SPEED and TIME parameters are included.
- For Upload on Terminal State is issued, only the AVG_SPEED, AGITATE_TIME, AVG_TEMP, and AMT_DUMPED reports are included.
- For Upload on TOC is issued, only the AVG_SPEED and AGITATE_TIME reports are included.

<table>
<thead>
<tr>
<th>AGITATE_1 (Control Strategies enabled; CS1 is the default control strategy)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameters</td>
</tr>
<tr>
<td>-----------</td>
</tr>
<tr>
<td>SPEED</td>
</tr>
<tr>
<td>TIME</td>
</tr>
<tr>
<td>LEVEL</td>
</tr>
<tr>
<td>TEMP</td>
</tr>
</tbody>
</table>
See also

Parameters and reports with FactoryTalk Batch on page 39
Electronic signatures

An electronic signature is an electronic representation of a signature and its associated data, including:

- Signoff meanings (such as review, approval, responsibility, or authorship)
- Signer comments
- Signer security requirements
- Date and time of signature

An electronic signature in the FactoryTalk Batch system can consist of signoffs by one, two, or three FactoryTalk users or groups.

See also

- Types of electronic signatures on page 55
- How electronic signatures work on page 56
- Signature template on page 57

Types of electronic signatures

Signatures can be configured for verification of these runtime FactoryTalk Batch events:

- Report parameters for automated and manual (eProcedure) phases.
- Changes to the recipe parameter values made by operators from the FactoryTalk Batch View or ActiveX controls.
- Completion of eProcedure steps.
- Batch or phase commands issued by operators from FactoryTalk Batch View or ActiveX controls.
- General usage that can be configured to display at any time when executing an automated or manual phase.

Electronic signatures are stored in the FactoryTalk Batch Event Journal with their corresponding electronic records. These electronic signatures include all the components necessary for compliance with 21 CFR Part 11.
How electronic signatures work

See also

Electronic signatures on page 55

The steps involved in configuring and executing electronic signatures span the FactoryTalk Batch product suite.

- The FactoryTalk Batch system administrator sets up electronic signature and signoff security.

  Important: Changing the FactoryTalk Directory Server after configuring electronic signatures using FactoryTalk Security is not supported. Doing so may cause Signature Signoffs to fail.

- In FactoryTalk Batch Equipment Editor, the FactoryTalk Event Archiver event filters for the signature events to capture for use in FactoryTalk Event Archiver with Reporting Services reports, or other reports.

- Create signature templates for the area model. To obtain electronic signatures for recipe parameters and report parameters of specific phase classes, configure the verification method and signature template to be used when the phase class is created or edited. To download all the limits information to the phase logic, map parameter tags and report tags to the appropriate limit tags on the phases. To require signature verification for the execution of specific commands, set up command verification policies in FactoryTalk Batch Equipment Editor using the signature templates that were created.

When the master recipe builds, the recipe parameter limits that trigger the verification policy default from the parameter limits specified on the equipment phase. Change these values at the recipe level in FactoryTalk Batch Recipe Editor. If configured a phase class to obtain a signature for deviation of report limits, specify those limits on the report for the phase class in FactoryTalk Batch Recipe Editor.

When a batch runs and parameter or report deviations occur requiring signature verification, or the operator issues commands requiring signature verification, a signature is added to the signature list in FactoryTalk Batch View, the Electronic Signature ActiveX control, or the eProcedure client, prompting the appropriate users to enter signatures.

All signatures record as Signature events in the FactoryTalk Batch Event Journal viewer in FactoryTalk Batch View.

Once signature event data generates, create FactoryTalk Event Archiver reports, or other reports that capture electronic signature data.
This flow chart shows the sequence of tasks in the setup and configuration, execution, viewing, and reporting of electronic signatures:

See also

Electronic signatures on page 55

Signature template on page 57

Signature template

Before setting up phase classes and recipes to require electronic signatures, establish a set of signature templates. Each 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. Create as many templates as needed to distinguish the different signature types the business processes requires.

The process of creating a signature template consists of three main procedures:

- Add a signature template
- Configure signature template signoffs
- Assign security permissions for a signoff

See also

- Edit signature templates dialog box on page 58
- Add a signature template on page 59
- Configure signature template signoffs on page 61
- Delete a signature template on page 66
- Modify a signature template on page 66

The **Edit Signature Templates** dialog box contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>General tab</strong></td>
<td></td>
</tr>
</tbody>
</table>
| Template Name       | Displays the name of the template selected in the **Signature Templates** list. If there are no templates in the template list, this is disabled.  
                      |  
                      | - Template names must be unique, and can range from 1 through 50 characters in length.  
                      | - The name must begin with an alpha character (a-z or A-Z) and may be followed with any combination of alphanumeric characters or under score.  
                      | - **Template Name** is not case sensitive.  
                      | - Spaces are not valid.                                                   |
| Template Index      | The unique identification number of the template selected in the **Signature Templates** list. This value must be a positive integer from 1 to 2,147,483,647. |
| Signoffs Required   | The number of signatures required in the verification policy this template represents. |
| Last Signoff        | A number indicating which signoff must be completed last, after other signatures have been entered. If **None** is selected, the signatures may be entered in any order. |
| New Template        | Opens the **Create Signature Template** dialog box where new templates can be added to the **Signature Templates** list. |
### Name
Delete Template

#### Purpose
Removes the selected Signature Template from the list. If the selected template is used by the verification policy of any commands, phase class parameters, or reports, the template cannot be deleted.

### Signoffs tab

<table>
<thead>
<tr>
<th>Template Name</th>
<th>Lists the templates that have been added on the General tab. Select the template to see its signoff information or to configure signoffs.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Signoff</td>
<td>Each number in the list represents a required signature for the selected template. These values are determined by the value selected in the General tab Signoffs Required box. Select the number of the Signoffs. Consider which value was selected in the General tab Last Signoff box.</td>
</tr>
<tr>
<td>Meaning</td>
<td>(optional) Type a short phrase describing the significance of the selected signoff. For example, Command Authorized by Signer, Done By Signer, or Checked by Signer. The Meaning displays as part of the signoff event recorded in the FactoryTalk Batch event journal. The Meaning box can range from 0 through 80 characters, and can contain any viewable and printable character. Special control or non-printable characters are not valid.</td>
</tr>
<tr>
<td>Comment</td>
<td>Select whether a comment is Optional, Required, or Not Allowed by the signer at the time the signoff is entered.</td>
</tr>
<tr>
<td>Security Permissions</td>
<td>Displays the users and groups of users that can complete the selected signoff. If this area is blank, no security permissions have been configured yet. Security Permissions must be configured for all signoffs before the Edit Signature Templates dialog box can be closed.</td>
</tr>
<tr>
<td>Add</td>
<td>Opens the Select User or Group dialog box. Select which users and groups have permission to complete the selected signoff.</td>
</tr>
<tr>
<td>Delete</td>
<td>Removes the selected user or group from the Security Permissions box, and removes the user or group permission to complete the selected signoff.</td>
</tr>
</tbody>
</table>

### See also

- [Delete a signature template on page 66](#)

### Add a signature template

Use these instructions to add a signature template.

#### To add a signature template:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Open the area model file.
3. Select Edit > Signature Templates.

4. Select New Template to display the Create Signature Template dialog box.

5. In the Template Name box, enter a unique name to identify the template. The name must begin with an alpha character (a through z or A through Z) and may be followed with any combination of alphanumeric characters or underscores. This box is not case sensitive. Spaces are not accepted as valid input.

   **Tip:** Choose a name that easily identifies the template properties or purpose. The template name is used to identify the signature template when the command verification policies, phase classes, and recipes are being configured.

6. (optional) A unique identification number is defaulted in the Template Index box. If needed, change this number (must be a positive integer).

   **Important:** If a decimal is entered in the Template Index box, the value is automatically rounded up or down to the nearest integer. If the nearest integer is already assigned to another template, a duplicate index value error message is opened.

7. Select OK. The Create Signature Template dialog box closes and the new template is listed in the Signature Templates list in the Edit Signature Templates dialog box.

8. In the Signoffs Required list, select the number of signatures that must be obtained for this signature template.
9. In the **Last Signoff** list, select the number of the signoff that corresponds to the signature that must be completed last (that is, after other signatures have been entered). If the signatures do not have to be completed in any particular order, select **None**.

![Edit Signature Template](image)

10. Configure the signoffs for the signature template.

**See also**

**Configure signature template signoffs** on page 61

*Configure signature template signoffs*

Use these instructions to configure the signature template signoffs.

**Before you begin:**

- Verify the signature template has been added.

**To configure signature template signoffs:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. From the **Edit Signature Templates** dialog box, select the **Signoffs** tab.

3. From **Template Name**, select the template to configure signoff information.

4. From **Signoff**, select the number that represents the signoff to configure.
Tip: This Signoff list corresponds to the Signoffs Required list on the General tab. When completing these steps, keep in mind the number selected for Last Signoff on the General tab.

5. (optional) In Meaning, enter a short phrase that describes the significance of this signoff.

Tip: The Meaning displays as part of the signoff event recorded in the FactoryTalk Batch event journal.

6. From Comment, select whether signer comments are Optional, Required, or Not Allowed.

7. Under Security Permissions, assign privileges to the users and groups authorized to complete this signoff.

When finished assigning security permissions for the signoff, the User and Group names are displayed in the Security Permissions area. (For detailed instructions, see Assign security permissions for a signoff.)
8. Select **Apply**.

9. If creating a template that requires more than one signoff, repeat steps 3 through 7 until all the signoffs have been configured for the template.

   **Tip:** When a signature requires multiple signoffs, each signoff must be made using a unique UserID and password.

10. When finished configuring all the signoffs for the template, either return to the **General** tab to add other signature templates, or select **OK** to close the dialog box.

**See also**

- [Add a signature template](#) on page 59
- [Assign security permissions for a signoff](#) on page 63
- [Remove signoff permissions for a user or group](#) on page 65

---

**Assign security permissions for a signoff**

When configuring signoffs for a signature template, specify which users or groups may complete each signoff.

**Before you begin:**

- Add a signature template.
- On the **Signoffs** tab, enter a **Template Name**, select a **Signoff** number and **Comment** option, and enter a **Meaning**.

**To assign security permissions for a signoff:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. From the **Edit Signature Templates** dialog box, select **Signoffs** tab.
3. From the **Security Permissions** area, select **Add**. The **Select User or Group** dialog box opens.

![Select User or Group dialog box]

4. In the **Select User or Group** dialog box, select **Show all** to view a list of all users and groups. The list displays all FactoryTalk Security users and groups and all Windows-linked users and groups defined in the FactoryTalk Directory Server.

   **Tip:** Changing the FactoryTalk Directory Server after adding users and groups to a signature template is not supported and may cause signature signoffs to fail.

5. Select the appropriate user or group.

![Select User or Group dialog box]

6. Select **OK**.

![Select User or Group dialog box]
Important: Changing the FactoryTalk Directory Server after configuring electronic signatures using FactoryTalk Security is not supported and may cause signature signoffs to fail.

See also

Add a signature template on page 59

Configure signature template signoffs on page 61

Remove signoff permissions for a user or group

Complete these steps to remove authority from an individual or group to complete a particular signoff.

To remove signoff permissions for a user or group:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select Edit > Signature Templates, select the template, and then select the Signoffs tab.

3. Under Security Permissions, select the individual or group.

4. Select Delete.

5. Select OK.
Delete a signature template

A signature template cannot be deleted if it is used in:

- The verification policy for any command.
- The verification policy for phase class parameters or reports.
- The approval process for a recipe.

Before you begin:

- Modify each command verification policy that uses the template to use a different signature template or to not require signature verification.
- Modify the verification policy for phase class parameters or reports to use a different signature template or no signature template, or the parameter or report must be deleted before the template currently in use can be deleted.
- Modify recipe approval step to have a different signature template assigned to its signoff.

To delete a signature template:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Open the area model file.

3. Select Edit > Signature Templates.

4. On the Edit Signature Templates dialog box, from Signature Templates, select the template to delete.

5. Select Delete Template. The template name is removed from the Signature Templates list.

6. Select OK.

See also

- Command verification policies on page 67
- Add a signature template on page 59
- Modify a signature template on page 66
Modify a signature template

The Template Name or Template Index cannot be modified if the signature template is currently used in the verification policy of any command, phase class parameters, or reports, or if assigned to a recipe approval step.

Before you begin:

- Modify each command verification policy that uses the template to use a different signature template or to not require signature verification.
- Modify the verification policy for phase class parameters or reports to use a different signature template or no signature template, or the parameter or report must be deleted before the template currently in use can be deleted.
- Modify recipe approval step to have a different signature template assigned to its signoff.

**Important:** If the number of Signoffs Required for a template is changed, the value of Last Signoff may also be affected. If the change results in the value of Last Signoff being greater than the value of Signoffs Required, the value of Last Signoff automatically changes to None. To enforce a different signature as the last signoff, select a different value in Last Signoff.

For example, the value of Signoffs Required is 3 and Last Signoff is 3. If Signoffs Required is changed to 2, the value of Last Signoff is automatically changed to None.

**To modify a signature template:**

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Open the area model file.

3. Select Edit > Signature Templates.

4. On the Edit Signature Templates dialog box, from Signature Templates, select the template to modify.

5. Select the General tab and modify boxes as needed.

6. Select OK.

See also

Command verification policies on page 67
Command verification policies

Set up command verification policies to obtain signatures for batch or phase commands issued by operators from FactoryTalk Batch View or ActiveX controls.

Before configuring command verification policies, create signature templates that describe the signoff policies to use for the commands.

**Important:** If signature verification for the **Remove** command is enabled, all AUTOREMOVE executes sent to the FactoryTalk Batch Server fail and generate an error message. The AUTOREMOVE execute tells the server to automatically remove a batch when its state reaches COMPLETE.

See also

- [Signature template](#) on page 57
- [Add a signature requirement to commands](#) on page 68
- [Change the verification policy for a command](#) on page 70
- [Remove the signature requirement for a command](#) on page 71

Use these instructions to add electronic signatures to commands.

To add a signature requirement to commands:

1. Select **Start**, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select **Equipment Editor**.

2. Select **Edit > Command Policies**.

![Command Verification Policies](image)
Tip: The ability to secure FactoryTalk eProcedure instruction files does not currently exist in FactoryTalk Security, require an electronic signature on Parameter Change to prevent unauthorized users from executing instructions.

3. Select the check box next to the command that requires an electronic signature. From Select Template, select the signature template that defines the signature policy to use for this command.

4. Select OK. The Select Signature dialog box closes, and the signature template displays in the Signature Template box next to the selected command.

5. Repeat these steps for each additional command that requires an electronic signature.

6. Select OK.

See also

- Edit command verification policies dialog box on page 69
- Change the verification policy for a command on page 70
- Remove the signature requirement for a command on page 71
Edit command verification policies dialog box

The components in the **Edit Command Verification Policies** dialog box include:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Command</td>
<td>This column displays each command that can be configured to require an electronic signature (view only).</td>
</tr>
<tr>
<td>Requires Signature</td>
<td>Opens the <strong>Select Signature Template</strong> dialog box. Select the check box next to each command that should require an electronic signature when it is executed.</td>
</tr>
<tr>
<td>Signature Template</td>
<td>Displays the signature template that defines the verification policy for the command. If the command does not require a signature, this box is disabled. To change the <strong>Signature Template</strong>, select browse and select a different template in the <strong>Select Signature Template</strong> dialog box.</td>
</tr>
</tbody>
</table>

See also

Add a signature requirement to commands on page 68

Change the verification policy for a command

If a command requires a signature and has a signature template specified, follow these instructions to change the verification policy for the command.

**Before you begin:**

- Verify a signature template already exists that describes the new signoff policies to use. If not, create one.

**To change the verification policy for a command:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. Select **Edit > Command Policies**.

3. Select browse next to the signature template to change.
4. On the **Select Signature Template** dialog box, from **Select Template**, select the signature template that defines the signature policy used for this command.

5. Select **OK**. The **Select Signature** dialog box closes. The signature template chosen displays in the **Signature Template** box next to the selected command.

6. Repeat these steps for each additional verification policy to change.

7. Select **OK**.

**See also**

- [Signature template](#) on page 57
- [Command verification policies](#) on page 67

---

**Remove the signature requirement for a command**

To remove the verification policy and signature requirement for a command, complete these steps.

**To remove the signature requirement for a command:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. Select **Edit > Command Policies**.

3. Clear the **Requires Signature** check box next to the appropriate command.

The **Signature Template** box is disabled and the template name appears shaded. This indicates that the signature template is not in use for this command and no signature verification is required.

**Tip:** To re-enable the signature requirement for a command, select the **Requires Signature** check box. The signature template previously used automatically activates.

4. Select **OK**.
Chapter 6  Electronic signatures

See also

Add a signature requirement to commands on page 68

Command verification policies on page 67
Chapter 7

Recipe approval process overview

The recipe approval process validates the development and maintenance of batch recipes. Using signature certification, the recipe approval process safeguards design workflow in a formalized manner. This ensures each recipe validates by authorized personnel before being released to production, or released as a component within a larger recipe.

Configured and enable the recipe approval process in FactoryTalk Batch Equipment Editor, then execute in FactoryTalk Batch Recipe Editor. When the approval process completes, the recipe is added to FactoryTalk Batch View, eProcedure, or the ActiveX controls Recipe List. At that point, production batches can be made from the recipe.

In addition to the primary approval process with up to six approval steps (three optional), a two-step expedited approval process is available. Both primary and expedited processes provide a revert option to allow forward and reverse progress through the recipe approval process.

See also

Recipe approval tasks on page 73
Configure recipe approval steps on page 74
Edit a recipe approval on page 76
Disable recipe approval on page 77

Recipe approval tasks

As a necessary first step, a FactoryTalk Batch system administrator sets up electronic signatures for signoff security.

Typically, an administrator, or a process engineer with full administrative rights, then defines signature templates in FactoryTalk Batch Equipment Editor. Signature templates define which users can sign off on recipe approval steps, either to approve or to revert the step.

Also in FactoryTalk Batch Equipment Editor, a process engineer configures recipe approval steps according to a defined approval process. The primary approval process comprises up to six steps (three optional). The expedited approval process involves just two approval steps.
Finally, the primary approval process (or expedited approval process) itself executes in FactoryTalk Batch Recipe Editor.

This flow chart shows this sequence of tasks in the setup, configuration, and execution of signatures for recipe approval.

See also

Configure recipe approval steps on page 74

Edit a recipe approval on page 76

**Configure recipe approval steps**

Use the **Recipe Approvals Configuration** dialog box to configure the steps used by the primary approval process and expedited approval process. This includes assigning step names, step order, step concurrency and signature templates to the selected approval steps. Upon saving the area model, FactoryTalk Batch Equipment Editor stores the current recipe approval process configuration.

**Tip:** The recipe approval process uses signoffs to progress through the primary approval process, the expedited approval process, or to revert back from a prior approval. Determine if existing signature templates are sufficient for the recipe approval process signoffs or if additional templates must be created.

**To configure recipe approval steps:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. Select **Edit > Recipe Approvals Configuration**. The **Recipe Approvals Configuration** dialog box opens.
3. Configure the steps for the approval process. Three steps are mandatory: the first listed step (default label: **Begin Approval**), **Release Recipe as Step**, and **Release Recipe to Production** must all have **Approval** and **Revert** signoffs configured before enabling recipe approval.

- For each step, select a Signature Template to Approve. Select browse ( ) and select a template from the list of templates on the **Select Signature Template** dialog box. Select **OK**.
- For each step, select a Signature Template to Revert. Select browse ( ) and select a template from the list of templates in the **Select Signature Template** dialog box. Select **OK**.

  **Tip:** Change the step order for the optional steps any time during or after configuration by selecting the desired number from the **Step Order** list and then selecting **Refresh Order**.

4. (optional) To enable one or more of the three optional steps in the primary approval process, select the check box next to the desired step. Once enabled, customize the name and description for these steps if desired. The optional steps can be named for any approval milestone the methodology requires.
5. When configured Approval and Revert signature templates for all enabled steps, select the **Enable Recipe Approvals** check box. The recipe approval process enables for all recipes in the current project.

6. Select **OK** to close the dialog box.

   **Tip:** If configured to have **View Only** rights in FactoryTalk Batch Equipment Editor, the **OK** button is disabled.

7. Save the area model and exit FactoryTalk Batch Equipment Editor. Open and verify all recipes against the updated area model (in FactoryTalk Batch Recipe Editor), to activate the approval process.

The actual approval of recipe steps is done in FactoryTalk Batch Recipe Editor. Either the primary approval process or expedited approval process is available.

**See also**

- [Recipe Approvals Configuration dialog box](#) on page 76
- [Signature template](#) on page 57

### Recipe Approvals Configuration dialog box

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step Order</td>
<td>The order that steps are approved. There can be multiple steps with the same number. When that happens, all steps of that number must be approved before the recipe can continue to the next step.</td>
</tr>
<tr>
<td>Step Name</td>
<td>Name of each approval step. Default names are provided. If needed, edit the name for the first and optional steps. Names must be unique, and must begin with 0 through 9, or A through Z. Maximum number of characters allowed is 32.</td>
</tr>
<tr>
<td>Approval</td>
<td>Approval signoff template.</td>
</tr>
<tr>
<td>Revert</td>
<td>Revert signoff template.</td>
</tr>
<tr>
<td>Description</td>
<td>An editable description of each approval step. Maximum number of characters allowed is 128.</td>
</tr>
</tbody>
</table>

**See also**

- [Configure recipe approval steps](#) on page 74
- [Edit a recipe approval](#) on page 76
Recipe approval process overview

Chapter 7

Edit a recipe approval

Use these instructions to edit a recipe approval.

To edit a recipe approval:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.


3. Make the necessary modifications. In this example one of the optional approval steps is enabled, and approval and revert signoffs are assigned:

4. Select OK.

5. Save the area model and select Invoke Recipe Editor ( ) to open FactoryTalk Batch Recipe Editor.

Tip: FactoryTalk Batch Recipe Editor must be restarted to see the updated area model and recipe approval changes made.

See also

Disable recipe approval on page 77

Disable recipe approval

Use these instructions to disable recipe approval.

To disable recipe approval:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

3. Uncheck **Enable Recipe Approvals**.

4. Select **OK**. A dialog box displays a warning that existing approval signoffs are removed from all recipes in the current working set when FactoryTalk Batch Recipe Editor is next opened.

5. Select **OK**.

6. Save the area model.

7. (optional) Select **Invoke Recipe Editor** ( )*to open FactoryTalk Batch Recipe Editor.

**Tip:** FactoryTalk Batch Recipe Editor must be restarted to see the updated area model and recipe approval changes made.

**See also**

[Edit a recipe approval](#) on page 76
Chapter 8

Security authority overview

FactoryTalk Batch Security Authority, when enabled, helps protect intellectual property (as held in the area model) and secures it from unauthorized use.

FactoryTalk Batch Equipment Editor, when directed by an authenticated user to secure an area model, reads the FactoryTalk Network Directory Security Authority Identifier (SAI) and stores a copy of it in the area model. Later, when FactoryTalk Batch Equipment Editor is directed to open and read a secured area model, the SAI of the area model is compared to the SAI of the current FactoryTalk Network Directory. If they match, FactoryTalk Batch Equipment Editor loads the area model. If they do not match, for example when running under another instance of FactoryTalk or if the SAI has been regenerated, the editing session does not have the authority to open the area model and stops the reading process.

Important: If the SAI in a FactoryTalk Network Directory is changed or lost, any area model that is bound to it can lose access. Rockwell Automation recommends that, before securing an area model, back up the FactoryTalk Network Directory, and save an unsecured version of the area model in a secure location.

See also

Security authority configuration on page 79

Secure an area model on page 81
Security authority configuration

This flow chart shows the sequence of tasks in the configuration and use of security authority.

Authority to secure area models and recipes is assigned in the FactoryTalk Administration Console. Security Authority is disabled by default.

Area models are secured within FactoryTalk Batch Equipment Editor using the **Security Authority** command. The FactoryTalk Network Directory Security Authority Identifier (SAI) is written into the area model schema. To subsequently open and edit the area model, the SAI in the area model must match that of the current FactoryTalk Network Directory. No match prevents the opening and editing of an area model and its associated recipes in FactoryTalk Batch Recipe Editor.

Recipes are secured within FactoryTalk Batch Recipe Editor using the **Security Authority** command. The FactoryTalk Network Directory SAI is written into the recipe header data. To subsequently open and edit the recipe in FactoryTalk Batch Recipe Editor, the SAI in the recipe must match the one in the current FactoryTalk Network Directory.

Import and export operations with secured recipes are restricted.

See also

- [Security authority overview](#) on page 79
- [Secure an area model](#) on page 81
Secure an area model

Area models are secured using the **Security Authority** command.

**To secure an area model:**

1. Open the area model to secure in FactoryTalk Batch Equipment Editor.

   **Important:** Before securing an area model to a specific FactoryTalk Network Directory, Rockwell Automation recommends backing up the FactoryTalk Network Directory and storing unsecured versions of the area model file in configuration (.cfg), XML (.axml), or text (.txt) formats in a secure location.

   For backup details, see **FactoryTalk Help**: Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Tools**, and then select **FactoryTalk Help**.

   Once an area model is secured with a particular FactoryTalk Network Directory, the area model cannot be opened if the security authority identifier associated with that directory no longer exists.

2. Select **Edit > Security Authority**. The **Security Authority** dialog box opens.

3. Select the check box to secure the area model. If the check box is not enabled, the logged in user account is not authorized to use this feature.

4. Select **OK**.

**See also**

[Security authority overview on page 79](#)
Chapter 9

Data servers overview

A FactoryTalk Batch data server is a reference to an OPC, FactoryTalk Live Data, or Common Industrial Protocol (CIP) data server program and one of its associated data sources. These data server references provide run-time information from the process-connected device (PCD) to FactoryTalk Batch View or the HMI software. Data servers are used to pass data between the FactoryTalk Batch Server and the PCD. Configure a separate FactoryTalk Batch data server for each data source, and its associated data server program, that communicates with the FactoryTalk Batch Server.

In FactoryTalk Batch Equipment Editor, add a data server definition for each data server program (and its associated data source) that communicates with the FactoryTalk Batch Server. This allows FactoryTalk Batch to access current tag data within the PCD, which is necessary to successfully run recipes, arbitrate equipment, and control single phases or operation sequences.

**Tip:** The FactoryTalk Live Data protocol is based on the OPC Data Access protocol. The FactoryTalk Batch Simulator imitates the functionality of a data server and can communicate with the FactoryTalk Batch Server using OLE for Process Control (OPC) communications. Sample programs are included with the FactoryTalk Batch Phase Simulator and can be run to simulate batch processes without being connected to a PCD.

See also

- Data server definition files on page 83
- Area model data servers on page 84
- OPC data servers on page 85

Data server definition files

Each data server defined in FactoryTalk Batch Equipment Editor must reference a data server definition file (DSDF). Configuration parameters for data servers are defined using DSDF files. Each data server program and associated data source that communicates with the FactoryTalk Batch Server must have its own associated DSDF file defining that data server type. The configuration values within the DSDF file define the communication parameters used by the Server when communicating with data servers of this type. DSDF files must be located in the Dataservers directory, which is located within the Batch root directory. These server types (.dsdf files) are included with FactoryTalk Batch:
• Simulator OPC
• PCPhase OPC
• Rockwell RSLinx Classic Gateway OPC
• RSOPC Gateway

• Rockwell RSLinx Classic OPC
• Logix5000 CIP
• RSLinx Enterprise FactoryTalk Live Data
• Instruction-based Server (eProcedure only)

Tip: A data server using the Instruction-based Server cannot be created. This data server is created automatically.

Tip: Data server definition files should never be edited directly. For questions regarding DSDF files, contact Rockwell’s Customer Support.

See also
Area model data servers on page 84

Area model data servers

One data server needs to be defined in the area model. Define a data server before adding phases or operation sequences to the area model.

Important: The Instruction-Based Server is for use with FactoryTalk eProcedure only. The data server automatically creates when an area model opens. A new data server using the InstructionBasedServer data server type cannot be created. If this data server is deleted, any phases or operation sequences and associated tags are also deleted. The next time the area model opens, the data server automatically creates again.

FactoryTalk Batch supports these data server protocols:

• OPC Data Access protocol
• FactoryTalk Live Data protocol
• Common Industrial Protocol (CIP)
The **Data Servers** dialog box provides a list of data servers defined for the active area model. The first data server in this list is the default. Select the **Name** header to sort the list in ascending or descending alphabetical order.

![Data Servers dialog box](image)

**See also**

- **Data Servers dialog box** on page 85

### Data Servers dialog box

The **Data Servers** dialog box contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The user-defined name of the data server. Select the <strong>Name</strong> header to sort the data servers in ascending or descending alphabetical order.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of data server, defined by a DSDF file. By selecting the <strong>Type</strong> header, the data server types can be sorted in ascending or descending alphabetical order.</td>
</tr>
<tr>
<td>Add</td>
<td>Opens the <strong>Add Data Server</strong> dialog box that is used to add new data servers to the area model.</td>
</tr>
<tr>
<td>Edit</td>
<td>Opens the <strong>Edit Data Server</strong> dialog box that is used to edit existing data servers.</td>
</tr>
<tr>
<td>Remove</td>
<td>Opens the <strong>Remove Data Server</strong> dialog box that is used to remove an existing data server. The dialog box displays a list of all associated phases or operation sequences, phase tags or operation sequence tags, and unit attribute tags that are removed.</td>
</tr>
<tr>
<td>Close</td>
<td>Closes the dialog box without applying a data server selection.</td>
</tr>
</tbody>
</table>

**See also**

- **Area model data servers** on page 84

### OPC data servers

These items are displayed for OPC data servers.

When adding a data server to the area model, these items must be configured in the **Add Data Server** dialog box:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The unique user-defined name for the data server definition.</td>
</tr>
</tbody>
</table>
## Data servers overview

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>The various choices of data server types are defined within separate DSDF files.</td>
</tr>
<tr>
<td>Default Item Access Path</td>
<td>The OPC item access path that is the default for the tags on all subsequently created phases that this data server is assigned to.</td>
</tr>
<tr>
<td></td>
<td>To reset the access path for all phase tags to this value, in the <strong>Edit Phase</strong> dialog box <strong>Tags</strong> tab, select <strong>Defaults</strong>. The tag item names are also reset to include the name of the phase.</td>
</tr>
<tr>
<td>Watchdog Item Access Path</td>
<td>The OPC watchdog item access path is for data servers that do not require separate read and write items. For data servers that require separate read and write items, this label changes to <strong>Watchdog Read Item Access Path</strong> and is used only for the read item access path. This is optional, depending on the data server.</td>
</tr>
<tr>
<td>Watchdog Item Name</td>
<td>The name of the OPC watchdog item for data servers that do not require separate read and write items. For data servers that require separate read and write items, this label changes to <strong>Watchdog Read Item Name</strong> and is used only for the read item name.</td>
</tr>
<tr>
<td>Location</td>
<td>Only used with OPC data servers. The location of the OPC data server. If the data server is located on the local computer, <strong>Local</strong> must be selected. If the data server is located on a remote network computer, <strong>Remote</strong> must be selected and the name of the computer where the data server is located must be provided. Enter the name of the remote computer using the naming scheme of the network transport. By default, Universal Naming Convention (UNC) names and Domain Names System (DNS) names are supported.</td>
</tr>
<tr>
<td>RSLinx Enterprise Server Path</td>
<td>Only used with RSLinx Enterprise FactoryTalk Live Data servers. A string that defines the path to a data server located in the FactoryTalk Directory. (See <strong>RSLinx Enterprise FactoryTalk Live Data server</strong> for more information.)</td>
</tr>
<tr>
<td>Watchdog Protocol</td>
<td>The type of watchdog protocol to be used by the data server. This can be either the <strong>Standard</strong> or <strong>Enhanced</strong> protocol.</td>
</tr>
<tr>
<td>Watchdog Write Item Access Path</td>
<td>The OPC access path for the watchdog write item. If the data server does not require separate read and write items or does not require an access path, this is disabled.</td>
</tr>
<tr>
<td>Watchdog Write Item Name</td>
<td>The OPC item name for the watchdog write item. If the data server does not require separate read and write items, this is disabled.</td>
</tr>
<tr>
<td>Scan Rates</td>
<td>The default <strong>On Control</strong> and <strong>Off Control</strong> scan rates for the data server, in milliseconds. The valid range of the <strong>On Control</strong> scan rate is 250 through 30,000 milliseconds. The valid range of the <strong>Off Control</strong> scan rate is 250 through 60,000 milliseconds. The usage of these rates is determined by the type of data server selected. Typically, these values do not need to be modified.</td>
</tr>
</tbody>
</table>
Add an OPC data server

These instructions only apply to OPC data servers.

Important: When using RSLinx Enterprise with a Live Data/OPC data server, the following entry must be added to the RSLinxEnterprise.dsdf in the \Batch\DataServers folder:

```
ForceBoolTrueToOne=YES
```

This is also required when updating from RSLinx Classic to RSLinx Enterprise.

To add an OPC data server:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.
2. Select Edit > Data Server. The Data Servers dialog box opens.
3. Select Add.
4. In Name, type a name for the data server.
5. From Type, select the data server definition that corresponds to the physical OPC data server being defined.
6. If necessary, type the appropriate values for the Default Item Access Path, Watchdog Item Access Path, and Watchdog Item Name.
7. Under Location, select the appropriate option. If Remote is selected, type the remote data server name.
10. If necessary, type the appropriate values for the Watchdog Write Item Access Path and the Watchdog Write Item Name.
11. Under **Scan Rates**, type the **On Control** and **Off Control** scan rates for the data server.

![Add Data Server dialog box](image)

12. Select **OK** to save the data server configuration and return to the **Data Servers** dialog box.

**See also**

- [OPC data servers on page 85](#)

**RSLinx Enterprise**

**FactoryTalk Live Data server**

The FactoryTalk Live Data protocol provides services that read and write real-time manufacturing data from OPC servers and controllers for all participating FactoryTalk-enabled products in the FactoryTalk Directory. FactoryTalk Live Data improves on OPC-Data Access (DA) by making communication with multiple servers easier.

To support FactoryTalk Batch as a FactoryTalk Live Data client, RSLinx Enterprise must be part of the FactoryTalk system. RSLinx Enterprise is a communication server that provides plant-floor device connectivity for Rockwell Automation applications and Logix processors.

**See also**

- [Add a data server on page 88](#)
- [RSLinx Enterprise server path on page 90](#)
- [Configure the RSLinx Enterprise server path on page 90](#)
Add a data server

The items necessary to configure an FactoryTalk Linx data server type in FactoryTalk Batch Equipment Editor are identical to OPC servers with the exception of the FactoryTalk Linx server path.

Before you begin:

Complete these prerequisites:

- Install and configure FactoryTalk Linx. (For more information, refer to FactoryTalk Linx Help.)
- Create an Application in the FactoryTalk Directory. (For more information, refer to FactoryTalk Help located in the FactoryTalk Administration Console.)
- Add an FactoryTalk Linx data server to the FactoryTalk Application.

Adding a data server to an application provides a link between a hardware device containing tags and the FactoryTalk Directory. (For more information, refer to FactoryTalk Help located in the FactoryTalk Administration Console.)

Important: When using FactoryTalk Linx with a Live Data/OPC data server, the following entry must be added to the FTLX.dsdf in the \Batch\DataServers folder:

```
ForceBoolTrueToOne=YES
```

This is also required when updating from RSLinx Classic to FactoryTalk Linx.

To add a data server:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.
2. Select Edit > Data Servers dialog box.
3. Select Add to open the Add Data Server dialog box.
4. In Name, type a name for the data server.
5. From Type, select FactoryTalk Linx. When the data server Type is FactoryTalk Linx, the FactoryTalk Linx Server Path text box replaces the Location text box in the Add Data Server dialog box.
6. If necessary, type the appropriate values for the Default Item Access Path and the Watchdog Item Access Path. (These are the device shortcut name(s) assigned in FactoryTalk Live Data.)
7. If necessary, type the appropriate values for the **Watchdog Item Name**.

8. Type the connection string for the **FactoryTalk Linx Server Path** found on the **FactoryTalk Administration Console** dialog box. Enter the path that is shown, but without the server name shown at the end of the path. (See **FactoryTalk Linx server path** for more information.)

9. Select **More** to view the watchdog write item access path and name and the on and off control scan rates for the data server.

10. Select **OK** to save the data server configuration and return to the **Data Servers** dialog box.

See also

**FactoryTalk Linx server path** on page 90

### RSLinx Enterprise server path

Once an RSLinx Enterprise data server is added to an Application in the FactoryTalk Directory, to view the RSLinx Enterprise Server Path connection string that is needed to configure the RSLinx Enterprise server type in FactoryTalk Batch Equipment Editor, see the **Communication Setup** title in the **Communication Setup** dialog box.

In this example, the RSLinx Enterprise Server Path connection string is: RNA://$Global/MyApplication/

(Note that the final name in the path (**RSLinx Enterprise**) does not show at the end as shown in the image.)

See also

**Configure the RSLinx Enterprise server path** on page 90
Configure the RSLinx Enterprise server path

Use these instructions to open the FactoryTalk Communication Setup dialog box.

To configure the RSLinx Enterprise server path:

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


3. Right-click Communication Setup and select Open.

The Communication Setup dialog box opens. The RSLinx Enterprise Server Path connection string displays as part of the title.

See also

RSLinx Enterprise FactoryTalk Live Data server on page 88

Edit a data server

When editing an OPC or RSLinx Enterprise FactoryTalk Live Data server associated with a unit attribute tag, a phase or any tag within a phase, Logix5000 CIP does not display in the Type list. One OPC type can be changed to another or changed from an OPC type to an RSLinx Enterprise FactoryTalk Live Data type (for example, for a Simulator OPC data server, change Type to Rockwell RSLinx Classic OPC), but an OPC or RSLinx Enterprise FactoryTalk Live Data type server cannot be changed to a Logix5000 CIP type data server.

To change the Type to Logix5000 CIP, delete the data server, and add a new one.

Important: When using FactoryTalk Linx with a Live Data/OPC data server, the following entry must be added to the FTLX.dsdf in the \Batch\DataServers folder:

```
ForceBoolTrueToOne=YES
```

This is also required when updating from RSLinx Classic to FactoryTalk Linx.

To edit a data server:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select Edit > Data Servers to open the Data Servers dialog box.

3. Select the data server definition edit, and then select Edit.
The **Edit Data Server** dialog box opens.

4. Make any required changes and select **OK** to return to the **Data Servers** dialog box.

All phase tags assigned to the modified data server definition are updated to reflect the changes.

5. Close the **Data Servers** dialog box.

**See also**

- [OPC data servers on page 85](#)
- [Delete a data server on page 92](#)

**Delete a data server**

Use these instructions to delete a data server.

**To delete a data server:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. Open the **Data Servers** dialog box.

3. Select the data server to delete, and then select **Remove**.

   - If the data server is not assigned to any phases or operation sequences, this message displays:

   ![Message](image)

   • If the data server is assigned to any phases, operation sequences, or unit attribute tags, the **Remove Server** dialog box opens. All the phases, operation sequences, and unit attribute tags associated with the data server are listed.

   **Tip:** The displayed phases or operation sequences and all of their tags, as well as the displayed unit attribute tags, are deleted when the data server is removed.

4. Perform one of the following steps:
• To delete the data server and remove all associated phases or operation sequences and tags, select Yes.
• To cancel the deletion and return to the Data Servers dialog box, select No.

5. Close the Data Servers dialog box.

See also

Assign data servers on page 93

Assign data servers
After a data server is added, assign it to either a phase, an operation sequence, or a unit.

<table>
<thead>
<tr>
<th>Phase or operation sequence</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>Assign a data server to a phase or operation sequence to link the equipment piece to the specified controller.</td>
<td>Assign the same data server to all phases or operation sequences subsequently created in a unit. Select the data server in the Edit Unit dialog box.</td>
</tr>
</tbody>
</table>

See also

Configure the general data for a process cell on page 113

Data servers overview on page 83
Chapter 10

Resources overview

Resource is a general term for equipment including process cells, units, phases and shared control modules.

Define the resources in the area model prior to assigning them as shared equipment. Create resource classes (such as PUMPS) and resource instances (such as PMP102). 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. To configure a resource instance, assign a name, equipment ID, the maximum number of owners, and any other equipment requirements for that resource. When the maximum number of owners are defined, the number of area model components that can simultaneously use the resource are also defined.

Resources can be simultaneously owned by one or more area model components, depending on the program. For example, if a pump is physically configured to charge an ingredient into three different vessels but it has the capacity to charge only two of the three vessels simultaneously, define the pump as a shared resource and specify the maximum number of owners as 2.

See also

Arbitration of resources on page 95

Before the recipe procedure executes, all resources needed by the process cells, units, phase classes, and operation sequence classes in which the recipe procedure is running must be acquired by the FactoryTalk Batch Server through arbitration. This example shows how each recipe level behaves when its corresponding area model component requires resources. Define resources as required by an area model component in the Edit Process Cell, Edit Unit, Edit Phase, or Edit Operation Sequence dialog boxes on the Arbitration tab.
<table>
<thead>
<tr>
<th>Area Model Component</th>
<th>Recipe Level</th>
<th>Recipe Name</th>
<th>Behavior of Recipes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process_Cell_A</td>
<td>Procedure</td>
<td>CLS_FRENCHVANILLA</td>
<td>The French vanilla recipe running in Process_Cell_A cannot begin execution until resources required by Process_Cell_A have been acquired by the FactoryTalk Batch Server.</td>
</tr>
<tr>
<td>Unit_A</td>
<td>Unit Procedure Operation</td>
<td>CLS_SWEETCREAM_UP:1 CLS_SWEETCREAM_OP:1</td>
<td>The unit procedure and operation sweet cream recipes running in Unit_A cannot begin execution until resources required by Unit_A have been acquired by the FactoryTalk Batch Server.</td>
</tr>
<tr>
<td>Phase_A</td>
<td>Recipe Phase</td>
<td>ADD_EGG:1</td>
<td>ADD_EGG:1 running in Phase_A cannot begin execution until the resources required by Phase_A have been acquired by the FactoryTalk Batch Server.</td>
</tr>
<tr>
<td>Operation_Sequence_A</td>
<td>Operation Sequence</td>
<td>OPERATION_CLS1</td>
<td>OPERATION_CLS1 running in Operation_Sequence_A cannot begin execution until the resources required by Operation_Sequence_A have been acquired by the FactoryTalk Batch Server.</td>
</tr>
</tbody>
</table>

**Tip:** When a resource is deleted, it is removed from all equipment lists in which it previously existed.

**See also**

- [Resources overview](#) on page 95
- [Define shared resources](#) on page 97
Define shared resources

Use these instructions to define shared resources.

To define shared resources:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select Edit > Resources.

3. To create a new resource class, select Add Class. The Edit Resource Class dialog box displays.
   a. In Name, type the new resource class name.
   b. Select OK to return to the Edit Resources dialog box. The new resource class displays in the Resource Classes column.

4. To add a resource instance to a resource class, select the new resource class, and then select Add. The Edit Resources: [Resource Name] dialog box displays.

The Edit Resources: [Resource Name] dialog box contains four tabs:

- General
- Arbitration
- Cross Invocation
- Hyperlink (eProcedure only)

See also

Configure general information for resources on page 98

Configure arbitration information for resources on page 98

Configure cross invocation information for resources on page 100
Configure general information for resources

Use these instructions to configure general information.

To configure general information for resources:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select Edit > Resources.

3. Under Resources, select the resource and then select Edit.

4. In the Edit Resources: [Resource Name] dialog box, select the General tab.

5. In Name, type the resource name (for example, Pump101, PV203).

6. Type the appropriate Equipment ID, or accept the default ID.

7. Select Apply to save changes and continue, or select OK to return to the Edit Resources dialog box.

See also

- Define shared resources on page 97
- Configure arbitration information for resources on page 98
- Configure cross invocation information for resources on page 100
- Configure hyperlink information for resources on page 101

Configure arbitration information for resources

On the Edit Resources dialog box Arbitration tab, the Equipment Allocation by Resource area contains a hierarchical view in the left pane that displays the current equipment configuration of the area model. The hierarchy has two roots: Process Cells and Resources. The right pane is the list of equipment on which this resource is dependent. Any equipment in the right pane must be acquired by the FactoryTalk Batch Server before any procedural element requiring this resource can begin execution.

Expand the Process Cells root to view all the process cells within the area model. From here, organization is hierarchical, so process cells contain units and units contain phases.

Expand the Resources root to view a list of all the configured resource instances within the area model.
To configure arbitration information for resources:

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. Select **Edit > Resources**.

3. Under **Resources**, select the resource and then select **Edit**.

4. In the **Edit Resources: [Resource Name]** dialog box, select the **Arbitration** tab.

5. In **Max Owners**, enter the number that represents how many process cells, units, phases, or other resources can maintain ownership of this resource at one time.

6. Under **Equipment Allocation by Resource**, browse the area model’s equipment hierarchy to specify equipment this resource needed. A plus sign (+) to the left of the equipment type indicates equipment of that type has been defined.
   - To add equipment to the right pane, select the equipment needed by the resource in the left pane and select **Add**. The equipment this resource needs to acquire displays in the right pane.
   - To remove equipment, in the right pane, select the equipment to delete and select **Remove**. The selected equipment is removed from the list.

7. Select **Apply** to save changes and continue, or select **OK** to return to the **Edit Resources** dialog box.

See also

- [Define shared resources](#) on page 97
- [Configure general information for resources](#) on page 98
Configure cross invocation information for resources

Configure cross invocation information for resources

Use the Cross Invocation tab to specify cross invocation labels for different resource types.

To configure cross invocation information for resources:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select Edit > Resources.

3. Under Resources, select the resource and then select Edit.

4. In the Edit Resources: [Resource Name] dialog box, select the Cross Invocation tab.

5. In Cross Invocation Label #, type the cross invocation string contents information.

6. Select Apply to save changes and continue, or select OK to save changes and return to the Edit Resources: [Resource Name] dialog box.

See also

Configure hyperlink information for resources on page 101
Configure hyperlink information for resources

This feature, allows hyperlinks to be associated with any of the equipment resources, used in conjunction with eProcedure only.

To configure hyperlink information for resources:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select Edit > Resources.

3. Under Resources, select the resource and then select Edit.

4. If hyperlinks are being used, in the Edit Resources: [Resource Name] dialog box, select the Hyperlink tab.

5. In Hyperlink Label # English, type a valid URL string.

6. Select Apply to save changes and continue, or select OK to return to the Edit Resources: [Resource Name] dialog box.

7. When finished creating the shared resources, select OK to return to the Design View area.

See also

Define shared resources on page 97
Configure general information for resources on page 98
Configure arbitration information for resources on page 98
Configure cross invocation information for resources on page 100
Chapter 11

Global binding requirements

Global unit binding requirements are sets of expressions that tell the FactoryTalk Batch Server what units are valid binding candidates for a recipe. A binding requirement can be evaluated against instances of a unit class to determine which instances of the class are legal bind candidates for a unit requirement. When the FactoryTalk Batch Server attempts to bind a unit requirement during recipe execution, the entire set of binding requirements defined for the unit requirement is evaluated against each potential binding candidate. Units unable to meet every binding requirement defined for the unit requirement are eliminated as potential binding candidates.

A global binding requirement is automatically applied to every unit requirement within a recipe. Pre-configuring global binding requirements in an area model, and then loading that area model within FactoryTalk Batch Recipe Editor, eliminates having to manually add the binding requirement to every unit requirement in every recipe.

For example, a global binding requirement may be $Days\_since\_CleanInPlace < 3$. This binding requirement might be imposed by corporate quality standards. This means that no unit is considered acceptable for use unless it has been cleaned within the last three days. As a global unit requirement, this requirement is required to be applied against every unit requirement in the plant.

See also

Unit binding expression builder on page 103
Create global binding requirements on page 106

Create expressions in the Unit Binding Expression Builder. The Unit Binding Expression Builder contains a type list on the left pane, which displays Recipe Header and Unit Attribute data that can be used to build an expression.

When Recipe Header is selected, a list of recipe header parameters displays in the list view (in the middle). The Recipe Header box names map to the same boxes defined in FactoryTalk Batch Recipe Editor within a Recipe Header.

When the Unit Attributes is selected, an alphabetic list of all global unit attributes existing within the area model displays in the list view.
Create or edit an expression by typing directly in the Expression Builder text box, selecting items in the list view and enumerations view, selecting various operators, or selecting a combination of items and operators. The Operator buttons of the Expression Builder are always enabled. When an Operator button is selected, the corresponding operator is inserted into the expression at the current cursor position and displays in the Expression Builder text box.

See also

- [Create a global unit attribute](#) on page 147
- [Expression operators](#) on page 104
- [Supported data types](#) on page 105
- [Operands](#) on page 105
- [Expression validation](#) on page 105

### Expression operators

Binding expressions support operators. The precedence of the execution depicts from highest to lowest. An operator with a higher precedence executes before an operator of lower precedence.

<table>
<thead>
<tr>
<th>Expression Operator</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>()</td>
<td>Expressions within parentheses are evaluated before expressions outside of parentheses.</td>
</tr>
<tr>
<td>NOT, -</td>
<td>Logical NOT, subtraction.</td>
</tr>
<tr>
<td>*, /, AND</td>
<td>Multiplication, division, and logical AND.</td>
</tr>
<tr>
<td>+, -, OR</td>
<td>Addition, subtraction, and logical OR.</td>
</tr>
<tr>
<td>&lt;, &lt;=, &gt;, &gt;=</td>
<td>Less than, less than or equal to, greater than, and greater than or equal to.</td>
</tr>
<tr>
<td>=, &lt;&gt;</td>
<td>Equal to, and not equal to.</td>
</tr>
</tbody>
</table>

See also

- [Unit binding expression builder](#) on page 103
Supported data types

The data types supported are integer, real, string, and enumeration. Data type examples:

**Integer:** 423

**Real:** 423.123456789012

**String:** The string constant must be in quotes: "READY".

**Enumeration:** As a string, the enumeration constant must be in quotes: "BUTTER_PECAN". As an integer, the ordinal for the enumeration may be: 4.

See also

- Unit binding expression builder on page 103

Operands

Operands used within a global unit binding expression can reference recipe header data, global unit attributes, and constants.

Examples of allowable expression entities and the required syntax:

<table>
<thead>
<tr>
<th>Expression Entity</th>
<th>Expression Syntax</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recipe header data</td>
<td>[Recipe header item]</td>
<td>[VERSION_NUMBER]</td>
</tr>
<tr>
<td>Global unit attribute</td>
<td>Unit attribute name</td>
<td>CAPACITY</td>
</tr>
<tr>
<td>Constant</td>
<td></td>
<td>100</td>
</tr>
</tbody>
</table>

See also

- Unit binding expression builder on page 103

Expression validation

Although an invalid expression cannot be created, a previously valid expression may become invalid through subsequent changes elsewhere in the area model. When the Edit Global Unit Binding Requirements dialog box is invoked all binding expressions are re-validated. If any current expressions have become invalid due to subsequent changes, an error message opens and the dialog displays an asterisk beside each invalid expression.
The expression validates for consistency and syntactical correctness. If the defined expression validates successfully, the dialog box closes. If the expression is invalid, then the location of where the first error encountered highlights, a description of the error displays in a message box, and the dialog box does not close.

**Validation rules:**

- An expression must resolve to either TRUE or FALSE and meet the criteria:
  - An expression must contain at least one operator (+ (addition), - (subtraction), * (multiplication), / (division), = (equals), > (greater than), < (less than), <= (less than, or equal to), >= (greater than, or equal to), <> (not equal to), Not, And, and Or.
  
  **Tip:** OR transitions fire if at least one side of the expression is TRUE.

- Separate each entity (recipe header, global unit attribute, or constant) with an operator.

- The operated entities must evaluate to acceptable data types (string with a string; integer with integer or real; real with integer or real).

- Maximum expression length must not exceed 1023 characters.

- Duplicate expressions are allowed.

**See also**

- [Unit binding expression builder on page 103](#)

- [Create global binding requirements on page 106](#)

- [Supported data types on page 105](#)

**Create global binding requirements**

Use these instructions to create global binding requirements.

**To create global binding requirements:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.
2. Select Edit > Global Binding Requirements.

3. Select Add Requirement.

4. In the Unit Binding Expression Builder dialog box, select options from the three different sections or type to build an expression.

   Tip: A binding requirement expression evaluates to either a TRUE or FALSE value, TEMPERATURE < 20.3, for example.

5. Select OK to save the expression and return to the Edit Global Unit Binding Requirements dialog box.

6. Continue building more binding requirements, or select OK to exit the dialog box.

See also

   Unit binding expression builder on page 103
   Expression validation on page 105
   Edit binding requirements on page 108
   Delete binding requirements on page 108
Edit binding requirements

Use these instructions to edit binding requirements:

To edit binding requirements:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select Edit > Global Binding Requirements.

3. Select browse ( ) next to the requirement to edit.

4. Make the desired changes, and then select OK.

5. Select OK to close the Edit Global Unit Binding Requirements dialog box.

See also

Create global binding requirements on page 106

Delete binding requirements on page 108

Delete binding requirements

Use these instructions to delete binding requirements:

To delete binding requirements:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select Edit > Global Binding Requirements.

3. Select Delete Requirement, and then select OK to confirm the deletion or Cancel to keep the requirement.

4. Select OK to close the Edit Global Unit Binding Requirements dialog box.
See also

Create global binding requirements on page 106

Edit binding requirements on page 108
Chapter 12

Process cell overview

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 a selected area, a process cell class must first be created. Once the process cell class has been created, an instance of that class can be created in the Design View area.

See also

Create a process cell class on page 111
Edit a process cell class on page 112
Create and configure process cell instances on page 112
Configure hyperlink data for a process cell on page 117

Create a process cell class

Use these instructions to create a process cell class.

**Tip:** If the icon column is empty, see Verify icon paths.

To create a process cell class:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. In the Class View area, verify that the column heading is Cell Classes. If not, select Go Up (▲) until the top level of the area model is shown.


4. In Name, enter a name for the process cell class.

5. Under Icon, select an icon.

6. Select OK.
See also

Verify icon paths on page 119

Process cell overview on page 111

Edit a process cell class

Use these instructions to edit a process cell class.

To edit a process cell class:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select the process cell class to edit.


4. Make the desired changes, and then select OK to return to the Class View area.

See also

Create and configure process cell instances on page 112

Verify icon paths on page 119

Unique icons on page 121

Create and configure process cell instances

Use these instructions to create and configure process cell instances.

To create and configure process cell instances:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select the desired process cell class.

The cursor changes to a c.

3. Place the cursor in the desired location in the Design View area and select.

The process cell icon displays in the Design View area with a default name of PROCESS_CELL# (where # is a sequentially assigned number starting with 1).

See also

Configure a process cell instance on page 113
Configure a process cell instance

Use these instructions to configure a process cell instance.

**To configure a process cell instance:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. Select a process cell icon in the **Design View** area.

3. Select **Edit > Properties**.

   The **Edit Process Cell** dialog box displays with the assigned defaults. The **Edit Process Cell** dialog box contains four tabs:
   - General
   - Arbitration
   - Cross Invocation
   - Hyperlink (eProcedure only)

See also

Configure the general data for a process cell on page 113

Configure the general data for a process cell

Use these instructions to configure the general data.

**To configure the general data for a process cell:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. Select a process cell icon in the **Design View** area.

3. Select **Edit > Properties**.

4. In the **Edit Process Cell** dialog box, select the **General** tab.

5. In **Name**, type a process cell name.
6. For RSBizWare BatchERP, in **ERP Alias**, type the SAP R/3 process cell name.

The ERP alias is the process cell name in SAP. It maps the process cell from FactoryTalk Batch to SAP.

7. In **Equipment ID**, type the equipment ID, or accept the default value.

8. Select **Apply** to save changes.

See also

**Edit process cell dialog box - General tab** on page 114

The **General** tab of the **Edit Process Cell** dialog box contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the process cell. (The default name is PROCESS_CELL#. The # is a consecutive number starting with 1 that is assigned to each process cell as it is added to an area.)</td>
</tr>
<tr>
<td>Icon</td>
<td>The icon selected for the process cell class (view-only).</td>
</tr>
<tr>
<td>Class</td>
<td>The name of the process cell class (view-only).</td>
</tr>
<tr>
<td>ERP Alias</td>
<td>(RSBizWare BatchERP only) The SAP name for the phase. The ERP alias maps the equipment module from FactoryTalk Batch to SAP.</td>
</tr>
<tr>
<td>Equipment ID</td>
<td>The equipment identification number identifies this process cell. (A default ID is automatically assigned to the process cell, based on the next available sequential number. The first item added to an area is assigned ID #1.) To request this process cell, the equipment ID must be the same as the equipment ID that is used by the phase logic.</td>
</tr>
</tbody>
</table>

See also

**Configure the general data for a process cell** on page 113

In the **Equipment Allocation by Resource** area, the left pane is a hierarchical display of the current equipment configuration of the area model. The hierarchy has two roots: Process Cells and Resources. The right pane is the list of equipment this process cell is dependent upon. Any equipment in this list must be acquired by the process cell before any procedural element linked to this process cell can begin execution.

Expand **Process Cells** to view all the process cells defined within the area model. From here, organization is hierarchical; process cells contain units and units contain phases.

Expand **Resources** to view a list of all the configured resources within the area model.
To configure the arbitration data for a process cell:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select a process cell icon in the Design View area.


4. In the Edit Process Cell dialog box, select the Arbitration tab.

5. In Max Owners, type the maximum number of owners.

6. Browse the area models equipment hierarchy to specify equipment the process cell must acquire. The lack of a plus or minus sign to the left of an equipment type indicates no equipment of that type has been defined, or the equipment type cannot be used as a needed equipment item at this level.

   • To add equipment, under the Equipment Allocation by Resource area, select the equipment needed by the process cell and select Add. The selected equipment displays in the right pane.

   • To remove equipment, in the right pane, select the equipment to delete and select Remove. The selected equipment is removed from the list.

7. Select Apply to save changes.

See also

Edit process cell dialog box - Arbitration tab on page 115
Edit Process Cell dialog box - Arbitration tab

The Edit Process Cell dialog box on the Arbitration tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the process cell (view-only). The default name is PROCESS_CELL#. The # is a consecutive number starting with 1 that is assigned to each process cell as it is added to an area.</td>
</tr>
<tr>
<td>Icon</td>
<td>The icon selected for the process cell class (view-only).</td>
</tr>
<tr>
<td>Class</td>
<td>The name of the process cell class (view-only).</td>
</tr>
<tr>
<td>Max Owners</td>
<td>The maximum number of owners (other process cells, units, phases, or other resources) that can simultaneously use this process cell. The default is 1.</td>
</tr>
</tbody>
</table>
| Equipment Allocation by Resource | A list of the equipment for which ownership is required by this process cell in order for any procedural element linked to the process cell to begin execution. The default is blank. In general, assign any required resources to the lowest possible level to allow for maximum efficiency when running multiple processes. The left pane is a hierarchical display showing the current equipment configuration of the area model. The hierarchy has two roots:  
  • Process Cells: Displays all the process cells within the area model. From here, organization is hierarchical: process cells contain units and units contain phases and operations sequences.  
  • Resources: Displays a list of all the configured resources within the area model.  
The right pane is the list of equipment this process cell is dependent upon. Any equipment in this list must be acquired by the process cell before the process cell itself can be acquired. |

See also

Configure the arbitration data for a process cell on page 114

Use these instructions to configure cross invocation data.

To configure cross invocation data for a process cell:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.
2. Select a process cell icon in the Design View area.
4. In the **Edit Process Cell** dialog box, select the **Cross Invocation** tab.

5. Type the cross invocation string information in the **Cross Invocation Label** boxes.

6. Select **Apply** to save changes.

See also

*Edit process cell dialog box - Cross Invocation tab* on page 117

**Edit process cell dialog box - Cross Invocation tab**

The **Edit Process Cell** dialog box on the **Cross Invocation** tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the process cell (view-only). The default name is <strong>PROCESS_CELL#</strong>, where # is a consecutive number starting with 1, assigned to each process cell as it is added to an area.</td>
</tr>
<tr>
<td>Class</td>
<td>The name of the process cell class (view-only).</td>
</tr>
<tr>
<td>Icon</td>
<td>The icon selected for the process cell class (view-only).</td>
</tr>
<tr>
<td>Cross Invocation Configuration</td>
<td>The cross invocation strings indicate the addition of an item to FactoryTalk Batch View and the ActiveX control shortcut menus. This includes the caption for each menu item and the data to be passed to the specified automation server when the menu item is selected. The default labels associated with each of the five fields are <strong>Cross Invocation Label #</strong> (where # is 1 through 5). These labels can be modified in the <strong>Server Options</strong> dialog box, and are seen in the <strong>Edit Process Cell</strong> dialog box.</td>
</tr>
</tbody>
</table>

See also

*Configure cross invocation data for a process cell* on page 116
Configure hyperlink data for a process cell

This feature is used in conjunction with FactoryTalk eProcedure only. Use these instructions to configure hyperlink data.

To configure hyperlink data for a process cell:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select a process cell icon in the Design View area.


4. In the Edit Process Cell dialog box, if hyperlinks are being used, select the Hyperlink tab.

5. In the Hyperlink Configuration boxes, type a valid URL string.

6. Select Apply to save changes.

7. Select OK to return to the Design View area.

See also

Edit process cell dialog box - Hyperlink tab on page 118
Edit process cell dialog box - Hyperlink tab

The Edit Process Cell dialog box Hyperlink tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the process cell (view-only). The default name is PROCESS_CELL#. The # is a consecutive number starting with 1 that is assigned to each process cell as it is added to an area.</td>
</tr>
<tr>
<td>Icon</td>
<td>The icon selected for the process cell class (view-only).</td>
</tr>
<tr>
<td>Class</td>
<td>The name of the process cell class (view-only).</td>
</tr>
<tr>
<td>Hyperlink Configuration</td>
<td>The URL associated with each hyperlink label. The default labels associated with each of the five fields are Hyperlink Label # English (where # is 1 through 5). These labels can be modified in the Server Options dialog box. The labels are used in the eProcedure Client Equipment view.</td>
</tr>
</tbody>
</table>

See also

Configure hyperlink data for a process cell on page 117

Remove a process cell instance

Use these instructions to remove a process cell instance.

To remove a process cell instance:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select a process cell icon in the Design View area.

3. Press Delete.

   The Remove Process Cell warning displays that lists the associated units and phases that are removed when deleted the process cell.

4. Select Yes to delete the process cell, or select No to cancel the deletion and return to FactoryTalk Batch Equipment Editor.

See also

Configure a process cell instance on page 113
Verify icon paths

The directory paths for each type of icon are set by the installation procedure; modify them only if the bitmaps have been moved to another directory.

To verify icon paths:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select Options > Directories.

3. Verify the directory paths for each type of icon.

4. Select OK to return to the Design View area.

See also

Create and configure process cell instances on page 112

Unique icons on page 121
Unique icons

Create a graphical representation of a process cell class, unit class, unit class background, or phase class using a graphic paint program. Create icons specific to the facility or process. Use the following rules when designing unique icons:

- Icons must be stored in the appropriate directory as a .bmp file.
- The icon must be a square or rectangle, with a width made up of an even number of pixels. Icons with an odd numbered pixel width will not show.
- The file must not exceed 7K in size (116 x 116 pixels). Icons exceeding the size limit are displayed as a blank icon with a red question mark.
- Icons are stored in one of these directories:
  - **Process Cell Class**
    - C:\Program Files\Rockwell Software\Batch\BMP\PROCCEL
  - **Unit Class**
    - C:\Program Files\Rockwell Software\Batch\BMP\UNIT
  - **Unit Class Backgrounds**
    - C:\Program Files\Rockwell Software\Batch\BMP\UNIT\VESSELS
  - **Phase Class**
    - C:\Program Files\Rockwell Software\Batch\BMP\PHASE

Tip: New icons that meet these criteria display in the Icon column for process cell classes, unit classes, and phase classes.

See also

Verify icon paths on page 119

Process cell overview on page 111
Chapter 13

Unit overview

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. The material contained in the central vessel is usually transformed either physically, chemically, or biologically. Units run one operation or unit procedure-level recipe at a time. Process &Instrumentation Diagrams (P&IDs) are a good source of information for determining unit boundaries.

Often process facilities contain multiple instances of similar equipment, such as unit vessels, that consist of the same or similar components and functionality. These groupings of similar equipment can be described as a unit class. In the area model database, create unit classes (such as MIXER_CLS) and one or many unit instances of a particular unit class (such as MIXER_A and MIXER_B).

Recipes can be configured to run within a specific unit instance or within a unit class. The action of specifying the particular unit in which a recipe runs is called unit binding. If a recipe is configured to run within a unit instance, the unit in which the recipe runs is specified at the time the recipe is created. If a recipe is configured to run within a unit class, the recipe can run in any unit instance of that unit class. The specific unit instance is determined:

- By the operator when the batch is created, prior to starting the batch.
- By the operator, after the batch is started, at the time a unit procedure level recipe is ready to run (the operator responds to a prompt).
- By the FactoryTalk Batch Server, after the batch is started, at the time a unit procedure level recipe is ready to run.

Class-based recipes:

- Minimize the number of recipes that need to be maintained.
- Allow the freedom to choose the most appropriate unit for binding the recipe at the time the batch is run.

Tip: To navigate to the unit level, double-click a Process Cell icon or select a Process Cell icon and select Go Down (↓).

See also

Create a unit class on page 124
Create a unit class

Use these instructions to create a unit class.

To create a unit class:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. In the Class View area, verify that the column heading is Unit Classes. The current unit classes for the selected process cell are displayed in the Class View area.

3. Select Class > New Unit Class. The Create Unit Class dialog box opens with the assigned defaults.
   - The Unit Attributes area lists non-global unit attributes.
   - The Global Unit Attributes area is view-only and lists unit attributes associated with every unit class in the area model.

4. In Name, type the unit class name.

5. Under Icon, select an icon.
   Tip: If the icon column is empty, see Verify icon paths.

6. If unit attributes have been defined and is used with this unit class, select the check box for the tag(s) to assign to the unit from the Unit Attributes list.
   Tip: Global unit attributes are associated automatically with the unit class being created.

7. Select OK.

See also

Verify icon paths on page 119
Add a unit attribute on page 138
Create a global unit attribute on page 147
Edit a unit class on page 124
**Edit a unit class**

Use these instructions to edit a unit class.

**To edit a unit class:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. In the **Class View** area, verify that the column heading is **Unit Classes**.
   
   The current unit classes for the selected process cell are displayed in the **Class View** area.

3. Select the unit class to edit and then select **Edit > Properties**.

4. In the **Edit Unit Class** dialog box, in **Name**, type a new unit class name.

5. From **Icon**, select a new unit class icon or modify the existing unit attribute tag selection. If any global unit attributes exist, they are listed in the **Global Unit Attributes** list (view-only).

6. Select **OK**.

**See also**

- [Create a unit instance](#) on page 125
- [Unit instance configuration](#) on page 126
- [Remove a unit instance](#) on page 133
- [Link unit instances](#) on page 134
- [Share units](#) on page 134

**Create a unit instance**

Use these instructions to create a unit instance.

**To create a unit instance:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. Select the **Unit Class** to be placed in the **Design View** area.

3. Place the cursor in the **Design View** area where the unit is to be located and select.
4. Configure the unit instance.

See also

Edit a unit class on page 124

Unit instance configuration

Unit configuration includes:

- Specifying a name for the unit.
- Creating unit attribute tags
- Assigning a data server.
- Specifying any equipment resources required by the unit during operation.

See also

Open the edit unit dialog box on page 126

Configure the general data for a unit on page 127

Configure unit attribute tags data on page 128

Configure arbitration data for a unit on page 129

Configure cross invocation data for a unit on page 131

Open the Edit Unit dialog box

Unit configuration is performed using the Edit Unit dialog box.

To open the Edit Unit dialog box:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. In the Design View area, select the unit icon and then select Edit Properties ( ). The Edit Unit dialog box displays with the unit defaults and these five tabs:

- General
- Attribute Tags
- Arbitration
- Cross Invocation
- Hyperlink (eProcedure only)

See also

Configure the general data for a unit on page 127
Configure the general data for a unit

Use these instructions to configure the general data for a unit.

To configure the general data for a unit:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. In the Design View area, select the unit icon and then select Edit > Properties.

3. In the Edit Unit dialog box, select General tab.

4. In Name, type the resource name.

5. In Class, type the class.

6. (For use with RSBizWare BatchERP only) In ERP Alias, type the SAP R/3 unit name.

7. In Equipment ID, type or accept the default ID.

8. Select the Data Server.

9. Select Apply to save changes and continue, or select OK to return to the Edit Unit dialog box.

See also

Configure unit attribute tags data on page 128
Configure arbitration data for a unit on page 129
Configure cross invocation data for a unit on page 131
Configure hyperlink data for a unit on page 132

Open the Edit Unit dialog box on page 126
Edit Unit dialog box - General tab on page 127
Edit Unit dialog box - General tab

The Edit Unit dialog box General tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the process cell. (The default name is PROCESS_CELL#. The # is a consecutive number starting with 1 that is assigned to each process cell as it is added to an area.)</td>
</tr>
<tr>
<td>Class</td>
<td>The name of the unit class.</td>
</tr>
<tr>
<td>ERP Alias</td>
<td>(RSBizWare BatchERP only) The SAP name for the phase. The ERP alias maps the equipment module from FactoryTalk Batch to SAP.</td>
</tr>
<tr>
<td>Equipment ID</td>
<td>The equipment identification number identifies this process cell. (A default ID automatically assigns to the process cell, based on the next available sequential number. The first item added to an area is assigned ID #1.) To request this process cell, the equipment ID must be the same as the equipment ID used by the phase logic.</td>
</tr>
</tbody>
</table>

See also

Configure the general data for a unit on page 127

Configure unit attribute tags data

Use these instructions to configure the unit attribute tag data.

To configure unit attribute tags data:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. In the Design View area, select the unit icon and then select Edit > Properties.

3. In the Edit Unit dialog box, select the Attribute Tags tab.

4. Under Unit Attribute Tags, select Add and create the required unit attribute tags.
Tip: If any global unit attributes have been created for the area model, their associated tags are also listed.

5. To edit an existing tag, select the tag in the Unit Attribute Tags area and select Edit. The Edit Unit Attribute Tag dialog box opens.

Tip: Unit attribute tags can only be assigned to OPC or RSLinx Enterprise FactoryTalk Live Data servers. They cannot be assigned to Logix5000 CIP data servers.

6. To remove an existing tag, select the tag in the Unit Attribute Tags area and select Remove.

Tip: It is not an option to cancel the removal of the unit attribute tag. However, select Cancel and any changes made to the unit definition are undone.

7. Select Apply to save changes.

See also

Edit Unit dialog box - Attribute Tags tab on page 129

Create a unit attribute tag on page 140

The Edit Unit dialog box Attribute Tags tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The unique name assigned to this unit (view-only).</td>
</tr>
<tr>
<td>Class</td>
<td>The unit class name (view-only). The icon associated with the unit class displays next to Name and Class.</td>
</tr>
<tr>
<td>Unit Attribute Tags</td>
<td>Process-connected device data associated with the unit that is used with recipes. Unit attribute tags are organized by unit attributes. Unit attribute tag instances are created at the time the unit is configured.</td>
</tr>
</tbody>
</table>

See also

Configure unit attribute tags data on page 128

Configure arbitration data for a unit

In the Equipment Allocation by Resource area, the left pane is a hierarchical display of the current equipment configuration of the area model. The tree has two roots: Process Cells and Resources. The right pane is the list of equipment this unit is dependent on. Any equipment in this list must be acquired by the unit before any procedural element linked to this unit can begin execution.

Expand Process Cells to view all the process cells within the area model. From here, organization is hierarchical, so process cells contain units and units contain phases.
Expand **Resources** to view a list of all the configured resources within the area model.

**Tip:** Units do not have a **Max Owners** box to define because a unit can execute only one procedural element (operation or unit-procedure level recipe) at a time. Therefore, all units can have only one owner.

To configure arbitration data for a unit:

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. In the **Design View** area, select the unit icon and then select **Edit > Properties**.

3. In the **Edit Unit** dialog box, select the **Arbitration** tab.

4. Browse the area models equipment hierarchy to specify equipment the unit must acquire. The lack of a plus or minus sign to the left of an equipment type indicates that no equipment of that type has been defined, or that the equipment type cannot be used as a needed equipment item at this level.

   - To add equipment, under the **Equipment Allocation by Resource** area, select the equipment needed by the unit and select **Add**. The equipment selected displays in the right pane.
   - To remove equipment, in the right pane, select the equipment to delete and select **Remove**. The equipment selected is removed from the list.

5. Select **Apply** to save changes.

See also

**Edit Unit dialog box - Arbitration tab** on page 131
The **Edit Unit** dialog box **Arbitration** tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The unique name assigned to this unit (view-only).</td>
</tr>
<tr>
<td>Class</td>
<td>The name of the class of the unit (view-only).</td>
</tr>
<tr>
<td>Class Icon</td>
<td>The unit class icon (view-only).</td>
</tr>
<tr>
<td>Max Owners</td>
<td>An equipment list requiring ownership by this unit in order for any procedural element linked to the unit to begin execution.</td>
</tr>
<tr>
<td>Equipment Allocation by Resource</td>
<td>A list of the equipment for which ownership is required by this unit to begin execution.</td>
</tr>
</tbody>
</table>

The left pane is a hierarchical display showing the current equipment configuration of the area model. The hierarchy has two roots:

- **Process Cells**: Displays all the process cells within the area model. From here, organization is hierarchical: process cells contain units and units contain phases and operations sequences.
- **Resources**: Displays a list of all the configured resources within the area model.

The right pane is the list of equipment this unit is dependent upon. Any equipment in this list must be acquired by the unit before the unit itself can be acquired.

See also

*Configure arbitration data for a unit* on page 129

**Configure cross invocation data for a unit**

Use these instructions to configure cross invocation data.

To configure cross invocation data for a unit:

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. In the **Design View** area, select the unit icon and then select **Edit > Properties**.

3. In the **Edit Unit** dialog box, select the **Cross Invocation** tab.
4. In the Cross Invocation Label boxes, type the cross invocation string information.

5. Select Apply to save changes.

See also

Edit Unit dialog box - Cross Invocation tab on page 132

The Edit Unit dialog box Cross Invocation tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The unique name assigned to this unit (view-only).</td>
</tr>
<tr>
<td>Class</td>
<td>The name of the class of the unit (view-only).</td>
</tr>
<tr>
<td>Class Icon</td>
<td>The unit class icon (view-only).</td>
</tr>
<tr>
<td>Cross Invocation Configuration</td>
<td>The cross invocation strings indicate the addition of an item to FactoryTalk Batch View and the ActiveX control shortcut menus. This includes the caption for each menu item and the data to be passed to the specified automation server when the menu item is selected. The default labels associated with each of the five fields are Cross Invocation Label # (where # is 1 through 5). These labels can be modified in the Server Options dialog box, and are seen in the Edit Process Cell dialog box.</td>
</tr>
</tbody>
</table>

See also

Configure cross invocation data for a unit on page 131

Configure hyperlink data for a unit

This feature is used in conjunction with FactoryTalk eProcedure only. Use these instructions to configure hyperlink data.

To configure hyperlink data for a unit:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. In the Design View area, select the unit icon and then select Edit > Properties.
3. If hyperlinks are being used, in the **Edit Unit** dialog box, select the **Hyperlink** tab.

![Edit Unit dialog box - Hyperlink tab](image)

4. In the **Hyperlink Configuration** boxes, type a valid URL string.

5. Select **Apply** to save changes.

6. Select **OK** to return to the **Design View** area.

**See also**

*Configure hyperlink data for a unit on page 132*

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The unique name assigned to this unit (view-only).</td>
</tr>
<tr>
<td>Class</td>
<td>The name of the unit class (view-only).</td>
</tr>
<tr>
<td>Class Icon</td>
<td>The unit class icon (view-only).</td>
</tr>
<tr>
<td>Hyperlink Configuration</td>
<td>The URL associated with each hyperlink label.</td>
</tr>
<tr>
<td></td>
<td>The default labels associated with each of the five fields are <strong>Hyperlink Label # English</strong> (where # is 1 through 5). These labels can be modified in the <strong>Server Options</strong> dialog box. The labels are used in the eProcedure Client Equipment view.</td>
</tr>
</tbody>
</table>

**See also**

*Edit Unit dialog box - Hyperlink tab on page 133*
Remove a unit instance

Use these instructions to remove a unit instance.

To remove a unit instance:

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. Select the unit to be removed and press the **Delete** key.

   A warning displays that when the unit is deleted, the phase instances it contains are deleted.

3. Select **Yes** to delete the unit, or select **No** to cancel the deletion and return to FactoryTalk Batch Equipment Editor.

See also

- Link unit instances on page 134
- Share units on page 134

Link unit instances

Links define the upstream and downstream relationships between units within a process cell. The FactoryTalk Batch Server uses the links to determine which units are available for selection when defining equipment requirements for procedures.

To link unit instances:

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. Select **Link Units**. The cursor changes to +.

3. Select and hold the mouse button on the source unit icon. Drag the cursor to the destination unit icon, and release the mouse button.

4. Repeat step 2 until all units link as needed.

See also

- Share units on page 134

Share units

When more than one process cell requires a single unit instance, it must be shared by all process cells that require it. Create the unit instance in one process cell. This unit instance is shared by the other process cell(s). Do not create a new instance of the same unit in the other process cells. Before sharing a unit instance, verify that the unit can physically be shared.
A unit may have only one owner at a time. Sharing a single unit allows any of the shared process cells to gain ownership of it during batch execution. But the unit cannot be owned by more than one process cell at the same time.

When a unit instance is shared, any changes made to it are reflected in each process cell that contains the shared unit instance, including its physical placement in the Design View area. When deleted an instance of a shared unit from one process cell, the unit instance is removed from the selected process cell, but it is still present in any other process cells to which it belongs.

To share units:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Create the unit to share.

3. Move to the process cell that does not contain this unit.

4. Select Share. The Share Unit Between Process Cells dialog box displays with a list of existing process cells.

   - A plus sign (+) to the left of the process cell indicates defined units within that process cell. The lack of a plus sign or minus sign to the left of a process cell indicates that no defined units are within that process cell.

   - Display the units in each process cell by selecting the plus sign (+) for the appropriate process cell folder.

5. Select the unit to share, and select OK.

The Share Unit Between Process Cells dialog box closes. The icon of the selected unit, with a small hand at the bottom left corner, displays in both process cells.
Tip: Deleting an instance of a shared unit only deletes the link to that unit. The unit is not deleted until the last instance of the unit is deleted.

See also

[Link unit instances on page 134](#)
Tags overview

FactoryTalk Batch uses tags to communicate with the process-connected device (PCD). For OPC and RSLinx Enterprise FactoryTalk Live Data phases, define tags in the area model equipment database. Area model tags point to a specific memory address or tag in the PCD. The FactoryTalk Batch Server and the PCD exchange information and communicate through these tags.

**Tip:** For PhaseManager phases assigned to a Logix5000 CIP data server, do not define tags. The tags create automatically when the area model synchronizes with the Logix Designer project.

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.

A phase uses phase tags to communicate with the engineered logic in the PCD. Every phase has a minimum of ten standard tags with which it is associated. When defining the phase class from which an OPC or RSLinx Enterprise FactoryTalk Live Data phase is created, indicate the number of tags required for parameter and report communications, as well as additional tags for requests from the PCD to the FactoryTalk Batch Server.

See also

- Unit attribute tags on page 138
- Create a unit attribute tag on page 140
- Phase tags on page 143
- Global unit attribute on page 146
Unit attribute tags

Unit attribute tag instances create at the time of unit instance configuration. Unit attributes (such as LEVEL_PV) organize Unit attribute tag instances (such as LI101_PV). Each unit attribute tag instance links to a specific tag in the process-connected device (PCD).

**Important:** Each unit can have only one unit attribute tag instance of each unit attribute.

A defined unit attribute tag instance is part of the area model database and can be used in recipe transition conditions for any recipe. In other words, the unit attribute tag instance can be used in a recipe that is bound to any unit in the area model (not just the unit for which the unit attribute tag was defined).

**Tip:** Prior to creating the unit attribute tags, compile a list of data servers and tag addresses to which FactoryTalk Batch connects. Create unit attribute tags that have an OPC or RSLinx Enterprise FactoryTalk Live Data server as their data source. Create unit attribute tags that represent tag information from a Logix5000 CIP data server.

**See also**

- Add a unit attribute on page 138
- Edit a unit attribute on page 140
- Delete a unit attribute on page 140
- Create a unit attribute tag on page 140

**Add a unit attribute**

Use these instructions to add a unit attribute.

**To add a unit attribute:**

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select Edit > Unit Attributes.
3. Select Add.

4. In **Name**, type a name for the new attribute.

5. From the **Type** list, select the appropriate tag type.

6. If the tag type is Enumeration, select the appropriate enumeration name from the **Enumeration** list.

7. Type the **Engineering Units** used for the tag.

8. (optional) Type a **Description** of the attribute.

9. Select **OK** to return to the **Edit Unit Attributes** dialog box.

10. Continue adding unit attributes, or select **OK** to exit. All changes are saved.

**See also**

- Unit attribute tags on page 138
- Edit a unit attribute on page 140
- Delete a unit attribute on page 140
- Create a unit attribute tag on page 140
**Edit a unit attribute**

Use these instructions to edit a unit attribute.

**Important:** Deletion or modification of global unit attributes may invalidate existing global unit binding expressions. Therefore, all global unit binding expressions configured in the area model are revalidated when OK is selected. If any expression is found to be invalid, open the Global Unit Binding Requirements dialog box to correct them.

To edit a unit attribute:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.
2. Select Edit > Unit Attributes.
3. Under Attributes, select an attribute and select Edit.
4. On the Edit Unit Attribute dialog box, modify as needed and then select OK to return to the Edit Unit Attributes dialog box.

See also

- [Delete a unit attribute](#) on page 140
- [Create a unit attribute tag](#) on page 140

**Delete a unit attribute**

Use these instructions to delete a unit attribute.

To delete a unit attribute:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.
2. Select Edit > Unit Attributes.
3. Under Attributes, select an attribute and select Remove.
4. To delete all tags and the selected attribute, select Yes.
5. To cancel the removal, select No.

See also

- [Create a unit attribute tag](#) on page 140
Create a unit attribute tag

Use these instructions to create a unit attribute tag.

To create a unit attribute tag:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. In the Design View area, right-click the appropriate unit.

3. Select Attribute Tags tab.

4. Under Unit Attribute Tags, select Add.

   • If a unit attribute exists, the Create Unit Attribute Tag dialog box opens. Otherwise, a prompt to create a unit attribute displays. If so, select Yes, and follow instructions in Add a unit attribute.

5. In Name, type the unit attribute tag name.

6. From Attribute, select the appropriate unit attribute.

7. From Type, select the appropriate tag type.

   If the tag type is Enumeration, from Enumeration, select the appropriate enumeration name.

8. In Engineering Unit, type the engineering units used for the tag.

9. If Static is selected, in Value, type the unit attribute tag value.

   Tip: The value of a dynamic unit attribute tag is held in the PCD. The value of a static unit attribute tag is held in the FactoryTalk Batch Server.

10. If Dynamic is selected:

   a. From Data Server, select the appropriate data server. The corresponding data server type automatically appears in Server Type.
Tip: Only OPC and RSLinx Enterprise FactoryTalk Live Data servers appear in the selection list. Logix5000 CIP data servers are not displayed.

b. In Access Path, type the appropriate access path.

c. In Item Name, type the appropriate item name.

d. If the data server requires separate read and write items, in Write Access Path, type the write access path.

e. If the data server requires separate read and write items, in Write Item Name, type the write item name.

11. Select OK to return to the Attribute Tags tab.

See also

Unit Attribute Tags dialog box on page 142

Add a unit attribute on page 138

Unit Attribute Tags dialog box

A unit attribute tag represents the value of a unit attribute for a specific unit. In previous releases of FactoryTalk Batch, unit attribute tags were known as unit tags.

When creating a unit attribute tag, these items are configured in the Create Unit Attribute Tags dialog box.

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Tag name for the unit attribute. Follow the naming conventions.</td>
</tr>
<tr>
<td>Attribute</td>
<td>Lists the unit attribute name options for the unit attribute tag.</td>
</tr>
<tr>
<td>Type</td>
<td>Lists the data type selections for the unit attribute tag. Valid types include:</td>
</tr>
<tr>
<td>Enumeration</td>
<td>Lists the appropriate enumeration selections. This list is only enabled if the tag type is Enumeration.</td>
</tr>
<tr>
<td>Engineering Unit</td>
<td>The engineering units associated with this tag.</td>
</tr>
<tr>
<td>Data Source</td>
<td>Defines the type of data source, Static or Dynamic.</td>
</tr>
<tr>
<td></td>
<td>• If data source is Static, the data is a constant value. The value entered here depends on the engineering units assigned and associates with a particular unit.</td>
</tr>
<tr>
<td></td>
<td>• If data source is Dynamic, the value comes from the PCD, and the data server information must be defined.</td>
</tr>
</tbody>
</table>
### Name | Purpose
--- | ---
**Data Server** | Displays the data server from which the tag is acquiring data. This list includes all configured data servers for the active area. Only OPC and RSLinx Enterprise FactoryTalk Live Data servers display in the selection list. Logix5000 CIP data servers are not displayed.

**Server Type** | Displays the server type that corresponds with the selected data server as defined in the data server definition (DSDF) files (view-only).

**Access Path** | The access path used with the read item when attempting to subscribe to the tag value or when performing read operations. This box changes to **Read Item Access Path** when the associated data server requires separate read and write items. This is disabled if the data server does not require an access path.

**Item Name** | The name of the item used when subscribing to a tag value or when performing read operations. This box changes to **Read Item Name** when the associated data server requires separate read and write items.

**Write Access Path** | The OPC access path for the write item. This disables if the data server does not require separate read and write items.

**Write Item Name** | The item name in the write items process-connected device. This disables if the data server does not require separate read and write items or does not require an access path.

### See also

*Create a unit attribute tag on page 140*

### Phase tags

The number of tags associated with an OPC or RSLinx Enterprise FactoryTalk Live Data phase depends on the phase class configuration from which the phase is created. Every phase has a minimum of ten standard tags. When defining the phase class from which the OPC or RSLinx Enterprise FactoryTalk Live Data phase is created, define additional tags required for parameter and report communication as well as additional request tags for requests from the process-connected device (PCD) to the FactoryTalk Batch Server.

A Request tag is one of the ten standard tags associated with every phase and is used for the PCD to request service from the server. Occasionally, the PCD needs to communicate more information to the server than can be accommodated by the single standard Request tag. Specify, on the phase class, the number of additional request tags needed for the phase.

If configured the phase class to require any of these additional tags, create corresponding tags in the OPC or RSLinx Enterprise FactoryTalk Live Data phase.
Tip: For PhaseManager phases assigned to a Logix5000 CIP data server, do not configure additional request tags. All required tags automatically create for Logix5000 CIP phases when the area model synchronizes with the Logix Designer project. FactoryTalk Batch Equipment Editor creates the same number of parameter tags as parameters that exist on the phase class, and the same number of report tags as reports that exist on the phase class.

See also

Edit tags from the menu on page 144

Default equipment names on page 145

Edit tags from the menu

The Edit Tags dialog box lists all of the tag classes configured for the selected area model, along with the associated tags.

To edit tags from the menu:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select Edit > Tags.

3. Under Tag Classes, select the appropriate tag class.

4. Under Tags, select the tag to edit

5. Select Edit Tag. The Edit [TagType] Tag dialog box displays, where [TagType] represents the type of tag being edited (phase or unit).

Tip: Instruction-based tags are view-only.

6. Modify as needed and then select OK.

See also

Phase tags on page 143
When creating a phase from a phase class in the area model, the default phase name is <phasename>1. For example, if the phase is AGITATE, the phase creates from this phase class with the default name of AGITATE1.

The phase’s tag items the default name is: <Equipment Module Name>< Tag Separator><Tag Extension>

For example, if the tag separator is a period (.), the Request tag (RQ) for phase AGITATE1 would be: AGITATE1.RQ

If the phase name changes, the tag item names do not automatically change to reflect the new phase name. For example, if the phase name changes from AGITATE1 to AGITATE_U2, the tag Names reflect the new name (AGITATE_U2), but the tag Item Names still reflect the original name of the phases (AGITATE1). The example request tag for the AGITATE_U2 phase would still be AGITATE1.RQ.

When a phase creates from a phase class in the area model, the tag access path is the Default Item Access Path specified for the assigned data server. Change the Access Path for an individual tag on the Edit Phase Tag dialog box.

See also

Set all tag items to default name and access path on page 145
Edit tags from the menu on page 144

Set all tag items to default name and access path

Use these instructions to globally change all the tag Item Names in the area model to reflect the names of their respective phases and reset the Access Path of each phase tag to the Default Item Access Path specified for its data server.

Tip: To change the tag names for only a single phase, change them on the Tags tab of the phase.

To set all tag items to default name and access path:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select Edit > Set Default Tag Items.

   This is an irreversible action! Once selected, all tag item names and access paths are automatically changed.

3. Select OK to continue.

   A message displays indicating all tag items have been set to the tag name and access paths have been reset.
4. Select **OK** to return to the **Design View** area.

**Tip:** The default character (tag separator) that separates the phase tag item name from its extension may vary depending on the tag separator defined in the `batcheqp.ini` file. The available characters are the underscore ( _ ), slash ( / ), or period ( . ). If changed, the default character is not overridden.

See also

- Set tag items to the default addresses on page 190
- Phase overview on page 181

**Global unit attribute**

FactoryTalk Batch associates units within unit classes that represent the common functionality of all the units in the class. This allows for the construction of class-based recipes, built against a unit class(es). This aggregation of common functionality allows for the authoring of recipes that can run against any instance of the unit class (with the exception of flowpath requirements and FactoryTalk Batch Material Manager functionality).

However, the set of functionality common across all instances of a unit class may be so small as to not allow for the creation of useful class-based recipe structures. There is likely to be significant commonality among some reactors that does NOT extend to the entire set. For example, some subset of reactors may contain agitators. Other subset of reactors may contain temperature sensors.

Also, there may be attributes of a unit, such as materials of construction, or temperature, that make it acceptable, unacceptable, desirable, or undesirable for use with certain recipes. The batch product has no mechanism for specification or enforcement of these requirements.

The ability to build class-based recipes that utilize subsets of commonality would be extremely useful in this case. Global unit attributes provides the ability to build class-based recipes that utilize subsets of commonality. A global unit attribute is a unit attribute configured to be automatically applied to all unit classes defined within the area model. This saves the area model author the effort of having to manually assign a custom unit attribute to every unit class.

See also

- Create a global unit attribute on page 147
- Edit a global unit attribute on page 148
- Delete a global unit attribute on page 149
- Remove a global unit attribute on page 149
Create a global unit attribute

Use these instructions to create a global unit attribute.

To create a global unit attribute:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select Edit > Unit Attributes. When an attribute is selected, it populates with the associated attribute tags. The first attribute tag is selected by default.

3. Under Attributes, select Add to open the Create Unit Attribute dialog box.

4. In Name, type a name for the new unit attribute.

5. From Type, select the appropriate tag type.

   - If the tag type is Enumeration, select the appropriate enumeration name from the Enumeration list.

6. In Engineering Units, type the engineering units used for the tag.

7. In Description, type a description of the tag class.
8. To be a global unit attribute that is applied to all new units, select **Global Attribute**.

**Tip:** If **Global Attribute** is selected, all unit classes within the area model, both existing and newly created, are associated with that unit attribute. In addition, all unit instances within the area model, both existing and those subsequently created, contain a unit attribute tag instance.

9. Select **OK** to return to the **Edit Unit Attributes** dialog box.

The new attribute is listed under **Attributes**. If it is a global unit attribute, it is associated with all existing unit classes. Attributes and attribute tags are upper-case and in alphabetic order.

---

**See also**

[Edit a global unit attribute](#) on page 148

---

**Edit a global unit attribute**

Use these instructions to edit a unit attribute.

**To edit a global unit attribute:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. Select **Edit > Unit Attributes**.

3. Under **Attributes**, select an attribute and select **Edit**.
4. Make the required modifications, and select **OK** to return to the **Edit Unit Attributes** dialog box.

See also

- **Delete a global unit attribute** on page 149
- **Remove a global unit attribute** on page 149

## Delete a global unit attribute

Use these instructions to delete a unit attribute.

### To delete a global unit attribute:

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. Select **Edit > Unit Attributes**.

3. Under **Attributes**, select a unit attribute and select **Remove**.

   A warning displays to verify the removal of all the unit attribute tags of the unit attribute.

4. To delete all attribute tags and the selected unit attribute, select **Yes**.

5. To cancel the removal, select **No**.

See also

- **Remove a global unit attribute** on page 149

## Remove a global unit attribute

Use these instructions to remove a global unit attribute.

### To remove a global unit attribute:

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. Select **Edit > Unit Attributes**.

3. Under **Attributes**, select the unit attribute that to edit and select **Edit**.

4. Clear the **Global Attribute** check box, and then select **OK**.

5. To remove the unit attribute, select **Yes**.

6. To cancel the removal, select **No**.
See also

Delete a global unit attribute on page 149

Create a global unit attribute on page 147
Chapter 15

Phase class overview

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.

Tip: For PhaseManager phases assigned to a Logix5000 CIP data server, do not define the number of parameter, report, and request tags. The required tags automatically configure when the area model synchronizes with the Logix Designer.

Every OPC or RSLinx Enterprise FactoryTalk Live Data phase automatically assigns ten standard tags to communicate with the engineered logic in the PCD. When defining the phase class from which an OPC or RSLinx Enterprise FactoryTalk Live Data phase is created, indicate the number of tags required for parameter and report communications, as well as additional tags for requests from the PCD to the FactoryTalk Batch Server.

See also

- Tags on page 152
- Create and configure phase classes on page 152
- Verification policies on page 169
- Material-enabled phases on page 176
- Disable material on a material-enabled phase on page 178
Tags

One of the ten standard tags associated with every OPC or RSLinx Enterprise FactoryTalk Live Data phase is a Request tag. The process-connected device (PCD) uses the Request tag to request service from the FactoryTalk Batch Server. Sometimes, for OPC and RSLinx Enterprise FactoryTalk Live Data phases, the PCD needs to communicate more information to the FactoryTalk Batch Server than can be accommodated by the single standard Request tag. For this purpose, specify on the phase class, the number of additional request tags needed for the phase.

Tip: For PhaseManager phases assigned to a Logix5000 CIP data server, do not configure request tags. The tags automatically create for Logix5000 CIP phases the area model synchronizes with the Logix Designer. FactoryTalk Batch Equipment Editor creates the same number of parameter tags as parameters that exist on the phase class, and the same number of report tags as reports that exist on the phase class.

See also

Configure general attributes on page 153

Create and configure phase classes

Use the Create Phase Class dialog box to create and configure phase classes.

To create and configure phase classes:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select a unit icon and then select Go Down (▼). The Design View area may display the Unit icon, and Classes displays as the column header in the Classes View area.

3. Select Add New Object (▼) and then select New Phase Class.

4. In the Create Phase Class dialog box, use the tabs to configure the phase class:

   - General
   - Parameters
   - Reports
   - Messages
   - Control Strategy (this tab is only available if the Control Strategy box is enabled on the General tab)

5. Select OK. The new phase class displays in the phase class list.
Tip: To edit the phase class, right-click the phase class to open the Edit Phase Class dialog box.

See also

Configure phase class general attributes on page 153

Configure phase class control strategies on page 166

Configure phase class messages on page 165

Configure phase class parameters on page 155

Configure phase class reports on page 160

Configure phase class general attributes

Use the Create Phase Class dialog box General tab to configure the phase class general attributes.

To configure phase class general attributes:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. With the Create Phase Class dialog box open, select the General tab.

3. In Name, type an appropriate phase class name.

4. In Type, type a phase class type.

   Tip: Use names that can easily be understood. Phase class names display to the operator during batch execution.

   a. Under Icon, select an appropriate icon. (If the icon area is empty, see Verify icon paths.)

   b. Under Additional Functionality, do the following:

      • Select Control Strategy to indicate whether control strategies are used for the phase class.

      • (optional) If FactoryTalk Batch Material Manager is installed, select Material Based Recipes to indicate whether the phase class is material-enabled.

   c. Under Parameter/Tag Locking, enable or disable locking for parameters and tags.

      • To edit the number of parameter and report tags associated with the phases created from this phase class, disable Parameter/Tag Locking. This option only applies to OPC and RSLinx Enterprise
FactoryTalk Live Data phases created from this phase class. For Logix5000 CIP phases, Parameter/Tag Locking is always enabled, regardless of the setting selected.

- If Parameter/Tag Locking is enabled, the Parameter and Report boxes are disabled and display the number currently enforced by Parameter/Tag Locking.
- Under Number of Tags, edit the number of tags in Parameter and Report and enter the appropriate number of tags in Request.

5. Select Apply to save changes, or select OK to save changes and return to the Classes View area.

The selected icon displays in the Classes View area.

See also

Create Phase Class dialog box - General tab overview on page 154

Verify icon paths on page 119

Create Phase Class dialog box - General tab

The phase class general attributes include the phase class name, phase class type, the number of phase class message partners, and the number of tags associated with each phase. Select an icon to graphically represent all phases throughout the area model based on the phase class. Enable or disable the use of control strategies and material-based recipes for the phase class.

The General tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Icon area</td>
<td>Displays all of the icons available for selection. Scroll through the icons using the scroll bar or the up and down arrows in the vertical scroll bar.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the phase class.</td>
</tr>
<tr>
<td>Type</td>
<td>An optional box that can be used to categorize phase classes for reporting purposes.</td>
</tr>
<tr>
<td>Control Strategy</td>
<td>Enables the use of separate control strategies for the phase class.</td>
</tr>
<tr>
<td>Material Based Recipes</td>
<td>(Used with FactoryTalk Batch Material Manager.) Enables the binding of material references to actual equipment. This makes the phase class a material-enabled phase class.</td>
</tr>
</tbody>
</table>
**Phase class overview**

**Parameter/Tag Locking**

To have the number of parameters and reports on the phase class to always equal the number of parameter tags and report tags on the associated phase classes, enable **Parameter/Tag Locking** on the phase class.

If **Parameter/Tag Locking** is disabled, specify on the phase class the number of tags for the parameters and reports that exist on the associated OPC or RSLinx Enterprise FactoryTalk Live Data phases. This number may be different from the number of defined phase class parameters and reports 1for OPC and RSLinx Enterprise FactoryTalk Live Data phases.

<table>
<thead>
<tr>
<th><strong>Number of Parameter Tags</strong></th>
<th>The number of memory registers within the phase to which FactoryTalk Batch can write parameter information. Generally equivalent to the number of parameters defined for the phase class.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Number of Report Tags</strong></td>
<td>The number of memory registers within the phase from which FactoryTalk Batch can read report tag information. Generally equivalent to the number of reports defined for the phase class.</td>
</tr>
<tr>
<td><strong>Number of Request Tags</strong></td>
<td>The number of memory registers within the phase to which the phase logic can write request information.</td>
</tr>
</tbody>
</table>
| **Number of Message Partners** | Message partners are two or more phases that need to communicate with each other during the execution of their respective phase logic for synchronization and timing purposes. In FactoryTalk Batch Recipe Editor, message partners are organized into Link Groups that inform the FactoryTalk Batch Server which phases require messaging.

If the phase is in a phase link group for communication or synchronization purposes, this is the number of additional phases with which the phase must be linked in order to operate properly.

**Tip:** Define the message partners in FactoryTalk Batch Equipment Editor before defining the corresponding phase link groups in FactoryTalk Batch Recipe Editor.

---

**See also**

- Configure general attributes on page 153

---

**Configure phase class parameters**

Use the **Create Phase Class** dialog box **Parameters** tab to configure phase class parameters.

**Tip:** Use the keyboard to move around in the grid. Pressing the **Tab** key moves focus to the grid. Pressing **Ctrl+Alt+Home** highlights the first editable cell in the grid.

**To configure phase class parameters:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.
2. With the Create Phase Class dialog box open, select the Parameters tab. A list displays of parameters associated with the phase class.

   Tip: If control strategies are enabled for the phase class, the Control Strategy list displays at the top of the Parameters tab. Only phase class parameters associated with the control strategy selected in the Control Strategy list are listed. If <All> is selected, then the High, Low, Default, or Enum/E.U. boxes are not editable for the phase class parameters. The material parameters, including the optional material parameters (if enabled), are added to each control strategy created. Edit the values for the MATERIAL, MATERIAL_CLASS and AMOUNT material parameters for each configured control strategy.

3. Select Add Parameter to add a new phase class parameter. A new parameter displays with default values.

4. Under Name, type a unique name for the parameter.

5. Under Type, select the appropriate parameter type.

   - If the Type is Real or Integer, add the High value, Low value, and Default value, then type the engineering units in the Enum/E.U. box.

   - If the Type is Enumeration, choose the appropriate enumeration set from the Enum/E.U. list and select the appropriate default enumeration from the Default list.

   - If the Type is String, type the engineering units in the Enum/E.U. box and the default string in the Default box.

6. If the parameter must scale within a recipe when a batch scales, select the Scale check box. (See Scale parameters for more information.)

   Tip: The DL on Start and DL on TOC check boxes are unavailable if not enough parameter tags are specified in Number of Parameter Tags on the General tab. Specify as many parameter tags as actual parameters to have these check boxes available for each parameter.

7. To add the parameter to the download on start subset, select the DL on Start check box.

8. To add the parameter to the download on transfer of control subset, select the DL on TOC check box.

   Tip: Material-enabled phase classes do not support transfer of control.

9. To assign a context or descriptor to the phase class parameter for the recipe, select one or more Context IDs from the Reporting Contexts column.

10. To require electronic signatures or generate deviation events in the FactoryTalk Batch event journal when deviations from the accepted range
of parameter values occur, see Configure the verification policy for parameters and reports for instructions.

11. To delete a parameter, select the parameter row header and select Delete Parameter. Select Yes to confirm the deletion.

12. Select Apply to save changes, or select OK to save changes and return to the Classes View area.

See also

Create Phase Class dialog box - Parameters tab overview on page 157

Scale parameters on page 46

Select context IDs on page 164

Configure the verification policy for parameters and reports on page 171

Create Phase Class dialog box - Parameters tab

Phase class parameters values pass from the FactoryTalk Batch Server to the phase logic during the execution of a recipe phase. For PhaseManager phases, which use a Logix5000 CIP data server, the phase definition includes its parameters. For other data servers, a phase parameter is defined for the phase class, and a corresponding parameter tag and tag address is configured in the process-connected device.

The verification methods of parameter values can be configured for real or integer phase parameters. The verification methods define the policies for verifying up to three sets of high and low limits of the parameter. They also define the actions to be taken if the parameter value falls outside the configured limits.

If Material Based Recipes is enabled, two parameters, MATERIAL and AMOUNT, can be defined. Add these optional parameters for material-enabled phases: CONTAINER, LOT, LABEL, and MATERIAL_CLASS.

Tip: If the FactoryTalk Batch Material Manager material database is unavailable, the MATERIALS and CONTAINERS enumeration sets are read from the Material Server when the FactoryTalk Batch Server starts up, and only contain the default enumerations.

The Parameters tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control Strategy</td>
<td>A list of configured control strategies used to filter the displayed parameters. Separate control strategies can be selected from the list or all control strategies can be viewed at the same time. This list is only visible if Control Strategy is enabled.</td>
</tr>
</tbody>
</table>
### Add Optional Material Parameters

Enables the optional material-based recipe parameters (CONTAINER, MATERIAL_CLASS, LOT, and LABEL). This check box is only visible if **Material Based Recipes** is enabled. This check box only enables if <All> is selected in the **Control Strategy** list.

### Name

The name of the phase class parameter. The name of the material parameters (MATERIAL, AMOUNT, CONTAINER, LOT, or LABEL) cannot be edited.

### ID

The identification number of the phase class parameter. If **Control Strategy** is enabled after parameters have been defined, the IDs are increased by one, as the control strategy itself always has the parameter ID of 1. For material-enabled phases, parameter ID fills in the first available integer greater than zero. The MATERIAL parameter is the first of the **Material Based Recipe** parameters to be assigned an ID (view-only).

### Type

The phase parameter data type: Real, Integer, String, and Enumeration. The data type for material-enabled phase parameters cannot be modified:

- MATERIAL, MATERIAL_CLASS and CONTAINER = ENUM
- AMOUNT = REAL
- LOT and LABEL = STRING

### Max

The maximum limit for the phase parameter. Defaults to 100.

### Min

The minimum limit for the phase parameter. Defaults to zero (0).

- Addition: Low = 0, High = positive number
- Distribution: Low = negative number, High = 0
- Both: Low = negative number, High = positive number

**Tip:** For the material-enabled phase parameter AMOUNT, the Min and Max values specify whether the phase is a material addition, distribution, or both.

### Default

The default value for the phase parameter. Defaults to zero (0). The value must be set between the Min and Max values. If a **Verification Method** of High/Low, High-High/Low-Low, or High-High-High/Low-Low-Low is selected, the default value must be within the high and low values.

For material-enabled phases, these are the defaults:

- MATERIAL defaults to NULL_MATERIAL
- MATERIAL_CLASS defaults to NULL_CLASS
- CONTAINER defaults to NULL_CONTAINER

The default value for the CONTAINER, LOT, or LABEL parameters cannot be modified.

### Enum/E.U.

The engineering units for the phase parameter. If the parameter **Type** is **Enumeration**, the Enum/E.U. value is the name of the enumeration set.

**Tip:** The Enum/E.U. for the MATERIAL, CONTAINER, LOT, LABEL, or MATERIAL_CLASS material-enabled phase parameters cannot be modified.
## Scale
Indicates whether the phase parameter is scalable with a recipe at batch creation time.

## DL on Start
If selected, the parameter can be downloaded by the FactoryTalk Batch Server in these situations:
- When the phase logic requests the download.
- When the phase is configured to automatically download on the START command. See Phase overview for more information.

## DL on TOC
If selected, the parameters are downloaded by the FactoryTalk Batch Server when a transfer of control takes place on the phase, when the phase logic requests the download.
Material-enabled phase classes do not support transfer of control.

## Reporting Contexts
Reporting contexts allows defining associations between a parameter and a set of context IDs. The defined context IDs are recorded by the FactoryTalk Batch Server in the Context ID box of the batch event journal records, whenever a parameter event logs. The default value for a new parameter is a null string. Select browse (browse icon) to open the Reporting Contexts dialog box, to select one or more context strings to be used for the particular parameter. Context IDs are members of the REPORTING_CONTEXTS enumeration set.

## Verification Method
The part of the verification policy that defines the set of acceptable limits for a parameter or report. The phase parameter type must be Real or Integer.

- **No Limits**: To configure a verification policy to enact when the parameter or report value is within the Minimum and Maximum limits. Also select this option to not use verification policies.
- **High/Low**: Allows configuration of up to two verification policies. In addition to the No Limits policy, specify a High and Low limit and a separate verification policy to use when the parameter or report value exceeds the High limit or falls short of the Low limit.
- **High-High/Low-Low**: Allows configuration of up to three verification policies. In addition to the No Limits policy and the High/Low policy, specify a second higher limit, a second lower limit, and a separate verification policy to use when the parameter or report value exceeds this second High limit or falls short of this second Low limit.
- **High-High-High/Low-Low-Low**: Allows configuration of up to four verification policies. In addition to the No Limits policy, the High/Low policy, and the High-High/Low-Low policy, specify a third higher limit, a third lower limit, and a separate verification policy to use when the parameter or report value exceeds this third High limit or falls short of this third Low limit.

## Add Parameter
Adds a new parameter to the list.

## Delete Parameter
Removes the selected parameter from the list.

### See also
Configure phase parameters on page 155
Configure phase class reports

Use the Create Phase Class dialog box Reports tab to configure phase class reports. A phase class report is defined for the phase class, while the corresponding report tag, including the address or tag in the process-connected device, is defined for the phase.

To configure phase class reports:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. With the Create Phase Class dialog box open, select the Reports tab. A list of phase class reports associated with the phase class is shown.

   Tip: Use the keyboard to move around in the grid. Pressing the Tab key moves the focus in the grid. Pressing Control+Alt+Home highlights the first editable cell in the grid.

3. Select Add Report to add a new phase class report.

   A new report displays with default values.

   a. Under Name, type a unique name for the report parameter.

   b. Under Type, select the appropriate parameter type.

      • If Type is REAL, STRING, or INTEGER, type the engineering units in Enum/E.U. If control strategies are enabled, edit the Enum/E.U. value for each configured control strategy.

      • If Type is ENUM, select the appropriate enumeration set in Enum/E.U.

      Tip: If there are not enough report tags specified on the General tab in the Number of Report Tags box, the UL on Terminal State and UL on TOC check boxes are unavailable. Specify as many report tags as actual reports to have these check boxes enabled for each report.

   c. Under Accumulate, select how the FactoryTalk Batch Server handles uploaded report values. These are the valid values:

      • None
      • Phase
      • Batch

   d. To add the report to the upload on terminal state subset, select UL on Terminal State.
c. To add the report to the upload on transfer of control subset, select UL on TOC.

**Tip:** Material-enabled phase classes do not support transfer of control.

f. To assign a context or descriptor to the report parameter, from the Reporting Contexts column, select one or more Context IDs.

g. To require electronic signatures or generate deviation events in the FactoryTalk Batch event journal when deviations from the accepted range of parameter values occur, see **Configure the verification policy for parameters and reports** for instructions.

h. (optional) In ERP Alias, type the SAP R/3 report name.

4. To delete a report, select the report row header and select **Delete Report**. Select Yes to confirm the deletion.

5. Select **Apply** to save changes, or select **OK** to save changes and return to the Classes View area.

See also

- **Configure the verification policy for parameters and reports** on page 171
- **Create Phase Class dialog box - Reports tab overview** on page 161
- **Select context IDs** on page 164

**Create Phase Class dialog box - Reports tab**

Phase class reports values pass from the process-connected device (PCD) to the FactoryTalk Batch Server. Report values contain data that indicates how the phase functioned. They are included in the event journal file (.*evt*) for the running batch and can be included in batch reports. For PhaseManager phases, which use a Logix5000 CIP data server, the phase definition includes its reports. For other data servers, a phase report is defined for the phase class, and a corresponding parameter tag and tag address is configured in the PCD.

The verification methods of report values can be configured for real or integer phase class reports. The verification methods define the policies for verifying up to three sets of high and low limits of the report. They also define the actions to be taken if the report value falls outside the configured limits.

If Material Based Recipes is enabled, two phase class reports (ACTUAL_AMOUNT and FEED_COMPLETE) are created automatically.
Tip: If control strategies are configured for the phase class, the Control Strategy list displays at the top of the Reports tab. Only reports associated with the control strategy selected from the Control Strategy list are displayed. If <All> is selected and the Type is Real, String, or Integer, then the Enum/E.U. box cannot be edited. This must be set for each control strategy.

The material-enabled phase reports are added to each control strategy created. Edit the Enum/E.U. value for each configured control strategy.

The Reports tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Control Strategy</td>
<td>A list of configured control strategies used to filter the displayed phase class reports. Separate control strategies can be selected from the list or all control strategies can be viewed at the same time. This list is only available if Control Strategy has been enabled on the General tab.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the phase class reports. <strong>Tip:</strong> The name of the material phase class reports ACTUAL_AMOUNT and FEED_COMPLETE cannot be edited.</td>
</tr>
<tr>
<td>ID</td>
<td>The identification number of the phase class reports. For material-enabled phase classes, parameter ID fills in the first available integer greater than zero. The ACTUAL_AMOUNT report is the first of the Material Based Recipe reports to be assigned an ID (view-only).</td>
</tr>
<tr>
<td>Type</td>
<td>The report data type: Real, Integer, String, and Enumeration. <strong>Tip:</strong> The data type for material-enabled phase reports ACTUAL_AMOUNT = REAL and FEED_COMPLETE = ENUM cannot be modified.</td>
</tr>
<tr>
<td>Enum/E.U.</td>
<td>If Enumeration Type is selected, shows the name of the enumeration set. Otherwise, shows the engineering units for the report parameter. <strong>Tip:</strong> The Enum/E.U. for the FEED_COMPLETE material-enabled phase class report cannot be modified.</td>
</tr>
<tr>
<td><strong>Accumulate</strong></td>
<td>Defines the phase class report parameter to add the value of uploaded values to its current value instead of just storing it.</td>
</tr>
<tr>
<td>--------------</td>
<td>--------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td><strong>None</strong>: The report parameter acts as a normal output parameter on the phase. When the phase logic uploads a value to FactoryTalk Batch Server using either the automatic upload or by PXREQ / Request (2000 series), the value is stored in the report and logged in the event journal. If the phase logic uploads multiple times on a single run of the phase, each value is recorded in the event journal and the last uploaded value displays in the report parameter.</td>
<td></td>
</tr>
<tr>
<td><strong>PHASE</strong>: The report value accumulates the value in the report every time the phase logic uploads a value to the report for the entire phase execution. The event journal records each uploaded value as a report increment and records the totaled value for the report value. The type of upload does not matter.</td>
<td></td>
</tr>
<tr>
<td><strong>BATCH</strong>: The report value accumulates the value in the report every time the phase logic uploads a value to the report for the entire batch. The event journal records each uploaded value as a report increment and records the totaled value for the report value. If the phase step executes again in the same batch, as in a loop, the value continues to totalize. The type of upload does not matter.</td>
<td></td>
</tr>
<tr>
<td>Changing the <strong>Accumulate</strong> setting also changes the <strong>UL on Terminal State</strong> setting:</td>
<td></td>
</tr>
<tr>
<td>If the <strong>Accumulate</strong> value is set to <strong>None</strong> (Default), the <strong>UL on Terminal State</strong> value is on and enabled.</td>
<td></td>
</tr>
<tr>
<td>If the <strong>Accumulate</strong> value is set to <strong>PHASE</strong>, the <strong>UL on Terminal State</strong> value is off and disabled.</td>
<td></td>
</tr>
<tr>
<td>If the <strong>Accumulate</strong> value is set to <strong>BATCH</strong>, the <strong>UL on Terminal State</strong> value is off and disabled.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>UL on Terminal State</strong></th>
<th>If selected, the reports are uploaded by the FactoryTalk Batch Server in these situations:</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>• When the phase logic requests the upload.</td>
</tr>
<tr>
<td></td>
<td>• When the phase is configured to automatically upload when the step changes to COMPLETE, STOPPED, or ABORTED. (See Create a phase for more information.)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><strong>UL on TOC</strong></th>
<th>If selected, the report can be uploaded when a transfer of control takes place on the phase, if the proper phase logic request is issued.</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Material-enabled phase classes do not support transfer of control.</td>
</tr>
</tbody>
</table>

| **Reporting Contexts** | Reporting context IDs allows defining of associations between a phase report and a set of context IDs. The defined context IDs are recorded by the FactoryTalk Batch Server in the Context ID box of the batch event journal records, whenever a report event logs. Select browse ( ) to open the Reporting Contexts dialog box, select one or more context strings to be used for the particular report. |

| **ERP Alias** | Used in conjunction with RSBizWare BatchERP only. The ERP alias is the SAP name for that report. The ERP alias maps the report from FactoryTalk Batch Server to SAP. |
Verification Method

The part of the verification policy that defines the set of acceptable limits for a parameter or report. The phase class parameter or report type must be **Real** or **Integer**.

**No Limits:** Allows configuration of a verification policy to enact when the parameter or report value is within the verification limits. Also select this option to not use verification policies.

**High/Low:** Allows configuration of up to two verification policies. In addition to the **No Limits** policy, specify a **High** and **Low** limit and a separate verification policy to use when the parameter or report value exceeds the **High** limit or falls short of the **Low** limit.

**High-High/Low-Low:** Allows configuration of up to three verification policies. In addition to the **No Limits** policy and the **High/Low** policy, specify a second higher limit, a second lower limit, and a separate verification policy to use when the parameter or report value exceeds this second **High** limit or falls short of this second **Low** limit.

**High-High-High/Low-Low-Low:** Allows configuration of up to four verification policies. In addition to the **No Limits** policy, the **High/Low** policy, and the **High-High/Low-Low** policy, specify a third higher limit, a third lower limit, and a separate verification policy to use when the parameter or report value exceeds this third **High** limit or falls short of this third **Low** limit.

<table>
<thead>
<tr>
<th>Add Report</th>
<th>Deletes a new report to the phase class.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Delete Report</td>
<td>Removes the selected report from the phase class.</td>
</tr>
</tbody>
</table>

**See also**

- **Create and configure phase classes** on page 152
- **Configure phase class reports** on page 160

**Select context IDs**

Context IDs, or descriptors, can be associated with each phase class report or parameter so that events recorded in the event journal may be more easily queried and sorted. This association makes report generation and analysis easier.

To assign a context ID to phase class parameters or reports, select one or more context IDs in the **Create Phase Class** dialog box.

**Important:** Context IDs are only assigned to recipe parameters and report parameters configured on a phase class. Thus, only phase class parameters may have Context IDs and they may not be changed within recipes.

To select context IDs:

1. With the **Create Phase Class** dialog box open, select the **Parameters** or **Reports** tab. A list of phase class parameters or reports associated with the phase class is shown.
Configure phase class messages

Use these instructions to add new messages, edit existing messages, and delete phase class messages from the list in the Create Phase Class dialog box Messages tab.

To configure phase class messages:

1. Select **Start**, point to **All Programs** > **Rockwell Software** > **FactoryTalk Batch Suite** > **FactoryTalk Batch**, and then select **Equipment Editor**.

2. With the Create Phase Class dialog box open, select the **Messages** tab. A list of messages associated with the phase class is shown.

3. Select **Add Message** to add a new phase class message. A new default message is added.

4. In the **Name** box, type the appropriate message.
5. In the ID box, type a message ID.

6. To delete a message, select the message row header, and then select Delete Message. Select Yes to confirm the deletion.

7. Select Apply to save changes, or select OK to save changes and return to the Classes View area.

See also

Create Phase Class dialog box - Message tab overview on page 166

Create Phase Class dialog box - Messages tab

Phase class messages are typically used for journal entry purposes and for troubleshooting phase logic problems. In some instances, the message displays to the operator. A phase class message has two pieces of data associated with it: the name and the message ID. Phase class message requests must be generated from within the phase logic (a 30NN or 130NN request) and a corresponding phase class message must be created in the area model for each phase class message request.

The Messages tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the message. This text also serves as the actual phase class message.</td>
</tr>
<tr>
<td>ID</td>
<td>The identification number of the report message.</td>
</tr>
<tr>
<td>Add Message</td>
<td>Adds a new phase class message to the phase class.</td>
</tr>
<tr>
<td>Delete Message</td>
<td>Removes the selected phase class message from the phase class.</td>
</tr>
</tbody>
</table>

See also

Configure phase class messages on page 165

Configure phase class control strategies

Use these instructions to edit the default control strategy, create new control strategies, and delete control strategies. To remove all of the control strategies, disable the Control Strategy option on the Create Phase Class dialog box General tab.

Tip: The Control Strategy tab is not visible unless the Control Strategy check box is selected in the Create Phase Class dialog box General tab.
Tip: Disabling control strategies for a phase class results in a warning message indicating the parameter information that is lost if control strategies are disabled. Confirmation is required to complete the process.

To configure phase class control strategies:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. With the Create Phase Class dialog box open, select the Control Strategy tab. A list displays of control strategies associated with the phase class.

3. Select Add Control Strategy to add a new control strategy.

4. On the Create Enumeration dialog box, in the Enumeration box, type a name for the new control strategy.

5. In the Ordinal box, type the ordinal identification value for the control strategy, or accept the default value and select OK.

   The new control strategy is added.

6. For each parameter in the Parameters table, select or clear the parameter’s check box within the column for the new control strategy.

   Important: If the target parameter is removed from the same control strategy as the report, the verification method for the report is automatically reset to No Limits.

7. For each report in the Reports table, select or clear the corresponding check box in the column for the new control strategy.

8. From the Set Default Control Strategy list, select the appropriate control strategy to configure that control strategy as the default for the phase class.

9. To edit or delete a control strategy, select the control strategy column and select Edit Control Strategy or Delete Control Strategy.

10. Select Apply to save changes, or select OK to save changes and return to the Classes View area.

See also

Create Phase Class dialog box - Control Strategy tab overview on page 167
Create Phase Class dialog box - Control Strategy tab

A control strategy identifies a subset of the phase class and report parameters configured on a phase class to be used at runtime. A control strategy value is assigned to the phase class and only the parameters and reports of that control strategy are available for use. A phase class can be configured to have one or more control strategies.

Individually configure each control strategy with a subset of the defined phase class parameters and reports. Define the value, range, and engineering unit for each parameter, and the engineering unit for each report independently for each control strategy. When creating master recipes using FactoryTalk Batch Recipe Editor, specify which control strategy to use for each instance of the phase class within the recipe.

If Material Based Recipes and Add Optional Material Parameters are enabled, the material parameters and reports for a material-enabled phase class are added to every control strategy created and cannot be deleted.

When the control strategy function is enabled, a default control strategy is created.

The Control Strategy tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set Default Control Strategy</td>
<td>A list of configured control strategies for the phase class. The selected control strategy is the default control strategy for any recipes based on the phase class.</td>
</tr>
<tr>
<td>Parameters</td>
<td>A table used to assign parameters to control strategies. Each row of the table represents an associated parameter, and each column of the table represents a different control strategy, as identified by the column header. Parameters can be assigned or unassigned to a control strategy by selecting or clearing the check box for the parameter in the appropriate control strategy column. A single parameter can be assigned to multiple control strategies.</td>
</tr>
<tr>
<td>Reports</td>
<td>A table used to assign reports to control strategies. Each row of the table represents an associated report, and each column of the table represents a different control strategy, as identified by the column header. Reports can be assigned to or removed from a control strategy by selecting or clearing the check box for the report in the appropriate control strategy column. A single report may be assigned to multiple control strategies.</td>
</tr>
<tr>
<td>Add Control Strategy</td>
<td>Adds a new control strategy to the phase class.</td>
</tr>
<tr>
<td>Edit Control Strategy</td>
<td>Edits the selected control strategy. Select the column header to select a control strategy.</td>
</tr>
<tr>
<td>Delete Control Strategy</td>
<td>Removes the selected control strategy from the phase class.</td>
</tr>
</tbody>
</table>
See also

Configure control strategies on page 166

Verification policies

Configure a separate verification policy for each parameter and report designated as a Real or Integer type.

Tip:

A verification policy for String or Enumeration parameters or reports cannot be configured. The Verification Policies dialog box still displays for Strings and Enumerations, but all boxes are view-only.

The Verification Method consists of four verification levels:

- **No Limits**: The selected policy is enacted when the parameter or report value is within the deviation limits.
- **High/Low**: The options selected in this area describe the actions to be taken when the parameter or report value is outside the calculated high-low deviation limits.
- **High-High/Low-Low**: The options selected in this area describe the actions to be taken when the parameter or report value is outside the calculated high-high/low-low deviation limits.
- **High-High-High/Low-Low-Low**: The options selected in this area describe the actions to be taken when the parameter or report value is outside the defined high-high-high/low-low-low limits.

See also

How verification policies work on page 169

Configure the verification policy for parameters and reports on page 171

How verification policies work with control strategies on page 175

How editing target parameters affects report verification policies on page 176

How verification policies work

When choosing a verification method, specify a calculation method used in determining the deviation limits. For each verification level, select a verification policy option:

Tip:

To use the signature template policy, first configure signature templates.

- **None**: A deviation event writes to the FactoryTalk Batch event journal when the parameter or report value is outside the calculated deviation limits, but no signature is required.
- **Signature Template**: Requires selection of a pre-configured signature template that describes the number of signoffs required and the user and groups permitted to enter an electronic signature.

- **Not Allowed**: If the parameter or report value is outside the calculated range, stops the batch from continuing its processing.

The verification method selected, along with the options selected within an area, affect the options available in the subsequently higher verification level areas. For example, the **No Limits** verification method is selected, any options for the **High/Low**, **High-High/Low-Low** or **High-High-High/Low-Low-Low** verification methods cannot be configured.

Verification options cannot be defined in a higher verification level less stringent than those defined in a lower verification level. For example, if the **High-High/Low-Low** verification method is selected, and the **Signature Template** option for the policy when the value is within limits is selected, the **None** option is disabled for the policy when the value exceeds the high/low level.

**Tip**: For recipe parameters, the highest and lowest limits must be within the minimum and maximum values specified for the parameter in the **Edit Phase** dialog box **Parameters** tab. This does not apply to reports. The default parameter value specified on the **Parameters** tab must be within the range of the low and high limits.

The sets of high and low limits must conform to this relationship:

\[
\text{Maximum} \geq \text{HHH} > \text{HH} > \text{H} = \text{Default} = \text{L} > \text{LL} > \text{LLL} \geq \text{Minimum}
\]
Configure the verification policy for parameters and reports

See also

Signature template on page 57

Configure the verification policy for parameters and reports on page 171

Use the Edit Phase Class dialog box to configure the verification policies for parameters and reports.

Before you begin:

- Verify set up of the electronic signature templates.

Tip: The phase class parameter or report type must be Real or Integer. A Verification Method for String or Enumeration parameters or reports cannot be configured.

To configure the verification policy for parameters and reports:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. In the Edit Phase Class dialog box, select the Parameters or Reports tab to configure a verification policy for either a parameter or report, respectively.

3. In the parameter or report row header, select the Verification Method browse to open the Verification Policies dialog box.

4. From the Verification Method list, select the verification method used for the parameter or report. The selected method determines which boxes in the dialog box become enabled.

- If the High/Low, High-High/Low-Low, or the High-High-High/Low-Low-Low verification method is selected, specify a calculation method. From the Calculation list, select the calculation method that is used to determine the deviation limits. (Absolute is the only available selection for parameters. If selected the No Limits verification method, the Calculation list is disabled. Continue with Step 5.)

- If the Absolute calculation method is selected, there is an option of selecting the Scale check box. The Scale option scales the absolute verification limits the same percentage as the Batch Scale value specified on the Batch Creation dialog box in the FactoryTalk Batch View.

- If the Percent or Relative calculation method is selected, select a parameter from the Percent/Relative Target Parameter list. The parameter value selected is used in the percentage or relative calculation that determines the report verification limit. The list only displays real
or integer recipe parameters assigned to the same control strategy as the report on the phase class being edited.

**Important:** Changes to the target parameter could automatically alter the verification policy of all referencing reports.

5. In the **Policy when value is within limits** area, select one of the policy options.

   - If **Signature Template** is selected, select browse () next to the **Signature Template**.
   - From the **Select Template** list, select the signature template that defines the signature policy to use for this parameter or report verification limit.
   - Select **OK**. The **Select Signature** dialog box closes, and the signature template chosen displays in the box next to the **Signature Template** option.

6. Repeat step 4 for each policy area enabled. In addition to these steps, for each enabled policy, enter values in each respective set of **Low** and **High** boxes that are used in calculating the verification limit.

   **Important:** If a parameter or report type is changed from Real to Integer, and the calculation method for the verification policy is Absolute or Relative, the values in each set of Low and High limits are automatically rounded to the nearest integer.

7. Select **OK** to save changes and return to the **Edit Phase Class** dialog box.

8. To download the verification limits to the phase logic, enable or define a set of limit tags. After setting up verification policies, enable or define these limit tags when configuring each phase for automatic (non-instruction based) phase classes.

**See also**

- **Signature template** on page 57
- **How editing target parameters affects report verification policies** on page 176
- **Verification Policies dialog box** on page 173
- **Parameter limit tags** on page 197
- **Report limit tags** on page 199
The **Verification Policies** dialog box contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| Verification Method | A list that allows selection of **Verification Methods**:  
|                     | **No Limits**: Define the actions taken when the parameter or report value is within the verification limits. To not use verification policies, select this option.  
|                     | **High/Low**: Define two verification policies. Specify a **High** and **Low** limit and a separate verification policy to use when the parameter or report value exceeds the **High** limit or falls short of the **Low** limit.  
|                     | **High-High/Low-Low**: Configure up to three verification policies. Specify a second higher limit, a second lower limit, and a separate verification policy to use when the parameter or report value exceeds this second **High** limit or falls short of this second **Low** limit.  
|                     | **High-High-High/Low-Low-Low**: Configure up to four verification policies. Specify a third higher limit, a third lower limit, and a separate verification policy to use when the parameter or report value exceeds this third **High** limit or falls short of this third **Low** limit. |
## Calculation

Verification limit calculation. Only enabled for reports. Select the type of equation used to calculate the verification limit for reports. The FactoryTalk Batch Server calculates the limits when the batch is added to the batch list or when the target parameter changes.

Example configuration settings:
- **Target Parameter** = 120
- **High Limit** = 10
- **Low Limit** = -10

**Absolute**: The deviation limit is the specified value defined in the recipe. The absolute values entered in the three sets of **Low** and **High** boxes are displayed as the default values for the **LLL**, **LL**, **L**, **H**, **HH**, **HHH** boxes in the FactoryTalk Batch Recipe Editor **Parameter Value Entry/Report Limit Entry** dialog box. This is the default and only type of limit calculation for recipe parameters.

At runtime, the Server calculates:
- **High limit** = 10
- **Low limit** = -10

**Percentage**: The Target Parameter value is multiplied by the percentage specified in the **Low** and **High** boxes. Only available for reports. For example:

At runtime, the FactoryTalk Batch Server calculates:
- **High Limit** = 120 + (.10)(120) = 132
- **Low Limit** = 120 + (-.10)(120) = 108

**Relative**: The Target Parameter value plus the value specified in the **Low** and **High** boxes. Only available for reports. For example:

At runtime, the Server calculates:
- **High Limit** = 120 + (10) = 130
- **Low Limit** = 120 + (-10) = 110

## Scale

Enabled only when the Calculation is Absolute. Indicates whether the absolute verification limits are scaled the same percentage as the Batch Scale value specified in the FactoryTalk Batch View **Batch Creation** dialog box.

## Percent/Relative Target Parameter

Enabled when the calculation is Percentage and Relative of report limits. Select the parameter whose value is to be used in the percentage or relative calculation that determines the report verification limit.

The list only displays real or integer recipe parameters assigned to the same control strategy as the report on the phase class being edited.

## None

A deviation event is recorded in the FactoryTalk Batch event journal, but no other action occurs (default).
### Signature Template
Indicates that a signature verification is required when parameter or report values fall outside of the specified range.

If selecting this option, select browse ( ) to open the Select Signature Template dialog box. Select the signature template that contains the Signoff groups who have the security privileges to enter signature verifications at this verification level.

### Not Allowed
Indicates the recipe cannot proceed when the values for the parameter or report are outside the defined limits. No event is recorded in the event journal.

<table>
<thead>
<tr>
<th>Low/High</th>
<th>Low-Low/High-High</th>
<th>Low-Low-Low/ High-High-High</th>
</tr>
</thead>
<tbody>
<tr>
<td>These boxes are used in calculating the default parameter deviation limits. These values can be changed in the recipe, but if no values are provided in the recipe parameters, the values entered here are used. The definition of this box differs depending on the Calculation method selected.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Absolute calculation: The actual values entered are the Low and High deviation limits. Deviation Limit = Absolute value.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Percent calculation: These values define the percentage of the Target Parameter used in calculating the Low and High deviation limits. These values are the Percentage in this equation: Deviation Limit = Target Parameter + (Target Parameter * Percentage). Enter positive and negative values. To define the Low limit as lower than the Target Parameter value, enter a negative value in the Low box.</td>
<td></td>
<td></td>
</tr>
<tr>
<td>• Relative calculation: The actual value entered is added to the Target Parameter value to determine the deviation limit. These values are the Relative value in this equation: Deviation Limit = Target Parameter + (Target Parameter + Relative value). Enter positive and negative values. To define the Low limit as lower than the Target Parameter value, enter a negative value in the Low box.</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

The selected policy is enacted when the parameter or report values are outside of the calculated deviation limits.

### See also
Configure the verification policy for parameters and reports on page 171

### How verification policies work with control strategies
If implementing control strategies, set up a different verification policy for each parameter and report in each control strategy.

If selected Percent or Relative as the calculation method for a report’s verification policy, select a different target parameter for each control strategy. The target parameter must be assigned to the same control strategy as the report. When enabled control strategies, the target parameter list only displays integer and real parameters assigned to the same control strategy as the report.
### Important:
If a report’s target parameters are removed from the same control strategy as the report, the verification method automatically resets to No Limits.

See also

*How editing target parameters affects report verification policies* on page 176

#### How editing target parameters affects report verification policies

If any of these changes are made to a parameter, the verification method automatically resets to No Limits for all reports that reference that parameter as the target parameter:

- Parameter name changes.
- Parameter type changes from Integer or Real to String or Enumeration.
- Removed Parameter from the same control strategy as a report.
- Deleted Parameter.

**Important:** If a report verification policy is set up with the Percent or Relative calculation method, changes to the target parameter could automatically alter the verification policy of the referencing report.

See also

*How verification policies work with control strategies* on page 175

### Material-enabled phases

If using FactoryTalk Batch Material Manager in conjunction with FactoryTalk Batch, two parameters (MATERIAL and AMOUNT) and two reports (ACTUAL_AMOUNT and FEED_COMPLETE) are automatically added to each material-enabled phase class.

Add optional parameters for material-enabled phase classes:

- CONTAINER
- LOT
- MATERIAL_CLASS
- LABEL

These optional parameters make data available to the phase class and to the client programs as the batch runs.

**Tip:** The optional material parameter values within FactoryTalk Batch Equipment Editor cannot be edited.
Because material-enabled phase classes are different from standard phase classes, be aware of these issues when configuring a phase class to be material-enabled:

- What to do with existing parameters and report parameters.
- What to do about the number of tags configured on the phase class.

If a standard phase class does not have matching parameter and report names reserved for materials, making this phase class material-enabled is quite simple. Once the **Material Based Recipes** check box is enabled, the two reserved parameters and two reserved report parameters are added. The same applies to **Add Optional Material Parameters**.

If disabled **Parameter/Tag Locking**, add the appropriate number of tags.

**See also**

- [Phase class overview](#) on page 151
- [Verification policies](#) on page 169
- [Disable material on a material-enabled phase](#) on page 178
- [Duplicate a phase](#) on page 178

If a standard phase class uses any of the reserved material parameter or material report names, they must be of the correct data type for a material-enabled phase class. In addition, if the MATERIAL parameter exists, it must also have the correct enumeration set selected.

**Tip:** The exceptions to this are the optional Material Parameters. The four parameters, CONTAINER, LOT, MATERIAL_CLASS, and LABEL, cannot already exist as non-system parameters. If the system finds parameters matching the reserved names, delete or rename the parameters before enabling the **Add Optional Material Parameters**.

If the data types of the existing parameters are not exactly what the material-based recipes subsystem expects, enabling a phase class to support materials could result in issues. The system checks for correct data types and displays a message to correct those with an invalid data type.

The parameter and report IDs previously configured remain unchanged. The other material-based recipe parameters and reports that have not been configured are added and assigned IDs using the first-available integer greater than zero.

**See also**

- [Standard phase class with control strategies configured](#) on page 178
Standard phase class with control strategies configured

If a standard phase class has control strategies configured and Material Based Recipes is enabled, the material parameters (including optional parameters, if enabled) and reports are added to every control strategy. If disabled Parameter/Tag Locking, add the appropriate number of tags.

See also

Material-enabled phase classes on page 176

Disable material on a material-enabled phase class

Disabling Material Based Recipes on a material-enabled phase class automatically deletes the two reserved parameters (and the optional parameters, if enabled) and the two reports. If Parameter/Tag Locking is enabled, remove the appropriate number of tags.

If multiple phases are created using the material-enabled phase classes, delete the extra phases in each unit. Then, disable the Material Based Recipes on the phase class.

See also

Material-enabled phase classes on page 176

Duplicate a phase class

Phase classes can be cut, copied, or pasted within the Classes View area using the FactoryTalk Batch Equipment Editor toolbar. Duplicate phase classes within an area model, including all of the phase parameters, reports, and messages. Use this feature to cut or copy a phase class from one area model to another.

See also

Copy a phase class on page 178
Cut and paste a phase class on page 179

Copy a phase class

Use these instructions to copy a phase class.

To copy a phase class:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.
2. From the Classes View area, select the phase class to be copied.
3. Select Edit > Copy Phase Class.
4. Select Edit > Paste Phase Class. A new phase class creates with the default name PHASE CLASS NAME#, where # is an integer (beginning with 1) that increments each pasted phase class.
See also

Cut and paste a phase class on page 179

Use these instructions to cut and paste a phase class.

To cut and paste a phase class:

Tip: To cut and paste between area models, open the first area model, cut or copy the appropriate phase class, and then open the second area model and paste the phase class into the Classes View area.

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. From the Class View area, select the phase class to be cut.

3. From the Edit > Cut Phase Class. The phase class is removed from the Classes View area and held in memory.

4. Select any phase class within the Classes View area and, select Edit > Paste Phase Class.

See also

Copy a phase class on page 178
Chapter 16

Configure phases

A phase is an instance of a specific phase class. Only one instance of a specific
phase class can be added to each unit, unless the phase class is material-enabled.
Once a phase class instance is added to a unit instance, the phase must be
configured.

A phase maps to the phase class in the engineered logic via tag addresses. If a single
instance of this phase class is required by multiple units, it may be shared between
the units. At a minimum, each tag in OPC or RSLinx Enterprise FactoryTalk Live
Data phases must be configured to indicate the tag's address location in the
process-connected device.

Tip: For PhaseManager phases assigned to a Logix5000 CIP data server, it is
not necessary to configure tag data.

Phase tags represent data associated with a specific phase during the execution of a
recipe. The number of tags associated with a phase depends on the phase class
configuration. Each phase has a minimum of ten pre-defined tags with which it is
associated. Additional tags may be configured. There are three types of additional
tags:

- Parameter
- Report
- Request

If the phase class is configured to require any of these additional tags, a
corresponding tag creates in the phase.

When using material-enabled phase classes, add multiple instances of the same
phase class to a single unit instance. Even though these phases are created from the
same phase class, they are distinguished from each other by the unique material
containers with which they are associated. Phases mapping to the same phase class
have independent resource IDs.
Tip: When a phase is selected in the Design View, the phase class on which it is based is highlighted. If the phase class on which they are based is modified, phases are automatically updated. Phases created prior to modifying the phase class are also updated.

When the same phase is required by multiple units, it can be shared between the units. However, a phase can only be owned by one unit at a time.

See also

Create a phase on page 182

Create a phase

Use these instructions to create a phase.

Tip: If using eProcedure, allow plenty of space between phase icons in the Design View area, as the eProcedure Equipment View displays the icons in the same location. In the eProcedure Equipment View, each equipment icon displays run-time data and up to five custom hyperlinks. If the icons are spaced too closely together, this information may overlap, making it difficult to see the data.

To create a phase:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Navigate to the phase class level of the area model (select Go Down ( ) to the lowest level).

3. Select the desired phase class icon in the Classes View area.

4. Place the cursor in the Design View area where the phase is to be located and select.

The Edit Phase dialog box opens. The Edit Phase dialog box contains these tabs:

- General
- Arbitration
- Cross Invocation
- Tags
- Hyperlink (eProcedure only)
- Containers (Material Manager only)
- Parameter Limit Tags
- Report Limit Tags

5. To add multiple instances of the same phase to the same unit instance, select again. Only material-enabled phase classes can be used for multiple phases.

See also

Configure the general data for a phase on page 183
Configure the general data for a phase

Use these instructions to configure the Edit Phase dialog box General tab.

To configure the general data for a phase:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select the Edit Phase dialog box General tab.

3. In the Name box, type the phase name.

4. (For use with RSBizWare BatchERP only) In ERP Alias, type the SAP R/3 phase name.

5. In Equipment ID, type the equipment ID, or accept the default value.

6. In Data Server, confirm that the correct data server that corresponds to the phase displays. If it does not, select the list to locate the appropriate server.

   Tip: If the data server does not correspond to the phase, a message displays stating that continuing resets the assignment to <None>.

7. (For use with FactoryTalk eProcedure only) Select browse ( ) next to the Instruction File box. The Instruction Selection dialog box opens.

8. Select the instruction file that applies to the phase. The instructions display in the Instructions area.

   Important: Define an instruction file for each instruction-based phase. If an instruction-based phase has no instruction file associated with its configuration, the eProcedure Server will not start.

9. Select OK to return to the Edit Phase dialog box.
10. To automatically download the appropriate parameters on a START command, enable the **Automatically Download Parameters Upon START** check box.

**Tip:** Automatic upload/download are configurable on a phase only when the associated phase class parameters and reports have at least one member of the upload/download parameter subsets. These features are disabled when an Instruction Based Server is selected.

11. To automatically upload the appropriate report values when the batch transitions to a terminal state (COMPLETE, STOPPED, or ABORTED), enable all or individual **Automatically Upload Parameters Upon** check boxes.

12. Select **Apply** to save changes.

See also

- *Edit Phase dialog box - General tab* on page 184
- *Automatic Upload/Download* on page 52

**Edit Phase dialog box - General tab**

The **Edit Phase** dialog box **General** tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the phase.</td>
</tr>
<tr>
<td>Phase</td>
<td>The name of the phase class from which this phase was created (view-only). The icon associated with the phase class is displayed next to the <strong>Name</strong> and <strong>Phase Class</strong>.</td>
</tr>
<tr>
<td>ERP Alias</td>
<td>(RSBizWare BatchERP only) The SAP name for the phase. The ERP alias maps the equipment module from FactoryTalk Batch to SAP.</td>
</tr>
<tr>
<td>Equipment ID</td>
<td>A unique number identifies this phase. It must be the same ID used in the phase logic to request this phase. When <strong>Dynamic Unit Allocation</strong> is enabled and the <strong>First Available</strong> method is selected, the FactoryTalk Batch Server performs a final ascending sort on this box and chooses equipment with the smallest ID number that meets the required criteria.</td>
</tr>
<tr>
<td>Data Server</td>
<td>Displays the currently assigned server. When the list opens, configured servers display.</td>
</tr>
<tr>
<td>Instruction File</td>
<td>Use this feature in conjunction with eProcedure only. Enables if the data server is not an Instruction Based Server. This maps to the instruction file associated with the phase.</td>
</tr>
</tbody>
</table>
Configure phases

Chapter 16

Configure the tag data for a phase

Configure the tag data for a phase. These instructions only apply to phases assigned to an OPC or RSLinx Enterprise FactoryTalk Live Data server.

Tip: For PhaseManager phases assigned to a Logix5000 CIP data server, it is not necessary to configure tag data.

To configure the tag data for a phase:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. In the Edit Phase dialog box, select the Tags tab.

3. In the On Control and Off Control boxes, type the On Control and Off Control scan rates for the phase.

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
</table>
| Automatically Download Parameters Upon START | When a START command issues, the FactoryTalk Batch Server downloads the parameters configured with the Download on Start option on the phase. Enable this function by selecting the check box. If no parameters exist on the phase, this option is dimmed.  
  - If an OPC or RSLinx Enterprise FactoryTalk Live Data server is assigned to the phase, this option is enabled when the number of parameter tags on the phase class is equal or greater than the number of parameters.  
  - If a Logix5000 CIP data server is assigned to the phase, this option enables when DL on Start is selected for at least one parameter on the phase class.  
  - If the Instruction Based Server is assigned to the phase, this function disables. |
| Automatically Upload Parameters Upon COMPLETE STOPPED ABORTED | When the batch transitions to a terminal state (COMPLETE, STOPPED, or ABORTED) the FactoryTalk Batch Server uploads the report values configured with the Upload on Terminal State option on the phase class. Enable this function so that it occurs for any or all of these terminal states by selecting the check box. If no reports exist on the phase class, these options are dimmed.  
  - If an OPC or RSLinx Enterprise FactoryTalk Live Data server is assigned to the phase, this option enables when the number of report tags on the phase class is equal or greater than the number of reports.  
  - If a Logix5000 CIP data server is assigned to the phase, this option enables when UL on Terminal State is selected for at least one report on the phase class.  
  - If the Instruction Based Server is assigned to the phase, this function disables. |

See also

Configure the general data for a phase on page 183
Configure phases

If the data server associated with the phase is an OPC or RSLinx Enterprise FactoryTalk Live Data server that allows for configurable scan rates, the default scan rates for the phase can be modified. If the values cannot be modified, the boxes disable.

4. Select **Apply** to save changes.

See also

**Edit Phase dialog box - Tag tab** on page 186

In the **Edit Phase** dialog box **Tags** tab, these boxes are available for phases assigned an OPC or RSLinx Enterprise FactoryTalk Live Data server:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Name</strong></td>
<td>The name of the phase (view-only).</td>
</tr>
<tr>
<td><strong>Phase Class</strong></td>
<td>The name of the phase class from which this phase was created (view-only). The icon associated with the phase class is displayed next to the <strong>Name</strong> and <strong>Phase Class</strong>.</td>
</tr>
<tr>
<td><strong>Phase Tags</strong></td>
<td>A list of all tags that must be defined for the phase.</td>
</tr>
<tr>
<td><strong>Defaults</strong></td>
<td>Returns all tag items associated with this phase to their default item names and the default access path that is configured on the data server that is assigned to the phase. The default tag item names include the name of the phase. (See <strong>Set tag items to the default addresses</strong> for more information.)</td>
</tr>
<tr>
<td><strong>Name</strong></td>
<td>The name of the highlighted tag (view-only).</td>
</tr>
<tr>
<td><strong>Type</strong></td>
<td>The phase tag data type (view only). The box below the <strong>Type</strong> box displays comments associated with the default tags. There are no comments associated with the report, parameter, or additional request tags (view-only).</td>
</tr>
<tr>
<td><strong>Class Description</strong></td>
<td>Displays a description of the currently selected tag class.</td>
</tr>
<tr>
<td><strong>On Control Scan Rate</strong></td>
<td>Displays the on control scan rate for the OPC or RSLinx Enterprise FactoryTalk Live Data server associated with the phase (view-only). The scan rate can be modified only within specific data server definition (DSDF) files. Typically, these values do not need to be modified. The valid range of the on control scan rate is <strong>250 to 30,000 milliseconds</strong>.</td>
</tr>
<tr>
<td><strong>Off Control Scan Rate</strong></td>
<td>Displays the off control scan rate for the OPC or RSLinx Enterprise FactoryTalk Live Data server associated with the phase (view-only). The scan rate can be modified only within specific data server definition (DSDF) files. Typically, these values do not need to be modified. The valid range of the off control scan rate is <strong>250 to 60,000 milliseconds</strong>.</td>
</tr>
</tbody>
</table>

See also

**Set tag items to the default addresses** on page 190
Phase tag

Phase tags automatically name and cannot be directly modified. If the phase name changes, the tag name changes. There are ten default tags created for each OPC or RSLinx Enterprise FactoryTalk Live Data phase. Additional tags create based on the number of report, parameter, and request tags specified.

After the phase creates, all phase tags must be configured to include the correct tag item. An item references the address location in which the tag data is stored. The data type for the ten default tags is integer and cannot be modified.

The default data type for any additional tags that have been configured is real. Change the data types for parameter and report tags when they are defined for the phase class. Change the data types for the additional request tags when the phase is edited.

When a phase is added to the area model, select the data server for assignment to each tag. Change the data server assigned to the tags when the phase is edited.

Tip: The data server for a limit tag must be the same as the data server for its associated parameter tag. If the data server for a parameter tag is changed, the data server for the associated limit tags is also automatically changed.

The data server for a limit tag cannot be changed. To change the data server for a limit tag, change the data server for its associated parameter tag.

See also

Phase tag naming conventions on page 187
Configure a phase tag on page 188

Phase tag naming conventions

These naming conventions used for phase tags are based on a phase called CHARGE1. The ## is a sequentially-assigned number for each user-defined parameter, report, and additional request configured on the phase class. Replace the ## with 01.

<table>
<thead>
<tr>
<th>Class Name</th>
<th>Tag Name</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Parameter Tags</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PARMTR##</td>
<td>[Phase_Name]P##</td>
<td>CHARGE1P01</td>
</tr>
<tr>
<td>Parameter Limit Tags</td>
<td></td>
<td></td>
</tr>
<tr>
<td>PARMTR##M</td>
<td>[Phase_Name]P##M</td>
<td>CHARGE1P01M</td>
</tr>
<tr>
<td>PARMTR##L</td>
<td>[Phase_Name]P##L</td>
<td>CHARGE1P01L</td>
</tr>
<tr>
<td>PARMTR##H</td>
<td>[Phase_Name]P##H</td>
<td>CHARGE1P01H</td>
</tr>
<tr>
<td>PARMTR##LL</td>
<td>[Phase_Name]P##LL</td>
<td>CHARGE1P01LL</td>
</tr>
<tr>
<td>PARMTR##HH</td>
<td>[Phase_Name]P##HH</td>
<td>CHARGE1P01HH</td>
</tr>
<tr>
<td>PARMTR##LLL</td>
<td>[Phase_Name]P##LLL</td>
<td>CHARGE1P01LLL</td>
</tr>
</tbody>
</table>
### Configure a phase tag

Use these instructions to configure a phase tag.

**To configure a phase tag:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. From the **Edit Tags** dialog box, double-click the desired phase tag in the **Tags** list box.

   The **Edit Phase Tag** dialog box opens.

---

<table>
<thead>
<tr>
<th>Class Name</th>
<th>Tag Name</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>PARMTR##HHH</td>
<td>[Phase_Name]P##HHH</td>
<td>CHARGE1P01HH</td>
</tr>
</tbody>
</table>

### Report Tags

<table>
<thead>
<tr>
<th>REPORT##</th>
<th>[Phase_Name]R##</th>
<th>CHARGE1R01</th>
</tr>
</thead>
</table>

### Report Limit Tags

<table>
<thead>
<tr>
<th>REPORT##M</th>
<th>[Phase_Name]R##M</th>
<th>CHARGE1R01M</th>
</tr>
</thead>
<tbody>
<tr>
<td>REPORT##L</td>
<td>[Phase_Name]R##L</td>
<td>CHARGE1R01L</td>
</tr>
<tr>
<td>REPORT##H</td>
<td>[Phase_Name]R##H</td>
<td>CHARGE1R01H</td>
</tr>
<tr>
<td>REPORT##LL</td>
<td>[Phase_Name]R##LL</td>
<td>CHARGE1R01LL</td>
</tr>
<tr>
<td>REPORT##HH</td>
<td>[Phase_Name]R##HH</td>
<td>CHARGE1R01HH</td>
</tr>
<tr>
<td>REPORT##LLL</td>
<td>[Phase_Name]R##LLL</td>
<td>CHARGE1R01LLL</td>
</tr>
<tr>
<td>REPORT##HHH</td>
<td>[Phase_Name]R##HHH</td>
<td>CHARGE1R01HHH</td>
</tr>
</tbody>
</table>

### Request Tags

<table>
<thead>
<tr>
<th>REQUEST##</th>
<th>[Phase_Name]Q##</th>
<th>CHARGE1Q01</th>
</tr>
</thead>
</table>

### Default Tags

<table>
<thead>
<tr>
<th>COMMAND</th>
<th>[Phase_Name]_OC</th>
<th>CHARGE1_OC</th>
</tr>
</thead>
<tbody>
<tr>
<td>FAILURE</td>
<td>[Phase_Name]_F</td>
<td>CHARGE1_F</td>
</tr>
<tr>
<td>OWNER</td>
<td>[Phase_Name]_W</td>
<td>CHARGE1_W</td>
</tr>
<tr>
<td>PAUSE</td>
<td>[Phase_Name]_P</td>
<td>CHARGE1_P</td>
</tr>
<tr>
<td>PAUSED</td>
<td>[Phase_Name]_PD</td>
<td>CHARGE1_PD</td>
</tr>
<tr>
<td>REQUEST</td>
<td>[Phase_Name]_RQ</td>
<td>CHARGE1_RQ</td>
</tr>
<tr>
<td>SINGLE_STEP</td>
<td>[Phase_Name]_SS</td>
<td>CHARGE1_SS</td>
</tr>
<tr>
<td>STATUS</td>
<td>[Phase_Name]_ST</td>
<td>CHARGE1_ST</td>
</tr>
<tr>
<td>STEP_INDEX</td>
<td>[Phase_Name]_SI</td>
<td>CHARGE1_SI</td>
</tr>
<tr>
<td>UNIT</td>
<td>[Phase_Name]_UN</td>
<td>CHARGE1_UN</td>
</tr>
</tbody>
</table>

---

See also

[Phase tag on page 187](#)
3. Select an appropriate data type from the **Type** list. OPC or RSLinx Enterprise FactoryTalk Live Data servers do not accept **Enumeration** data types. Use **Integer** instead. If the phase class has enabled **Parameter/Tag Locking**, the type cannot be modified.

4. In **Access Path**, type the appropriate access path.

5. In **Item Name**, type the appropriate item name.

6. If the data server requires separate read and write items, type the write access path in the **Write Access Path** box.

7. If the data server requires separate read and write items, type the write item name in the **Write Item Name** box.

8. From the **Data Server** list, if the default is not correct, select the appropriate data server.

9. Select **OK** to return to the **Edit Tags** dialog box.

10. Select **OK** to close the **Edit Tags** dialog box.

**See also**

*[Edit Phase Tag dialog box on page 189]*

---

**Edit Phase Tag dialog box**

Phase tags are configured using the **Edit Phase Tag** dialog box, which contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The automatically assigned name of the phase tag (view-only).</td>
</tr>
<tr>
<td>Tag Class</td>
<td>The phase tag class (view only).</td>
</tr>
<tr>
<td>Type</td>
<td>The phase tag data type. Parameters and reports defined as enumerations default to <strong>Integer</strong> as the tag type. <strong>Type</strong> should never be set to <strong>Enumeration</strong> because an OPC or RSLinx Enterprise FactoryTalk Live Data server does not recognize this tag type. Parameters and reports defined as enumerations should use <strong>Integer</strong> as the tag type. If the phase <strong>Parameter/Tag Locking</strong> is enabled (selected), the <strong>Type</strong> cannot be modified.</td>
</tr>
<tr>
<td>Enumeration</td>
<td>This should not be used. OPC or RSLinx Enterprise FactoryTalk Live Data servers do not recognize enumeration tag types. Parameters and reports defined as enumerations must use <strong>Integer</strong> as the tag type. If the phase <strong>Parameter/Tag Locking</strong> is enabled (checked), the <strong>Enumeration</strong> cannot be modified.</td>
</tr>
<tr>
<td>Access Path</td>
<td>The OPC access path used with the read item when the data server attempts to subscribe to the tag's value, or when performing read operations. This box changes to <strong>Read Item Access Path</strong> when the associated OPC data server requires separate read and write items.</td>
</tr>
</tbody>
</table>
Set tag items to the default addresses

When creating a phase from a phase class in the area model, the default phase name is <phasename>1. For example, if the phase is AGITATE, the phase creates from this phase class with the default name of AGITATE1.

The phase’s tag items the default name is: <Equipment Module Name>< Tag Separator><Tag Extension>

For example, if the tag separator is a period (.), the Request tag (RQ) for phase AGITATE1 would be: AGITATE1.RQ

If the phase name changes, the tag item names do not automatically change to reflect the new phase name. For example, if the phase name changes from AGITATE1 to AGITATE_U2, the tag Names reflect the new name (AGITATE_U2), but the tag Item Names still reflect the original name of the phases (AGITATE1). The example request tag for the AGITATE_U2 phase would still be AGITATE1.RQ.

Re-establish the default address information for a phase by using the Defaults button. The Item, Read Access Path, Read Item Name, Write Item Path, and Write Item Name are returned to the appropriate default values specified for the data server assigned to the phase. The phase tag names are changed to reflect the current name of the phase.

This is an irreversible action! Once Defaults is selected, all tag item names and access paths are automatically changed. (See Set all tag items to default name and access path for more information.)

To set tag to the default address:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.
Configure phases

Configure the arbitration data for a phase

2. In the **Edit Phase** dialog box **Tags** tab, select **Defaults**.

![](image)

3. Select **Yes** to set the defaults and return to the **Edit Phase** dialog box.

**See also**

- [Set all tag items to default name and access path](#) on page 145

Use these instructions to configure the arbitration data.

**To configure the arbitration data for a phase:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. Select the **Edit Phase** dialog box **Arbitration** tab.

3. Browse the area models equipment hierarchy to specify the equipment acquired by the equipment phase. The lack of a plus or minus sign to the left of an equipment type indicates no defined equipment. It can also indicate that the equipment type cannot be used as a needed equipment item at this level.

   - To add equipment, under the **Equipment Allocation by Resource** area, select the equipment needed by the phase and select **Add**. The selected equipment displays in the right pane.
   - To remove equipment, in the right pane, select the equipment to delete and select **Remove**. The selected equipment is removed from the list.

4. Select **Apply** to save changes.

**Tip:** Phases do not have a **Max Owners** box to define because a phase can only execute one recipe phase class at a time. For that reason, all phases can have only one owner.

**See also**

- [Edit Phase dialog box - Arbitration tab](#) on page 191
The **Edit Phase** dialog box on the **Arbitration** tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the phase (view-only). The icon associated with the phase class displays next to the Name and Phase Class.</td>
</tr>
<tr>
<td>Phase Class</td>
<td>The name of the phase class from which this phase was created (view-only).</td>
</tr>
</tbody>
</table>
| Equipment Allocation by Resource | A list of the equipment for which ownership is required by this phase in order for any procedural element linked to the phase to begin execution. The left pane is a hierarchical display showing the current equipment configuration of the area model. The hierarchy has two roots:  
  • **Process Cells**: Displays all the process cells within the area model. From here, organization is hierarchical: process cells contain units and units contain phases and operations sequences.  
  • **Resources**: Displays a list of all the configured resources within the area model. The right pane is the list of equipment this phase is dependent upon. Any equipment in this list must be acquired by the phase before the phase can begin execution. |

**Add**  
Add the selected equipment requirement to the **Equipment Needed** list.

**Remove**  
Remove the selected equipment requirement from the **Equipment Needed** list.

See also

Configure the arbitration data for a phase on page 191

Use these instructions to configure cross invocation data.

**Tip:** Cross invocation is not supported in FactoryTalk eProcedure.

To configure cross invocation data for a phase:

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. Select the **Edit Phase** dialog box **Cross Invocation** tab.

3. In **Cross Invocation Configuration**, type the cross invocation string information.

4. Select **Apply** to save changes.
See also

Edit Phase dialog box - Cross Invocation tab on page 193

The Edit Phase dialog box Cross Invocation tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the phase (view-only).</td>
</tr>
<tr>
<td>Phase Class</td>
<td>The name of the phase class from which this phase was created (view-only). The icon associated with the phase class is displayed next to the Name and Phase Class.</td>
</tr>
<tr>
<td>Cross Invocation Configuration</td>
<td>The cross invocation strings indicate the addition of an item to FactoryTalk Batch View and the ActiveX control shortcut menus. This includes the caption for each menu item and the data to be passed to the specified automation server when the menu item is selected. The default labels associated with each of the five fields are Cross Invocation Label # (where # is 1 through 5). These labels can be modified in the Server Options dialog box, and are seen in the Edit Process Cell dialog box.</td>
</tr>
</tbody>
</table>

See also

Configure cross invocation data for a phase on page 192

This feature is used in conjunction with eProcedure only. Use these instructions to configure hyperlink data.

To configure hyperlink data for a phase:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. In the Edit Phase dialog box, select the Hyperlink tab.

3. In the Hyperlink Configuration boxes, type a valid URL string.

4. Select Apply to save changes.

See also

Edit Phase dialog box - Hyperlink tab on page 193
Edit Phase dialog box - Hyperlink tab

This feature is used in conjunction with eProcedure only.

The Edit Phase dialog box on the Hyperlink tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the phase (view-only).</td>
</tr>
<tr>
<td>Phase Class</td>
<td>The name of the phase class from which this phase was created (view-only). The icon associated with the phase class is displayed next to the Name and Phase Class.</td>
</tr>
<tr>
<td>Hyperlink Configuration</td>
<td>The URL associated with each hyperlink label. The default labels associated with each of the five fields are Hyperlink Label # English (where # is 1 through 5). These labels can be modified in the Server Options dialog box. The labels are used in the eProcedure Client Equipment view.</td>
</tr>
</tbody>
</table>

See also

Configure hyperlink data for a phase on page 193

Configure container data

The Containers tab is visible with material-enabled phase classes only.

Configure the set of containers from which this phase can select material, or into which it can distribute material, or both.

Tip

If the Material Server is unavailable, the Container and Storage Location Associations area contains an error message stating the container data is unavailable. The CONTAINER column is unavailable but contains the associations previously configured.

To configure container data:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Select the Edit Phase dialog box Containers tab.

3. Browse the container and storage location trees to specify associations with the phase. The lack of a plus or minus sign to the left of Containers or Storage Locations indicates no containers or storage locations have been defined. Define these in the Material Editor.

Select the container or storage location needed by the phase under the Container and Storage Location Associations area and select Add. The item selected displays in the right pane under the Container/Storage
Configure phases

Location column. Adding a container that already exists in the Container/Storage Location column is not allowed.

To remove a container or storage location, in the right pane, select the item to delete and select Remove. The selected item is removed from the list.

4. Select the appropriate check box to indicate if the container is a source of material (material is drawn from this container), a material destination (material enters this container), or both.

Tip: At least one check box, addition, or distribution, must be selected for the configuration to be accepted, and it must agree with the container behavior. The High and Low values configured in the phase class parameters specify the container behavior, whether the phase class is a material addition, distribution, or both.

Addition - Low = 0, High = positive number
Distribution - Low = negative number, High = 0
Both - Low = negative number, High = positive number

5. Select Apply to save changes.

6. Select OK to close the Edit Phase dialog box.

See also

Edit Phase dialog box - Containers tab on page 195

The Edit Phase dialog box Containers tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the phase (view-only).</td>
</tr>
<tr>
<td>Phase Class</td>
<td>The phase class name from which this phase was created (view-only). The icon associated with the phase class displays next to the Name and Phase Class.</td>
</tr>
</tbody>
</table>
| Container and Storage Location Associations | A list of containers and storage locations to which this phase connects. The two columns in the right pane indicate whether the container is a source or a destination for material. Icons are used to represent a material source (container with a plus sign), and a material destination (container with a minus sign). The left pane is a hierarchical display of the set of containers and storage locations currently configured in the material database. The hierarchy has two roots:  
  - **Containers:** Displays all the containers within the material database not associated with a storage location. Containers within storage locations are not displayed because the phase makes an association only with independent containers or storage locations, never with containers within storage locations.  
  - **Storage Locations:** Displays a list of all the storage locations configured within the Material Editor. |
## Name

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Add</td>
<td>Add a selected container or storage location to the Container/Storage Location column.</td>
</tr>
<tr>
<td>Remove</td>
<td>Remove the selected container or storage location from the Container/Storage Location column.</td>
</tr>
</tbody>
</table>

### See also

[Configure container data](#) on page 194
If enabled, **Parameter/Tag Locking** enables FactoryTalk Batch Equipment Editor to determine and build the tags required for downloading all the parameter limits and verification configuration details to the phase logic. If **Parameter/Tag Locking** is disabled, have all the limits information downloaded to the parameters chosen by selecting the appropriate set of parameter limit tags for those parameters. The **Parameter Limit Tags** tab is not displayed for instruction-based phases.

**See also**

* Enable parameter limit tags on page 197
* Edit Phase dialog box - Parameter Limit Tags tab on page 197
* Map parameter tags to parameter limit tags on page 198

### Parameter limit tags

Use the **Edit Phase** dialog box **Parameter Limit Tags** tab to enable parameter limit tags. The **Parameter/Tag Locking** option must be enabled in the **Edit Phase** dialog box **General** tab.

When parameter limit tags are enabled, the required verification limit tags for the parameters that have signature verification policies defined are created.

**To enable parameter limit tags:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. In the **Edit Phase** dialog box, select the **Parameter Limit Tags** tab.

3. Select the **Enable Limit Tags** check box.

   The limit tags required for all parameters with signature verification policies in this phase are automatically created and displayed in the **Limit Tags** area.

4. Select **OK** to close the **Edit Phase** dialog box or **Apply** to keep the dialog box open.

**See also**

* Edit Phase dialog box - Parameter Limit Tags tab on page 197

### Enable parameter limit tags

Automatically create and maintain the necessary phase parameter tags for the parameters defined in the phase class.

If the **Parameter/Tag Locking** option is enabled on the phase class from which the phase was created, the system automatically creates the required verification limit tags for the parameters that have signature verification policies defined.
The **Parameter Limit Tags** tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the phase.</td>
</tr>
<tr>
<td>Phase Class</td>
<td>Name of the phase class from which the phase was created.</td>
</tr>
<tr>
<td>Enable Limit Tags</td>
<td>Creates the necessary parameter limit tags for the verification policies configured on the phase class. This option is only available if the <strong>Parameter/Tag Locking</strong> option was selected in the <strong>Edit Phase Class</strong> dialog box <strong>General</strong> tab. If the <strong>Parameter/Tag Locking</strong> option was not selected, define the limit tag sets and map the parameter tags to the limit tags.</td>
</tr>
</tbody>
</table>

**Tip:** These areas are enabled only if **Parameter/Tag Locking** is disabled.

### Parameter Tag to Limit Tag Mapping area

<table>
<thead>
<tr>
<th>Parameter Tag</th>
<th>Displays the tag name for the user-defined parameter that was created in the phase class (view only). (See <strong>Phase tags</strong> for information on phase tag naming conventions.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Displays the data type for the parameter (view only).</td>
</tr>
<tr>
<td>Limit Tag Set</td>
<td>Creates the necessary set of verification limit tags for the parameter. Select the same verification method that is configured in the <strong>Edit Phase Class</strong> dialog box <strong>Parameters</strong> tab.</td>
</tr>
</tbody>
</table>

### Limit Tags area

<table>
<thead>
<tr>
<th>Limit Tags scroll box</th>
<th>Lists the set of phase tag classes created for the selected <strong>Limit Tag Set</strong>.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Displays the phase tag defined for the selected tag class (view only).</td>
</tr>
<tr>
<td>Type</td>
<td>Displays the phase tag data type. The area below the <strong>Type</strong> box describes the phase tag (view only).</td>
</tr>
</tbody>
</table>

See also

[Enable parameter limit tags on page 197](#)

**Map parameter tags to parameter limit tags**

Use the **Edit Phase** dialog box **Parameter Limit Tags** tab to map parameter tags to parameter limit tags, if **Parameter/Tag Locking** is not enabled on the phase class and parameter tags are created manually. These instructions only apply to OPC or RSLinx Enterprise FactoryTalk Live Data phases.

If the **Parameter/Tag Locking** option is disabled on the phase class, manually select the appropriate limit tag set for each parameter tag.

**To map parameter tags to parameter limit tags:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. In the **Edit Phase** dialog box, select the **Parameter Limit Tags** tab.
3. Place the cursor in the Limit Tag Set box that is in the same row as the parameter to map.

   **Tip:** Remember to update the Number of Tags boxes in the Edit Phase Class dialog box General tab before mapping parameter tags to limit tags. Even if parameters are added to the phase class and verification methods are configured for them, parameter tags do not display in the Parameter Tag to Limit Tag Mapping area if the number of tags entered on the phase class is too low.

4. From the Limit Tag Set list, select the option that matches the verification method defined for the parameter on the phase class.

   **Tip:** If the phase class on which the phase is based uses control strategies, select the verification method of the control strategy that has the largest number of limits. For example, if a phase class has two control strategies, and Strategy 1 has a verification method of H/L, and Strategy 2 has a verification method of HHH/LLL, choose the HHH/LLL limit tag set.

   All limit tags required for the selected Limit Tag Set are created and displayed in the Limit Tags area.

5. Repeat steps 2 and 3 for each parameter that has a verification policy with parameter verification limits specified.

6. Select OK to close the Edit Phase dialog box or Apply to keep the dialog box open.

**See also**

- Parameter limit tags on page 197
- Edit Phase dialog box - Parameter Limit Tags tab on page 197

---

**Report limit tags**

If Parameter/Tag Locking is enabled, it allows FactoryTalk Batch Equipment Editor to determine and build the tags required for downloading the report limits and verification configuration details to the phase logic. If Parameter/Tag Locking is disabled, have all the limits information downloaded to the reports chosen by selecting the appropriate set of report limit tags for those parameters. The Report Limit Tags tab is not displayed for instruction-based phases.

**See also**

- Edit Phase dialog box - Report Limit Tags tab on page 199

---

**Edit Phase dialog box - Report Limit Tags tab**

When the Enable Limit Tags check box is enabled, the necessary phase parameter and report tags are created and maintained for the parameters and reports defined in the phase.

The Report Limit Tags tab contains:
### Configure phases

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Name of the phase.</td>
</tr>
<tr>
<td>Phase Class</td>
<td>Name of the phase class from which the phase was created.</td>
</tr>
<tr>
<td>Enable Limit Tags</td>
<td>Creates the necessary parameter limit tags for the verification policies configured on the phase class. This option is only available if the Parameter/Tag Locking option was selected in the Edit Phase Class dialog box General tab. If the Parameter/Tag Locking option was not selected, define the limit tag sets and map the parameter tags to the limit tags.</td>
</tr>
</tbody>
</table>

**Tip:** These areas are enabled only if Parameter/Tag Locking is disabled.

### Report Tag to Limit Tag Mapping area

<table>
<thead>
<tr>
<th>Report Tag</th>
<th>Displays the tag name for the user-defined report that was created in the phase class (view only). (See Configure a phase tag for information on phase tag naming conventions.)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type</td>
<td>Displays the report data type (view only).</td>
</tr>
<tr>
<td>Limit Tag Set</td>
<td>Creates the necessary set of verification limit tags for the report. Select the same verification method configured in the Edit Phase Class dialog box Reports tab.</td>
</tr>
</tbody>
</table>

### Limit Tags area

<table>
<thead>
<tr>
<th>Limit Tags scroll box</th>
<th>Lists the set of phase tag classes created for the selected Limit Tag Set.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Displays the phase tag defined for the selected tag class (view only).</td>
</tr>
<tr>
<td>Type</td>
<td>Displays the phase tag data type. The area below the Type box describes the phase tag (view only).</td>
</tr>
</tbody>
</table>

### See also

- Report limit tags on page 199
- Configure a phase tag on page 188

### Enable report limit tags

Use the Edit Phase dialog box Report Limit Tags tab to enable limit tags. The Parameter/Tag Locking option must be enabled on the Edit Phase dialog box General tab to enable limit tags.

When the Enable Limit Tags check box is enabled, the necessary phase parameter and report tags are created and maintained for the parameters and reports defined in the phase.

**To enable report limit tags:**

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. In the Edit Phase dialog box, select the Report Limit Tags tab.

3. Select the Enable Limit Tags check box.
The limit tags required for all reports with signature verification policies in this phase are automatically created and displayed in the Limit Tags area.

4. Select OK to close the Edit Phase dialog box or Apply to keep the dialog box open.

See also

Map report tags to report limit tags on page 201

Edit Phase dialog box - Report Limit Tags tab on page 199

Map report tags to report limit tags

If Parameter/Tag Locking is disabled and parameter tags are created manually, follow this procedure to create parameter limit tags. These instructions only apply to OPC or RSLinx Enterprise FactoryTalk Live Data phases.

If Parameter/Tag Locking is disabled, manually select the appropriate Limit Tag Set for each Report tag.

To map report tags to report limit tags:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. In the Edit Phase dialog box, select Report Limit Tags tab.

3. Place the cursor in the Limit Tag Set box that is in the same row as the parameter that to map.

   Tip: Remember to update the Number of Tags boxes in the Edit Phase dialog box General tab before mapping report tags to limit tags. Even if reports to the phase and configured verification methods for them are added, report tags do not display in the Report Tag to Limit Tag mapping area if the number of tags entered on the phase is too low.

4. From the Limit Tag Set list, select the option that matches the verification method defined for the report on the phase class.
Tip: If the phase class on which the phase is based uses control strategies, select the verification method of the control strategy that has the largest number of limits. For example, if a phase class has two control strategies, and Strategy 1 has a verification method of H/L, and Strategy 2 has a verification method of HHH/LLL, choose the HHH/LLL limit tag set.

All limit tags required for the selected Limit Tag Set are created and displayed in the Limit Tags area.

5. Select OK to close the Edit Phase dialog box or Apply to keep the dialog box open.

See also

Enable report limit tags on page 200

Configure parameter and report limit tags

After enabling or mapping the parameter and report limit tags, configure the Access Path and Item Name for each limit tag. These are configured in the same Edit Phase Tag dialog box where other phase tags are configured.

Tip: The data server cannot be changed for limit tags. The data server for a limit tag must be the same as the data server for the parameter or mapped report tag. When changing the data server for a parameter or report, the data server for its corresponding limit tags automatically changes to match.

See also

Phase tags on page 143
How edits to parameters, reports, and phase classes affect limit tags

After limit tags are enabled or mapped, if the verification method of the parameter or report changed, the limit tags are automatically adjusted to match the new verification method. New tags are automatically created or deleted. There is no need to alter the number of limit tags that exist.

| Important | After limit tags are enabled or mapped, if the data server changed for the phase to the instruction server, all limit tags are automatically deleted. Limit tags are not needed for instruction-based phases. |

If the **Parameter/Tag Locking** option is enabled:

- When the phase class contains control strategies, the number of limit tags for the parameter is the largest limit set for any of the control strategies to which the parameter is assigned. When a control strategy is deleted, the system automatically recalculates the limit tag set required for each parameter and adjusts the number of limit tags accordingly. If the control strategy that defined the largest set of limits for the parameter is deleted, the number of limit tags are automatically decreased.

- If **Parameter/Tag Locking** was enabled when limit tags were created for the phase, and then **Parameter/Tag Locking** is disabled, the existing tags are preserved and not modified.

- If a parameter or report is deleted, the corresponding set of limit tags are automatically deleted. The parameter tag to limit tag set is maintained for the other remaining parameters or reports, and the item names assigned to each tag remain the same.

| Important | If **Parameter/Tag Locking** is not enabled and a parameter or report is deleted from the phase class and the number of tags in the **Edit Phase** dialog box **General** tab is decreased, re-map the remaining parameter tags to limit tag sets. The parameter tag names and limit tag associations are shifted; they are not maintained. |

See also

Configure parameter and report limit tags on page 202

Overview of sharing phases and operation sequences between units

When a single phase or operation sequence is required by more than one unit, it must be shared by all units that require it. Do not create a new instance of the same phase or operation sequence in the other unit(s). Prior to sharing a phase or operation sequence, verify the phase or operation sequence is physically able to be shared.

A phase or operation sequence may have only one owner at a time. Sharing a single phase or operation sequence allows any shared unit to gain ownership of it during
batch execution. A phase or operation sequence cannot be owned by more than one unit at the same time.

When a phase or operation sequence is shared, any changes made to it are reflected in each unit that contains the shared phase or operation sequence, including its physical placement in the Design View area. When an instance of a shared phase or operation sequence is deleted from one unit, the phase or operation sequence is removed from the selected unit but it is still present in any other units to which it belongs.

See also

Share a phase or operation sequence on page 204

Use these instructions to share a phase or operation sequence.

To share a phase or operation sequence:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Open the area model.

3. Navigate to the phase or operation sequence level of the unit in the area model that does not contain the phase or operation sequence to be shared.

4. Select Share.

The Share Phases and Operation Sequences Between Units dialog box displays. Navigate to the phases and operation sequences by selecting the plus (+) sign next to the unit names.

5. Select the phase or operation sequence to share with the current unit, then select OK.

The icon of the selected phase or operation sequence displays in the Design View area. A small hand is at the bottom left corner, in both the active unit and the unit in which the phase or operation sequence was originally defined.

See also

Overview of sharing phases and operation sequences between units on page 203
Chapter 17

Operation sequence configuration

An operation sequence is an instance of a specific operation sequence class. Only one instance of a specific operation sequence can be added to each unit. Once an operation sequence instance is added to a unit instance, the operation sequence must be configured.

Tip: It is not necessary to configure tag data, as SequenceManager Equipment Operation Sequences are assigned to a Logix5000 CIP data server.

When an operation sequence is selected in the Design View, the operation sequence class on which it is based is highlighted. If the operation sequence class on which they are based is modified, operation sequences are automatically updated. Operation sequences created prior to modifying the operation sequence class are also updated.

See also

Configure the general data for an operation sequence on page 206

Configure the arbitration data for operation sequences on page 207

Create an operation sequence

Use these instructions to create an operation sequence.

To create an operation sequence:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Open the area model.

3. Navigate to the operation sequence level of the area model (select Go Down (.Split) to the lowest level).

4. In the Classes area, select the operation sequence and place it in the Design View area where you want the instance to reside.

5. Use the Edit Operation Sequence dialog box to view and configure the properties in the General, Arbitration, and Tags tabs.
The operation sequence general attributes include the operation sequence class name, equipment ID, and the current assigned data server.

The **Edit Operation Sequence** dialog box **General** tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the operation sequence.</td>
</tr>
<tr>
<td>Operation Class</td>
<td>The name of the operation sequence class from which this operation sequence was created (view-only).</td>
</tr>
<tr>
<td>Equipment ID</td>
<td>A unique number, greater than zero, that identifies this operation sequence. It must be the same equipment ID used in the phase logic to request this operation sequence.</td>
</tr>
<tr>
<td>Data Server</td>
<td>Displays the currently assigned server. When the list opens, configured servers are shown.</td>
</tr>
</tbody>
</table>

**See also**

- [Configure the general data for an operation sequence](#) on page 206
- [Configure the arbitration data for operation sequences](#) on page 207

**Edit Operation Sequence dialog box - General tab**

Use these instructions to configure general data for an operation sequence, using the **Edit Operation Sequence** dialog box **General** tab.

**To configure the general data for an operation sequence:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.
2. With the **Edit Operation Sequence** dialog box open, select the **General** tab.
3. In **Name**, type the operation sequence name.
4. In **Equipment ID**, enter the equipment ID, or accept the default value.
5. In **Data Server**, confirm that the correct data server is shown. If it does not, select the list to locate the appropriate server.
6. Select **Apply** to save changes.

**See also**

- [Edit Operation Sequence dialog box - General tab](#) on page 206

---

**Configure the general data for an operation sequence**

Use these instructions to configure general data for an operation sequence, using the **Edit Operation Sequence** dialog box **General** tab.

**To configure the general data for an operation sequence:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.
2. With the **Edit Operation Sequence** dialog box open, select the **General** tab.
3. In **Name**, type the operation sequence name.
4. In **Equipment ID**, enter the equipment ID, or accept the default value.
5. In **Data Server**, confirm that the correct data server is shown. If it does not, select the list to locate the appropriate server.
6. Select **Apply** to save changes.

**See also**

- [Edit Operation Sequence dialog box - General tab](#) on page 206
The Edit Operation Sequence dialog box Arbitration tab configures resources the operation sequence instance must acquire to be run.

The Arbitration tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the operation sequence (view-only).</td>
</tr>
<tr>
<td>Operation Class</td>
<td>The name of the operation sequence class from which this operation sequence was created (view-only).</td>
</tr>
<tr>
<td>Equipment Allocation by Resource</td>
<td>A list of the equipment for which ownership is required by this operation sequence in order for any procedural element linked to the operation sequence to begin execution. The left pane is a hierarchical display showing the current equipment configuration of the area model. The hierarchy has two roots:  &lt;ul&gt;  &lt;li&gt;Process Cells: Displays all the process cells within the area model. From here, organization is hierarchical: process cells contain units and units contain phases and operations sequences.&lt;/li&gt;  &lt;li&gt;Resources: Displays a list of all the configured resources within the area model. The right pane is the list of equipment this operation sequence is dependent upon. Any equipment in this list must be acquired by the operation sequence before the operation sequence can begin execution.&lt;/li&gt; &lt;/ul&gt;</td>
</tr>
<tr>
<td>Add</td>
<td>Add the selected equipment requirement.</td>
</tr>
<tr>
<td>Remove</td>
<td>Remove the selected equipment requirement.</td>
</tr>
</tbody>
</table>

See also

Configure the arbitration data for operation sequences on page 207

Use the Edit Operation Sequence dialog box Arbitration tab to configure arbitration data for an operation sequence.

To configure the arbitration data for an operation sequence:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. With the Edit Operation Sequence dialog box open, select the Arbitration tab.

3. Browse the area model’s equipment hierarchy to specify the equipment acquired by the operation sequence. The equipment hierarchy can be navigated by opening equipment to view the contained resources.
• To add equipment, under the **Equipment Allocation by Resource** area, select the equipment needed by the operation sequence and select **Add**. The selected equipment displays in the right pane.

• To remove equipment, in the right pane, select the equipment to delete and select **Remove**. The selected equipment is removed from the list.

4. Select **Apply** to save changes.

**See also**

*Edit Operation Sequence dialog box - Arbitration tab* on page 207

The **Edit Operation Sequence** dialog box **Tags** tab displays the configuration of CIP communication data for the operation sequence.

The **Tags** tab contains:

<table>
<thead>
<tr>
<th>Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The name of the operation sequence (view-only).</td>
</tr>
<tr>
<td>Operation Class</td>
<td>The name of the operation sequence class from which this operation sequence was created (view-only).</td>
</tr>
<tr>
<td>Operation sequence tags</td>
<td>A list of all the tags that are defined for the operation sequence (view-only). The tags shown are configured in the <strong>Operation Sequence Class</strong> dialog box in the <strong>Reports</strong> tab and <strong>Parameters</strong> tab.</td>
</tr>
<tr>
<td>Name</td>
<td>The name of the highlighted tag (view-only).</td>
</tr>
<tr>
<td>Usage</td>
<td>The type of parameter (view-only), either <strong>Input</strong> or <strong>Output</strong>.</td>
</tr>
<tr>
<td>Type</td>
<td>The operation sequence tag data type (view only).</td>
</tr>
</tbody>
</table>

**Tip:** Parameters and reports that use the ENUM type are shown as the INTEGER type for the Logix5000 controller.

**See also**

*Configure operation sequence class reports* on page 210

*Configure operation sequence class parameters* on page 208
Configure operation sequence class parameters

Use the Create Operation Sequence Class dialog box Parameters tab to configure operation sequence class parameters.

Tip: Use the keyboard to move around in the grid. Press the Tab key to move focus to the grid. Press Ctrl+Alt+Home to highlight the first editable cell in the grid.

To configure operation sequence class parameters:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. With the Create Operation Sequence Class dialog box open, select the Parameters tab. A list of parameters associated with the operation sequence class displays.

3. Select Add Parameter to add a new operation sequence class parameter. A new parameter displays with default values.

4. In Name, type a unique name for the operation sequence class parameter.

   Important: The parameter name must match the name of the sequencing input parameter belonging to the equipment sequence in the controller.

5. From the Type list, select the appropriate parameter type. The parameter type is equivalent to the sequencing input parameter type belonging to the equipment sequence in the controller. See Create Operation Sequence Class dialog box - Parameters tab for sequencing input parameter type information.

   • If the Type is Real or Integer, add the Max value, Min value, and Default value, and then optionally type the engineering units in the Enum/E.U. box.

   • If the Type is Enumeration, choose the appropriate enumeration set from the Enum/E.U. list and select the appropriate default enumeration from the Default list.

   • If the Type is String, type the engineering units in the Enum/E.U. box and the default string in the Default box.

6. If the parameter must scale within a recipe when a batch scales, select the Scale check box. (See Scale parameters for more information.)
7. To assign a context or descriptor to the recipe operation sequence class parameter, select one or more **Context IDs** from the **Reporting Contexts** column.

8. To delete a parameter, do the following:
   a. Select the parameter row header, then select **Delete Parameter**.
   b. Select **Yes** to confirm the deletion.

9. To save changes, select **Apply**.

10. To save changes and return to the **Class View** area, select **OK**.

**See also**

[Scale parameters on page 46](#)

---

### Configure operation sequence class reports

Use the **Create Operation Sequence Class** dialog box **Reports** tab to configure operation sequence class reports. An operation sequence class report is defined for the operation sequence class, while the corresponding report tag is automatically built for the operation sequence instance.

**To configure operation sequence class reports:**

1. Select **Start**, point to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Equipment Editor**.

2. With the **Create Operation Sequence Class** dialog box open, select the **Reports** tab. A list of operation sequence reports associated with the operation sequence is shown.

   **Tip:** Use the keyboard to move around in the grid. Press the **Tab** key to move the focus in the grid. Press **Control+Alt+Home** to highlight the first editable cell in the grid.

3. Select **Add Report** to add a new operation sequence report.

   A new report displays with default values.
   a. In **Name**, type a unique name for the report parameter.
b. From the **Type** list, select the appropriate parameter type. The report type is equivalent to the sequencing output parameter type belonging to the equipment sequence in the controller. See **Create Operation Sequence Class dialog box - Reports tab** for sequencing output parameter type information.

- If **Type** is REAL, STRING, or INTEGER, type the engineering units in the **Enum/E.U.** box. If control strategies are enabled, edit the **Enum/E.U.** value for each configured control strategy.
- If **Type** is ENUM, select the appropriate enumeration set from the **Enum/E.U.** box.

c. To assign a context or descriptor to the report parameter, from the **Reporting Contexts** column, select one or more **Context IDs**.

4. To delete a report, do the following:

a. Select the report row header, and then select **Delete Report**.

b. Select **Yes** to confirm the deletion.

5. To save changes, select **Apply**.

6. To save changes and return to the **Class View** area, select **OK**.

See also

- **Configure the verification policy for parameters and reports** on page 171
- **Select context IDs** on page 164

**Overview of sharing phases and operation sequences between units**

When a single phase or operation sequence is required by more than one unit, it must be shared by all units that require it. Do not create a new instance of the same phase or operation sequence in the other unit(s). Prior to sharing a phase or operation sequence, verify the phase or operation sequence is physically able to be shared.

A phase or operation sequence may have only one owner at a time. Sharing a single phase or operation sequence allows any shared unit to gain ownership of it during batch execution. A phase or operation sequence cannot be owned by more than one unit at the same time.
When a phase or operation sequence is shared, any changes made to it are reflected in each unit that contains the shared phase or operation sequence, including its physical placement in the Design View area. When an instance of a shared phase or operation sequence is deleted from one unit, the phase or operation sequence is removed from the selected unit but it is still present in any other units to which it belongs.

**See also**

[Share a phase or operation sequence on page 204](#)

Use these instructions to share a phase or operation sequence.

**To share a phase or operation sequence:**

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor.

2. Open the area model.

3. Navigate to the phase or operation sequence level of the unit in the area model that does not contain the phase or operation sequence to be shared.

4. Select Share.

The Share Phases and Operation Sequences Between Units dialog box displays. Navigate to the phases and operation sequences by selecting the plus (+) sign next to the unit names.

5. Select the phase or operation sequence to share with the current unit, then select OK.

The icon of the selected phase or operation sequence displays in the Design View area. A small hand is at the bottom left corner, in both the active unit and the unit in which the phase or operation sequence was originally defined.

**See also**

[Overview of sharing phases and operation sequences between units on page 203](#)
Supported area model changes

This section lists properties of an area model that can be edited without affecting a Warm or Warm-All restart of the FactoryTalk Batch Server.

**Important**

If an option or parameter is not listed in this section, it is not considered safe to modify (create, edit, or delete) that object in order to ensure a successful Warm or Warm-All restart of the FactoryTalk Batch Server.

See also

- [Area model object warm restart support on page 213](#)
- [Resource classes warm restart support on page 213](#)
- [Resources warm restart support on page 214](#)
- [Unit attributes warm restart support on page 214](#)
- [Unit attribute tags warm restart support on page 215](#)

**Area model object warm restart support**

A FactoryTalk Batch area model configuration file (.CFG file) always contains a single area object by definition.

To edit, select **Edit > Area**.

<table>
<thead>
<tr>
<th>Area Model Object</th>
<th>Warm Restart Supported</th>
</tr>
</thead>
<tbody>
<tr>
<td>Version</td>
<td>Yes</td>
</tr>
</tbody>
</table>

See also

- [Supported area model changes on page 213](#)

**Resource classes warm restart support**

Select **Edit > Resources** to created or edited resource classes.

Warm restart support for adding resource classes within an area model setting:

<table>
<thead>
<tr>
<th>Resource Class Action</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create new resource class</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Warm restart support for modifying resource class properties setting:
### Resources warm restart support

<table>
<thead>
<tr>
<th>Resource Class Property</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Resource class name</td>
<td>Yes</td>
</tr>
</tbody>
</table>

See also [Supported area model changes on page 213](#)

Select **Edit > Resources** to create and edit resources.

Warm restart support for adding resources defined within an area model:

<table>
<thead>
<tr>
<th>Resource Action</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creation</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Warm restart support for modifying resource properties:

<table>
<thead>
<tr>
<th>Resource Property</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Yes</td>
</tr>
<tr>
<td>Max Owners</td>
<td>The value of this property can be increased without affecting warm restart functionality.</td>
</tr>
<tr>
<td>Dependent Resources</td>
<td>Resources can be removed from a resource’s list of dependent resources without affecting warm restart functionality.</td>
</tr>
<tr>
<td>Cross Invocation Labels</td>
<td>Yes</td>
</tr>
<tr>
<td>Hyperlink Labels</td>
<td>Yes</td>
</tr>
</tbody>
</table>

See also [Supported area model changes on page 213](#)

### Unit attributes warm restart support

Select **Edit > Tags** to create and edit unit attributes.

Warm restart support for adding unit attributes defined within an area model:

<table>
<thead>
<tr>
<th>Unit Attribute Action</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Create tag classes</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Existing warm restart support for modifying unit attribute properties:

<table>
<thead>
<tr>
<th>Unit Attribute Property</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Engineering Unit</td>
<td>Yes</td>
</tr>
<tr>
<td>Description</td>
<td>Yes</td>
</tr>
</tbody>
</table>

See also [Supported area model changes on page 213](#)
Unit attribute tags warm restart support

Create, delete, and edit unit attribute tags using the properties dialog of a unit object.

Warm restart support for editing unit attribute tags within an area model:

<table>
<thead>
<tr>
<th>Unit Attribute Tag Action</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creation</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Warm restart support for modifying the properties of a unit attribute tag:

<table>
<thead>
<tr>
<th>Unit Attribute Tag Property</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Engineering Unit</td>
<td>Yes</td>
</tr>
<tr>
<td>Data Source</td>
<td>Yes</td>
</tr>
<tr>
<td>Access Path</td>
<td>Yes</td>
</tr>
<tr>
<td>Item Name</td>
<td>Yes</td>
</tr>
<tr>
<td>Write Access Path</td>
<td>Yes</td>
</tr>
<tr>
<td>Write Item Name</td>
<td>Yes</td>
</tr>
<tr>
<td>Server</td>
<td>Yes</td>
</tr>
</tbody>
</table>

See also

- Units warm restart support on page 218
- Unit overview on page 123

Equipment phase tags warm restart support

Equipment phase tags are automatically created by FactoryTalk Batch Recipe Editor when equipment phases are created.

Warm restart support for modifying the properties of an equipment phase tag:

<table>
<thead>
<tr>
<th>Unit Attribute Tag Property</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Access Path</td>
<td>Yes</td>
</tr>
<tr>
<td>Item Name</td>
<td>Yes</td>
</tr>
<tr>
<td>Write Access Path</td>
<td>Yes</td>
</tr>
<tr>
<td>Write Item Name</td>
<td>Yes</td>
</tr>
</tbody>
</table>

See also

- Supported area model changes on page 213

Enumeration sets warm restart support

Select Edit > Enumeration Sets to create and edit enumeration sets.

Warm restart support for adding enumeration sets defined within an area model.

- Enumeration Set Action: Creation
- Warm Restart Support: Yes
See also

Supported area model changes on page 213

Data servers warm restart support

Select Edit > Server to create and edit data servers.

Warm restart functionality for adding data servers to an area model:

<table>
<thead>
<tr>
<th>Server Action</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creation</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Warm restart support for modifying data server properties:

<table>
<thead>
<tr>
<th>Server Property</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Server Type</td>
<td>Yes</td>
</tr>
<tr>
<td>Watchdog Protocol</td>
<td>Yes</td>
</tr>
<tr>
<td>Watchdog Item Access Path</td>
<td>Yes</td>
</tr>
<tr>
<td>Watchdog Item Name</td>
<td>Yes</td>
</tr>
<tr>
<td>Watchdog Write Item Access Path</td>
<td>Yes</td>
</tr>
<tr>
<td>Watchdog Write Item Name</td>
<td>Yes</td>
</tr>
<tr>
<td>Server Location</td>
<td>Yes</td>
</tr>
<tr>
<td>On Control Scan Rate</td>
<td>Yes</td>
</tr>
<tr>
<td>Off Control Scan Rate</td>
<td>Yes</td>
</tr>
</tbody>
</table>

See also

Supported area model changes on page 213

Process cell classes warm restart support

Process cell classes are created within the main window of FactoryTalk Batch Equipment Editor. Properties of a process cell class can be edited by right-clicking on the icon that represents a process cell class.

Warm restart support for adding process cell classes:

<table>
<thead>
<tr>
<th>Process Cell Class Action</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creation</td>
<td>Yes</td>
</tr>
</tbody>
</table>

Warm restart support for modifying the properties of a process cell class:

<table>
<thead>
<tr>
<th>Process Cell Class Property</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>Yes</td>
</tr>
<tr>
<td>Icon</td>
<td>Yes</td>
</tr>
</tbody>
</table>

See also

Supported area model changes on page 213
Process cells warm restart support

Process cells are created and deleted using the main window of FactoryTalk Batch Equipment Editor. Properties of a process cell can be edited by right-clicking on the icon that represents a process cell.

Warm restart support for adding process cells:

<table>
<thead>
<tr>
<th>Process Cell Action</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creation</td>
<td>Process cells can be added to an area model.</td>
</tr>
</tbody>
</table>

This table describes warm restart support for modifying the properties of a process cell:

<table>
<thead>
<tr>
<th>Process Cell Property</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERP Alias</td>
<td>Yes</td>
</tr>
<tr>
<td>Max Owners</td>
<td>The value of this property can be increased without affecting warm restart functionality.</td>
</tr>
<tr>
<td>Dependent Resources</td>
<td>Resources can be removed from a process cell's list of dependent resources without affecting warm restart functionality.</td>
</tr>
<tr>
<td>Cross Invocation Labels</td>
<td>Yes</td>
</tr>
<tr>
<td>Hyperlink Labels</td>
<td>Yes</td>
</tr>
</tbody>
</table>

See also

- [Area models on page 19](#)
- [Process cell overview on page 111](#)

Unit classes warm restart support

Unit classes are created within the main window of FactoryTalk Batch Equipment Editor. Properties of a unit class can be edited by right-clicking on the icon that represents a unit class.

Warm restart support for adding unit classes:

<table>
<thead>
<tr>
<th>Unit Class Action</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creation</td>
<td>Unit classes can be added to an area model.</td>
</tr>
</tbody>
</table>

This table describes warm restart support for modifying the properties of a unit class:

<table>
<thead>
<tr>
<th>Unit Class Property</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Icon</td>
<td>Yes</td>
</tr>
<tr>
<td>Assigned Unit Attributes</td>
<td>Unit attributes can be added to the assigned set of the unit class.</td>
</tr>
</tbody>
</table>
Appendix A  Supported area model changes

See also

Supported area model changes on page 213

Units warm restart support

Create units within the main window of FactoryTalk Batch Equipment Editor. Right-click the icon representing a unit to edit the unit properties.

Warm restart support for adding units:

<table>
<thead>
<tr>
<th>Unit Action</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creation</td>
<td>Yes</td>
</tr>
</tbody>
</table>

This table describes warm restart support for modifying unit properties:

<table>
<thead>
<tr>
<th>Unit Property</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERP Alias</td>
<td>Yes</td>
</tr>
<tr>
<td>Unit Attribute Tags</td>
<td>(See Unit attribute tags for information on which properties can be modified.)</td>
</tr>
<tr>
<td>Dependent Resources</td>
<td>Removing resources from the units list of dependent resources can be done without affecting warm restart functionality.</td>
</tr>
<tr>
<td>Cross Invocation Labels</td>
<td>Yes</td>
</tr>
<tr>
<td>Hyperlink Labels</td>
<td>Yes</td>
</tr>
</tbody>
</table>

See also

Unit attribute tags warm restart support on page 215

Unit overview on page 123

Parameters warm restart support

Right-click on a Phase icon to create Parameters for a phase in the Edit Phase dialog box Parameters tab.

Warm restart support for adding parameters to a phase setting:

<table>
<thead>
<tr>
<th>Parameter Action</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creation</td>
<td>Parameters can be added to a recipe phase without affecting warm restart functionality, with the exception of material parameters.</td>
</tr>
</tbody>
</table>
Important: If the Edit Phase dialog box General tab Material Based Recipes option is selected, the optional material parameters, MATERIAL and AMOUNT, are added to the Parameters tab. The warm restart functionality does NOT support these material parameters. Adding these material parameters and then performing a warm restart causes running batches to be removed from the Batch List.

Warm restart support for modifying the properties of a parameter associated with a phase settings:

<table>
<thead>
<tr>
<th>Parameter Property</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>High</td>
<td>Raise the value of this property.</td>
</tr>
<tr>
<td>Low</td>
<td>Lower the value of this property without affecting warm restart functionality.</td>
</tr>
<tr>
<td>Default</td>
<td>Yes</td>
</tr>
<tr>
<td>Engineering Units</td>
<td>Yes</td>
</tr>
</tbody>
</table>

See also

**Added parameters warm restart support** on page 219

When building a unit operation level recipe, specify a value for each parameter in the recipe phases added as steps to the unit operation. The value can be:

- Actual value
- Operator Prompt
- Deferral to a higher level parameter

If the FactoryTalk Batch Server detects that a new parameter has been added to the recipe phase (during the restoration of batches to the batch list during a warm restart), then there is no value for the parameter that can be found in the loaded recipe structure.

When no value is found, the server sets the value on the parameter for the step to Operator Prompt.

See also

**Supported area model changes** on page 213
Report warm restart support

Right-click on a phase class icon to create phase reports in the Edit Phase Class dialog box Reports tab.

Warm restart support for adding reports to a phase class setting:

<table>
<thead>
<tr>
<th>Report Action</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creation</td>
<td>Reports can be added to a phase class without affecting warm restart functionality, with the exception of material reports.</td>
</tr>
</tbody>
</table>

**Important:** When the Material Based Recipes option is selected in the Edit Phase dialog box General tab, the optional material reports ACTUAL_AMOUNT and FEED_COMPLETE are added to the Reports tab. The warm restart functionality does not support these material reports. Adding these material reports and then performing a warm restart causes running batches to be removed from the batch list.

Warm restart support for modifying report properties associated with phase class settings:

<table>
<thead>
<tr>
<th>Report Property</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Engineering Units</td>
<td>Yes</td>
</tr>
<tr>
<td>Verification Method</td>
<td>Yes</td>
</tr>
<tr>
<td>ERP Alias</td>
<td>Yes</td>
</tr>
</tbody>
</table>

See also

[Added reports warm restart support](#) on page 220

Added reports warm restart support

When a unit operation level recipe is loaded into the FactoryTalk Batch Server, the value of each report for each recipe phase step is set to Bad Value. As the phase logic uploads report values, the Bad Value is replaced with the actual value reported by the phase logic.

If the FactoryTalk Batch Server detects that a new report has been added to the recipe phase (while restoring batches to the batch list during a warm restart), then the report is treated as if it was never uploaded and assigned a current value of Bad Value.

See also

[Report warm restart support](#) on page 219
Configured messages warm restart support

Right-click on a recipe phase class icon to create phase class messages in the Edit Phase Class dialog box, Messages tab.

Existing warm restart support for adding configured messages to a phase class settings:

- **Message Action**: Creation
- **Warm Restart Support**: Messages can be added to a phase class without affecting warm restart functionality.

See also

Supported area model changes on page 213

Recipe phases warm restart support

Right-click on a Recipe Phase icon to create and delete Recipe phases.

Warm restart support for the adding of recipe phases setting:

<table>
<thead>
<tr>
<th>Recipe Phase Action</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creation</td>
<td>Recipe Phases can be added to an area model without affecting warm restart functionality.</td>
</tr>
</tbody>
</table>

Warm restart support for modifying the properties of a recipe phase settings:

<table>
<thead>
<tr>
<th>Recipe Phase Property</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Parameter Tags</td>
<td>An increase in this value is safe for warm restart functionality.</td>
</tr>
<tr>
<td>Number of Report Tags</td>
<td>An increase in this value is safe for warm restart functionality.</td>
</tr>
<tr>
<td>Parameters</td>
<td>(See Parameters for information on which properties can be modified.)</td>
</tr>
<tr>
<td>Reports</td>
<td>(See Reports for information on which properties can be modified.)</td>
</tr>
<tr>
<td>Messages</td>
<td>(See Configured messages for information on which properties can be modified.)</td>
</tr>
</tbody>
</table>

See also

Parameters warm restart support on page 218

Report warm restart support on page 219

Configured messages warm restart support on page 221
Equipment phases warm restart support

From the main window in FactoryTalk Batch Equipment Editor, right-click on an equipment phase icon to create equipment phases.

Warm restart support for adding equipment phases setting:

<table>
<thead>
<tr>
<th>Equipment Phase Action</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>Creation</td>
<td>Equipment phases can be added to an area model without affecting warm restart functionality.</td>
</tr>
</tbody>
</table>

Warm restart support for modifying properties of an equipment phase settings:

<table>
<thead>
<tr>
<th>Equipment Phase Property</th>
<th>Warm Restart Support</th>
</tr>
</thead>
<tbody>
<tr>
<td>ERP Alias</td>
<td>This property can be changed without affecting warm restart functionality.</td>
</tr>
<tr>
<td>Automatic Download on Start</td>
<td>Yes</td>
</tr>
<tr>
<td>Automatic Upload on Complete</td>
<td>Yes</td>
</tr>
<tr>
<td>Automatic Upload on Stopped</td>
<td>Yes</td>
</tr>
<tr>
<td>Automatic Upload on Aborted</td>
<td>Yes</td>
</tr>
<tr>
<td>Server</td>
<td>Yes</td>
</tr>
<tr>
<td>Tags</td>
<td>(See Equipment phase tags for information on which properties can be modified.)</td>
</tr>
<tr>
<td>Dependent Resources</td>
<td>Removing resources from a list of dependent resources can be done without affecting warm restart functionality.</td>
</tr>
<tr>
<td>Cross Invocation Labels</td>
<td>Yes</td>
</tr>
<tr>
<td>Hyperlink Labels</td>
<td>Yes</td>
</tr>
</tbody>
</table>

See also

Equipment phase tags warm restart support on page 215
Troubleshoot issues

Issues that may occur include:

- Import/export area model errors
- COM-related exception errors
- Signature template errors

See also

Unable to access instruction file on page 223

Illegal characters encountered in writing instruction file on page 224

Unable to access Material Server for material support per container on page 224

COM-related exception errors on page 224

Invalid signoff on page 228

Unable to access instruction file

The XML schema requires that the contents of any instruction files must be included in the XML file. FactoryTalk Batch Equipment Editor must access the instruction file at the time of the export. If the instruction files cannot be accessed, this error message displays and the export does not occur.

See also

Troubleshoot issues on page 223

Illegal characters encountered in writing instruction file on page 224

Unable to access Material Server for material support per container on page 224
Illegal characters encountered in writing instruction file

Because the contents of instruction files are included (that is, putting them into a CDATA section), it is possible that the particular character sequence `]]>`, which is illegal in CDATA sections, might be encountered.

If this character sequence is encountered, this error message displays and the export does not occur.

See also

Troubleshoot issues on page 223

Unable to access Material Server for material support per container

There is a requirement that material support per container must be included in the XML file. FactoryTalk Batch Equipment Editor must make a call to the Material Server at the time of the export. It is possible that the Material Server cannot be accessed at that time.

If the Material Server cannot be accessed, this error message displays and the export does not occur.

See also

Troubleshoot issues on page 223
COM-related exception errors

General COM-related system or program errors can occur for any number of reasons in processing the import or export request. A generic error handling displays the routine that encountered the error and the reason for the error as supplied by the originator (function) of the error.

If this occurs, an error message displays and the import or export does not occur. Examples include:

See also

XML parser error messages on page 225

Material-specific errors on page 225

XML parser error messages

The Microsoft XML parser returns different error messages for different error conditions. Examples of error conditions and error messages:

<table>
<thead>
<tr>
<th>Error Condition</th>
<th>Error Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>Invalid Data</td>
<td>The element: &lt;Element Name&gt; has an invalid value according to its data type.</td>
</tr>
<tr>
<td>Invalid Element Content</td>
<td>The element content is invalid according to its Schema. Expecting: &lt;Element Name&gt;.</td>
</tr>
<tr>
<td>Undefined Key Reference</td>
<td>Undefined values for KeyRef identity constraint: &lt;Reference Constraint Name&gt;.</td>
</tr>
<tr>
<td>Uniqueness Violation</td>
<td>Duplicated values for unique/key identity constraint: &lt;Uniqueness Constraint Name&gt;.</td>
</tr>
</tbody>
</table>

See also

Troubleshoot issues on page 223

COM-related exception errors on page 224
Material-specific errors

When the referenced material, material class, or container does not exist in the material database, a dialog box similar to the one shown here opens with a list of the missing phase class parameter values:

Select:

- **Continue**: Change the missing material, material class, or container with NULL_MATERIAL, NULL_CLASS, or NULL_CONTAINER, respectively, and continue the process.
- **Ignore**: Continue the process without any changes. A prompt displays to fix the problem.
- **Cancel**: Quit the process.

If the Material Server is installed, but not accessible, or if the material system enumeration sets (MATERIALS, MATERIAL_CLASSES, and CONTAINERS) are included in a material-enabled recipe phase class and are not configured as NULL, the import occurs without generating an error message.

A material-enabled recipe phase class must contain:

- $MATERIAL
- $AMOUNT

A material-enabled recipe phase class must contain:

- $ACTUAL_AMOUNT
- $FEED_COMPLETE

A material-enabled recipe phase class may, as an option, contains four recipe parameters. They must be included together or not at all:

- $CONTAINER
- $LOT
• $LABEL
• $MATERIAL_CLASS

See also

COM-related exception errors on page 224
Invalid signoff

If required signoff information for a signature template is unknown or missing, the Invalid Signoffs dialog box opens when starting or using FactoryTalk Batch Equipment Editor:

This error notification can occur:

- When FactoryTalk Batch Equipment Editor starts after a group or user account used in signature template(s) is deleted.
- In FactoryTalk Batch Equipment Editor, Edit Signature Templates dialog box was exited without configuring all required signature information.

See also

Resolve invalid signoff caused by deleted group or user on page 228
Resolve invalid signoff when required information is omitted on page 229

Resolve invalid signoff caused by deleted group or user

Use these instructions to troubleshoot the Invalid Signoff dialog box triggered by:

- Signature templates are created in FactoryTalk Batch Equipment Editor and then saved to the area model.
- FactoryTalk Batch Equipment Editor is closed.
- A group or user account used in one or more signature templates is deleted in the FactoryTalk Administration Console.
- FactoryTalk Batch Equipment Editor is re-started.
- The Invalid Signoff dialog box opens.
To resolve invalid signoff caused by deleted group or user:

- To delete the account from the signature template(s), select **OK**, save the area model and restart FactoryTalk Batch Equipment Editor. The error message no longer opens.

- To not delete the account from the signature template(s), select **OK**, and exit FactoryTalk Batch Equipment Editor without saving the area model. Determine the deleted group or user account in the Event Viewer (Start>Administrative Tools>Event Viewer> Applications and Services Logs> FactoryTalk Diagnostics). Then re-add the account using the FactoryTalk Administration Console.

See also

**Invalid signoff** on page 228

**Resolve invalid signoff when required information is omitted**

Use this information to troubleshoot the Invalid Signoff dialog box triggered by these sequence of events:

1. Signature templates are created in the **Edit Signature Templates** dialog box in FactoryTalk Batch Equipment Editor.

2. Required information for a valid signoff is omitted or forgotten (for example, multiple signoffs are configured but security permissions are not specified for all signoffs).
3. Selected OK or Apply.

4. The Invalid Signoffs dialog box opens.

To resolve invalid signoff:

1. Examine the information presented in the Invalid Signoffs dialog box. Listed beneath the Signature Template column is the name of the template or templates not fully configured.

2. Select OK and then make the necessary change(s) in the Edit Signature Templates dialog box. For example, configure security permissions for all signoffs, select OK to save the signature template configuration, then use the File > Save command to save the signature templates in the area model.

See also

Invalid signoff on page 228

No Signature templates configured within the Area Model

If no signature templates have been configured in the area model, attempted use of Signature Template Selection Browse produces this error message:

Define templates in Edit > Signature Templates and retry the operation.

See also

Troubleshoot issues on page 223
If applied security authority to the area model, and there is a security mismatch between the security authority identifier (SAI) in the area model and the FactoryTalk Network Directory SAI, this error message opens in FactoryTalk Batch Equipment Editor:

Make note of the FactoryTalk Network Directory and host computer information, then select **OK** to close the error message.

If there is a backup of the FactoryTalk Network Directory SAI, or of the area model in an unsecured form, these options are available:

- Restore the FactoryTalk Network Directory SAI to which the area model is secured. The name of the computer that hosts that SAI is provided in the dialog box. Use the FactoryTalk Administration Console to restore a saved backup of the SAI. The **Restore** command is available by right-clicking on the top node in the Explorer pane. When the SAI restores, use **File > Open** in FactoryTalk Batch Equipment Editor to open the secured area model.

- Open an unsecured copy of the area model.

**See also**

[Troubleshoot issues on page 223](#)

When attempting to open FactoryTalk Batch Recipe Editor, and an SAI mismatch exists between the area model and the FactoryTalk Network Directory, this error message opens:
Make note of the FactoryTalk Network Directory and host computer information, then select **OK** to close the error message and FactoryTalk Batch Recipe Editor.

To recover, backup of the FactoryTalk Network Directory SAI, or of the area model in unsecured form.

- Restore the FactoryTalk Network Directory SAI to which secures the area model. The name of the computer that hosts that SAI is provided in the dialog box. Use the FactoryTalk Administration Console to restore a saved backup of the SAI. The **Restore** command is available by right-clicking on the top node in the Explorer pane. When the SAI with which the area model is associated is restored, open FactoryTalk Batch Recipe Editor.

- Open an unsecured copy of the area model in FactoryTalk Batch Equipment Editor, then select **File > Save**. Then, open FactoryTalk Batch Recipe Editor.

**See also**

[Troubleshoot issues](#) on page 223
Legal Notices

Copyright notice

Copyright © 2018 Rockwell Automation Technologies, Inc. All Rights Reserved. Printed in USA.

This document and any accompanying Rockwell Software products are copyrighted by Rockwell Automation Technologies, Inc. Any reproduction and/or distribution without prior written consent from Rockwell Automation Technologies, Inc. is strictly prohibited. Please refer to the license agreement for details.

End User License Agreement (EULA)

You can view the Rockwell Automation End-User License Agreement ("EULA") by opening the License.rtf file located in your product's install folder on your hard drive.

Trademark Notices

Allen-Bradley, CompactLogix, ControlLogix, Connected Components Workbench, eProcedure, FactoryTalk, GuardLogix, GuardPLC, Micro800, MicroLogix, MobileView, MobileView Guard, PanelBuilder, PanelView, PhaseManager, PlantLink, PlantPAx, PLC-2, PLC-3, PLC-5, PowerMonitor, RSBizWare, RSBizWare BatchHistorian, RSBizWare Historian, RSBizWare MaterialTrack, RSLinx, Rockwell Automation, Rockwell Software, SLC, SoftLogix, Studio 5000, and X Mare trademarks of Rockwell Automation, Inc.

Any Rockwell Automation software or hardware not mentioned here is also a trademark, registered or otherwise, of Rockwell Automation, Inc.

For a complete list of products and their respective trademarks, go to: http://www.rockwellautomation.com/rockwellautomation/legal-notices/overview.page?#tab4

Warranty

This product is warranted in accordance with the product license. The product’s performance may be affected by system configuration, the application being performed, operator control, maintenance, and other related factors. Rockwell Automation is not responsible for these intervening factors. The instructions in this document do not cover all the details or variations in the equipment, procedure, or process described, nor do they provide directions for meeting every possible contingency during installation, operation, or maintenance. This product’s implementation may vary among users.
This document is current as of the time of release of the product; however, the
accompanying software may have changed since the release. Rockwell Automation,
Inc. reserves the right to change any information contained in this document or
the software at any time without prior notice. It is your responsibility to obtain the
most current information available from Rockwell when installing or using this
product.

Environmental compliance

Rockwell Automation maintains current product environmental information on
its website at

Contact Rockwell Automation

Customer Support Telephone — 1.440.646.3434

Online Support — http://www.rockwellautomation.com/support/
Index

A
add
  data servers 214
  enumeration sets 213
  OPC data server 86
  process cell classes 214
  process cells 215
  resource classes 211
  resources 212
  unit attribute tags 213
  unit attributes 212
  unit classes 215
  units 216
arbitration information
  process cell 115
  shared resources 98
  unit 127
area model
  create 19
  edit 211
  import XML file 27
  migration 21
  name 19
  open 20
  secure 80
automatic upload/download 51

B
batch commands, configure verification policies 67
binding expressions
  data types 105
  operators 104
binding requirements 103
  example 103

C
calculation method 167
classes, unit 215
command verification policies dialog box, field descriptions 69
command verification policy
  change 70
  configure 67
  remove 71
  configure a recipe approval 74
  container information
    equipment module 192
  control strategies
    and verification policies 173
    example 46
    implement 47
    overview 46
    with upload/download parameter subsets 52
copy a phase 176
create
  global unit attribute 144
  create area model 19
  create phase classes 150
  create process cell class 111
cross invocation
  definition of 129
cross invocation strings
  equipment modules 190
  process cells 129
  shared resources 100
  unit 129
cut and paste a phase 177

D
data server
  add 84, 214
  default item access path 85
  delete 92
  edit 91
  FactoryTalk Linx Live 88
  location 85
  name 85
  OPC 85
  scan rates 85
  types 85
  watchdog item access path 85
  watchdog item name 85
  watchdog protocol 85
  watchdog write item access path 85
  watchdog write item name 85
data server definition files 83
data server properties
  modify 214
data server protocols
  CIP  84
  FTD  84
  OPC  84
data servers
  assign  92
data types
  binding expressions  105
  enumeration  105
  examples  105
    integer  105
    real  105
    string  105
default tags
  naming conventions  185
delete
  unit instance  131
development limits  167
  DSDF  83
duplicate a phase  176

e
edit command verification policies dialog box, field
descriptions  69
edit signature templates dialog box, field
descriptions  58
electronic signature template  58
  add  59
    configure signoffs  61
delete  66
  modify  66
electronic signatures
  how they work  56
  sequence of tasks  56
  types  55
  use  55
enable
  parameter limit tags  194
  report limit tags  197
enhanced phase logic request  48, 51
enumeration sets  31
  add  213
  delete  36
description  31
  example  31
  negative ordinal values  31
  work with  33
enumerations
  material database unavailable  32
equipment dependencies
  process cells  115
  unit  127
Equipment Editor
  interface  15
  open  13
  security  13
equipment module
  container information  192
  cross invocation strings  190
  hyperlink strings  191
  OPC tags  185
tag naming conventions  185
tags  141, 185
equipment phase tag properties
  modify  213
equipment phases
  modify  219
example
  unit attribute binding  103
expression builder
  unit binding  103
  validation-Recipe Editor  105
expressions
  operands-Equipment Editor  105
  operators-Equipment Editor  104
  validation-Equipment Editor  105
F
FactoryTalk Linx  88
FactoryTalk Linx Live Data server  88
FactoryTalk Linx Live Data tags  40
FactoryTalk Linx Server Path  85, 90
FactoryTalk Live Data  88
G
general information
  process cells  182
  shared resources  98
global binding requirements  103
global unit attribute
  create  144
groups, configure for electronic signatures  61
H
High/Low 167
High-High/Low-Low 167
High-High-High/Low-Low-Low 167
hyperlink strings
equipment module 191
shared resources 101
unit 130

I
icons
create 120
path verification 119
import
enumerations (XML) 27
system phases (XML) 27
XML area model 27
import errors
material-specific 223
XML parser 223

L
limit tags
map to parameter tags 195
map to report tags 198
name conventions 185
tag locking disabled 195, 198
tag locking enabled 194, 197
link
units 132

M
material database
area model 23
enumerations 32
material-enabled phases
configure 174
disable 176
material-specific errors 223
menu bar 17
message partners 152
messages
modify 218
modify
area object 211
data server properties 214
equipment phase tag properties 213
equipment phases 219
messages 218
process cell class 214
process cell properties 215
recipe parameter properties 216
recipe phases 219
report parameter properties 217
resource class properties 211
resource properties 212
unit attribute properties 212
unit attribute tag properties 213
unit attribute tags 213
unit class properties 215
unit properties 216

N
No Limits 167
none, verification policy 167
not allowed 167

O
operands
binding expression-Equipment Editor 105
operators, expressions-Equipment Editor 104

P
parameter deviations, select verification method 45
parameter limits 45
parameter tags
map to limit tags 195
name conventions 185
parameters
overview 39
phase 42
report 43
scale 45
target 174
permissions, configure for electronic signatures 61
phase
configuring to material-enabled 174
copy 176
cut 177
disable material-enabled 176
Index

parameters
  definition 42, 216
  paste 177
phase commands, configure verification policies 67
phase link group 152
phase logic
  enhanced request 48, 51
phase parameters
  definition 216
  policy
    change for a command 70
    remove for a command 71
    verification 67
process cell
  add 215
  arbitration information 115
class
  create 111
  edit 112
  create 112
  cross invocation strings 129
definition 215
  general information 182
process cell classes
  add 214
  modify 214
process cell properties
  modify 215

R
recipe approval 74
recipe parameter properties
  modify 216
recipe parameters
  warm restart 217
recipe phases
  modify 219
report parameter properties
  modify 217
report parameters
  warm restart 218
report tags
  map to limit tags 198
  name conventions 185
reports
  phase 43
request tags, name conventions 185
resource class properties
  modify 211
resource classes
  add 211
resource properties
  modify 212
resources
  add 212
re-verify recipes 19

S
save area model 19
scale parameters 45
scan rates
  data server 85
security permissions, configure for electronic signatures 61
server options 17
share
  units 132
shared resources
  define 97
  description 95
signature requirement, remove for a command 71
signature template
  add 59
  configure signoffs 61
  create 58
  delete 66
  modify 66
  verification policy 167
signature templates dialog box, field descriptions 58
signatures
  electronic 55
  how they work 56
  sequence of tasks 56
types 55
signoff policies, configure for commands 67
signoffs, configure 61
Single Sign-On Equipment Editor 13

T
tags
  delete tag class 147
  description 135
index

edit 142
edit from menu 142
edit tag class 146
equipment module information 183
equipment module tags
  create 141
FactoryTalk Linx Live Data 40
map to limit tags 195, 198
naming conventions 185
set to default addresses 188
set to default names 143
unit attribute 138
unit attribute tags 136
  information 126
work with 135
target parameters, and report verification policies 174
template
  add 59
  configure signoffs 61
delete 66
modify 66
signature 58
toolbar-Equipment Editor 17
transitions
  data types-Equipment Editor 105

U
unique icons 120
unit
  arbitration information 127
  configure arbitration 127
  configure cross invocation 129
  configure hyperlinks 130
  configure tag information 126
  create a unit class 122
  create a unit instance 123
  create tags 138
  cross invocation strings 129
definition 216
delete 131
edit a unit class 122
equipment dependencies 127
hyperlink strings 130
link 132
remove a unit instance 131
share 132
tags information 126
work with 121
unit attribute binding
  example 103
unit attribute properties
  modify 212
unit attribute tag properties
  modify 213
unit attribute tags 126
  add 213
  create 138
  modify 213
unit attributes
  add 212
unit binding expression builder-Equipment Editor 103
unit class properties
  modify 215
unit classes
  add 215
unit instance
  create 123
  remove 131
unit properties
  modify 216
unit tag class
  See 136
units
  add 216
upload/download parameter subsets 48
  use with control strategies 52
user groups, configure for electronic signatures 61

V
validation
  binding expressions-Equipment Editor 105
verification method, select 45
verification policies 167
  and control strategies 173
  change for a command 70
  configure 167
configure for commands 67
configure for parameters and reports 169
dialog box fields 171
remove for a command 71
Index

W

watchdog
  item access path  85
  item name  85
  protocol  85
Windows-linked user  13

X

XML
  import enumerations  27
XML parser errors  223
Rockwell Automation support

Rockwell Automation provides technical information on the web to assist you in using its products. At [http://www.rockwellautomation.com/support](http://www.rockwellautomation.com/support) you can find technical and application notes, sample code, and links to software service packs. You can also visit our Support Center at [https://rockwellautomation.custhelp.com](https://rockwellautomation.custhelp.com) for software updates, support chats and forums, technical information, FAQs, and to sign up for product notification updates.

In addition, we offer multiple support programs for installation, configuration, and troubleshooting. For more information, contact your local distributor or Rockwell Automation representative, or visit [http://www.rockwellautomation.com/services/online-phone](http://www.rockwellautomation.com/services/online-phone).

Installation assistance

If you experience a problem within the first 24 hours of installation, review the information that is contained in this manual. You can contact Customer Support for initial help in getting your product up and running.

<table>
<thead>
<tr>
<th>United States or Canada</th>
<th>1.440.646.3434</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outside United States or Canada</td>
<td>Use the Worldwide Locator available at <a href="http://www.rockwellautomation.com/locations">http://www.rockwellautomation.com/locations</a>, or contact your local Rockwell Automation representative.</td>
</tr>
</tbody>
</table>

New product satisfaction return

Rockwell Automation tests all of its products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning and needs to be returned, follow these procedures.

<table>
<thead>
<tr>
<th>United States</th>
<th>Contact your distributor. You must provide a Customer Support case number (call the phone number above to obtain one) to your distributor to complete the return process.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outside United States</td>
<td>Please contact your local Rockwell Automation representative for the return procedure.</td>
</tr>
</tbody>
</table>

Documentation feedback

Your comments will help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete the feedback form, publication [RA-DU002](http://www.rockwellautomation.com/locations).

---

**www.rockwellautomation.com**
Important User Information

Solid-state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation, and Maintenance of Solid State Controls (publication SGI-1.1 available from your local Rockwell Automation sales office or online at http://www.rockwellautomation.com/literature/) describes some important differences between solid-state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid-state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

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.

**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.

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

---

Allen-Bradley, Rockwell Software, and Rockwell Automation ControlLogix, sProcedure, FactoryTalk, RSBeWare, RSBeWare BatchHistorian, RSBeWare Historian, RSBeWare MaterialTrack, and RSLink are trademarks of Rockwell Automation, Inc.

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

### Chapter 1

**FactoryTalk Batch Recipe Editor introduction**
- Open FactoryTalk Batch Recipe Editor ................................................................. 11
- Set FactoryTalk Batch Recipe Editor options ....................................................... 12
- FactoryTalk Batch Recipe Editor interface .......................................................... 13
- FactoryTalk Batch Recipe Editor menu bar .......................................................... 15
- FactoryTalk Batch Recipe Editor toolbar ............................................................. 16
- Recipe construction toolbox ............................................................................... 17
- Recipe creation overview ................................................................................... 18
- Additional resources ............................................................................................ 20

### Chapter 2

**Sequential Function Chart overview**
- SFC components .................................................................................................. 23
- SFC steps .................................................................................................................. 24
- Transition expressions ............................................................................................ 25
  - Transition operators ............................................................................................. 27
  - Parameter expression functions .......................................................................... 27
- Data types ................................................................................................................ 28
- Linked elements ...................................................................................................... 29
- SFC limitations ........................................................................................................ 30
- SFC execution .......................................................................................................... 31
- Null procedures ...................................................................................................... 32
  - Synchronize execution of steps overview ............................................................ 33
  - Arbitration .......................................................................................................... 33
  - Subarbitration ...................................................................................................... 35
- Add a null procedure .............................................................................................. 37
- Timer steps .............................................................................................................. 37
  - Timer step hold behavior .................................................................................... 38
- Recipe comments ................................................................................................... 38

### Chapter 3

**Table-based recipes overview**
- Recipe structure icons .......................................................................................... 42
- Table view transition expressions ......................................................................... 43
- Parallel structures ................................................................................................... 43
  - Simple parallelism example ................................................................................ 44
  - Complex parallelism example ............................................................................ 45

### Chapter 4

**Recipe creation**
- Procedure levels .................................................................................................... 47
- Unit requirements .................................................................................................. 48
## Table of contents

- Unit-based recipes ........................................................................................................... 49
- Unit class-based recipes ................................................................................................. 49
- Material-based recipes ................................................................................................ 50
- Material class-based recipes ......................................................................................... 50
- Dynamic unit allocation ................................................................................................. 50
  - How dynamic unit allocation affects unit binding ..................................................... 51
- Binding methods ........................................................................................................... 52
  - Criteria for unit selection ......................................................................................... 52
- Unit requirement name ................................................................................................. 53
- Downstream unit requirements .................................................................................. 54
- Enable dynamic unit allocation .................................................................................. 54
- Open recipe .................................................................................................................. 55
- Cannot open recipe error message ............................................................................. 57
- Create operations and unit procedures ......................................................................... 58
  - Change operation/unit procedure unit requirements .............................................. 60
- Create a procedure ....................................................................................................... 60
  - Unit requirement name example ............................................................................. 63
  - Change procedure unit requirements .................................................................. 64
- Smart binding .............................................................................................................. 65
  - Binding requirements .............................................................................................. 65
    - Configure binding requirements ........................................................................... 66
  - Binding preferences ................................................................................................ 67
    - Configure binding preferences ............................................................................ 68
- Recipe header data ...................................................................................................... 70
  - Add recipe header data ........................................................................................... 73
- Recipe storage .............................................................................................................. 74
- Copy recipe options .................................................................................................... 75
  - Copy a recipe using the backup directory ............................................................... 76

## Chapter 5

### Build a Sequential Function Chart

- Insert steps into an SFC .............................................................................................. 79
- Insert timer steps ....................................................................................................... 80
- Manually add steps to an SFC .................................................................................. 82
- Add a parallel step to an SFC ................................................................................... 82
  - When parallel steps require the same dedicated resource ........................................ 83
- Add a branch structure to an SFC ............................................................................. 84
- Add transitions to an SFC ......................................................................................... 85
- Define a step ............................................................................................................... 86
- Link SFC components ............................................................................................... 87
  - Remove a link from an SFC ................................................................................... 87
- Assign a transition expression .................................................................................. 88
- Remove a step from an SFC ..................................................................................... 88
- Create material loops ............................................................................................... 89
- Recipe comments ...................................................................................................... 90
  - Add comments to an SFC ....................................................................................... 91
## Table of contents

### Chapter 6

**Build a table**
- Insert steps into a table ................................................................. 95
- Add a parallel step to a table ............................................................ 96
- Add a step before and after parallel steps ........................................ 97
- Define a step .................................................................................. 97
- Remove a step from a table ............................................................... 98

### Chapter 7

**Bind Expressions**
- Expression operators expanded ....................................................... 100
  - Expression data types ................................................................. 100
  - Operands ..................................................................................... 101
- Expression validation .......................................................................... 101

### Chapter 8

**Phase, operation sequence, and recipe parameters**
- Phase parameters ............................................................................. 103
- Operation sequence parameters ...................................................... 104
- Recipe parameters ........................................................................... 104
  - Defer parameters ........................................................................ 105
- Step formula values and phase report limits .................................... 106
  - Material recipe parameters ........................................................ 107
  - Material report parameters ......................................................... 108
- Formula Value Entry/Report Expression settings .............................. 109
- Assign parameter values and phase report limits ............................ 112
- Assign formula values to Timer steps ............................................. 113
- Recipe properties ............................................................................ 115
  - Parameters ................................................................................ 115
  - Assign recipe formula parameters ............................................. 116
- Reports .......................................................................................... 116
  - Assign report parameters .......................................................... 117
  - Aggregate report values ............................................................. 118

### Chapter 9

**Expressions in parameters and reports**
- Evaluation of recipe parameter expressions .................................. 120
- Validation of parameters with expressions ..................................... 120
- Calculated recipe parameter expression value override .................. 121
- Evaluation of report parameter expressions ................................... 121
  - Override a calculated report expression ..................................... 121
Table of contents

Expression value configurations ................................................................. 122
  Configure expression values in recipe and report parameters .......... 122
  Expression operators ............................................................................. 125
  Parameter expression functions ............................................................ 125

Chapter 10

Phase link groups

Message partners ...................................................................................... 128
  Application of message partners example ......................................... 128
Link group rules ....................................................................................... 129
Phase communication ............................................................................... 130
Create a phase link group ...................................................................... 130
Delete a phase link group ...................................................................... 131

Recipe approval

Recipe approval process overview

Configure recipe approval ....................................................................... 134
Approve a recipe ..................................................................................... 135
Revert a recipe approval ......................................................................... 137
Automatic system signoff ($System) ......................................................... 139

Chapter 11

Recipe versioning

Recipe versioning overview

Recipe versioning .................................................................................. 141
How recipe versions are named .............................................................. 142
Restrictions for recipe versions ............................................................... 144
Enable recipe versioning option .............................................................. 145
Disable recipe versioning option ............................................................. 145
Create a recipe version .......................................................................... 146
Recipe version control ............................................................................ 146
  Recipe version history ................................................................. 146
  Obsoleted recipe versions ............................................................. 150
Author recipes using version control and recipe approvals .................. 151

Chapter 12

Security authority configuration .............................................................. 156
Secure recipe .......................................................................................... 157

Chapter 13

Complete and maintain recipe

Verify recipe ........................................................................................... 159
  What gets verified? ............................................................................. 160
SFC validation ........................................................................................ 161
  Run SFC validation .......................................................................... 162
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Set allowable SFC permutations</td>
<td>163</td>
</tr>
<tr>
<td>SFC validation error types</td>
<td>164</td>
</tr>
<tr>
<td>Parallel activation of a linear segment</td>
<td>165</td>
</tr>
<tr>
<td>Unreachable terminal step</td>
<td>166</td>
</tr>
<tr>
<td>Parallelism with terminal step</td>
<td>167</td>
</tr>
<tr>
<td>Linear segment cannot reach terminal step</td>
<td>168</td>
</tr>
<tr>
<td>Unreachable linear segment</td>
<td>169</td>
</tr>
<tr>
<td>Release Recipe as Step option</td>
<td>170</td>
</tr>
<tr>
<td>Release a recipe to production option</td>
<td>170</td>
</tr>
<tr>
<td>Rebuild the recipe directory</td>
<td>170</td>
</tr>
<tr>
<td>Recipe maintenance</td>
<td>171</td>
</tr>
<tr>
<td>Find Recipe References overview</td>
<td>171</td>
</tr>
<tr>
<td>Find recipe references</td>
<td>172</td>
</tr>
<tr>
<td>Page setup</td>
<td>173</td>
</tr>
<tr>
<td>Generate reports option</td>
<td>174</td>
</tr>
<tr>
<td>Print a working set of recipes</td>
<td>174</td>
</tr>
<tr>
<td>Print a single recipe</td>
<td>176</td>
</tr>
<tr>
<td>Remove a recipe</td>
<td>177</td>
</tr>
<tr>
<td>Translation</td>
<td>178</td>
</tr>
</tbody>
</table>

**Chapter 15**

Import recipes .................................................................................................................. 179
Select a directory ............................................................................................................ 182
Select a SQL Server database ......................................................................................... 183
Export recipes ................................................................................................................... 183
Import conflicts .............................................................................................................. 185
Resolve duplicate name conflict .................................................................................... 186
Resolve recipe basename conflict .................................................................................. 186
Resolve recipe version conflict ..................................................................................... 187
Resolve recipe conflict when rebuilding the recipe directory .................................. 188
Import and export restrictions for secured recipes ..................................................... 188

**Appendix A**

Recipe formats

RDB format .......................................................................................................................... 191
Create the MasterRecipes database ................................................................................. 192
Add a Windows user to the MasterRecipeAuthor and MasterRecipeViewer local groups .. 193
XML format .......................................................................................................................... 193
XML recipe schema ............................................................................................................ 195
Set recipe file format ....................................................................................................... 195
# Appendix B

## Import/Export error issues
- Improperly secured recipe ineligible for selection
- Verification results for a versioned recipe
- Solutions for common verification warnings and errors
  - Modify the area model to reinstate an obsolete procedure example
  - Use Check Out and Redefine Step to remove an area model conflict example
- Invalid recipe folder or directory path warning

## Legal Notices
- Legal Notices

## Index
- Legal Notices
FactoryTalk Batch Recipe Editor introduction

Use FactoryTalk Batch Recipe Editor to create and configure master recipes for use in batch automation. A master recipe is a type of recipe that accounts for equipment capabilities and may include process cell-specific information.

The interface contains IEC 61131-3 sequential function charts to graphically organize recipes into procedures, unit procedures, operations, and phases. Build recipes using either the SFC format or a table-based format.

Procedural levels:

Recipes use the ISA S88.01 Batch Control Standards for configuration and display, which define these levels for the procedural model:

- **Batch control**: Consists of a sequence of one or more steps (phases) that must be performed in a defined order for a finite period of time to process finite quantities of input material to produce finished product.

  Examples: Make Product A  Make Product B

- **Procedure**: The strategy for carrying out a process. In general, it refers to the strategy for making a batch within a process cell. It may refer to a process that does not result in the production of a product.

  Examples: Emulsification  Dehydrogenation

- **Unit Procedure**: A strategy for carrying out a contiguous process within a unit. It consists of contiguous operations and the algorithm necessary for the initiation, organization, and control of those operations.

  Examples: Filtration  Reaction

- **Phase**: The lowest level of the procedural element in the procedural control model. A phase consists of the control steps and the algorithm necessary for the initiation, organization, and control of those steps.
The recipe phase is the lowest level within a recipe. The phase provides an interface to basic control. The recipe phase maps directly to the engineered logic on the plant floor. Base an added phase on a phase already defined in the area model using the FactoryTalk Batch Equipment Editor. Based on the unit requirements specified in the recipe, a phase in the FactoryTalk Batch Equipment Editor maps to a specific phase (an instance of a phase class). This phase is mapped (using tags) to equipment phase defined in the engineered logic.

An operation consists of one or more recipe phases, and it must run within a single unit in the area model. Combine multiple operations into a single unit procedure if each operation runs in the same unit in the area model. Unit procedures combine to create a procedure, the highest recipe level. Procedures can run across multiple units, allowing for unit-to-unit transfers.

See also

Open FactoryTalk Batch Recipe Editor on page 10
Set FactoryTalk Batch Recipe Editor options on page 12
FactoryTalk Batch Recipe Editor interface on page 13
Recipe creation overview on page 18
Open FactoryTalk Batch Recipe Editor

Follow these instructions to open the FactoryTalk Batch Recipe Editor.

To open FactoryTalk Batch Recipe Editor:

1. Select **Start**, navigate to **All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch**, and then select **Recipe Editor**.

2. If the **Log on to FactoryTalk** dialog box opens, enter the user name and password and select **OK**.

   **Tip:** If **SingleSignOn** is configured, the **Log On to FactoryTalk** dialog box does not open. Proceed to step 5.

3. If the FactoryTalk user name and password that is not configured to run the FactoryTalk Batch Recipe Editor, this message opens.

   ![User Name and password is invalid...](image)

   4. If the user is not found, a **Logon Message** opens. Select **OK** to close this message, check the user credentials, and retry or enter other credentials.

5. If Security Authority is applied to the area model, and there is a mismatch between the security authority identifier (SAI) in the area model and the (SAI) in the current FactoryTalk Network Directory, this error message opens:

   ![The area model is secured to a different FactoryTalk Network Directory](image)
Make note of the FactoryTalk Network Directory and host computer information, then select **OK** to close the error message and exit the FactoryTalk Batch Recipe Editor.

6. If the recipe directory defined in the FactoryTalk Batch Equipment Editor contains binary recipes from a previous version of FactoryTalk Batch, a dialog box opens to translate the recipes to the new schema. After translation, verify the recipes.

   **Tip:** If upgrading from a FactoryTalk Batch version older than the previous major release (such as from version 10.x or earlier to version 12.x), contact the Rockwell Customer Support Representative to have recipes upgraded to the new recipe schema.

7. The **Recipes Requiring Verification** dialog box may open when the FactoryTalk Batch Recipe Editor opens. Select **Verify All** to verify any unverified recipes. Select **Verify and Validate All** to verify and validate the SFCs of any unverified recipes.

   **Important:** FactoryTalk users with ViewOnly permissions to the FactoryTalk Batch Recipe Editor cannot open the FactoryTalk Batch Recipe Editor to verify recipes. The FactoryTalk Batch Recipe Editor closes.

The FactoryTalk Batch Recipe Editor displays the **Procedure View** and **Recipe Construction** panes.

**See also**

- [Set FactoryTalk Batch Recipe Editor options on page 12](#)

---

**Set FactoryTalk Batch Recipe Editor options**

Configure the general FactoryTalk Batch Recipe Editor options, such as enabling/disabling support for dynamic unit allocation and parallel operations, as well as configuring the pane size in the FactoryTalk Batch Recipe Editor window. Selected options apply to all recipes saved after the options are applied.

**Tips:**

- FactoryTalk Batch procedures can have parallel unit procedures and operations can have parallel equipment phases, regardless of this options setting.
- To refer to the area model, open the FactoryTalk Batch Equipment Editor.

**To set FactoryTalk Batch Recipe Editor options:**

1. Select **View > Options**. The **Options** dialog box opens.

2. Set the FactoryTalk Batch Recipe Editor options as required.
3. Select **Apply** to save selections.

![Options dialog box]

These options are available in the **Options** dialog box:

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Support Dynamic Unit Allocation</td>
<td>Dynamic unit allocation enables the creation of Unit Class-based recipes with specific unit requirements (such as downstream unit requirements) and allows the choice of unit binding type to apply. If selected, the Procedure - Unit Requirements dialog box displays. The Unit Requirement menu selection becomes active in a procedure, and the dialog for defining a step includes a unit requirement selection.</td>
</tr>
<tr>
<td>Allow Parallel Operations</td>
<td>Select this option to allow the creation of AND convergences and divergences, which allow operations to run simultaneously within unit procedures.</td>
</tr>
<tr>
<td>Font Size</td>
<td>Set the font size for text displayed in the procedure view outline (left frame). The default is 7 points.</td>
</tr>
<tr>
<td>Number of Decimal Places for the Display Parameter</td>
<td>This option adjusts the number of decimal places displayed for real numbers in the SFC View. This value defaults to 2. The actual value of the parameter is unaltered. Only the number displayed in the FactoryTalk Batch Recipe Editor in the SFC View is affected.</td>
</tr>
<tr>
<td>Maximum SFC Permutations</td>
<td>The maximum number of SFC permutations (caused by OR Divergences within a recipe) that the SFC Validation Tool allows. Recipes structures exceeding this limit are too complex to verify. Please note that the higher this setting, the longer SFC validation takes.</td>
</tr>
</tbody>
</table>

See also

[FactoryTalk Batch Recipe Editor interface on page 13](#)

**FactoryTalk Batch Recipe Editor interface**

The FactoryTalk Batch Recipe Editor interface provides a workspace and tools to graphically construct recipes and specify a sequence of phases in an operation. Configure the actual phase logic in the process-connected device (PCD). Configure the interface to the PCD in the FactoryTalk Batch Equipment Editor.
The Procedure View Pane contains a hierarchical list of the current recipe components. Selecting a component from the list highlights the corresponding step in the Recipe Construction pane.

**Tip:** To resize the panes in the FactoryTalk Batch Recipe Editor, select and drag the splitter bar between the two panes to the desired size.

The Recipe Construction pane provides a way to construct and view recipe structures using a sequential function chart (SFC) or a table. The SFC view and the Table view display exclusively. Tile the Recipe Construction pane to display both views at the same time. Selecting a component within either view highlights the corresponding item in the Procedure View pane.

**Tip:** When using tiled views, selecting a recipe step in one view highlights the corresponding item in the other view. The selected step in the active view highlights in dark blue. The step in the inactive view highlights in light blue. Additionally, the status bar displays the word Table or SFC accordingly.

See also

- FactoryTalk Batch Recipe Editor menu bar on page 15
- FactoryTalk Batch Recipe Editor toolbar on page 15
### FactoryTalk Batch Recipe Editor menu bar

Opened recipe menu bar commands:

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>File</strong></td>
<td>Lists options for creating new recipes (New Top Level), open existing recipes (Open Top Level), close recipes, and save recipes. Use the File menu to create recipe versions (Check In), work-in-progress (WIP) copies (Check Out), remove recipes, rebuild the recipe directory, verify recipes, validate SFCs, import recipes, and export recipes. Secure an open recipe using Security Authority. The Page Setup (printer/paper choices), Generate Reports (print options), and Exit commands available from the File menu.</td>
</tr>
<tr>
<td><strong>Edit</strong></td>
<td>Select, add, or remove a recipe step within the SFC View and Table View. Select, add, or remove transitions, links, and material loops within the SFC View. The Edit menu contains the option to add, link, or delete recipe comments (Text Box). These commands are also available in the Recipe Construction Toolbox.</td>
</tr>
<tr>
<td><strong>Recipe</strong></td>
<td>Enter or view recipe header information, view recipe version parent information, approve or revert recipe approvals, add/delete recipe formula parameters and reports, edit unit/bind requirements, set bind preferences, and verify recipes.</td>
</tr>
<tr>
<td><strong>Step</strong></td>
<td>Allows redefine, enter formula values, parameter values, and report limits for a selected step.</td>
</tr>
<tr>
<td><strong>Link</strong></td>
<td>Create, edit, and delete phase link groups.</td>
</tr>
<tr>
<td><strong>Format</strong></td>
<td>Modify the magnification of the SFC view and align SFC elements to a grid.</td>
</tr>
</tbody>
</table>
| **View** | Switch between the SFC View, Table View, or to tile the Recipe Construction Pane for displaying both views. Customize the FactoryTalk Batch Recipe Editor by hiding/showing the Toolbar, Status Bar, and recipe construction Toolbox. Additionally, it allows show/hide the page boundaries and set the FactoryTalk Batch Recipe Editor options:  
  - Supporting dynamic unit allocation  
  - Parallel operations  
  - Formula parameter decimal places  
  - Maximum number of SFC validation OR permutations |
| **Help** | Provides online help and information about FactoryTalk Batch. |

See also

- FactoryTalk Batch Recipe Editor interface on page 13
FactoryTalk Batch Recipe Editor toolbar

The FactoryTalk Batch Recipe Editor toolbar contains buttons used to perform FactoryTalk Batch Recipe Editor commands. Repositioned or detached the toolbar from the window by selecting and dragging it to the desired position. When the toolbar is detached from the window, moved or hide by right-clicking it and selecting *Hide or Move* from the shortcut menu.

This table describes the FactoryTalk Batch Recipe Editor toolbar buttons and their corresponding keyboard shortcuts:

<table>
<thead>
<tr>
<th>Graphic</th>
<th>Name</th>
<th>Description</th>
<th>Keyboard Shortcut</th>
</tr>
</thead>
<tbody>
<tr>
<td><img src="image" alt="New Top Level" /></td>
<td>New Top Level</td>
<td>Creates a new procedure, unit procedure, or operation.</td>
<td>Ctrl+N</td>
</tr>
<tr>
<td><img src="image" alt="Open Top Level" /></td>
<td>Open Top Level</td>
<td>Opens an existing recipe file.</td>
<td>Ctrl+O</td>
</tr>
<tr>
<td><img src="image" alt="Save" /></td>
<td>Save</td>
<td>Saves the current recipe.</td>
<td>Ctrl+S</td>
</tr>
<tr>
<td><img src="image" alt="Check In" /></td>
<td>Check In</td>
<td>Check in and version the recipe element (create a recipe version).</td>
<td>None</td>
</tr>
<tr>
<td><img src="image" alt="Check Out" /></td>
<td>Check Out</td>
<td>Check out the versioned recipe to edit (create a work-in-progress (WIP) copy).</td>
<td>None</td>
</tr>
<tr>
<td><img src="image" alt="Generate Reports" /></td>
<td>Generate Reports</td>
<td>Opens the <em>Generate Reports</em> dialog box for printing and reporting options.</td>
<td>Ctrl+T</td>
</tr>
<tr>
<td><img src="image" alt="Delete Selection" /></td>
<td>Delete Selection</td>
<td>Deletes the currently selected steps, transitions, or text boxes.</td>
<td>Delete</td>
</tr>
<tr>
<td><img src="image" alt="Go Down" /></td>
<td>Go Down</td>
<td>Moves down through the recipe levels and displays the related recipe structure for the step.</td>
<td>None</td>
</tr>
<tr>
<td><img src="image" alt="Go Up" /></td>
<td>Go Up</td>
<td>Moves up through the recipe levels and displays the related recipe structure for the step.</td>
<td>None</td>
</tr>
<tr>
<td><img src="image" alt="Header Data" /></td>
<td>Header Data</td>
<td>Enter or change header data for the viewed recipe level, including the name.</td>
<td>Ctrl+H</td>
</tr>
<tr>
<td><img src="image" alt="Version History" /></td>
<td>Version History</td>
<td>View information for the currently open recipe and its previous version.</td>
<td>None</td>
</tr>
<tr>
<td><img src="image" alt="Recipe Formula Parameters" /></td>
<td>Recipe Formula Parameters</td>
<td>Create, edit, and delete recipe formula parameters.</td>
<td>Ctrl+R</td>
</tr>
<tr>
<td><img src="image" alt="Verify" /></td>
<td>Verify</td>
<td>Verifies recipes for proper construction and displays a list of unresolved problems with a recipe. Also saves the recipe in the process.</td>
<td>Ctrl+L</td>
</tr>
<tr>
<td><img src="image" alt="Verify and Validate" /></td>
<td>Verify and Validate</td>
<td>Verifies recipes based on the current unit requirements and validates the SFC structure. Also saves the recipe in the process.</td>
<td>Ctrl+N</td>
</tr>
<tr>
<td><img src="image" alt="Bind Requirements" /></td>
<td>Bind Requirements</td>
<td>Add or delete binding requirements assigned to a unit requirement. Available only with enabled Dynamic Unit Allocation.</td>
<td>Ctrl+M</td>
</tr>
</tbody>
</table>
The Recipe Construction toolbox contains the command buttons for creating and editing steps, transitions, text boxes, and links in the recipe construction pane. Reposition or detach the toolbox from the window by selecting and dragging it to the desired position. When the toolbox is detached from the window, move or hide it. Right-click the toolbox and select the appropriate option from the shortcut menu. Press Ctrl+Q to add or remove the toolbox from the FactoryTalk Batch Recipe Editor window.

The toolbox command buttons and the keyboard shortcut keys:

<table>
<thead>
<tr>
<th>Graphic</th>
<th>Name</th>
<th>Description</th>
<th>Keyboard Shortcut</th>
</tr>
</thead>
<tbody>
<tr>
<td>🗠️</td>
<td>Selection Tool</td>
<td>Selects a step in the Procedure View pane, Table view, or SFC view. Select transitions in the SFC view. Select and drag the selection pointer over several SFC elements to select a group of elements.</td>
<td>Shift+P</td>
</tr>
<tr>
<td>📋</td>
<td>Step Tool</td>
<td>Adds a new step to the SFC.</td>
<td>Shift+S</td>
</tr>
</tbody>
</table>

See also

FactoryTalk Batch Recipe Editor interface on page 13
## Graphic Name Description Keyboard Shortcut

<table>
<thead>
<tr>
<th>Graphic</th>
<th>Name</th>
<th>Description</th>
<th>Keyboard Shortcut</th>
</tr>
</thead>
<tbody>
<tr>
<td>+</td>
<td>Transition Tool</td>
<td>Adds a new transition to the SFC.</td>
<td>Shift+T</td>
</tr>
<tr>
<td></td>
<td>Link Tool</td>
<td>Links recipe components in the SFC view. Select a step and drag the cursor from the step to a transition to link the two components.</td>
<td>Shift+L</td>
</tr>
<tr>
<td></td>
<td>Remove Link Tool</td>
<td>Unlinks recipe components in the SFC view. Select a step and drag the cursor from the step to a transition to unlink the two components.</td>
<td>Shift+R</td>
</tr>
<tr>
<td></td>
<td>Text Box Tool</td>
<td>Adds a text box for recipe comments.</td>
<td>Shift+D</td>
</tr>
<tr>
<td></td>
<td>Add Step</td>
<td>Adds a step and transition after the selected recipe element. The Add Step tool is valid for initial, sequential, and parallel steps, as well as transitions.</td>
<td>Shift+A</td>
</tr>
<tr>
<td></td>
<td>Insert Step</td>
<td>Inserts a step and transition before the selected recipe element. The Insert Step tool is valid for sequential, parallel, and terminal steps, as well as transitions.</td>
<td>Shift+I Shift+Insert</td>
</tr>
<tr>
<td></td>
<td>Add Parallel</td>
<td>Adds a step in parallel with the selected step. The Add Parallel tool is valid for sequential and parallel steps only.</td>
<td>Shift+E</td>
</tr>
<tr>
<td></td>
<td>Add Branch</td>
<td>Adds a step and additional transitions to form a branch structure. The Add Branch tool is valid for regular steps only.</td>
<td>Shift+B</td>
</tr>
<tr>
<td></td>
<td>Remove Step</td>
<td>Removes a step from the SFC structure, automatically removing the transitions and rearranging the SFC. The Remove Step tool is valid for sequential and parallel steps only.</td>
<td>Shift+M</td>
</tr>
<tr>
<td></td>
<td>Insert Step Before Parallel</td>
<td>Inserts a step before a parallel structure. Select a step within the parallel structure to enable the tool.</td>
<td>Shift+O</td>
</tr>
<tr>
<td></td>
<td>Add Step After Parallel</td>
<td>Adds a step after a parallel structure. Select a step within the parallel structure to enable the tool.</td>
<td>Shift+F</td>
</tr>
<tr>
<td></td>
<td>Create Material Loop</td>
<td>Creates a material loop for the selected material enabled step. The step must have the preceding and following elements attached before creating the material loop.</td>
<td>Shift+N</td>
</tr>
<tr>
<td></td>
<td>Undo Material Loop</td>
<td>Removes a material loop just created, before any other editing takes place.</td>
<td>Shift+X</td>
</tr>
</tbody>
</table>

See also

[FactoryTalk Batch Recipe Editor interface on page 13](#)

### Recipe creation overview

A common best practice is to design a recipe top-down and build it bottom-up, focusing on high-level procedures that accurately depict the process used to make the product. In determining where to segment the process into Unit Procedures and Operations, consider breaking the process into reusable modules.

Each recipe level consists of descriptive information, formula information, unit requirements, and the required processes to make the batch. A batch is a running control recipe. The material that is being produced or that has been produced by a single execution of a recipe is also considered a batch.
A parameter is used to allow for flexibility in the recipe creation process. Parameter values are used in transition conditions or for substitution of phase parameters. A parameter is created for these recipe levels: operation, unit procedure, or procedure. Values are assigned when a unit procedure or operation step is added to a recipe. A recipe parameter is specific to a recipe and is on the recipe phase and downloaded to the equipment phase at runtime.

A phase link group identifies phases that may communicate and work together.

The processes involved in building a recipe:

**Recipe Creation**

For recipe creation:

- Determine the procedure levels.
- Specify unit requirements.
- Enter recipe header data.
- Define recipe parameters.
- Save the recipe.

**Tip:** After creating a new blank recipe, build the recipe structure by adding and configuring recipe steps and transitions. Represent the recipe structure using a SFC (sequential function chart), a table, or both. (See Build a Sequential Function Chart or Build a Table for more information.)

**SFC or Table Creation and Configuration**

For SFC or Table creation and configuration:

- Create a sequential function chart or table.
- Define the steps.
- Configure the step parameters.
- Define the transitions (SFC only).
- Create phase link groups.

**Recipe Completion**

To complete recipe creation:

- Verify the recipe and correct errors.
- (optional) Create a version of the recipe to set it to read-only and protect it from further modification.
• (optional) Complete the Recipe Approvals process, either in conjunction with the recipe version process, or on its own.

• Set the recipe property **Release recipe as step** to true, or signoff on it using the approval process.

• Set the recipe property **Release recipe to production** to true, or signoff on it by using the approval process.

• (optional) Secure the recipe using Security Authority. When secured, the recipe is bound to the Security Authority Identifier in the current FactoryTalk Network Directory. Afterwards, the same FactoryTalk Network Directory opens the recipe.

**Recipe Maintenance**

For recipe maintenance:

• Find recipe references.

• Create additional recipe versions, or work-in-progress (WIP) copies of a versioned recipe, to handle modifications to the area model, or changes to process and formula requirements.

• Print the recipe documentation.

• Remove a recipe file.

• Translate recipes.

**See also**

- [Build a Sequential Function Chart](#) on page 79
- [Build a Table](#) on page 95
- [Find Recipe References overview](#) on page 171

**Additional resources**

These documents contain additional information concerning related Rockwell Automation products.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FactoryTalk Batch ActiveX Controls User Guide</td>
<td>Quick and easy access to information and instructions required to use the FactoryTalk Batch ActiveX Custom Controls.</td>
</tr>
<tr>
<td>FactoryTalk Batch Administrator Guide</td>
<td>Instructions for configuring security and services, and implementation and use of components not normally accessed or used by batch operators, such as the FactoryTalk Batch Server.</td>
</tr>
</tbody>
</table>
Resource | Description
--- | ---
FactoryTalk Batch Equipment Editor User Guide | The FactoryTalk Batch Equipment Editor is a graphical interface used to create and maintain an equipment database. This process equipment database is called an area model. The area model is stored in a file with a .cfg file extension and is available to all other FactoryTalk Batch programs, including the Recipe Editor, View, and Phase Simulator.

FactoryTalk Batch Components Upgrade and Installation Guide | Provides information and procedures needed for installing FactoryTalk® Material Manager.

FactoryTalk Batch View User Guide | FactoryTalk Batch View is used to initiate and execute FactoryTalk Batch automation processing. FactoryTalk Batch View secured objects are located in the FactoryTalk Diagnostics and are modified using the FactoryTalk Administration Console. A system administrator can customize FactoryTalk Batch View security to meet the needs of the facility. FactoryTalk Batch View is used in conjunction with a Human-Machine Interface (HMI).

View or download publications at [http://www.rockwellautomation.com/literature](http://www.rockwellautomation.com/literature). To order paper copies of technical documentation, contact the local Rockwell Automation distributor or sales representative.
Chapter 2

Sequential Function Chart overview

In the FactoryTalk Batch Recipe Editor, create a sequential function chart (SFC) to represent the recipe logic flow. SFC provides a more flexible alternative to creating recipes. SFC methodologies define each recipe procedure and equipment.

Benefits of SFC:

- Unambiguous representation of logic.
- Support for all important logic structures: decisions, loops, and parallelism.
- Standardized in IEC 61131-3 specification.

See also

- SFC components on page 23
- Steps on page 24
- Transition expressions on page 25
- SFC limitations on page 30
- SFC execution on page 31

SFC components

Sequential function charts are composed of six graphical structures linked to form a chart. Each of the graphical structures represents an element in the recipe. The rules for execution of an SFC describe how the procedure executes. The table describes each recipe element and the graphical structure used in the SFC.

<table>
<thead>
<tr>
<th>Step</th>
<th>Purpose</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Initial Step</td>
<td>The logical start of the SFC.</td>
<td></td>
</tr>
<tr>
<td>Step</td>
<td>A reference to a subordinate recipe element.</td>
<td></td>
</tr>
<tr>
<td>Final Step</td>
<td>The logical end of the SFC.</td>
<td></td>
</tr>
<tr>
<td>Transition</td>
<td>Defines how recipe control moves from step to step.</td>
<td></td>
</tr>
<tr>
<td>OR Divergence</td>
<td>Represents a decision, where recipe control passes to only one of the subsequent steps.</td>
<td></td>
</tr>
</tbody>
</table>
### Sequential Function Chart overview

<table>
<thead>
<tr>
<th>Step</th>
<th>Purpose</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>AND Divergence</td>
<td>Represents multiple procedures processed concurrently.</td>
<td><img src="AND_Divergence" alt="Diagram" /></td>
</tr>
<tr>
<td>OR Convergence</td>
<td>Indicates where two optional execution paths converge back into one execution path.</td>
<td><img src="OR_Convergence" alt="Diagram" /></td>
</tr>
<tr>
<td>AND Convergence</td>
<td>Indicates where two simultaneous execution paths converge back into one execution path.</td>
<td><img src="AND_Convergence" alt="Diagram" /></td>
</tr>
</tbody>
</table>

See also

*Sequential Function Chart overview* on page 23

### SFC steps

A recipe can contain up to four procedure levels:

- Procedure
- Unit Procedure
- Operation
- Phase

Each of the first three procedure levels can contain one or more steps, each referencing a lower level procedure. A step in a procedure references a unit procedure. A step in a unit procedure references an operation. And a step in an operation references a phase. Define phases in the area model using the FactoryTalk Batch Equipment Editor. A phase is a specific instance of a phase class. The phase maps to the equipment phase, which is the engineered logic residing in the process-connected device (PCD).

This diagram depicts the relationships between the recipe procedure levels and the phase logic in the PCD. Equipment phases and phases are the links between the recipe and the engineered logic.
See also

**SFC components** on page 23

Transitions are the criteria that must be met before the recipe continues to the next logical step. Transition expressions are evaluated for syntax at the time they are created and are designated in the SFC with the name Tn, where n represents a unique number. The expression associated with each transition follows the traditional expression evaluation for the arithmetic operators (+, -, *, and /), the logical operators (AND, OR, and NOT), and the functions (ABS, MOD, TRUN, RND, and RDUP). The expression must always evaluate to either TRUE or FALSE.

With transfer of control, a running equipment phase receives notifications as expressions become TRUE that new input parameter values are available. This is modeled by a series of steps referencing the phase to run continuously. Each step contains the new set of inputs. The transitions between the steps test the process conditions. The equipment phase does not stop running until the transfer of control series is COMPLETE.

**Tip:**

The maximum length of a transition expression is 1023 characters. If the transition expression exceeds this limit, reorganize the parallel steps in the recipe.

<table>
<thead>
<tr>
<th>Transition Identifiers</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRUE</td>
<td>PREMIX_A.L101</td>
</tr>
<tr>
<td>FALSE</td>
<td></td>
</tr>
<tr>
<td>&lt;unit&gt;.&lt;unit tag&gt;</td>
<td></td>
</tr>
<tr>
<td>&lt;unit&gt; represents one of the pre-configured units, and &lt;unit tag&gt; is a legal tag for the &lt;unit&gt;</td>
<td></td>
</tr>
</tbody>
</table>
### <unit class>.<tag class>
For class-based recipes, specify the name of the `<unit class>` and an associated `<tag class>`

<table>
<thead>
<tr>
<th><code>&lt;unit class&gt;.&lt;tag class&gt;</code></th>
<th>PREMIX.LEVEL</th>
</tr>
</thead>
</table>

### `<step name>.<step attribute>`
`<step name>` is the name of one of the steps in the same level of the recipe
`<step attribute>` is one of the following:

<table>
<thead>
<tr>
<th><code>&lt;step name&gt;.&lt;step attribute&gt;</code></th>
<th>XFR_OUT_1:1.FAILURE = 0</th>
</tr>
</thead>
<tbody>
<tr>
<td>FAILURE</td>
<td></td>
</tr>
<tr>
<td>The FAILURE attribute is zero if no error has been encountered. Otherwise, its value is greater than zero.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><code>&lt;step name&gt;.&lt;step attribute&gt;</code></th>
<th>XFR_OUT_1:1.OWNERID = PROGRAM</th>
</tr>
</thead>
<tbody>
<tr>
<td>OWNERID</td>
<td></td>
</tr>
<tr>
<td>The OWNERID attribute can be either PROGRAM or EXTERNAL, where PROGRAM signifies the batch is controlling the step and EXTERNAL means that the batch is not controlling the step.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><code>&lt;step name&gt;.&lt;step attribute&gt;</code></th>
<th>XFR_OUT_1:1.PAUSE = TRUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAUSE</td>
<td></td>
</tr>
<tr>
<td>Set by the FactoryTalk Batch Server. Used to command the phase to pause at the next programmed pause transition. Can have the value of TRUE or FALSE.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><code>&lt;step name&gt;.&lt;step attribute&gt;</code></th>
<th>XFR_OUT_1:1.PAUSED = FALSE</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAUSED</td>
<td></td>
</tr>
<tr>
<td>Set by phase logic when the phase reaches a pause location. Can have the value of TRUE or FALSE</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><code>&lt;step name&gt;.&lt;step attribute&gt;</code></th>
<th>XFR_OUT_1:1.SINGLESTEP= TRUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>SINGLESTEP</td>
<td></td>
</tr>
<tr>
<td>Set by the FactoryTalk Batch Server. Allows the phase to pause at each location. Can have the value of TRUE or FALSE</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><code>&lt;step name&gt;.&lt;step attribute&gt;</code></th>
<th>XFR_OUT_1:1.STATE = COMPLETE</th>
</tr>
</thead>
<tbody>
<tr>
<td>STATE</td>
<td></td>
</tr>
<tr>
<td>Legal STATE identifiers are: ABORTING, HOLDING, STOPPING, STARTING, RESTARTING, RUNNING, HELD, COMPLETE, STOPPED, ABORTED, IDLE, and READY</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><code>&lt;step name&gt;.&lt;step attribute&gt;</code></th>
<th>XFR_OUT_1:1.STEPINDEX= 10</th>
</tr>
</thead>
<tbody>
<tr>
<td>STEPINDEX</td>
<td></td>
</tr>
<tr>
<td>An integer which represents the current step index of the active phase.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><code>&lt;step name&gt;.&lt;report parameter&gt;</code></th>
<th>XFR_OUT_1:1.AMOUNTTRANSFERRED</th>
</tr>
</thead>
<tbody>
<tr>
<td>If the step represents a recipe level, phase, or operation sequence, it may have report parameters. Report parameters contain values of tags written from a PCD/phase to FactoryTalk Batch.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><code>&lt;step name&gt;.&lt;recipe formula parameter&gt;</code></th>
<th>XFR_OUT_1:1.AMOUNT_TO_TRANSFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recipe formula parameters for a step contain the values for the recipe level, phase, or operation sequence represented by the step.</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th><code>&lt;recipe formula parameter&gt;</code></th>
<th>AMOUNT_TO_TRANSFER</th>
</tr>
</thead>
<tbody>
<tr>
<td>Each level of the recipe may have recipe formula parameters. The values for these parameters may be found in higher levels of the recipe.</td>
<td></td>
</tr>
</tbody>
</table>

---

**See also**

- [Transition operators](#) on page 26
- [Transition expression data types](#) on page 28
Transition operators

Transition expressions support these operators. The precedence of the execution depicts from highest to lowest. An operator with a higher precedence executes before an operator of lower precedence.

<table>
<thead>
<tr>
<th>Transition Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( )</td>
<td>Expressions within parentheses evaluates before expressions outside of parentheses.</td>
</tr>
<tr>
<td>NOT, unary minus</td>
<td>Logical NOT user, and &quot;-&quot;.</td>
</tr>
<tr>
<td>*, /, AND</td>
<td>Multiplication, division, and logical AND.</td>
</tr>
<tr>
<td>+, -, OR</td>
<td>Addition, subtraction, and logical OR.</td>
</tr>
<tr>
<td>&lt;, &lt;=, &gt;, &gt;=</td>
<td>Less than, less than or equal to, greater than, and greater than or equal to.</td>
</tr>
<tr>
<td>=, &lt;&gt;</td>
<td>Equal to, and not equal to.</td>
</tr>
</tbody>
</table>

See also

Parameter expression functions on page 27

Sequential Function Charts overview on page 23

Parameter expression functions

Functions determine how the expression parser handles Real and Integer data types used in a parameter expression. This table lists available functions and their behavior on positive and negative values:

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
<th>Behavior: Value</th>
<th>Behavior: Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>RND( )</td>
<td>Round – Numeric values round to the nearest integer.</td>
<td>6.7 6.5 6.3 -6.7 -6.5 -6.3</td>
<td>7 7 6 -7 -6 -6</td>
</tr>
<tr>
<td>RDUP( )</td>
<td>Round up – Numeric values round to the next larger integer.</td>
<td>6.7 6.5 6.3 -6.7 -6.5 -6.3</td>
<td>7 7 7 -6 -6 -6</td>
</tr>
<tr>
<td>TRNC( )</td>
<td>Truncate – Retains only the integer portion of the numeric value.</td>
<td>6.7 6.5 6.3 -6.7 -6.5 -6.3</td>
<td>6 6 6 -6 -6 -6</td>
</tr>
</tbody>
</table>
## Data types

The data types supported are integer, real, string, and enumeration. The following are data type examples:

**Integer:** 423

**Real:** 423.123456789012

**String:** The string constant must be in quotes: "READY".

**Enumeration:** As a string, the enumeration constant must be in quotes: "BUTTER_PECAN". As an integer, the ordinal for the enumeration may be: 4.

**Tip:** If the result of the expression is an Integer, the values used to build the expression must be Integers — Real is not compatible with an Integer. However, using division in an expression always results in the value being a Real number.

See also

- *Transition operators* on page 26
- *Transition expressions* on page 25
- *Sequential Function Charts overview* on page 23
Linked elements

Sequential function charts must follow a step-transition-step pattern to be valid; a step links to a transition and a transition links to a step. Recipe elements link to form a simple sequence (step-transition-step), or take any of these paths:

**SFC Example #1**

Decision logic (OR path) allows an SFC to flow in one direction or another depending on the choice made by the transition statement. A single horizontal line connects the two statements. Create the decision construct of an SFC by selecting two parallel steps and connecting them to the step above them with two separate transitions.

Parallel logic (AND path) allows more than one step to be active concurrently. A double horizontal line connects the two statements. Create the parallel construction of an SFC by selecting one transition and connecting it to two parallel steps.

The double lines of a parallel structure do not count as either a step or a transition.
SFC Example #2

Loop logic (REPEAT path) allows a process to flow from beginning to end and back again to a previous step where it repeats the loop. Create the loop construction of an SFC by joining a transition to a previous step. The single lines of a decision branch do not count as either a step or a transition.

Tip: To create a material rebinding loop, select a material-enabled step and then select the Create Material Loop tool. Loop logic allows a process to flow back to a previous step where it repeats the loop. The system automatically includes a null procedure in the material loop for rebinding.

See also

Recipe construction toolbox on page 17
Create material loops on page 89
Subarbitration on page 35

SFC limitations

Use the FactoryTalk Batch Recipe Editor to build an SFC of any size. For very large procedures, divide into multiple procedures each with fewer steps.

The maximum length of a transition expression is 1023 characters. If the transition expression exceeds this limit, reorganize the parallel steps in the recipe.

Important: If a batch has more than nine material-enabled, class-based phases running in parallel, the FactoryTalk Batch Server will run out of memory and will not write any messages to the Server log.

See also

Transition expressions on page 25
SFC execution

This is an overview of SFC execution:

- **Step 1:** Upon activation of this recipe, the initial step is active (indicated by the step color).

- **Step 2:** Control passes to Step A after Transition 1 becomes TRUE.

- **Step 3:** After Transition 2 becomes TRUE, Step A deactivates and Steps B and C become active.

- **Step 4:** After Transition 3 becomes TRUE, Step B deactivates and Step D activates.

- **Step 5:** When Transition 4 becomes TRUE, Steps C and D deactivate and Step E activates.
• Step 6: When Transition 5 becomes TRUE, Step E deactivates and the final step activates. This indicates that the entire recipe is complete.

See also

SFC limitations on page 30

Null procedures

A null procedure contains no parameters and is not associated with a phase on the plant floor. Use null procedures to follow the step-transition-step pattern in a sequential function chart (SFC) and enable the SFC to do arbitration that is more flexible. This creates otherwise impossible SFC structures, such as separating consecutive transitions.

Add null procedures at the unit procedure, operation, and phase levels. Null procedures automatically provided in the Unit Procedure Select, Operation Select, and Select Phase dialog boxes; there is no need to create null phases in the FactoryTalk Batch Equipment Editor.

See also

Synchronize execution of steps overview on page 32

Arbitration on page 33

Subarbitration on page 35

Add null procedures on page 36
Synchronize execution of steps overview

One powerful application of null procedures is to synchronize the execution of steps. In this picture, the CHARGE_A_CLS:1 step and the CHARGE_B_CLS:1 step execute in parallel. The execution of the CHARGE_B_CLS:1 step is does not occur until the CHARGE_A_CLS:1 step has already charged 1500 liters. To accomplish this timing, place a null procedure at the beginning of the parallel structure. A new transition creates before the CHARGE_B_CLS:1 step. This new transition contains a transition condition that allows the CHARGE_B_CLS:1 step to execute only after the CHARGE_A_CLS:1 step has charged 1500 liters.

See also

- Arbitration on page 33
- Subarbitration on page 35
- Add a null procedure on page 36

Arbitration

Use Null procedures are to regulate the arbitration of resources. A step cannot release any allocated resources it owns until the transition after the step becomes true. When this step is part of a parallel structure, a delay in releasing allocated resources may occur while the step waits for the other steps within the parallel structure to complete. Similarly, if the step is a shared phase, the phase is not be released until the other parallel steps have completed.

This picture illustrates the arbitration problem that can occur in a parallel SFC structure. If the AGITATE_PM_CLS:1 step is a shared phase, it is not released until CHARGE_B_CLS:1 and HEAT_CLS:1 have completed. Similarly, if AGITATE_PM_CLS:1 has acquired any shared resources, those resources are not released until all steps within the parallel structure are complete. A shared resource is a resource used in parallel by an unlimited number of steps at a time inside a
recipe structure. Any other procedures waiting for resources owned by AGITATE_PM_CLS:1 are forced to wait for those resources, even though the resources may be available.

This picture illustrates the same parallel structure, using a null procedure to correct the delay. The null phase inserts below the AGITATE_PM_CLS:1 step, allowing a transition above the null phase. This transition allows the AGITATE_PM_CLS:1 step to complete, even if other phases within the parallel structure have not, thereby releasing the phase if shared, and releasing any shared resources.

See also

Subarbitration on page 35
Subarbitration

Through the process of subarbitration, the FactoryTalk Batch Server determines how dedicated resources within a recipe procedure allocate when there are common resources requested at the same time. This can occur when a recipe with a parallel structure has parallel steps that require the same dedicated resource. If the parallel steps occur just after an AND Divergence or just before an AND Convergence, must add null procedures in order to run batches successfully.

This image shows an AND Divergence of two parallel steps requesting the same resource. This structure results in the batch HELD until the conflict is resolved and the errors clear. Because both steps are requesting the same dedicated resource at exactly the same time, the resource does not release to either one of them.

This image shows the same parallel structure with the addition of two null phases after the AND Divergence. This changes the timing just enough to allow one side to process before the other. The FactoryTalk Batch Server can then sub-arbitrate the resources required for ADD_CREAM:1 and ADD_CREAM:2.
This image shows an AND Convergence of two parallel steps requesting the same dedicated resource. In this case, the recipe appears to continue running, but the AGITATE:1 step will not release its resource. This prevents the AGITATE:2 step from acquiring the resource. The batch will not progress to the convergence transition.

This image shows the same parallel structure with the addition of two null procedures before the AND Convergence. The Server can then release the resources and transition past the convergence.

See also

Add a null procedure on page 36
Add a null procedure

Add a null procedure instructions are similar to insert a step. The NULL option is always available in the Unit Procedure Select, Operation Select, and Select Phase dialog boxes.

To add a null procedure:

1. Using the Selection Tool, select a step or transition that either precedes or follows the new step.

2. Select Insert Step to insert a new step before the selected element, or select Add Step to insert a new step after the selected element. A new UNDEFINED step displays on the SFC with a blue box outlined in a black dashed line. Additionally, the SFC structure automatically rearranges to make room for the new elements. One of these dialog boxes opens, based on the recipe level: Select Phase, Operation Select, or Unit Procedure Select.

3. Select the $Null option.

4. Select OK.

See also

Insert steps into an SFC on page 79

Define a step on page 86

Timer steps

Use timer steps to create a step that causes a branch in a recipe to pause for a specified period or to create a step that monitors the amount of time a parallel step executes. The TIMER option is always available in the Unit Procedure Select, Operation Select, and Select Phase dialog boxes.
The two TIMER types have these parameters and reports:

<table>
<thead>
<tr>
<th>Timer Type</th>
<th>Parameters</th>
<th>Reports</th>
</tr>
</thead>
<tbody>
<tr>
<td>COUNT_DOWN</td>
<td>• TIMER_TYPE&lt;br&gt;• HOLD_BEHAVIOR (CONTINUE, RETENTIVE, and RESET)&lt;br&gt;• SETPOINT</td>
<td>• ELAPSED_TIME&lt;br&gt;• REMAINING_TIME</td>
</tr>
<tr>
<td>COUNT_UP</td>
<td>• TIMER_TYPE&lt;br&gt;• HOLD_BEHAVIOR (CONTINUE, RETENTIVE, and RESET)</td>
<td>ELAPSED_TIME</td>
</tr>
</tbody>
</table>

See also

- Insert timer steps on page 80
- Phase and recipe parameters on page 103
- Timer step hold behavior on page 38

**Timer step hold behavior**

When a Timer step restarts:

- If configured the Timer as CONTINUE, the Timer step will continue timing using the initial start time while in the HELD state. The Timer procedure accumulates time in the RUNNING and HELD states based on the initial start time.
- If configured the Timer as RESET, the Timer step is set to restart timing from 0.
- If configured the Timer as RETENTIVE, upon entering the RUNNING state, the timer will continue timing from the last known ELAPSED_TIME value from when the timer went into HELD.

Once these actions are completed, the Timer procedure transitions to the RUNNING state. If the Timer procedure cannot determine the type of hold behavior configured, an error generates, causing the recipe to hold based on the configured hold propagation.

See also

- Timer steps on page 37

**Recipe comments**

Use recipe comments to give written instructions to operators, share notes with other engineers, or access a reference. Recipe commenting associates data with a step, transition, or the entire recipe. The comment is viewable at design and at run time.
The comment, and its associated information, is a text box. Above each text box, the assigned reference name displays as a C followed by a number. View added SFC text boxes in the SFC views of FactoryTalk Batch View, FactoryTalk Procedure, and ActiveX Control.

See also

Sequential Function Chart overview on page 23

Recipe creation overview on page 18
Chapter 3

Table-based recipes overview

When you build a recipe in the FactoryTalk Batch Recipe Editor you can represent the logic flow of the recipe using a sequential function chart (SFC) or a table. Table-based recipes provide a mechanism for creating simple recipes that do not require a complex recipe structure or complex transition expressions. Additionally, they allow you to view and edit all recipe parameters without having to navigate between steps. The logic flow of the recipe is represented using a sequential function chart (SFC) or a table. Table-based recipes provide a mechanism for creating simple recipes that do not require a complex recipe structure or complex transition expressions. Additionally, they allow viewing and editing of all recipe parameters without having to navigate between steps.

Recipe structures created in the Table view provide a list-based representation of recipe logic flow. Steps execute in order, starting with the top row, and proceeding downward through the table. Each row consists of an individual phase and its associated parameters. When selecting a phase row, the column headings show the parameter names for that phase. When selecting a different phase, the column headings change to reflect the selected phase parameters.

Steps are executed starting with row one, and proceeding downward until the end of the recipe. Steps located within a single parallel structure execute at the same time. Every step within the Table view displays the name of the Procedure, Unit Procedure, Operation, or Phase represented by the step. All parameters associated with the selected step display as the column header. The parameter value displays below the parameter name and to the right of the step name. To edit parameter values, select the table cell where the parameter displays and enter a new value (if enumerations are created for the selected parameter, a list containing the enumerations is provided).
Recipe structure icons

The Table view provides a graphical representation of a recipe structure in the form of a structure icon. The Table view is only capable of displaying the graphical recipe structure for simple recipes. Simple recipes are recipes that do not have complex recipe structures such as loops and branches. Simple recipes only have simple parallel structures. No structure icon displays for complex recipe structures viewed in the Table view.

Every step within the Table view displays the name of the Procedure, Unit Procedure, Operation, or Phase represented by the step.

The Table view can contain any of these recipe structure icons:

<table>
<thead>
<tr>
<th>Graphic</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Simple Step</td>
<td>Represents a single step within the recipe path.</td>
</tr>
<tr>
<td></td>
<td>Begin Parallel Step</td>
<td>Represents the first step in a parallel structure containing multiple steps.</td>
</tr>
<tr>
<td></td>
<td>Parallel Step</td>
<td>Represents the step located within a parallel structure containing multiple steps.</td>
</tr>
<tr>
<td></td>
<td>End Parallel Step</td>
<td>Represents the final step in a parallel structure containing multiple steps.</td>
</tr>
</tbody>
</table>

See also

Recipe structure icons on page 42
Assign recipe formula parameters on page 116

Table-based recipes overview on page 41
Table view transition expressions

Each recipe step has an associated transition. Transitions designate criteria that must be true before the recipe continues to the next logical step. The transition expression must always evaluate to either TRUE or FALSE. Transitions generate automatically for all table-based recipe steps when created. By default, all table-based recipe transitions contain the transition expression `<STEP>.STATE = COMPLETE`, where `<STEP>` is the step associated with the transition.

Tip: Transition expressions cannot be edited using the Table view. To edit a transition expression, use the Transition Expression Builder dialog box in the SFC view.

See also

Assign a transition expression on page 88

Table-based recipes overview on page 41

Parallel structures

The FactoryTalk Batch Recipe Editor allows creation of simple parallel structures using the Table view. These structures can have multiple steps in parallel, but cannot have more than one step located on the same path. That is, all steps located within the parallel structure must be preceded by the same AND divergence and followed by the same AND convergence.

More complex parallel structures created using the SFC view are considered undetermined when viewed in the Table view. Recipe structure icons do not display for the entire recipe level in which the parallel structure is located.

See also

Simple parallelism example on page 43

Complex parallelism example on page 44
Simple parallelism example

This table contains an example of a simple parallel structure displayed in the SFC view and in the Table view.

### SFC View:

```
SFC View:
```

### Table View:

<table>
<thead>
<tr>
<th></th>
<th>STEPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CHARGE_C_CLS:1</td>
</tr>
<tr>
<td>2</td>
<td>AGITATE_CLS:1</td>
</tr>
<tr>
<td>3</td>
<td>MILL_CLS:1</td>
</tr>
</tbody>
</table>

**Tip:** If parallel steps require the same dedicated resource, the FactoryTalk Batch Server automatically determines how the resources allocate among steps when the batch runs (called subarbitration). If parallel steps occur just after an AND divergence or just before an AND convergence, add null procedures to the recipe in order for batches to run successfully.

**See also**

[Parallel structures on page 43](#)
Complex parallelism example

This table contains an example of a complex parallel structure created in the SFC view and how that same structure displays in the Table view. Notice that there are no recipe structure icons displayed for the entire recipe level in the Table view.

SFC View:

Table View:

<table>
<thead>
<tr>
<th></th>
<th>STEPS</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CHARGE C CLS:1</td>
</tr>
<tr>
<td>2</td>
<td>AGITATE CLS:1</td>
</tr>
<tr>
<td>3</td>
<td>CHARGE D CLS:2</td>
</tr>
<tr>
<td>4</td>
<td>MILL CLS:1</td>
</tr>
</tbody>
</table>

See also

Parallel structures on page 43
Recipe creation

Before building an SFC or Table structure to define the logical flow of a recipe, first create the recipe in FactoryTalk. The recipe creation process includes:

- Determine the recipe level
- Specify unit requirements
- Define recipe formula parameters
- Enter recipe header data

See also

Procedure levels on page 47
Unit requirements on page 48
Dynamic unit allocation on page 50
Open recipe on page 55
Create operations and unit procedures on page 58

Procedure levels

The first step in planning a new recipe is determining the level of the procedure. A batch recipe may consist of up to four procedure levels. These levels follow the S88.01 Procedural Model:

- Procedure
- Unit Procedure
- Operation
- Phase
Procedure-level recipes contain steps called unit procedures. Unit procedure-level recipes contain steps called operations. Operation-level recipes contain steps called phases. Because of this hierarchical structure of the S88.01 Procedural Model, plan recipes from the top down, but build recipes from the bottom up. First, create the operation, then create the unit procedure, and lastly create the procedure.

If a recipe is run as a single unit (no unit-to-unit transfers are involved), then building a recipe as an operation may be adequate. If the recipe intends to operate across multiple units, create a procedure containing a unit procedure for each unit involved. Each unit procedure may consist of one or more operations.

To begin, have a good definition of what the recipe needs to do, then decide what process levels are needed. Start by building the operations and configuring the phases. Add the operations to the unit procedures, then add the unit procedures to the top-level procedure.

Tip: Every unit procedure must have at least one operation, and every procedure must have at least one unit procedure.

See also

Unit requirements on page 48

Unit requirements

When initially building a recipe, define two things:

- The procedure level: procedure, unit procedure, or operation.
- The recipe unit requirements.

To properly define the recipe requirements, have a good understanding of the area model, the equipment classes, and the actual equipment required for recipe execution. If phases are material-enabled, know what materials and containers needed for the recipe.

Prior to creating a recipe, determine if the recipe is unit-based, class-based, or a combination of the two. If Dynamic Unit Allocation is enabled and the recipe is a procedure containing some or all class-based unit procedure and operation level recipes, choose a unit allocation method.

See also

Unit-based recipes on page 49

Unit class-based recipes on page 49

Material-based recipes on page 49

Material class-based recipes on page 50
Dynamic unit allocation on page 50

**Unit-based recipes**

A unit-based recipe includes information regarding the specific units used for the recipe. During recipe creation, a unit-based recipe is required. Unit-based recipes use static binding and are bound to a specific unit instance that defined in the area model.

Also, unit-based recipes include information regarding the specific unit instances used for the recipe. Define unit instances in the area model. During recipe creation, the recipe author specifies the required unit instance. As a result, unit-based recipes use a binding method called static binding. Static binding means that the recipe can run in only one unit instance. At batch creation time, unit-based recipes automatically bind to the specified unit instance.

See also

**Unit requirements on page 48**

**Unit class-based recipes**

Create class-based recipes for a unit class rather than for a specific unit instance. Class-based recipes are useful if a unit class contains many unit instances and a particular recipe runs in most of those unit instances. The use of class-based recipes reduces the number of recipes created and maintained. Create only one recipe per unit class rather than one recipe for each unit instance. At batch creation time, the operator assigns the specific unit instance in which the class-based recipe runs, or FactoryTalk Batch Server availability determines assignment of the unit.

When creating a recipe, the phases that are not common to all the unit instances built from a particular unit class are available for selection. When adding a non-common phase to the recipe, the FactoryTalk Batch Recipe Editor automatically adds a Require Phase bind requirement to the unit requirements so that only the units having the phase are selectable at run time.

Enabling Dynamic Unit Allocation on a procedure level class-based recipe allows either the FactoryTalk Batch Server or the operator to select which unit instance of the unit class to use for each class-based unit procedure contained in the procedure. Select the unit instance either at batch creation or after the batch has started, depending on the selected unit allocation method.

See also

**Dynamic unit allocation on page 50**
Material-based recipes

Material-based recipes contain steps that specify the materials needed to run the recipe without requiring a definition of the equipment that is required to supply those materials. Each phase in the FactoryTalk Batch Equipment Editor contains configurations for material additions and distributions. Material-based recipes, unit classes, and the FactoryTalk Batch Material Editor drastically reduced the number of different recipes needed.

See also

Unit requirements on page 48

Material class-based recipes

Material class-based recipes contain steps configured to utilize any of several similar materials from a common class of materials. Configure material classes and specific materials in the FactoryTalk Batch Material Editor.

A material-enabled phase is a phase configured by the FactoryTalk Batch Equipment Editor and stored in the area model, which is enabled to support the specification of a material as a means to find appropriate equipment and bind to that equipment in a control recipe.

Configure material-enabled phases to be material class-based in the FactoryTalk Batch Equipment Editor. In the FactoryTalk Batch Recipe Editor, insert these material class-based phases into a recipe. When adding the recipe to the batch list in the FactoryTalk Batch View, the operator selects a specific material from the class of materials. Using material class-based recipes reduces the number of recipes needed. In the FactoryTalk Batch Recipe Editor, create one recipe and change the configuration of that recipe to be material class-based or material-specific.

See also

Unit requirements on page 48

Dynamic unit allocation

Dynamic unit allocation is a method for binding specific units to class-based unit procedure-level recipes that are contained within a procedure-level recipe. Dynamic Unit Allocation allows different unit procedures with the same unit requirements within the procedure. Define any downstream units required by unit instances and unit classes.

Downstream unit requirements may be defined for class-based and instance-based unit procedures. Downstream unit requirements are used to control batch flow and to make sure that the appropriate unit is selected during unit-binding.
This is an example procedure-level PREMIX recipe:

When not enabling Dynamic Unit Allocation, allocate a unit to every unit procedure contained within the procedure-level recipe during batch creation. (A batch creates when it is on the batch list in the FactoryTalk Batch View or custom HMI application.) This presents complications if this procedure-level recipe takes thirty days to complete from the moment the first unit procedure is initiated until the last unit procedure runs to completion. If some or all of the unit procedures contained within the procedure are class-based, it may be difficult to predict, at batch creation time, which units are available when needed.

Dynamic Unit Allocation solves this problem because it allows the bound unit to be defined when the unit procedure is ready to run, rather than when the batch is created. Late binding is postponing the unit definition until the unit procedure requires it to run.

See also

- How dynamic unit allocation affects unit binding on page 51
- Binding methods on page 52
- Unit requirement name on page 53
- Downstream unit requirements on page 54
- Enable dynamic unit allocation on page 54
Late binding is a binding method where a step is bound to equipment just before it is used.

- For unit procedure steps (also called dynamic unit allocation), two types of late unit binding are supported: First Available and Prompt binding.
- For material phase steps, two types of late phase binding are supported: Automatic and Prompt binding.

See also

- [Binding methods](#) on page 52
- [Unit requirement name](#) on page 53
- [Downstream unit requirements](#) on page 54
- [Enable dynamic unit allocation](#) on page 54

### Binding methods

These four binding methods are options for class-based unit procedures:

- **At Batch Creation** - The operator defines the units at batch creation. This is similar to what occurs with disabled Dynamic Unit Allocation.

- **First Available** - The FactoryTalk Batch Server assigns the unit for binding when the unit procedure is ready to run (FactoryTalk Batch Server-defined late binding).

  **Tip:** Once the unit binds using the First Available binding method, the FactoryTalk Batch Server sorts through Equipment ID numbers in ascending order and selects the Equipment ID with the smallest ID number.

- **Prompt** - The operator responds to a prompt to assign the unit for binding when the unit procedure is ready to run (operator-defined late binding).

- **Operator Choice** - When the batch creates, the operator can choose the unit binding methods: At Batch Creation, First Available, or Prompt.

See also

- [Criteria for unit selection](#) on page 52

### Criteria for unit selection

When automatically selecting a unit for binding, the FactoryTalk Batch Server tries to use the unit that the recipe can acquire first. The unit selected must meet these criteria:

- The acquired unit must belong to the unit class of the unit procedure step.
- Recipes can configure upstream, downstream, or both upstream and downstream dependencies that define a series of unit classes that a recipe
requires as a recipe executes. The acquired unit supports the flow path to other units.

- Recipes containing material phase steps impose additional requirements when selecting a unit for binding:
  - The acquired unit must be capable of fulfilling the material requirements of all material phase steps in the recipe to be run in that unit.
  - Support any recipe to be run upstream or downstream flow path.

See also

[Binding methods](#) on page 52

**Unit requirement name**

A unit requirement name is a user-defined name assigned to a unit class with a defined binding method. When creating the procedure-level recipe, the recipe author assigns the appropriate unit requirement name to each unit procedure step in the recipe. Assigning a unit requirement name solves potential complications when using Dynamic Unit Allocation.

In the example PREMIX Procedure-level recipe, assume that both of these class-based unit procedures have the PREMIX_A_CLS unit class assigned:

- PREMIX_A
- XFR_UP1_PREMIX

Additionally, there are five unit instances of the PREMIX_A_CLS unit class and unit procedures, PREMIX_A and XFR_UP1_PREMIX that can bind to any of the five unit instances. But they must bind to the same unit instance.

If the selected binding method is First Available, define a unit requirement name for both unit procedures. This is so the Batch Server can bind both unit procedures to the same unit instance.

For example, in the sample recipe, the recipe author must assign the same unit requirement name to both unit procedures PREMIX_A and XFR_UP1_PREMIX. This same unit requirement identifies to the Batch server that whatever unit instance it binds to unit procedure PREMIX_A, it must also bind to unit procedure XFR_UP1_PREMIX.

If implementing Unit Attribute Binding, the recipe author can further define the binding requirements and preferences. The author can configure binding to units meeting specific criteria, for example, warmest available unit, capacity restrictions, or non-agitator units.

A procedure-level recipe can contain a mix of class-based unit procedures and unit-specific unit procedures. If Dynamic Unit Allocation is enabled, all unit
procedures use a unit requirement name, even those that are unit-specific. Therefore, assign a unit requirement name to a unit instance (rather than to a unit class) and define the binding method as Static. Static binding means that any unit procedure steps must always be bound to the specific unit identified for this unit requirement name.

See also

Dynamic unit allocation on page 50

Downstream unit requirements

A downstream unit is a unit that follows another specified unit during batch execution. Downstream unit requirements mirror the linking of units represented in the area model. Specifying required downstream units allows control of batch flow and to make sure to select the appropriate unit during unit-binding. Configure the area model and recipes as follows to enforce these flow paths:

- Link the units in the area model using the FactoryTalk Batch Equipment Editor.
- Define downstream unit requirements in the recipe using the FactoryTalk Batch Recipe Editor.

When these configurations are in place, the FactoryTalk Batch Server enforces these flow paths when binding units to a class-based unit procedure or when displaying a unit selection list to the operator for unit binding.

See also

Dynamic unit allocation on page 50

Enable dynamic unit allocation

To use enable dynamic unit allocation enable it in FactoryTalk Batch Recipe Editor.

To enable dynamic unit allocation:

1. From the View menu, select Options to display the Options dialog box.
2. Select Support Dynamic Unit Allocation.
3. Select Apply.

Important: The allocation setting applies to ALL procedure-level recipes created, edited, or saved.

See also

Dynamic unit allocation on page 50
Open recipe

Open existing recipes for viewing and editing from within the FactoryTalk Batch Recipe Editor. Open entire recipe procedures, as well as individual operations or unit procedures.

To open a recipe:

1. Select Open to display the Open [Type] Recipe dialog box, where [Type] is the recipe storage type (binary, XML, or RDB, configured in the FactoryTalk Batch Equipment Editor Server Options dialog box).

2. From the Recipe Name list, double-click a recipe to open it, or select the appropriate recipe and select Open.

Recipe header information for a selected recipe displays in the Recipe Details tab.

In the Approvals Process area, examine the approval process state for each selected recipe to see which recipes may require approval attention and action.
In the **Recipe Filter** tab, set filters as desired to narrow the number of recipe names. Multiple filters have an additive effect in narrowing the list shown. For example, effectively search for recipes that require particular approval step signoffs by setting the **Step Name** and the **Step State** filters.

<table>
<thead>
<tr>
<th>Item</th>
<th>Available Selections</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recipe Type</td>
<td>All Procedures, Operation, Unit Procedure, or Procedure.</td>
</tr>
<tr>
<td>Equipment Type</td>
<td>Select <strong>ALL</strong> to view recipes regardless of equipment associations in the area model. To limit the displayed recipes to those that use particular equipment, select <strong>Unit Class</strong> or <strong>Unit</strong>. Then select from the corresponding classes or instances in the <strong>Class/Instance</strong> drop-down list.</td>
</tr>
<tr>
<td>Class/Instance</td>
<td>Unit classes defined in the area model display when Equipment Type is <strong>Unit Class</strong>. Similarly, unit instances display when Equipment Type is <strong>Unit</strong>.</td>
</tr>
<tr>
<td>Approval Process Type</td>
<td>ALL, Primary Approval Process, or Expedited Approval Process. If Recipe Approval Process is disabled, this list along with Step Name and Step State are unavailable.</td>
</tr>
</tbody>
</table>
### Item | Available Selections
--- | ---
Step Name | ALL, any defined step in the Primary approval process, Release Recipe as Step, and Release Recipe to Production (the latter two are the only steps used for Expedited Approval).
Step State | ALL, Not Started, In Progress, Complete, Reverting, and $System Signoff Pending (for translated recipes).
Version Type | Versioned & Unversioned, ALL, and Obsoleted.
Basename | ALL, or any basename (name of the recipe element without the ~V# or _WIP extension). Obsoleted recipes display in the list, but the obsoleted status is not visible from the recipe name.

#### See also
- [Recipe header data](#) on page 70
- [Cannot open recipe error message](#) on page 57

### Cannot open recipe error message

If Security Authority is enabled for a recipe, and there is a mismatch between the Security Authority Identifier (SAI) it contains and the SAI in the current FactoryTalk Network Directory, this error message opens:

Make note of the FactoryTalk Network Directory host computer information, then select **OK** to close the error message.

To recover the recipe, saved either as a backup of the FactoryTalk Network Directory SAI or of the recipe in unsecured form:

- Restore the FactoryTalk Network Directory SAI which secures the area model. The name of the computer that hosts that SAI provides in the dialog box. Use the FactoryTalk Administration Console to restore a saved backup of the SAI. Then open the secured recipe.
- Open an unsecured copy of the recipe if saved a copy of the recipe prior to applying Security Authority.
Create operations and unit procedures

Define unit requirements for unit procedures and operations when creating the recipe. Only indicate the specific unit instance or unit class used for this particular recipe.

To create operations and unit procedures:

1. Select **New**. The **New** dialog box opens.

2. Select **Unit Procedure** or **Operation**, and select **OK**. The **Create Unit Procedure** or **Create Operation** dialog box opens.

3. Enter recipe information and select **OK**. **Procedure Identifier** is the only required box. The **Unit Requirement** dialog box opens.

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Procedure Identifier</td>
<td>The unique name that displays when working with a recipe in the FactoryTalk Batch View. It is also used as the file name when the recipe is saved. Follow methodologies for assigning product identification codes. The Procedure Identifier can contain letters (A-Z), numbers (0-9), and underscores (_). For Procedures, the Procedure Identifier can begin with either a letter or a number. For Unit Procedures and Operations, it must begin with a letter.</td>
</tr>
<tr>
<td>Version Number</td>
<td>Legacy, user-set version number (not related to the Recipe Versioning feature introduced in FactoryTalk Batch version 12). Manually update this version number as desired.</td>
</tr>
<tr>
<td>Version Data</td>
<td>Defaults to today's date, cannot be changed.</td>
</tr>
<tr>
<td>Author</td>
<td>The name of the individual that created the recipe.</td>
</tr>
<tr>
<td>Product Code</td>
<td>A short label that uniquely identifies the product.</td>
</tr>
<tr>
<td>Procedure Description</td>
<td>Product or process description at the procedure level. Indicate whether this is a class-based recipe, or any other distinguishing information required.</td>
</tr>
<tr>
<td>Procedure Abstract</td>
<td>A summary or outline of the steps of the procedure.</td>
</tr>
<tr>
<td>Version Description</td>
<td>Legacy, version description (not related to the Recipe Versioning feature introduced in FactoryTalk Batch version 12).</td>
</tr>
</tbody>
</table>

4. Select either the **Class Based** or **Instance Based** unit requirement. The **Class Based** group box shows the unit classes defined in the area model as class-based. The **Instance Based** group box shows the unit instances defined in the area model as unit-based.

If supporting Dynamic Unit Allocation, **Bind Requirements** and **Bind Preferences** are enabled.
5. When binding materials, select either **Inventory** or **Configuration**. The Factory Talk FactoryTalk Batch Server and Material Server determines the appropriate set of binding candidates at runtime.

   - **By Inventory**: Containers assigned to a unit must have material in them.
   - **By Configuration**: Containers assigned to a unit can have zero quantity of material in them.

6. Choose either a **Unit Class** or **Unit** from the appropriate list.

7. Select **OK** to return to the **Procedure View** pane, which shows the default procedure name (**UNIT_PROCEDURE#** or **OPERATION#**) on the title bar. The **SFC** view contains the initial and final steps of the sequential function chart, and the **Table** view contains a blank table header.

See also

- **Binding requirements** on page 65
- **Insert steps into an SFC** on page 79
Change operation/unit procedure unit requirements

Edit the unit requirements within a unit procedure or operation at any time.

To change operation/unit procedure unit requirements:

1. Select the appropriate recipe level. Double-click the unit procedure or operation to edit. Make sure it displays in the Recipe Construction pane.

2. From the Recipe menu, select Unit Requirements. If not using the unit procedure or operation in a higher level recipe, the appropriate Unit Requirement dialog box opens.

3. Select Class Based or Instance Based and then determine the unit class or unit instance from the appropriate list. If the unit procedure or operation has been used in a higher level recipe, a warning message displays indicating what higher level recipes needs to be modified the unit requirements are changed. Select Proceed to updated affected recipes.

4. Select OK to return to the Procedure View pane.

Tip: If modifying the unit requirements for a unit procedure or operation used in a higher level recipe, then make the required adjustments to the unit requirements of higher level recipes. If not making corrections, the Released to Production check boxes clears for higher level recipes.

See also

Create operations and unit procedures on page 58

Create a procedure

When creating a procedure, indicate all unit instances and unit classes used within the procedure. If the Support Dynamic Unit Allocation option is enabled in the View > Options dialog box, each unit instance or unit class must have a unit requirement name assigned when used in a procedure.

To create a procedure:


2. Select Procedure and select OK. The Create Procedure dialog box opens.

3. Enter recipe information and select OK. Procedure Identifier is the only required box.
### Item - Description

**Procedure Identifier**  
The unique name that displays when working with a recipe in the FactoryTalk Batch View. It is also used as the file name when the recipe is saved. Follow methodologies for assigning product identification codes.

The Procedure Identifier can contain letters (A-Z), numbers (0-9), and underscores (_). For Procedures, the Procedure Identifier can begin with either a letter or a number. For Unit Procedures and Operations, it must begin with a letter.

**Version Number**  
Legacy, user-set version number (not related to the Recipe Versioning feature introduced in FactoryTalk Batch version 12). Manually update this version number as desired.

**Version Date**  
Defaults to today's date, cannot be changed.

**Author**  
The name of the individual that created the recipe.

**Product Code**  
A short label that uniquely identifies the product.

**Procedure Description**  
Product or process description at the procedure level. Indicate whether this is a class-based recipe, or any other distinguishing information required.

**Procedure Abstract**  
A summary or outline of the steps of the procedure.

**Version Description**  
Legacy, version description (not related to the Recipe Versioning feature introduced in FactoryTalk Batch version 12).

If **Support Dynamic Unit Allocation** is enabled, the **Procedure-Unit Requirements** dialog box opens.

### Procedure - Unit Requirements

<table>
<thead>
<tr>
<th>Unit Requirement Name</th>
<th>Classifiers</th>
<th>Class</th>
<th>Existing Method</th>
<th>Deactivate Tools</th>
</tr>
</thead>
<tbody>
<tr>
<td>M1S1_MDI000_CLS</td>
<td>MDI000_CLS</td>
<td></td>
<td>Batch Creation</td>
<td></td>
</tr>
</tbody>
</table>

Click the [Add Unit Requirement] button to add new unit requirements.
4. Select **Add Unit Requirement** to display the **Add Unit Requirement** dialog box.

5. Type a unique name in the **Name** box. The name can contain alphanumeric characters (A through Z, 0 through 9), an underscore (_), or a colon (:), but it **must** begin with an alphanumeric character.

   Any unit procedure mapped to this unit requirement name uses the same unit to which this unit requirement name is bound. Bind the unit requirement name to the unit class or to a unit instance.

6. Select **Class Based** or **Instance Based** to enable the corresponding group box.

7. Select the required unit class or unit from the appropriate list.

8. If adding Class Based requirements, select the appropriate binding method from the **Binding Method** list.

9. If needed, add applicable **Bind Requirements** and **Bind Preferences**.

10. In the **Available Downstream Unit Requirements** list, select the unit requirement name for the unit that is downstream from the unit requirement being defined and select the right arrow. The unit requirement is added to the **Selected Downstream Unit Requirements** list.
Tip: Specify **Downstream Unit Requirements** only after all additional unit requirements are specified. The list box contains the unit requirement names for all available units defined in the **Procedure-Unit Requirements** dialog box (except the one currently being defined). To return all selected downstream units to the available downstream units, select **Remove All**.

11. Select **OK** to return to the **Procedure-Unit Requirements** dialog box.

12. Repeat steps 4 through 10 to define additional unit requirements for this procedure.

13. Select **Close** in the **Procedure-Unit Requirements** dialog box to return to the **Procedure View** pane, which now shows the default procedure name on the FactoryTalk Batch Recipe Editor title bar. The **SFC** view contains the initial and the final steps of the sequential function chart, and the **Table** view contains a blank table header.

![FactoryTalk Batch Recipe Editor - [OPERATION]](image)

See also

- **Unit requirement name example** on page 63
- **Dynamic unit allocation** on page 50
- **Binding requirements** on page 65
- **Configure binding requirements** on page 66

### Unit requirement name example

The unit requirement name is a label and acts as a lookup table for the FactoryTalk Batch Server. When a recipe is on the Batch List and the server encounters the unit requirement name, it looks up all the unit classes (or unit instances) mapped to the unit requirement name.

For example, a plant has two mixers that used to run one recipe. When the recipe runs, the operator or the FactoryTalk Batch Server (depending on the binding
method) must pick which mixer to use. In the area model, these mixers are each assigned a unit instance (MIXER_1 and MIXER_2) from the same unit class (MIXER). In the recipe, create a unit requirement name (MIXERS) configured to the MIXER unit class, with the binding method at batch creation. When the recipe is on the Batch List, the FactoryTalk Batch Server encounters the label MIXERS, which tells it that piece of equipment is required for this recipe. The server looks for all the unit instances of the unit class defined with this unit requirement name and prompts the operator to select either MIXER_1 or MIXER_2.

Any unit procedure, including its subordinate operations, mapped to this unit requirement name uses the specified unit class and the associated binding method. Because recipes using dynamic unit allocation require that all unit procedures use a unit requirement name, assign a unit requirement name to all instance-based unit procedures and define the binding method used as Static.

See also

Create a procedure on page 60

If Support Dynamic Unit Allocation is enabled, edit unit requirements within a procedure at any time. If disabled, the Unit Requirement dialog box is not available at the procedure level. Remove a unit requirement from a procedure at any time, as long as no steps within this procedure define this unit requirement.

To change procedure unit requirements:

1. Open an existing procedure.

2. Select the procedure level of the recipe.

3. From the Recipe menu, select Unit Requirements to display the Procedure-Unit Requirements dialog box.

4. To edit a unit requirement, select the row containing the unit requirement, and then select Edit Unit Requirement to display the Edit Unit Requirement dialog box.

5. Make the required modifications to the unit requirement. Select OK to return to the Procedure-Unit Requirements dialog box.

   Tip: Modifying from a class-based requirement to an instance-based requirement deletes all of the binding requirements and preferences.

6. To delete an existing unit requirement, select the row containing the unit requirement and select Delete Unit Requirement. If the procedure has, any steps defined using this unit requirement, a message displays stating that the requirement cannot be deleted. In this case, delete all steps defined
against the unit requirement first. Deletion of a unit requirement removes
the unit from any referenced downstream lists.

See also

Create a procedure on page 60

Smart binding

Smart binding, also referred to as unit attribute binding, enhances Dynamic Unit
Allocation by allowing custom, user-defined binding requirements and binding
preferences. With smart binding, specify:

- A binding requirement, such as "The unit must be in service."
- A binding preference, such as "I want the mixer cleaned less than 3 days
  ago."

Configure custom unit attributes such as VESSEL_STATUS in the FactoryTalk
Batch Equipment Editor. The recipe author then configures binding requirements
and binding preferences based on those custom attributes in the FactoryTalk
Batch Recipe Editor.

Tip: Binding requirements and binding preference
expressions do not support parameters with deferred
values or expression values.

See also

Dynamic unit allocation on page 50

Binding requirements on page 65

Binding preferences on page 67

Create binding expressions on page 69

Binding requirements

Bind requirements define subsets of class-based unit procedures and operations.
Configure subsets as one of these options:

- **Expression** – Used to reference Unit Attributes assigned to the Unit Class
  associated with the Unit Requirement. Expression objects can also reference
  Recipe Header values such as BATCH_SIZE, output parameters on the
  parent recipe, and report and formula parameters on the peer steps within
  the recipe.

- **Require Phase** – Used when a Unit Requirement must be bound to a Unit
  that supports a specified recipe phase.

- **Reject Phase** – Used when a Unit Requirement must be bound to a Unit
  that does not support a specified recipe phase. This requirement prevents
  recipes from using units with unneeded capabilities.
• **Require Attribute** – Used when a Unit Requirement must be bound to a Unit that supports a specified Unit Attribute. It is a demand for a Unit that provides an attribute tag for support of the specified unit attribute.

• **Reject Attribute** – Used when a Unit Requirement must be bound to a unit that does not support a specified Unit Attribute. This requirement prevents recipes from using units with unneeded attributes.

See also

- Binding preferences on page 67
- Create binding expressions on page 69

**Configure binding requirements**

Once the appropriate custom Unit Attribute definitions, assignments, and configurations are complete in the FactoryTalk Batch Equipment Editor, define the bind requirements in the FactoryTalk Batch Recipe Editor.

**To configure binding requirements:**

1. Open a new or existing unit procedure or operation recipe.

2. From the **Recipe** menu, select **Unit Requirements** to display the **Unit Requirement** dialog box.

3. Select **Bind Requirements** to display the **Bind Requirements for Unit Requirement: [CLASS_NAME]** dialog box.

   **Tip:** Any global unit attributes list automatically and are view-only. Configure global unit attributes in the FactoryTalk Batch Equipment Editor.

4. Select **Add Requirement**. A new row displays containing the unit procedure/operation name and boxes for the bind Type and Description.
5. Configure the **Type** and **Description** columns with the applicable bind requirements.

![Image of Recipe Requirements for Unit Requirement: MIXER](image)

**Tip:** The **Description** information changes based on the **Type** selected. If the **Type** is an Expression, select Browse to open the **Unit Binding Expression Builder**.

6. Select **OK** twice to return to the **Procedure View Pane**.

**See also**

- [Binding preferences](#) on page 67
- [Create binding expressions](#) on page 69

**Binding preferences**

A bind preference is an object that evaluates against an instance of a Unit Class in order to sort the legal bind targets for a Unit Requirement into a most preferred order. A bind preference can specify a preferred Phase or Unit Attribute, an expression to minimize or maximize, or a Phase or Attribute to avoid. The priority number indicates the order of evaluation for the FactoryTalk Batch Server.

Configure these preferences in recipes as using one of these options:

- **Expression** – A boolean expression used to reference unit attributes assigned to the unit class associated with the Unit Requirement. Expression objects can also reference Recipe Header values such as **BATCH_SIZE**.

- **Minimize Expression** – An expression that evaluates to either an integer or real value. Legal bind targets for which the expression evaluates to a **lower** value are more **preferred** bind targets than those for which the expression evaluates to a higher value.

- **Maximize Expression** – An expression that evaluates to either an integer or real value. Legal bind targets for which the expression evaluates to a **higher** value are more **preferred** bind targets than those for which the expression evaluates to a lower value.

- **Prefer Phase** – Used as a recipe phase inclusion for bind preferences. The preferred unit supports the specified phase.

- **Avoid Phase** – Used as a recipe phase exclusion for bind preferences. The preferred unit does not support the specified phase.


- **Prefer Attribute** – Used as a recipe phase inclusion, it is a request for a Unit that provides an attribute tag for support of the specified unit attribute.

- **Avoid Attribute** – Used as a recipe phase exclusion, it is a request for a Unit that lacks support for the specified recipe phase.

See also

- [Smart binding](#) on page 65
- [Configure binding preferences](#) on page 68

**Configure binding preferences**

Use this procedure to configure binding preferences.

To configure binding preferences:

1. Open a new or existing unit procedure or operation recipe.

2. From the **Recipe** menu, select **Unit Requirements** to display the **Unit Requirement** dialog box.

3. Select **Bind Preferences** to display the **Bind Preferences for Unit Requirement: [CLASS_NAME]** dialog box.

4. Select **Add Preference**. A new row displays containing the unit procedure/operation name and drop-down boxes for the **Bind Type** and **Description**.

5. Configure the **Type** and **Description** columns with the applicable bind requirements.
Tip: The Description information changes based on the Type selected. If the Type is an Expression, select browse to open the Unit Binding Expression Builder.

6. Select OK twice to return to the Procedure View Pane.

See also

Create binding expressions on page 69

Create binding expressions

When selecting one of the Expression types as the Type of bind preference or requirement, define that expression in the Unit Binding Expression Builder dialog box.

Tip: Binding requirements and binding preference expressions do not support parameters with deferred values or expression values.

To create binding expressions:

1. From the Bind Requirements or Bind Preferences dialog box, select Add Preference or Add Requirement.

2. From the Type list, select the type of expression, and then select browse to the right of the Description to display the Unit Binding Expression Builder dialog box.

   Tip: Bind requirement and preference expressions can reference items that are unit class attributes or a recipe procedure’s recipe header, and step parameters.

3. From the tree view in the left pane, select the value type for building the expression, unit attributes or recipe header, or step parameters.
4. From the middle pane, select the operand used for the expression. The example shown lists operands like CAPACITY and GLASS_LINED.

5. Select an operator button to enter the desired operation into the expression. Type the operation directly into the Expression text box. The example uses the greater than or equal to operator.

6. If the operand is an Enumeration, double-click the value in the right pane to add the value to the expression in the text box; otherwise, type the desired value directly into the expression.

7. Select OK to save and validate the expression, and then return to the dialog box from which the Expression Builder was opened. The new bind requirement or preference displays.

8. Select OK to close the Bind Requirement or Bind Preference dialog box.

See also

Bind Expressions on page 99

Recipe header data

Header data is general information about the recipe. The Header Data dialog box contains:

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Procedure Identifier</td>
<td>The unique name that displays when working with a recipe in the FactoryTalk Batch View. It is also used as the file name when the recipe is saved. Follow methodologies for assigning product identification codes. The Procedure Identifier can contain letters (A-Z), numbers (0-9), and underscores (_). For Procedures, the Procedure Identifier can begin with either a letter or a number. For Unit Procedures and Operations, it must begin with a letter.</td>
</tr>
<tr>
<td>Version Number</td>
<td>Legacy, user-set version number (not related to the Recipe Versioning feature introduced in FactoryTalk Batch version 12). Manually update this version number as desired.</td>
</tr>
<tr>
<td>Version Date</td>
<td>Defaults to today's date, cannot be changed.</td>
</tr>
<tr>
<td>Author</td>
<td>The name of the individual that created the recipe.</td>
</tr>
<tr>
<td>Approved By</td>
<td>The name of the individual who approved the recipe.</td>
</tr>
<tr>
<td>Product Name</td>
<td>The product name the recipe produces.</td>
</tr>
<tr>
<td>Product Code</td>
<td>A short label that uniquely identifies the product.</td>
</tr>
<tr>
<td>Item</td>
<td>Description</td>
</tr>
<tr>
<td>--------------------------</td>
<td>---------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
</tbody>
</table>
| Batch Size               | • Min - a numeric value indicating the minimum size of the completed batch.  
                         | • Default - a numeric value indicating the default size of the completed batch.  
                         | • Max - a numeric value indicating the maximum size of the completed batch.  
                         | • Units of Measure - the type of measurement used to designate the minimum, default, and maximum batch sizes, such as kilograms, pounds, liters, and other units. |
| Estimated                | Approximate length of time it takes to run and complete a batch using the recipe (minutes).                                                  |
| Procedure Description    | Product or process description at the procedure level. Indicate whether this is a class-based recipe, or any other distinguishing information required. |
| Procedure Abstract       | A summary or outline of the steps of the procedure.                                                                                       |
| Area Model File Name     | Indicates the location and filename of the area model file against which the recipe was last verified. If the entire path and filename is not visible, place the cursor in the box and use the right arrow key to scroll to the right. |
| Date/Time Stamp of Area Model | Date and time the area model was last saved.                                                                                       |
| Time of Verification     | Date and time the recipe was verified. If the recipe was never verified, this information is also indicated.                              |
| File Name                | Recipe filename.                                                                                                                          |
### Approval Process Area (recipe approvals enabled)

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
</table>
| Approval Process Area (recipe approvals enabled) | The title of this area changes to reflect the active Approval Process for the recipe:  
  - Approval Process Not Started  
  - Primary Approval Active  
  - Expedited Approval Active  
  - Pending Recipe Selection  
Recipe Approval Process steps, and their current state, are listed in this area. Hover the cursor over an approval process step to view a description of the approval step.  
- Check boxes are read-only and controlled by the Approval Process.  
- When checked, **Release Recipe as Step** signifies the recipe (or unit procedure or unit operation) is approved for inclusion as a step in another recipe.  
- When checked, **Release Recipe to Production** signifies the recipe is approved for inclusion on the Recipe List in the FactoryTalk Batch View, in eProcedure, or in custom applications that use Control Recipe List ActiveX controls.  
The state of each approval step:  
- Not Started - approval sign off not started.  
- In Progress - approval started but one or more signoffs still required.  
- Reverting - revert started but one or more signoffs still required.  
- $System Signoff Pending - translated (migrated) recipes, or recipes re-entering an approval process, are set to begin with the Expedited approval process. Prior to verification, the Expedited approval steps for these recipes are set to a pending state.  
- Completed - all signoffs made and approval is complete. |

(Recipe approvals disabled)  
This area contains check boxes indicating the states of the **Release Recipe as Step** and **Release Recipe to Production** recipe properties.  
The check boxes are active and can be selected to toggle between true and false.  
- **Release Recipe as Step**, when set to true, signifies the recipe (or unit procedure or unit operation) can be included as a step in another recipe, but batches cannot be created from it alone.  
- **Release Recipe to Production**, when set to true, signifies the recipe can be added to the Recipe List in the FactoryTalk Batch View, in eProcedure, or in custom applications that use Control Recipe List ActiveX controls.

See also

[Add recipe header data on page 72](#)
Add recipe header data  Use this procedure to add recipe header information to a recipe.

To add recipe header data:

1. With the appropriate recipe level open in the FactoryTalk Batch Recipe Editor, select **Header Data**. The **Header Data** dialog box opens.

2. Type the required data (**Procedure Identifier**, **Version Number**, and **Author**) in the **Header Data** dialog box.

3. (optional) Enter information in the remaining boxes.
Tip: If Approval Process is enabled for the area model, then the approval process, along with configured approval steps, displays in the bottom area of the dialog box. The two check boxes are read-only (disabled) but shown to reflect their current state.

4. If disabled Recipe Approvals process, then the two check boxes enable. Select one or both of the boxes as required.

- Select **Release Recipe As Step** to use the recipe or operation as a component in another recipe. With just this box checked, the recipe does not appear in the Recipe List.

- Select **Release Recipe To Production** for the recipe to appear in the FactoryTalk Batch View, eProcedure, or ActiveX control Recipe List, to create production batches.

5. Select **OK**.

See also

**Recipe header data** on page 70

Recipe storage

After building the recipe, save the recipe by selecting **Save**. The recipe is saved in the location and format defined in the FactoryTalk Batch Equipment Editor **Server Options** dialog box.

If recipes are stored using binary files, the recipes save separate files according to the recipe level:

- Procedures are stored as **.bpc** files
- Unit Procedures are stored as **.upc** files
- Operations are stored as **.uop** files

If recipes are stored using XML, the recipes save separate files according to the recipe level:

- Procedures are stored as **.pxml** files
- Unit Procedures are stored as **.uxml** files
- Operations are stored as **.oxml** files

If recipes are stored using the RDB format, recipes are stored in a single SQL Server database (MasterRecipes is the default database name).

See also

**Copy recipe options** on page 75
Copy recipe options

Copy recipes by using the **Save As** function or using the backup directory. When selecting a procedure level or unit procedure level recipe and the **Save As** function is used, only the selected recipe level (procedure or unit procedure) saves with a new name. Recipe Approval steps and their states do not carry over to the new copy.

**Important:** Any lower level recipes (unit procedure, operation, or phase) are not copied with the selected recipe level. Any changes made to those lower levels modify the original version. To modify the original versions, perform a **Save As** on each of the lower level recipes.

Many recipes use deferred parameters in order to minimize the number of different recipes maintained. Because deferred parameter values assign at a higher level recipe, using the **Save As** command to copy the recipes with deferred parameters does not affect the original recipe.

Operations can be copied using **Save As** because they are the lowest level and do not point to upper levels. Changes made to the copy do not affect the original operation.

**Tip:** With Recipe Approvals enabled, using **Save As** clears all signature-approved approval steps in the new copy. The approval steps in the original recipe are not affected.

In this example, performing a **Save As** on the procedure level saves a new copy of that procedure only. The new procedure still points to the original unit procedures, operations, and phases. If changing a parameter value on a phase in the copied recipe, that parameter value changes in the original recipe. The same applies to the operations and unit procedures; all changes affect the original recipe.
Copy a recipe using the backup directory

See also

Copy a recipe using the backup directory on page 76

Use this procedure to copy a recipe using the backup directory.

To copy a recipe using the backup directory:

1. Open Windows Explorer to create a backup directory for recipes.
2. Copy all recipes to duplicate to the backup directory.
3. In the FactoryTalk Batch Equipment Editor, open the Server Options dialog box and change the Recipe Directory to point to the backup directory.
4. Open the FactoryTalk Batch Recipe Editor. If it was already open, exit the FactoryTalk Batch Recipe Editor and restart it.
5. Open the recipe to copy. The phases, operations, and unit procedures display in the Procedure View pane hierarchical list.
6. Select a phase in the first operation of the hierarchical list. From the Recipe menu, select Header Data to display the operation information.
7. Change the Procedure Identifier to a new name, and then select OK. Change the other information if needed.
8. A message indicates recipes affected by the change. Select Proceed, and then select OK.
9. Select the operation that just renamed. Open the Header Data and change the Procedure Identifier to a new unit procedure name. Select OK, and a message lists recipes affected by the change. Select Proceed, and then select OK.
10. Repeat steps 6 through 9 for each operation and unit procedure.
11. Select any unit procedure. Open the Header Data and change the Procedure Identifier to a new name for the procedure. Select OK and a message displays the recipes affected by the change. Select Proceed.
12. Save the procedure.
13. Export the renamed recipe back to the original recipe directory.
14. In the FactoryTalk Batch Equipment Editor, open the Server Options dialog box and change the Recipe Directory to point back to the original recipe directory.
15. Open the FactoryTalk Batch Recipe Editor to display the newly created recipe copies.

See also

Copy recipe options on page 75
Chapter 5

Build a Sequential Function Chart

The creation of an SFC involves the addition of steps, transitions, and links. During the recipe creation process, an initial SFC displays. This SFC consists of the initial step, initial transition, final transition, and final step. Add steps to the chart and then define the steps appropriately. If needed, reconfigure transitions that require special expressions and include any necessary comments.

**Tip:** In addition to using the mouse for navigation, keyboard controls are also available. When focus is in the SFC view, use the arrow keys to navigate between recipe elements. Use the arrow keys while pressing the Ctrl key to move between selected recipe elements. When deleting a selected step with the delete key, the SFC is not automatically rearranged. The active step outlines in dark blue and SFC displays on the status bar at the bottom of the screen when the SFC view has focus.

See also

- Insert steps into an SFC on page 79
- Insert timer steps on page 80
- Manually add steps to an SFC on page 82
- Add a parallel step to an SFC on page 82
- Add a branch structure to an SFC on page 84

**Insert steps into an SFC**

When constructing a recipe, insert steps into an SFC using the Add Step tool or the Insert Step tool.

- The Add Step tool creates a new step after the selected recipe element.
- The Insert Step tool creates a new step before the selected element.

If required, a new transition automatically creates for the new step and a default common transition expression automatically configures. If a new transition is not required, the existing transition modifies to reflect the new step.
Tip: Add material-enabled phases to operations just like other phases. When adding a material-enabled phase to the recipe, define the material parameters.

To insert steps into an SFC:

1. Using the Selection Tool select a step or transition that either precedes or follows the new step.

   Select Insert Step or Add Step to insert a new step before or after the selected element. A new UNDEFINED step displays on the SFC with a blue box outlined in a black dashed line. Additionally, the SFC structure automatically rearranges to make room for the new elements. One of these dialog boxes opens, based on the recipe level: Select Phase, Operation Select, or Unit Procedure Select.

2. Make the appropriate selection and select OK.

3. Repeat Steps 1 through 3 until all of the required steps are on the SFC.

   Tip: The word UNDEFINED displays inside the step until configured.

See also

Recipe construction toolbox on page 17

Define a step on page 86

Insert timer steps

Timer steps are available at all levels of a recipe: operations, unit procedures, and procedure.

To insert timer steps:

1. Using the Selection Tool select the transition that either precedes or follows the new step.
2. Select Insert Step to insert a new step before the selected element, or select Add Step to insert a new step after the selected element. One of these dialog boxes opens, based on the recipe level: Select Phase, Operation Select, or Unit Procedure Select. These instructions are for using the Select Phase dialog box. For information on inserting a step into a Unit Procedure or Operation, see Define a step.

3. Select TIMER from the dialog box and select OK to display the Select Timer - TIMER dialog box.

4. Enter these values:
   a. Type a new name for the Timer step.
   b. Select the Timer type (COUNT_DOWN or COUNT_UP).
   c. Select the units of measure (seconds, minutes, hours, or days).
   d. Select OK. The new Timer step displays in the SFC.

   Tip: If a COUNT_UP timer in parallel with existing steps displays, the transition expression does not change. If it is the only step, then the transition defaults to TRUE.

5. Set parameter values for the Timer step. (See Assign formula values to Timer steps for more information.)

See also

Define a step on page 86

Assign formula values to Timer steps on page 113
Manually add steps to an SFC

Manually add a step to a sequential function chart using the Step Tool on the Recipe Construction toolbox. When using the Step Tool, no new transitions are added, no existing transitions are modified, no links are created for the step, and the SFC structure is not automatically rearranged to make room for the new step.

To manually add steps to an SFC:

1. Select Step Tool.
2. In the SFC view, place and click the cursor in the desired location.

   The step is placed on the sequential function chart and is outlined in blue (indicated by the dashed lines) until another step is placed or another tool is selected. The step displays in red until configured. UNDEFINED is displayed inside the step until it is defined.

3. Repeat Step 2 until all of the required steps for the process display on the sequential function chart.

See also

Recipe construction toolbox on page 17
Define a step on page 86

Add a parallel step to an SFC

Use the Add Parallel tool to add a step to an SFC in parallel with another. The Add Parallel tool creates a parallel step, the appropriate links, and redefines the existing transition to reflect the new parallel structure.

To add a parallel step:

1. Use the Selection Tool to select the step with which the new step is to run in parallel.
2. Select **Add Parallel**. A new UNDEFINED step creates in parallel with the selected step on the SFC and outlines in blue. Additionally, the SFC structure automatically rearranges to make room for the new elements. One of these dialog boxes opens, based on the recipe level: **Select Phase, Operation Select**, or **Unit Procedure Select**. Make the appropriate selection and select **OK**. The existing transition modifies to reflect the new step and the parallel structure.

See also

- **Recipe construction toolbox** on page 17
- **Define a step** on page 86

**When parallel steps require the same dedicated resource**

If parallel steps require the same dedicated resource, the FactoryTalk Batch Server automatically determines how the resources allocate between steps when the batch runs. This is subarbitration. If parallel steps requiring the same resource occur just after an AND divergence or just before an AND convergence, add null procedures to the recipe in order for batches to run successfully.

This example shows the correct placement of null procedures in a sequential function chart (SFC) that contains parallel steps requiring the same phase.
In this SFC, each set of parallel steps is requesting the same phase.

- The first set of steps after the AND Divergence, ADD_CREAM:1 and ADD_CREAM:2, require a set of null procedures before them so subarbitration can work. Without the null procedures, the batch goes into a HELD state before it reaches these steps.
- The FactoryTalk Batch Server automatically subarbitrates for the second set, ADD_EGG:1 and ADD_EGG:2.
- The third set, AGITATE:1 and AGITATE:2, requires a set of null procedures before the AND Convergence so subarbitration can work. Without the null phases, the batch appears as though it is still running, but it never is able to transition past these steps to the Convergence.

See also

Add a parallel step to an SFC on page 82

Add a branch structure to an SFC

Add a branch structure to an SFC using the **Add Branch** tool, which creates a new step, a new transition preceding the step, and a new transition following the step. The new preceding transition contains the transition condition FALSE and the new following transition contains the transition condition **STEP.STATE=COMPLETE**.
Tip: Due to the variation and complexity of SFC branch structures, it is necessary to redefine one or both transitions.

To add a branch structure:

1. Use the Selection Tool to select the step on which the new branch structure is to border.

2. Select Add Branch A new step creates on the SFC and outlines in blue (indicated by the dashed lines). A new transition precedes the new step, and a new transition follows the new step. Additionally, the SFC structure automatically rearranges to make room for the new elements. One of these dialog boxes opens, based on the recipe level: Select Phase, Operation Select, or Unit Procedure Select. Make the appropriate selection and select OK.

3. Define the new preceding and following transition expressions, if necessary.

See also

Recipe construction toolbox on page 17
Assign a transition expression on page 88

Add transitions to an SFC

Use the Transition Tool to add transitions to an SFC. The Transition Tool creates a single transition requiring a manual link to the appropriate steps and defines the transition expression.

To add a transition:

1. Select Transition Tool

2. Place the cursor in the desired location in the SFC View and click. The transition, with its name (T followed by a number), is placed on the SFC.
3. Link the transition to the appropriate steps.

4. Assign a transition expression.

**Tip:** In FactoryTalk eProcedure recipes, all loops and transitions must branch to the right side of the SFC steps.

---

**Define a step**

Newly added steps are UNDEFINED. Only valid lower level recipes are used in the step definition. When a process for a step is selected, that process is assigned to that particular step in the recipe, and the step is marked with the process name. Redefine a step at any time.

**To define a step:**

1. Double-click a step marked UNDEFINED. One of these dialog boxes opens, based on the recipe level: Select Phase, Operation Select, or Unit Procedure Select.

2. To define a phase or operation either select from the option list and then select OK. Or, select New Phase or New Operation to create a new phase or operation. See Create operations and unit procedures.

3. To define a unit procedure:
   a. From the Alias list, select the alias to associate with this unit procedure.
   b. The Alias list contains a list of all units defined in the procedures unit requirements.
      a. If this unit runs in parallel with other units, and if each of the units at this level must acquire in order to initiate this level of the recipe, then select Acquire unit prior to starting unit procedure.
b. If the units running in parallel can run independently of each other, clear this check box for all of the units at this level. The recipe then runs the individual units acquired, regardless of whether the other units acquire.

c. Select the appropriate unit procedure to associate with this step. Or, select New Unit Procedure to create a new unit procedure.

d. Select OK.

See also

Create operations and unit procedures on page 58
Add a null procedure on page 36

Link SFC components

Use this procedure to link SFC components.

To link SFC components:

1. Select Link Tool.

2. Place the cursor at the links starting step. Drag the link to the connecting step, and release the mouse button to complete the link. The elements are now connected.

   Tip: In FactoryTalk Procedure recipes, all loops and transitions must branch to the right side of the SFC steps.

See also

Recipe construction toolbox on page 17
Remove a link from an SFC on page 87

Remove a link from an SFC

Follow these instructions to remove a link from an SFC.

To remove a link from an SFC:

1. Select Remove Link Tool. The cursor changes to a + sign.

2. Place the cursor at the links starting step. Drag the cursor to the connecting step.

3. Release the mouse button to remove the link.
See also

Recipe construction toolbox on page 17

Link SFC components on page 87

Assign a transition expression

After adding a transition to the SFC, assign a transition expression.

To assign a transition expression:

1. Double-click a transition in the SFC view. The Transition Expression Builder dialog box appears.

   ![Transition Expression Builder](image)

2. Enter the expression for the transition condition or build an expression by selecting options from the three different sections. Press Enter to continue the expression on the next line.

3. Select OK to return to the SFC view.

See also

Transition expressions on page 25

Remove a step from an SFC

Remove a step from an SFC using the Selection tool.

To remove a step from an SFC:

1. Using the Selection Tool select the step to remove.

2. Select Remove Step.

   The selected step is removed and all affected transitions are removed or modified to reflect the new SFC structure. Additionally, the SFC structure rearranges automatically to recover the space created by any removed elements. If there are no recipe elements bordering the removed step (if the
step was added to the SFC view without being linked), a blank space is left in the SFC structure. Any blank spaces left within the SFC structure must be linked manually using the Link Tool.

Tip: Transitions following parallel structures can be extremely complex so they are not automatically reconfigured. If the transition following the removed parallel step requires reconfiguration, it must be configured manually.

See also

Manually add steps to an SFC on page 82

Create material loops

If the SFC contains a material-enabled step, create a material loop that allows the step to automatically rebind when a feed is not completed. Because material loops disable the ability to create complex recipe structures, add them after the SFC is set up with AND and OR convergences and divergences. For example, to create three material equipment phases in parallel, create the three parallel recipe phases first, and then add the material loop.

To create material loops:

1. Using the Selection Tool, select the material enabled step to add to the loop.

2. Select Create Material Loop. The material loop is added, including the two required null procedures.

To remove a material loop immediately after adding it, select Undo Material Loop.

This example shows how a material loop appears when the material-enabled step is preceded and followed by a transition. The material loop may look different and contain more than one null procedure if it is preceded or followed by a convergence or a divergence.
Recipe comments

Use recipe comments to give written instructions to operators, share notes with other engineers, or access a reference. Recipe commenting associates data with a step, transition, or the entire recipe. The comment is viewable at design and at run time.

The comment, and its associated information, is a text box. Above each text box, the assigned reference name displays as a C followed by a number. View added SFC text boxes in the SFC views of FactoryTalk Batch View, FactoryTalk Procedure, and ActiveX Control.

See also

Sequential Function Chart overview on page 23
Recipe creation overview on page 18
Add comments to an SFC

Use these instructions to add comments to an SFC.

To add comments to an SFC:

1. Select Text Box Tool.

2. Move the cursor (text box) to the desired location, click, and the text box is displayed. When placing the text box, pick an area in the SFC that does not interfere with viewing existing steps and transitions as these elements do not relocate when the text box is generated. If necessary, text boxes, steps, and transitions can be moved by selecting and dragging them to new locations.

3. Use the selection tool to add the required information inside the text box. When finished typing the text, select outside the box. The text box resizes to accommodate the comment. If needed, use the Selection Tool to edit existing text.

Tip: The maximum number of characters contained in a text box is 1024. If more capacity (space) is required for a recipe comment, use additional text boxes and associated them with the step or transition.

See also

Recipe construction toolbox on page 17

Associate recipe comments on page 91

Disassociate recipe comments on page 92
**Associate recipe comments**

When a text box is created, the default is a general recipe procedure comment. The associated information section is blank.

**To associate a recipe comment with a step or transition:**

1. Select Link Tool select inside the text box, and drag to the desired step or transition. Completing this step in reverse order is also an option.

2. Release the button, and the bottom portion of the text box displays the associated step or transition name.

**Tip:** A text box can only be associated with one step or transition. Linking a text box to a step or transition that is already associated to another element results in assigning the most recently associated step or transition name.

**See also**

- Recipe construction toolbox on page 17
- Disassociate recipe comments on page 92

**Disassociate recipe comments**

Use these instructions to disassociate a comment.

**To disassociate a recipe comment from a step or transition:**

1. Select Remove Link Tool select inside the text box, and drag to the associated step or transition. This step can be performed in reverse order with the same result.

2. The associated step name is removed from the text box and now associated with the entire recipe.
Tip: If a step or transition associated with a text box is deleted, the text box becomes associated with the entire recipe procedure and the association label is blank.

See also

Recipe construction toolbox on page 17

Delete recipe comments on page 93

Delete recipe comments

Follow these instructions to delete recipe comments.

To delete recipe comments:

1. Select the text box to be deleted.

2. From the Edit menu, select Delete Selection or press Delete.

The comment and text box are removed from the SFC.

See also

Add comments to an SFC on page 91
Chapter 6

Build a table

The creation of a table involves the addition and configuration of steps to the table structure. Add steps to the table and then define the steps appropriately.

Tip: While in the Table view, the active step is highlighted in dark blue and Table is displayed on the status bar at the bottom of the screen when the Table view has focus. Delete a selected step in the table by pressing the Delete key.

See also

- Insert steps into a table on page 95
- Add a parallel step to a table on page 96
- Add a step before and after parallel steps on page 97
- Define a step on page 86
- Remove a step from a table on page 98

Insert steps into a table

When constructing a recipe, steps can be inserted into a table using the Add Step tool or the Insert Step tool. The Add Step tool creates a new step after the selected step, while the Insert Step tool creates a new step before the selected step.

Tip: Material-enabled phase steps are added to operation recipes similar to all other phase steps.

To insert steps into a table:

1. Select the table row that either precedes or follows the new step.

2. Select Insert Step to insert a new step before the selected row, or select Add Step to insert a new step after the selected row. A new UNDEFINED step is created in the table and highlighted in dark blue. One of these dialog boxes opens, based on the recipe level: Select Phase, Operation Select or Unit Procedure Select.

3. Make the appropriate selection and select OK.
4. Repeat steps 1 through 3 until all of the required process steps are placed in
the table.

![Table](image)

**Tip:** The word UNDEFINED displays inside the step until it is
configured.

**See also**

- Recipe construction toolbox on page 17
- Build a table on page 95
- Add a parallel step to a table on page 96

## Add a parallel step to a table

To add a step to a table in parallel with another step use the **Add Parallel** tool.

**To add a parallel step to a table:**

1. Select the step that is to run in parallel with the new step.

2. Select **Add Parallel**. A new UNDEFINED step is created in parallel
with the selected step and highlighted in blue. To the left of the step, a
parallel icon signifies that the step is a parallel step. One of these dialog
boxes opens, based on the recipe level: **Select Phase, Operation Select** or
**Unit Procedure Select**.

3. Make the appropriate selection and select **OK**.

![Table](image)

**Tip:** For parallel steps that request the same resource, the
FactoryTalk Batch server determines how the resources
are allocated through the subarbitration process. If
parallel steps require the same resource to occur just
after an AND divergence or just before an AND
convergence, add null procedures to the recipe in order
for batches to run successfully.

**See also**

- When parallel steps require the same dedicated resource on page 83
Add a step before and after parallel steps

Use the Insert Step Before Parallel and Add Step After Parallel tools to add steps before and after parallel steps.

To add a step before and after parallel steps:

1. Select the parallel step that is to run in parallel to the new step.

2. Select Insert Step Before Parallel. A new UNDEFINED step is created before the selected parallel step and is highlighted in blue. One of these dialog boxes opens, based on the recipe level: Select Phase, Operation Select or Unit Procedure Select.

3. Make the appropriate selection and select OK.

4. Select the parallel step that runs after the new step.

5. Select Add Step After Parallel. A new UNDEFINED step is created after the selected parallel step and highlighted in blue. One of these dialog boxes opens, based on the recipe level: Select Phase, Operation Select or Unit Procedure Select.

6. Make the appropriate selection and select OK.

See also

Define a step on page 86

Define a step

Newly added steps are UNDEFINED. Only valid lower level recipes are used in the step definition. When a process for a step is selected, that process is assigned to that particular step in the recipe, and the step is marked with the process name. Redefine a step at any time.

To define a step:

1. Double-click a step marked UNDEFINED. One of these dialog boxes opens, based on the recipe level: Select Phase, Operation Select, or Unit Procedure Select.
2. To define a phase or operation either select from the option list and then select **OK**. Or, select **New Phase** or **New Operation** to create a new phase or operation. See **Create operations and unit procedures**.

3. To define a unit procedure:
   a. From the **Alias** list, select the alias to associate with this unit procedure.
   b. The **Alias** list contains a list of all units defined in the procedures unit requirements.
      a. If this unit runs in parallel with other units, and if each of the units at this level must acquire in order to initiate this level of the recipe, then select **Acquire unit prior to starting unit procedure**.
      b. If the units running in parallel can run independently of each other, clear this check box for all of the units at this level. The recipe then runs the individual units acquired, regardless of whether the other units acquire.
   c. Select the appropriate unit procedure to associate with this step. Or, select **New Unit Procedure** to create a new unit procedure.
   d. Select **OK**.

**See also**

Create operations and unit procedures on page 58

Add a null procedure on page 36

Remove a step from a table

Use these instructions to remove a step from a table.

**To remove a step from a table:**

1. Select the step to remove.

2. From the **Recipe Construction** toolbox, select **Remove Step** or press **DELETE**.

**See also**

Insert steps into a table on page 95
Chapter 7

Bind Expressions

Use the Unit Binding Expression Builder to build bind requirement and bind preference expressions. If the bind requirement or preference type is Expression, selecting browse next to the Expression box displays the Unit Binding Expression Builder dialog box. The Unit Binding Expression Builder dialog box contains a tree view in the left pane. The tree view shows Recipe Header and Unit Attribute data used to build an expression.

Select Recipe Header, a list of recipe header parameters displays in the list view (in the middle). The Recipe Header names map to the same boxes defined within a Recipe Header.

Create or edit an expression by typing directly in the Expression Builder text box, select items in the list view and enumerations view, or select various operators. The operator buttons of the Expression Builder are always enabled. The selected operator is inserted into the expression at the current cursor position and displayed in the Expression Builder text box.

When the expression is complete, selecting OK validates and saves the expression and returns to the Bind Requirements or Bind Preferences dialog box.

There are three types of Bind Expressions: Boolean, Minimize, Maximize. Boolean expressions must evaluate to true or false. Minimize and maximize expressions evaluate to real or integer values.

See also

Expression operators expanded on page 100
Parameter expressions support these operators. The precedence of the execution depicts from highest to lowest. An operator with a higher precedence executes before an operator of lower precedence.

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( )</td>
<td>Expressions within parentheses are evaluated before expressions outside of parentheses</td>
</tr>
<tr>
<td>NOT, -</td>
<td>Logical NOT, negation (of a single argument)</td>
</tr>
<tr>
<td>*, /, AND</td>
<td>Multiplication, division, logical AND</td>
</tr>
<tr>
<td>+, -, OR</td>
<td>Addition, subtraction, logical OR</td>
</tr>
<tr>
<td>&lt;, &lt;=, &gt;, &gt;=</td>
<td>Less than, less than or equal to, greater than, greater than or equal to</td>
</tr>
<tr>
<td>=, &lt;&gt;</td>
<td>Equal to, not equal to</td>
</tr>
</tbody>
</table>

**Tip:** If the result of the expression is an Integer, the values used to build the expression must be Integers — Real is not compatible with an Integer. However, using division in an expression always results in the value being a Real number.

**Expression data types**

The data types supported are integer, real, string, and enumeration. These are data type examples:

**Integer:** 423

**Real:** 423.123456789012

**String:** The string constant must be in quotes: "READY".

**Enumeration:** As a string, the enumeration constant must be in quotes: "BUTTER_PECAN". As an integer, the ordinal for the enumeration may be: 4.
### Tip:
If the result of the expression is an Integer, the values used to build the expression must be Integers — Real is not compatible with an Integer. However, using division in an expression always results in the value being a Real number.

**See also**

**Operands**
Operands used within a binding expression can reference Recipe Header data, Unit Attributes, and constants. This table illustrates examples of allowable expression entities and the required syntax.

<table>
<thead>
<tr>
<th>Expression Entity</th>
<th>Expression Syntax</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Recipe Header data</td>
<td>[Recipe Header item]</td>
<td>[VERSION_NUMBER]</td>
</tr>
<tr>
<td>Global Unit Attribute</td>
<td>Unit Attribute Name</td>
<td>CAPACITY</td>
</tr>
<tr>
<td>Constant</td>
<td></td>
<td>100</td>
</tr>
</tbody>
</table>

**See also**

**Bind Expressions** on page 99

### Expression validation
The expression validates for consistency and syntactical correctness. If the defined expression validates successfully, the dialog box closes. If the expression is invalid, then the location of where the first error encountered highlights, a description of the error displays in a message box, and the dialog box does not close.

### Validation rules:

- An expression must resolve to either TRUE or FALSE and meet the criteria:
  - An expression must contain at least one operator (+ (addition), - (subtraction), * (multiplication), / (division), = (equals), > (greater than), < (less than), <= (less than, or equal to), >= (greater than, or equal to), <> (not equal to), Not, And, and Or.
  - **Tip:** OR transitions fire if at least one side of the expression is TRUE.
- Separate each entity (recipe header, global unit attribute, or constant) with an operator.
- The operated entities must evaluate to acceptable data types (string with a string; integer with integer or real; real with integer or real).
- Maximum expression length must not exceed 1023 characters.
- Duplicate expressions are allowed.
See also

Bind Expressions on page 99
Phase, operation sequence, and recipe parameters

Within a recipe, there are three kinds of parameters:

- Phase parameters
- Operation sequence parameters
- Recipe parameters

See also
- Phase parameters on page 103
- Recipe parameters on page 104
- Operation sequence parameters on page 104
- Assign parameter values and phase report limits on page 112
- Recipe properties on page 115

Phase parameters

When adding recipe phases to an operation-level recipe, specify the formula values for its associated phase parameters. Recall that a recipe phase links to an equipment phase configured in the area model. The recipe phase parameters originate from the parameters defined in the FactoryTalk Batch Equipment Editor for the associated equipment phase. Therefore, the recipe phase formula value parameters are the phase parameters defined for the associated equipment phase in the area model. These parameters pass data directly to the process connected device (PCD).

Configure the phase parameter values in one of four ways:

- Specify them in the FactoryTalk Batch Recipe Editor for the recipe phase at recipe creation time.
- Specify them prior to the execution of the recipe phase in a running recipe after batch creation. (Done most often with an operator prompt.)
- Specify them as a parameter expression.
Chapter 8  Phase, operation sequence, and recipe parameters

- Defer them to a recipe parameter, which allows defining at batch creation time.

See also

Recipe parameters on page 104

Operation sequence parameters

When adding an operation sequence to a unit procedure recipe, specify the formula values for its associated operation sequence parameters.

Configure to have values in one of five ways:

- Assign values to operation sequence parameters using the FactoryTalk Batch Recipe Editor.
- Specify them before the phase or operation sequence step runs, by assigning a value from an FactoryTalk Batch user control.
- Specify them when an unacknowledged prompt requests the value for a phase parameter. The Origin value for the parameter is Operator.
- Specify the value is calculated from a parameter expression. The Origin value for the parameter is Expression.
- Defer them to a recipe parameter, which allows defining at batch creation time. The Origin value for the parameter is Defer.

See also

Recipe parameters on page 104

Phase parameters on page 103

Formula Value Entry/Report Expression settings on page 108

Recipe parameters

Recipe parameters are parameters associated with the recipe as a whole, not a particular phase. Recipe parameters:

- Are created and defined in the FactoryTalk Batch Recipe Editor.
- Are specific to the recipe.
- Define any recipe level: operation, unit procedure, or procedure.

The values of the highest level recipe parameters within a batch are defined at batch creation time.

Recipe parameters are:

- Used in transition conditions.
• Used to assign parameter values to lower level steps in a process called deferring.

• Phase formula value parameter values may defer to the value of Operation-level recipe parameters.

• Operation-level recipe parameters may defer to the value of Unit Procedure-level recipe parameters.

• Unit Procedure-level recipe parameters may be deferred to the value of Procedure-level recipe parameters.

  Tip: Recipe parameters of unit procedures and operation steps can also use expressions to reference parent and peer step data (for example, to calculate a value that other parameter expressions reference).

• Procedure-level recipe parameters define at batch creation.

<table>
<thead>
<tr>
<th>Recipe Formula Parameters</th>
<th>Formula Values</th>
</tr>
</thead>
<tbody>
<tr>
<td>Procedure</td>
<td>Batch creation</td>
</tr>
<tr>
<td>Unit Procedure</td>
<td>Procedure</td>
</tr>
<tr>
<td>Operation</td>
<td>Unit Procedure</td>
</tr>
<tr>
<td>Phase in FactoryTalk Batch Equipment Editor</td>
<td>Operation</td>
</tr>
</tbody>
</table>

Tip: The grayed cells indicate the function cannot perform within the FactoryTalk Batch Recipe Editor.

See also

Defer parameters on page 105

Defer parameters

Define the values of the highest level recipe parameters within a batch at batch creation time. For example, assume:

• A phase parameters value defers to an operation-level recipe parameter defined for its containing operation-level recipe.

• The operation-level recipe parameters value defers to a unit-procedure-level recipe parameter defined for its containing unit procedure-level recipe.

When creating a unit procedure-level recipe instance, the operator defines the value for the unit-procedures recipe parameter at batch creation time because this is the highest level recipe of the batch. The value the operator defines for the unit-procedure’s recipe parameter passes down to the operation-level recipe parameters value, which then passes down to the phase parameters value at batch creation time. When the FactoryTalk Batch Server downloads parameters to the process-connected-device prior to starting the recipe phase, the phase parameters value is sent to the process-connected device.

Deferring phase parameter values to recipe parameters provides a way to define the recipe parameter value at batch creation time rather than waiting for the Server to
initiate the phase within the recipe after batch creation time. This frees the operator from closely monitoring the running recipe for phase parameter value prompts that would not occur until the Server initiates the phase.

Deferring phase parameters also provides a way for third-party scheduling packages or other applications working through the Server API (such as the ActiveX Controls) to create batches and define the values of the corresponding recipe parameters for those batches at batch creation time.

See also

Recipe parameters on page 104

Step formula values and phase report limits

Assign formula values by specifying the values and choosing the step parameter displayed within an operation. Report parameters pass phase information to the FactoryTalk Batch server for reports. Typically, this information is a process value. The parameters and report limits that display in the Formula Value Entry/Report Expression dialog box are the configured phase parameters and report limits defined in the area model. Configure the step formula values and modify the report limits in the Formula Value Entry/Report Expression dialog box.

If control strategies are configured for the phase class, the first item in the Parameters values area is named CONTROL_STRATEGY and the Value column for this item contains a list of configured control strategies. Only the parameters and reports associated with the selected control strategy display. Select the appropriate control strategy before configuring parameters and report limits. When configuring parameters, the parameter values may be assigned as literal values or deferred to another parameter at a higher level in the recipe.
If the step selected is a material-enabled phase class, the Formula Value Entry/Report Expression dialog box also contains the recipe parameters (MATERIAL and AMOUNT) and the report parameters (ACTUAL_AMOUNT and FEED_COMPLETE) associated with material phases. If optional parameters are enabled, it will also contain the optional material recipe parameters CONTAINER, LOT, LABEL, and MATERIAL_CLASS.

Tip: To enable the optional parameters in the FactoryTalk Batch Equipment Editor, double-click the phase class to open the Edit Phase Class dialog box. On the Parameters tab, make sure the Add Optional Material Parameters box is selected.

See also

Material recipe parameters on page 107

Material report parameters on page 108

Parameter Value Entry/Report Limit settings on page 108

Material recipe parameters:

The values of the MATERIAL, LOT and LABEL parameters are combined to create the material specification. The material specification is used to determine the container from which a material is drawn, or into which a material is distributed, when a batch is run. The material specification always contains a material name. Optionally, the specification can include a lot, a label, or both.

Material recipe parameters:

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Purpose</th>
</tr>
</thead>
<tbody>
<tr>
<td>AMOUNT</td>
<td>Indicates the quantity of the material to be used, or created, in the recipe. A positive number or zero (0), indicates a material addition. A negative number or zero (0), indicates a material distribution. If needed, defer the value assignment for the AMOUNT parameter to a higher recipe level or to the operator.</td>
</tr>
<tr>
<td>CONTAINER</td>
<td>Cannot be modified. This parameter is used only at run time. Each time a step is bound to a container, the FactoryTalk Batch Server stores the container's Controller ID (the CONTAINER's enumeration set ordinal value) and string into this parameter. The container associations are defined for each phase in the area model.</td>
</tr>
<tr>
<td>LABEL</td>
<td>Indicates a specific sublot of material to be used in the recipe, either as a material addition or a material distribution. If a specific sublot is not needed, leave the value of the parameter blank and the FactoryTalk Batch Server selects a sublot based on the material specified, the unit in which the phase is running, and the phases container associations. When distributing, the value of the LABEL parameter is assigned to the Distributed Sublot. The LABEL parameter value assignment cannot be deferred.</td>
</tr>
</tbody>
</table>
LOT Indicates the lot to be used in the recipe either as an addition or a distribution. If the recipe does not require a specific lot, leave the parameter value blank and the FactoryTalk Batch Server selects a lot based on the material specified, the unit in which the phase is running, and the phase’s container associations. In distributions, the LOT parameter value is assigned to the Distributed Sublot. The LOT parameter value assignment cannot be deferred.

MATERIAL Indicates a specific material to be used in this particular phase. A value must be assigned; and cannot defer to the value assignment. If the operator selects from a class of materials at the time the batch is run, select NULL_MATERIAL in order to enable the MATERIAL CLASS parameter.

MATERIAL CLASS Indicates a class of materials to be used in this particular recipe. Assign a value; the value assignment cannot be deferred. If a material class is selected, the operator is prompted to select a specific material from the material class at the time the batch is added to the batch list in FactoryTalk Batch View. Only one specific material is to be used in this phase, select NULL_CLASS in order to enable the MATERIAL parameter.

Tip: Materials and material classes are configured in the FactoryTalk Batch Material Editor. Use the same recipe to change back and forth between a material-specific recipe and a material class-based recipe by changing the MATERIAL and MATERIAL_CLASS parameters.

See also

- Step formula values and phase report limits on page 106
- Material report parameters on page 108
- Parameter Value Entry/Report Limit settings on page 108

Material report parameters

The FEED_COMPLETE parameter reports that a material addition or distribution is complete.

The ACTUAL_AMOUNT report parameter records the actual quantity of the material produced or consumed.

See also

- Parameter Value Entry/Report Limit settings on page 108
The **Formula Value Entry/Report Limit Entry** dialog box is used to define the phase, operation sequence, or recipe parameters and reports associated with a recipe step. The **Formula Value Entry/Report Limit Entry** dialog box contains:

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Parameters value entry</strong></td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>The formula parameter name associated with this step. (view-only)</td>
</tr>
<tr>
<td>Type</td>
<td>The parameter data type as configured in the area model (for phase parameters) or the FactoryTalk Batch Recipe Editor (for recipe parameters). Valid types include Real, Integer, Enumeration, and String. (view-only)</td>
</tr>
<tr>
<td>Origin</td>
<td>A single word description of where the value for this parameter is assigned. Valid entries for the Origin box are: Value: This assigns a default fixed value for the parameter (not valid for Timer steps). Operator: Only available for timer, phase, or operation sequence step formula values. Postpones the value assignment to the operator. A prompt is generated in FactoryTalk Batch View when the recipe is run. The operator is prompted for a value when the phase or operation sequence is executing. The operator must enter a value for the parameter for that phase or operation sequence to continue running. Defer: Defers the value assignment to the next higher recipe level. When Defer is selected, a list of appropriate recipe parameters to which this phase or recipe parameter may be deferred is presented in the Value column. This list only includes existing recipe formula parameters that have the same data type as the lower level phase or recipe parameter, and a range less than or equal to the lower level phase or recipe parameter. These material parameters cannot be deferred: MATERIAL, LOT, LABEL, or MATERIAL CLASS. Expression: Allows assignment of an expression result as the value for the parameter. Select browse (/open) to open the Parameter Expression Builder.</td>
</tr>
<tr>
<td>Min</td>
<td>The minimum value for this parameter. This is determined when the phase parameter is defined in the area model or the recipe parameter is defined in FactoryTalk Batch Recipe Editor. (view-only)</td>
</tr>
<tr>
<td>Value</td>
<td>The value that is to be assigned to this parameter. This is dependent on the selected Origin. For material-enabled phases: MATERIAL — contains a list of materials retrieved from the material database. AMOUNT — dependent on the selected Origin. May be set to a material amount if the Origin box is Value. The sign of this value specifies whether the phase is a material distribution (amount is negative) or a material addition (amount is positive or zero). The default value is zero.</td>
</tr>
<tr>
<td>Item</td>
<td>Description</td>
</tr>
<tr>
<td>---------------------</td>
<td>-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------</td>
</tr>
<tr>
<td>Parameters value entry</td>
<td>The maximum value for this parameter. This is determined when the phase parameter is defined in the area model or the recipe parameter is defined in FactoryTalk Batch Recipe Editor. (view-only)</td>
</tr>
<tr>
<td>Max</td>
<td>The maximum value for this parameter. This is determined when the phase parameter is defined in the area model or the recipe parameter is defined in FactoryTalk Batch Recipe Editor. (view-only)</td>
</tr>
<tr>
<td>Enum/EU</td>
<td>Enumeration/Engineering Units. The engineering units associated with the parameter value. This is assigned when the phase parameter is defined in the area model or the recipe parameter is defined in FactoryTalk Batch Recipe Editor. (view-only)</td>
</tr>
<tr>
<td>Display</td>
<td>Indicates display of this parameter value in the step in the SFC. Only one parameter may be displayed per step.</td>
</tr>
</tbody>
</table>
| Verification Method | The algorithm that specifies which policy and set of limits verify the parameter value validity. There are four possible verification methods:  
  • High-High-High/Low-Low-Low  
  • High-High/Low-Low  
  • High/Low  
  • No Limits  
  The Verification Method is set in FactoryTalk Batch Equipment Editor. (view-only) |
| LLLL                | The Low-Low-Low limit value for the parameter (Real, Integer). This is only enabled if the Verification Method is High-High-High/Low-Low-Low.                                                                 |
| LL                  | The Low-Low limit value for the parameter (Real, Integer). This is only enabled if the Verification Method is High-High-High/Low-Low-Low or High-High/Low-Low.                                                        |
| L                   | The Low limit value for the parameter (Real, Integer). This is disabled if the Verification Method is No Limits.                                                                                              |
| H                   | The High limit value for the parameter (Real, Integer). This is disabled if the Verification Method is No Limits.                                                                                              |
| HH                  | The High-High limit value for the parameter (Real, Integer). This is only enabled if the Verification Method is High-High-High/Low-Low-Low or High-High/Low-Low.                                                        |
| HHH                 | The High-High-High limit value for the parameter (Real, Integer). This is only enabled if the Verification Method is High-High-High/Low-Low-Low.                                                                     |
| Container Binding area (material-enabled phase only) | Specifies the method used to bind the material-enabled phase to a container.  
  **Automatic**: The FactoryTalk Batch Server selects the appropriate container based upon the container associations configured in the area model and the material specification.  
  **Prompt**: The operator is prompted to choose the container and phase to which the phase is to be bound.                                      |
### Phase, operation sequence, and recipe parameters

#### Chapter 8

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feed Type area</td>
<td>The selection available is dependent upon the phase parameter values defined in the area model. If the phase is an addition (material is added to a recipe), <strong>Addition</strong> is selected and cannot be edited. If the phase is a distribution (material is added to inventory), <strong>Distribution</strong> is selected and cannot be edited. If the phase is configured as both, specify the material feed type associated with this step. The <strong>Max</strong> and <strong>Min</strong> values adjust accordingly.</td>
</tr>
</tbody>
</table>

Many aspects of recipe configuration are dependent upon accessing data from the material database using the FactoryTalk Batch Material Server. If the Material Server is not available, the MATERIALS, CONTAINERS, and MATERIAL CLASS enumeration sets are recreated with the NULL_MATERIAL, NULL_CONTAINER, and NULL_CLASS, respectively. reconfigure the materials after the server becomes available.

#### Report Limits area

<table>
<thead>
<tr>
<th>Name</th>
<th>The parameter name. (view only)</th>
</tr>
</thead>
<tbody>
<tr>
<td>ID</td>
<td>The number, unique to the phase that identifies this parameter. (view only)</td>
</tr>
<tr>
<td>Type</td>
<td>The parameter data type. Valid types include: Real, Integer, Enumeration, and String. (view only)</td>
</tr>
<tr>
<td>Accumulate</td>
<td>Displays the type of accumulation for the report as defined in the phase class in the area model. (view only)</td>
</tr>
<tr>
<td>Verification Method</td>
<td>The type of verification method used for the parameter. This method sets the limits and policies for verifying report values (view-only).</td>
</tr>
<tr>
<td>Limit Calculation</td>
<td>The equation type used to calculate the limit. There are three ways to calculate this limit: <strong>Absolute</strong>: Absolute value for the parameter. <strong>Percentage</strong>: Target Parameter + (Target Parameter * Percentage) <strong>Relative</strong>: Target Parameter + Value (view-only)</td>
</tr>
<tr>
<td>Target Parameter</td>
<td>The recipe parameter used in calculating the limit when the Limit Calculation is Percent or Relative. (view-only)</td>
</tr>
<tr>
<td>LLL</td>
<td>The Low-Low-Low limit for the parameter (Real, Integer). This is only enabled if the Verification Method is High-High-High/Low-Low-Low.</td>
</tr>
<tr>
<td>LL</td>
<td>The Low-Low limit for the parameter (Real, Integer). This is only enabled if the Verification Method is High-High-Low-Low-Low or High-High-Low-Low.</td>
</tr>
<tr>
<td>L</td>
<td>The Low limit for the parameter (Real, Integer). This is disabled if the Verification Method is No Limits.</td>
</tr>
<tr>
<td>H</td>
<td>The High limit for the parameter (Real, Integer). This is disabled if the Verification Method is No Limits.</td>
</tr>
<tr>
<td>HH</td>
<td>The High-High limit for the parameter (Real, Integer). This is only enabled if the Verification Method is High-High-Low-Low-Low or High-High-Low-Low.</td>
</tr>
</tbody>
</table>
| HHH | The High-High-High limit for the parameter (Real, Integer). This is only enabled if the Verification Method is High-High-High-Low-Low.
Assign parameter values and phase report limits

Use these instructions to assign parameter values and phase report limits.

To assign parameter values and phase report limits:

1. Select a step in either the SFC view or Table view.

2. Select Value Entry to display the Formula Value Entry/Report Expression dialog box.

3. If the step is a phase with configured control strategies, select the appropriate control strategy from the Value list in the row named CONTROL_STRATEGY. Only parameters and reports associated with the selected control strategy display.
4. For each formula parameter, select the appropriate origin from the **Origin** list. (If the step is a phase or operation sequence, the list includes **Operator**.)

5. In the **Value** box, type a value for each parameter.

6. If the data **Type** is **Enumeration**, select an enumeration from the **Value** list based on the selected origin.

   **Tip:** The list of enumerations displays in alphabetical order and configure in the area model.

7. Select **Display** to see the step formula value for this parameter in the SFC view step (step formula values always display in the **Table** view).

8. To change the default parameter deviation limit values defined for the phase in the area model, edit the limit values for this recipe in the LLL, LL, L, H, HH, and HHH boxes. The values entered determine deviation events at run time.

9. If this is a material-enabled phase, select the type of binding in the **Container Binding** area.

10. If this is a material-enabled phase, and the parameter is configured for addition and distribution, specify the material feed type associated with this step in the **Feed Type** area.

11. In the **Report Limits** area, enter the report deviation limit values for this recipe in the LLL, LL, L, H, HH, and HHH boxes. The values entered determine deviations in a report value at run time.

12. Select **OK** to return to the FactoryTalk Batch Recipe Editor window.

**See also**

- [Step formula values and phase report limits](#) on page 106
- [Formula Value Entry/Report Expression settings](#) on page 108

---

**Assign formula values to Timer steps**

There are three parameters available for a COUNT_DOWN timer and two parameters available for a COUNT_UP timer.

**To assign formula values to Timer steps:**

1. Select a timer step in either the **SFC** view or **Table** view.
2. Select **Value Entry**. The **Parameter Value Entry/Report Limit Entry** dialog box appears.

![Parameter Value Entry/Report Limit Entry dialog box](image)

3. For the **HOLD_BEHAVIOR** formula parameter, select the appropriate origin from the **Origin** list.

- The origin of the SETPOINT and HOLD_BEHAVIOR parameters of a COUNT_DOWN Timer can be set to OPERATOR, DEFER, or to the result of an EXPRESSION. When a COUNT_DOWN Timer parameter is OPERATOR, the FactoryTalk Batch Server prompts the operator on start of the Timer step to supply a runtime value for the parameter. When a COUNT_DOWN Timer parameter is DEFER, the FactoryTalk Batch Server looks to the parent procedure (or upper level recipe) for the value as defined on the **Recipe Formula Parameter** list.

- Select one of three values for HOLD_BEHAVIOR: CONTINUE, RETENTIVE, and RESET. The default value is CONTINUE.

- The origin of the HOLD_BEHAVIOR parameter of a COUNT_UP Timer can be OPERATOR, DEFER, or EXPRESSION. When a COUNT_UP Timer parameter is OPERATOR, the FactoryTalk Batch Server prompts the operator on start of the COUNT_UP Timer step to supply a runtime value for the parameter. When a COUNT_UP Timer parameter is set to DEFER, the FactoryTalk Batch Server looks to the recipe parameter to supply the value for the step parameter.

- When the **Origin** is VALUE, at runtime, the value for the parameter comes from the value specified in the **Value** column.

4. The second parameter displayed is **SETPOINT**. The **SETPOINT** determines how long the timer step runs. The default value for the **SETPOINT** parameter is the default value configured for the **SETPOINT**
parameter of the TIMER system phase. The SETPOINT parameter is of the type Real.

5. The third parameter displayed is TIMER_TYPE. It is set to the enumeration value of COUNT_UP or COUNT_DOWN. If all the boxes of this parameter disable; default values cannot change.

6. Select the Display check box to display the SETPOINT, TIMER_TYPE, or HOLD_BEHAVIOR parameter values on a timer step.

7. Select OK to return to the FactoryTalk Batch Recipe Editor window.

See also

Expressions in parameters and reports on page 119

Recipe properties

Use the Procedure Properties dialog box to configure parameters and reports for recipe procedures and steps. There are two tabs on the Procedure Properties dialog box – Parameters and Reports.

See also

Parameters on page 115

Reports on page 116

Aggregate report values on page 118

Parameters

Recipe formula parameters provide flexibility when creating recipes. Use recipe parameters in transition conditions or to substitute for lower-level phase or recipe parameters. Create a recipe parameter for Operation, Unit Procedure, or Procedure recipe levels.

Recipe parameter values assign as follows:

- If the operation is the highest level recipe for the batch, operation recipe parameter values assign when the operation step is added to a unit procedure-level recipe or at batch creation.
- If the unit procedure is the highest level recipe for the batch, unit procedure recipe parameter values assign when the unit procedure step is added to a procedure-level recipe or at batch creation.
- Procedure-level recipe parameter values always assign at batch creation because this is the highest possible recipe level for the batch.

A recipe formula parameter is a parameter specific to a recipe. A parameter can be used to pass values from one level of a recipe to the next lower level. A recipe formula parameter is configured in the FactoryTalk Batch Recipe Editor on a
procedure or operation step. Recipe parameters can be deferred to recipe formula parameters.

See also

Assign recipe formula parameters on page 116

Assign recipe formula parameters

Use these instructions to assign recipe formula parameters on the **Properties** dialog box, **Parameters** tab.

To assign recipe formula parameters:

1. Select **Recipe Formula Parameters**

2. On the **Parameters** tab, select **Add Parameter** to create a new recipe formula parameter. A new parameter row displays and is populated with default values.

3. Type the appropriate information in the boxes.

4. Repeat steps 2 and 3 until all recipe formula parameters are defined.

5. Select **Apply** to save changes and keep the **Properties** dialog box open, select **OK** to save any changes and return to the FactoryTalk Batch Recipe Editor.

6. To delete a recipe formula parameter, select the row to delete and select **Delete**. Select **OK** to save any changes and return to the FactoryTalk Batch Recipe Editor.

See also

Parameters on page 115

Reports

Use report parameters in transition conditions or to substitute for lower-level phase or recipe parameters. Create a report parameter for Operation, Unit Procedure, or Procedure recipe levels. Reports on operation, unit procedure, and procedure recipes accumulate data from their child steps to aid report generation activities. Report values are calculated and recorded in the batch event journal when the state of the procedure is Complete, Aborted, or Stopped.
See also

Assign report parameters on page 117

Assign report parameters

Use these instructions to assign report parameters on the Properties dialog box Reports tab.

Assign report parameters:

1. Select Recipe Formula Parameters.


3. Enter a name for the report.

4. Select the data type from the list in the Type column, and enter the appropriate engineering units in the Enum/EU column.

If the Type is Enumeration, select an enumeration from the Enum/EU list.

5. Select browse next to the Report Expression box to display the Report Expression Builder dialog box.
6. Create the desired expression to use as the report value, and then select OK to save the expression into the Report Expression box.

7. Repeat steps 2 through 6 until all reports are defined.

8. Select Apply to save changes and keep the Properties dialog box open, select OK to save any changes and return to the FactoryTalk Batch Recipe Editor.

9. To delete a report, select the row to delete and select Delete. Select OK to save any changes and return to the FactoryTalk Batch Recipe Editor.

See also

Expression value configurations on page 122

Aggregate report values

The report parameters of phases cannot have expressions because a control program running outside the FactoryTalk Batch Server supplies these values. There is one exception—the ability to accumulate the values uploaded by phase logic. The phase logic does not sum integer and real values for reporting purposes.

See also

Reports on page 116
Expressions in parameters and reports

A parameter expression is an arithmetic expression assigned to a recipe parameter or report parameter of a recipe or a step. Assign a parameter expression to all levels of the procedural control hierarchy (phase, operation, unit procedure, batch procedure) where it is appropriate.

Parameters provide a means to perform calculations in the structure of the recipe. The use of parameter expressions avoids complex and awkward downloading and uploading of parameter data so the control can perform needed mathematical operations. A key application is the ability to aggregate report (output) parameter data into key process indicators (KPIs) at the appropriate level of the recipe. A secondary application is to enable flexibility in calculating recipe (input) parameters.

The terms of the expression may reference several different sources of data. Recipe and report parameters behave differently because their expected roles are different. A recipe parameter supplies data to the process. A report parameter reports a result of the process. The two kinds of parameters evaluate and the result is stored as the value of the parameter at different times.

A parameter expression contains one or more terms with arithmetic operations connecting them. A term may be a constant value, a data reference, or another expression. For recipe parameters, the data references are subscribed. As the data sources change their values, the expression is notified. The expression evaluates the value and stores the new value for the recipe parameter.

For report parameters, evaluation of the expression does not update as references change, but as the expression triggers. Conceptually, report parameters intend to record data resulting from the execution of the process the step represents. This data typically comes from below the recipes level of the recipe hierarchy (child data). Or the data references the current level (peer data or data on the same step) and may include recipe header data, equipment data, and runtime data.

Configure report parameter expressions on operation, unit procedure, and batch procedure recipes. Expressions on phase report parameters are not supported.
Evaluated of recipe parameter expressions

At runtime, parameter expressions constantly evaluate. Subscriptions create to
each data reference. As these values update, the expression evaluates and updates
the value of its parameter. To avoid overwhelming the event journal with frivolous
events, not every change in value of an input expression records in the event
journal.

It is possible for an expression to evaluate to a value outside the range limits of the
recipe parameter. When this happens, the parameter assigns the value
BAD_VALUE. There are several ramifications for a parameter with the
BAD_VALUE assignment:

- A download request to a controller will cause the requesting equipment
  phase to be HELD.
- A transition expression referencing the BAD_VALUE parameter will itself
evaluate to BAD_VALUE, which will prevent the transition from firing.
- A binding expression referencing the BAD_VALUE parameter is unable to
  find binding candidates. Binding is pending and the recipes execution does
  not advance.
- An input parameter expression referencing a parameter with a
  BAD_VALUE will evaluate its expression to BAD_VALUE.

See also

Evaluation of recipe parameter expressions on page 120
Evaluation of report parameter expressions on page 121
Expression value configurations on page 122

Validation of parameters with expressions

When a parameter expression evaluates, the result compares to the verification
policy (Hi/Low, Hi-Hi/Low-Low, or Hi-Hi-Hi/Low-Low-Low). If the value falls
outside the expected range, the appropriate verification policy action triggers. A
value of BAD VALUE does not trigger a verification policy.

Tip: Parameter validation limits, which can result in the invocation of an
electronic signature, only support phases.

See also

Validation of parameters with expressions on page 120
Calculated recipe parameter expression value override on page 121
Calculated recipe parameter expression value override

The value of a parameter having an expression cannot change unlike parameters having an origin of VALUE. The Batch View and Batch ActiveX controls provide the ability to override the expression and assign a value when responding to extraordinary circumstances. The new displayed value is marked as using the Override function.

See also

Validation of parameters with expressions on page 120

Evaluation of report parameter expressions

At runtime, report parameter expressions differ from recipe parameter expressions when evaluated. Recipe parameter expressions update continuously. Report parameter expressions are configured to update on triggers. The report values are uploaded to the FactoryTalk Batch Server based on the configured report parameter to upload on terminal state, or if an upload is requested by the phase logic.

There are two scenarios for phases:

- One, the phase step transitions to a terminal state and the values automatically upload.
- Two, an upload is requested by the phase logic (2XXX and 12XXX upload commands). So the upload is made on demand. Thus, report parameter expressions update when their step transitions to a terminal state (COMPLETE, STOPPED, and ABORTED).

For operation sequences, report parameter expressions update when their step transitions to a terminal state (COMPLETE, STOPPED, and ABORTED).

See also

Override a calculated report expression on page 121

Override a calculated report expression

The operator under any circumstances may not change calculated report parameters. This corrupts the record of the execution of a procedure. There are two exceptions where the value of a report expression may be changed:

1. A transition expression references the report parameter and its value is preventing the transition from firing. In this case, use the Force Transition function to force the transition to fire.

2. A recipe parameter is dependent on a report parameter to provide its value. If the report parameter is incorrect or undesirable, there is cause to change it. In this case, use the Override function to change the recipe parameter's value.
Expression value configurations

A parameter expression is an arithmetic expression that may be assigned to an input parameter or output parameter of a phase, operation, or unit procedure step that can reference other parameters and recipe header data within a recipe. The expression is evaluated and the result stored as the value of the parameter.

A recipe parameter expression may also reference recipe/formula parameters of the parent operation recipe, data in the operations recipe header, the phases own recipe parameters, and the phases own report parameters. The FactoryTalk Batch Server will evaluate the expression and store the result in the phase steps parameter.

For example, in a control system, the quantity of sugar added is a function of the quantity of water added by another phase. Configure a recipe parameter referencing the ADD_WATER report parameter of another phase and use an expression to calculate the amount of sugar added as the ADD_SUGAR formula parameter on this phase.

See also

Configure expression values in recipe and report parameters on page 122

Parameter expression functions on page 27

Use these instructions to configure expression values.

Before you begin:

Make sure:

- The data types of the referenced parameters in the expression are compatible.
- The data types real, integer, enumeration, and string are supported (no string manipulation is allowed).
- The expression reference phase parameters and report parameters within the open active step.

To configure expression values in recipe and report parameters:

1. Select a step in either the SFC view or Table view.
2. Select Value Entry to display the Parameter Value Entry/Report Limit Entry dialog box.

3. Select a recipe parameter or report.

4. Select Expression from the Origin list, and then select browse in the Value column. (If the step is a phase, the list will include Operator.) The Parameter Expression Builder opens.

5. From the tree view in the left pane, select the step containing the recipe or report parameter.

6. From the right pane, double-click the recipe or report parameter used for the expression to copy it to the text box (or select it and select Paste).

7. Select an operator or function button to enter the desired operation or math function into the expression. Type directly into the Expression text box.
For this example, a value of 2 times the amount of milk added is used to determine the amount of sugar to add:

```
For this example, a value of 2 times the amount of milk added is used to determine the amount of sugar to add:
```

**Tip:** The maximum length of a parameter expression is 1023 characters.

8. Select **OK** to save and validate the expression and to return to the **Parameter Value Entry/Report Limit Entry** dialog box. The report parameter expression displays.

9. Select **OK** to close the **Parameter Value Entry/Report Limit Entry** dialog box and validate the expression.

In a running batch, the parameter expression continuously evaluates. In the example used in these instructions, the calculation and resulting value is the amount for the ADD_SUGAR phase. This amount would be 2 times the water added (50 KG) for a total amount of 100 KG of sugar.

**See also**

- Expression value configurations on page 122
- Expression operators on page 124
- Parameter expression functions on page 27
Expression operators

Parameter expressions support these operators. The precedence of the execution depicts from highest to lowest. An operator with a higher precedence executes before an operator of lower precedence.

<table>
<thead>
<tr>
<th>Operator</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>( )</td>
<td>Expressions within parentheses are evaluated before expressions outside of parentheses</td>
</tr>
<tr>
<td>*, /</td>
<td>Multiplication, division</td>
</tr>
<tr>
<td>+, -</td>
<td>Addition, subtraction</td>
</tr>
</tbody>
</table>

Tip: If the result of the expression is an Integer, the values used to build the expression must be Integers — Real is not compatible with an Integer. However, using division in an expression always results in the value being a Real number.

See also

Parameter expression functions on page 27

Parameter expression functions

Functions determine how the expression parser handles Real and Integer data types used in a parameter expression. This table lists available functions and their behavior on positive and negative values:

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
<th>Behavior: Value</th>
<th>Behavior: Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>RND( )</td>
<td>Round – Numeric values round to the nearest integer.</td>
<td>6.7 6.5 6.3 -6.7 -6.5 -6.3</td>
<td>7 7 6 -7 -6 -6</td>
</tr>
<tr>
<td>RDUP( )</td>
<td>Round up – Numeric values round to the next larger integer.</td>
<td>6.7 6.5 6.3 -6.7 -6.5 -6.3</td>
<td>7 7 7 -6 -6 -6</td>
</tr>
<tr>
<td>TRNC( )</td>
<td>Truncate – Retains only the integer portion of the numeric value.</td>
<td>6.7 6.5 6.3 -6.7 -6.5 -6.3</td>
<td>6 6 6 -6 -6 -6</td>
</tr>
<tr>
<td>ABS( )</td>
<td>Absolute – Numeric values are positive values. For example, if the Real or Integer is 6.7, there is no effect. If the Real or Integer is -6.7, it multiplied by -1 and is 6.7.</td>
<td>6 6.7 -6 -6.7</td>
<td>6.7 6 6.7</td>
</tr>
</tbody>
</table>
### Chapter 9  Expressions in parameters and reports

<table>
<thead>
<tr>
<th>Function</th>
<th>Description</th>
<th>Behavior: Value</th>
<th>Behavior: Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>MOD( )</td>
<td><strong>Modulo</strong> – Returns the modulo, or remainder, of a division. (Integers only)</td>
<td>7 MOD 6</td>
<td>1</td>
</tr>
</tbody>
</table>

See also

[Transition operators on page 26](#)
Phase link groups

Phase link groups identify phases that may communicate and work together for these reasons:

- **Synchronization** – For example, one phase running logic will not proceed until another phase is at a certain point within its own phase logic.
- **Permissives** – For example, one phase must pass a certain point in its phase logic before its partners can begin their phase logic.
- **Data Transfer** – For example, moving data from one phase to another.

Phases that need to communicate with each other are message partners. Message partners organize into link groups in the FactoryTalk Batch Recipe Editor. When a recipe containing message partner phases is on the Batch List, the link group definition informs the FactoryTalk Batch Server which phases need to communicate with each other.

There are three programs involved in implementing message partners:

- **FactoryTalk Batch Equipment Editor** – defines the number of message partners for a phase class within the Area Model.
- **FactoryTalk Batch Recipe Editor** – defines the grouping of message partner phases into link groups.
- **Phase Logic** – where the actual communication between the phases occurs.

**Tip:** Before defining phase link groups, define the corresponding message partners in the area model.

A phase link group must contain the same number of phases as there are message partners plus one (the phase itself), configured in the area model for each of the phases that are included in the group. For example, if each phase in a phase link group is configured with three message partners, then there must be a total of four phases in the phase link group (each phase has three message partners plus itself). Message partners are configured in the FactoryTalk Batch Equipment Editor.

**See also**

- [Message partners](#)
- [Create a phase link group](#)
Message partners

When a recipe containing message partner phases is on the Batch List, the link group definition informs the FactoryTalk Batch Server which phases need to communicate with each other.

There are three programs involved in implementing message partners:

- **FactoryTalk Batch Equipment Editor** – defines the number of message partners for a phase class within the Area Model.
- **FactoryTalk Batch Recipe Editor** – defines the grouping of message partner phases into link groups.
- **Phase Logic** – where the actual communication between the phases occurs.

**Tip:** Before defining phase link groups, define the corresponding message partners in the area model.

A phase link group must contain the same number of phases as there are message partners plus one (the phase itself), configured in the area model for each of the phases that are included in the group. For example, if each phase in a phase link group is configured with three message partners, then there must be a total of four phases in the phase link group (each phase has three message partners plus itself). Configure message partners in the FactoryTalk Batch Equipment Editor.

**See also**

Application of message partners example on page 128

A common message partner application is with a Transfer-Out phase on one unit and a Transfer-In phase on another unit running in parallel within a Procedure-level recipe.

When the Transfer-Out phase on a unit is ready to send its material to the next unit, the Transfer-Out phase’s running logic sets its _RQ value equal to 5501 (where 01 is the message number). The Transfer-Out phase’s running logic, if coded to do so, halts and waits for the Transfer-In phase to set its _RQ value. When the Transfer-In phase is ready to receive the material from the previous unit, the Transfer-In phase’s running logic sets its _RQ value equal to 5201 (where 01 is the message number) and, if coded to do so, halts and waits.

The FactoryTalk Batch Server then coordinates the messaging when it sees that both message partners have either a 55xx or a 52xx value in their respective _RQ memory registers and their xx value (the message number) is the same. The FactoryTalk Batch Server then commands the PLI to reset both message partners’ _RQ values to 0. Once both message partners see their _RQ registers reset to 0, their running logic (which had halted while they were waiting for their respective message partner) then continues to execute.
The sequence of events is as follows:

- **T1**: Transfer-Out phase is ready; sets _RQ = 5501 and halts its running logic.
- **T2**: Transfer-In phase is ready; sets _RQ = 5201 and halts its running logic.
- **T3**: FactoryTalk Batch Server commands the PLI to reset both _RQ registers to 0.
- **T4**: PLI resets both _RQ registers to 0.
- **T5**: All running logic resumes execution.

See also

- [Create a phase link group](#) on page 130
- [Delete a phase link group](#) on page 131

## Link group rules

The rules involving link groups are as follows:

- Up to 200 link groups per recipe.
- Up to 20 phases per link group.
- Define all phases in a link group with the same number of message partners in the area model's phase class definition.
- The total number of phases within a link group must be equal to the number of configured message partners plus one.
- Assign each phase instance to only one link group per recipe.
- Phases in phase link groups must have unique names. If the phase names are not unique, it will result in missing message partners in the linked groups.

Message partners must run in parallel within a recipe and the phase logic must run in parallel so the phases can synchronize.

**Tip:** When using phase link groups in conjunction with control strategies, the number of phase link groups cannot vary between control strategies configured for a single phase. For example, if a phase configures with two control strategies, CONTROL_STRATEGY_1 and CONTROL_STRATEGY_2, and the phase has two configured phase link groups, then both control strategies must also have two phase link groups. CONTROL_STRATEGY_1 cannot have a different number of phase link groups than CONTROL_STRATEGY_2.

See also

- [Phase link groups](#) on page 127
Phase communication

The actual communication between the phases is programmed in the phase logic using the _RQ = 5Xxx request category. Commonly implemented between two individual phases using 55xx and 52xx where:

- **55xx** means Request to Wait for a message from another phase.
- **52xx** means Send Message and Wait for One Receiver.
- **xx** means the message number from 00 to 99. All message partners must utilize the exact same message number.

One phase is responsible for sending a message to its partners at the appropriate time. The synchronization occurs when one phase in the group sends the message and the appropriate number of phase partners wait for the message.

See also

- [Create a phase link group](#) on page 130
- [Delete a phase link group](#) on page 131

Create a phase link group

Use these instructions to create a phase link group.

**Tip:** A phase may only be contained in one phase link group.

**To create a phase link group:**

1. From the **Link** menu, select **Link Groups**.

![Phase Link Group][1]

2. Select the appropriate group column heading. The cells in the column highlight.

3. Select the appropriate phase in the **Procedure View** pane, or select a phase step in the **Recipe Construction** pane.

4. Select **Add**. The selected phase is added to the link group.

5. Repeat steps 3 and 4 until all phases have been added to the group.
6. Select **OK** to save changes.

**See also**

- [Link group rules](#) on page 129
- [Delete a phase link group](#) on page 131

---

**Delete a phase link group**

Follow these instructions to remove a phase link group.

**To remove a phase link group:**

1. From the **Link** menu, select **Link Groups** to display the **Phase Link Group** dialog box.

2. Select the column header of the group to delete. The cells in the column are highlighted.

3. Select **Delete**. The selected group is removed.
Tip: Phase link groups can exist at any level of a recipe, but only the root level link groups can be seen or edited.

All of these recipe elements can contain phase link groups. To edit the phase link group contained in Operation A, open Operation A by itself.

See also

Create a phase link group on page 130
Recipe approval process overview

Use the recipe approval process to validate the development and maintenance of batch recipes. Signature certification allows the recipe approval process to safeguard the design workflow in a formalized manner. This ensures validation of each recipe by authorized personnel before released to production, or released as a component within a larger recipe.

Configure and enable Recipe Approval in the FactoryTalk Batch Equipment Editor, and executed in the FactoryTalk Batch Recipe Editor.

In addition to the Primary Approval process with up to six approval steps (three optional), a two-step Expedited Approval process is available.

Use the Expedited Approval process instead of the Primary Approval process, for example in the early stages of recipe design. A recipe can go through initial review using the expedited approval process. After the recipe is deemed ready, the Expedited approval process can be reverted to its starting point, and then the Primary approval process can be used to validate it for release as a step (procedure or operation) used in a larger recipe, or to release it to production.

Revert option

Both primary and expedited processes provide a Revert option to allow forward and reverse progress through the recipe approval process.

- Revert into the initial Not Started state. The Not Started state allows restart of the recipe approval process, obtain area model recipe approval process changes, or switch between the primary and expedited approval processes.
- Revert returns to the beginning of any incomplete approval step and/or to any previous approval step without restarting the entire approval process.
- The revert process operates the same for both the primary approval process and the expedited approval process.

Release Recipe to Production

With Recipe Approval enabled, Release Recipe to Production is a recipe approval step, and must be approved to allow the recipe to generate production batches. Similarly, with Recipe Approval enabled, Release Recipe as Step is a
Recipe approval step, and must be approved to allow the recipe to be incorporated into other recipes.

With Recipe Approval disabled, **Release Recipe to Production** is a recipe property and must be checked (set to true) to allow the recipe to generate production batches. The same is true for **Release Recipe as Step**—when this recipe property is set to true, the recipe can be incorporated into other recipes.

When Recipe Approval changes from disabled to enabled in the area model, a recipe property set to true becomes an approved step, with Recipe Editor using the $System approver.

**See also**

- [Configure recipe approval](#) on page 134
- [Approve a recipe](#) on page 135
- [Revert a recipe approval](#) on page 137
- [Automatic system signoff ($System)](#) on page 139

**Configure recipe approval**

This chart shows the sequence of tasks in the setup, configuration, and execution of recipe approval.

Configure the signature templates used in the Recipe Approval process within the FactoryTalk Batch Equipment Editor. Approval steps and their signoffs are also configured in the FactoryTalk Batch Equipment Editor. The process configuration becomes part of the specific Area Model used by the recipe elements. Manage the Approval process, primary or expedited, within the FactoryTalk Batch Recipe Editor.
See also

Recipe version control on page 146

Recipe approval process overview on page 133

Revert a recipe approval on page 137

Automatic system signoff ($System) on page 139

Approve a recipe

Use these instructions to approve a recipe.

Before you begin:

- In the FactoryTalk Batch Equipment Editor:
  - Enable and configure the Recipe Approval process.
  - Create the required signature templates in the appropriate area model.
  - Save the area model.

To approve a recipe:

1. Open the FactoryTalk Batch Recipe Editor. If prompted to translate the recipes, select Yes.
   - (optional) Select Verify All Recipes.
   - Select Accept when the verification has finished.
   - If Auditing is enabled, add any desired audit comments. Select OK to close the Audit Comments dialog box.
   - Select Close to close the Verification Process dialog box.

2. From the File menu, select Open Top Level.

3. From the Recipe Name list, double-click the recipe to open, or select the appropriate recipe and select Open. The selected recipe opens in the Procedure View pane.

4. From the Recipe menu, select either Primary Approval or Expedited Approval depending on the current approval state of the recipe. If an approval process has not started, either process is available.

5. Depending on which process chosen, the Primary Approval or Expedited Approval dialog box opens to the Approve tab, showing the next step available for approval (marked with a pencil icon). If all step approvals are completed, the dialog box opens to the Summary tab. Signature credentials are entered in the area below the signoff step list.
• In the **User** box, type the name of a user with the required permissions for the signoff (configured in the FactoryTalk Batch Equipment Editor under the **Edit** menu).

• Enter the correct password in the **Password** box.

• Enter a **Comment** (mandatory or optional, as configured in the area model).

• If a signoff is required to be last in a series of signoffs for an approval step, it is marked with a checkered flag 🟢 icon.

• If a FactoryTalk Security Permission is invalid, it is preceded by a warning icon ⚠ in the **Security Permission(s)** column. Hover over an invalid permission to show any valid permissions for that signoff.

6. **Select Approve.**

Continue with the approval process for all steps to approve. If a different user is required to sign off on an approval step, save and close the recipe after any approval step. The recipe can then be opened by the next authorized signer.

When a signature requires multiple signoffs, each signoff must be made using a unique UserID and password. Reuse of the same UserID and password results in this warning:

![Release Recipe to Production](image)

Be sure to provide a unique UserID and password (different from the UserID quoted in the warning message) and continue with the signoff.
Tip:  • With the completion of the first signoff on an approval step, the recipe becomes read-only. To make the recipe editable, revert all approvals back to the not started state.
  • Any time a recipe is copied using Save As, the Release Recipe as Step and Release Recipe to Production states are cleared in the copy.
  • Approval steps and their status are included when a recipe is exported, and retained during import.

See also

Revert a recipe approval on page 137

Revert a recipe approval

Any completed approval step can be reverted. When a step is reverted, all completed approval steps that follow it in the approval process are also reverted to their Not Started state. Reverting the first formal step has the effect of reverting an entire recipe approval back to its Not Started state. To revert an approved step:

To revert a recipe approval:

1. Open the desired recipe in the FactoryTalk Batch Recipe Editor.

2. From the Recipe menu, select either Primary Approval or Expedited Approval, determined by which process was used to approve the recipe.

   Tip: Use keyboard shortcuts for the approval process:
   • Primary approval process is Ctrl+J
   • Expedited approval process is Ctrl+E

3. Select the Revert tab. The Revert tab displays steps that are available to be reverted.
4. Select the step to revert by selecting on the signature line below the step name. Gray shading signifies that the step is selected.

5. Enter the signoff credentials to revert that step in the bottom area of the dialog box and select **Revert**. If a FactoryTalk Security Permission is invalid, it is indicated by a warning icon preceding it in the Security Permission(s) column. Hover over an invalid permission to show any valid permissions for that signoff.

6. If the signoff fails, select **Clear** to empty all values in this area, re-enter the credentials, and retry the operation.

7. (optional) Select the **Summary** tab. The Summary shows the selected approval step marked as Not Started (when a single signature is required) or Reverting (if more than one signature is required).

8. If more than one signature is required, the other signers must also sign off in the **Revert** tab. The Revert tab displays the next required signature with that line highlighted. The final required signature is marked with a checkered flag icon.

When all signatures are entered and the approval process step is reverted, make and save the necessary edits to the recipe. Navigate to the **Approval** tab of the desired approval process, and continue the approval process. Reverting back to the first step provides the option of switching between approval processes (primary to expedited, or vice versa).
Tip: Changing the recipe approval process from enabled to disabled in the area model has the effect of removing all signoffs from the current recipe set.

When Recipe Approval is disabled in the area model, and recipes in the current working set have existing signoffs, the Inconsistency between Area Model and Recipes dialog box opens.

- **Remove** opens the recipe set with all approval step signoffs removed.
- **Exit** closes the FactoryTalk Batch Recipe Editor and retains existing signoffs. Return to the FactoryTalk Batch Equipment Editor, re-enable recipe approvals, and continue with the recipe approval process in the FactoryTalk Batch Recipe Editor.

See also

- [Recipe approval process overview](#) on page 133
- [Automatic system signoff ($System)](#) on page 139

### Automatic system signoff ($System)

Following recipe translation from FactoryTalk Batch version 11.x to FactoryTalk Batch version 12, and with Recipe Approvals enabled, the FactoryTalk Batch Recipe Editor automatically creates the two Expedited Approvals process steps.

The FactoryTalk Batch Recipe Editor creates the **Release Recipe to Production** step when:

- The recipe was migrated from a previous FactoryTalk Batch version that did not support recipe approvals.
- Recipe Approvals are enabled in the area model.

If the **Release to Production** recipe property in the recipe was previously set to true, then the $System signoff is applied to approve the step.

Similarly, **Release Recipe as Step** is added (always with approval signoff $System) when:

- The recipe was migrated from a previous FactoryTalk Batch version that did not support recipe approvals.
• Recipe Approvals are enabled in the area model.

Check if the $System signoff has been applied to a recipe in the FactoryTalk Batch Recipe Editor. Open the Recipe > Expedited Process dialog box, then select the Summary tab. Look under the Approver column:

**Release Recipe as Step** and **Release Recipe to Production** are managed differently depending on whether Recipe Approvals are enabled or not:

*When Recipe Approvals are enabled, **Release Recipe as Step** is a recipe approval step and is approved as part of a formal or expedited approval process. When approved, the recipe can be used within other recipes.*

*With Recipe Approvals disabled, **Release Recipe as Step** is a recipe property, and set to true or false by the user in the Recipe > Header Data dialog box. When true, the recipe can be used within other recipes.*

*When Recipe Approvals are enabled, **Release Recipe to Production** is a recipe approval step and is approved as part of a formal or expedited approval process. When approved, the recipe is placed on the Recipe List in FactoryTalk Batch View, FactoryTalk eProcedure, or FactoryTalk Batch ActiveX controls, and the recipe can be used to generate production batches.*

*With Recipe Approvals disabled, **Release Recipe to Production** is a recipe property, and set to true or false by the user in the Recipe > Header Data dialog box. When true, the recipe is placed on the Recipe List in FactoryTalk Batch View, FactoryTalk eProcedure, or FactoryTalk Batch ActiveX controls, and the recipe can be used to generate production batches.*

**See also**

Recipe approval process overview on page 133
Recipe versioning overview

Recipe Versioning enhances management of recipes. A versioned recipe is a saved, read-only snapshot of the recipe taken at a particular point in time. Recipe versioning is useful when an author, or a team of authors, need to store and protect unique versions of the recipe at chosen development milestones.

Recipe versioning starts with a new or existing recipe before it has had a version of it created. When the recipe has reached a development state the recipe author wants to protect, the Check In command creates the first instance of a versioned recipe. At that point, the recipe version is saved, set to read-only, and can no longer be edited.

Using the Check Out command, an editable work-in-progress (WIP) copy of a versioned recipe can be created. Typically, a WIP recipe undergoes further development before being checked in as the next version of the recipe. Subsequently, a new WIP copy can be made and used in the next iteration of recipe development.

If a versioned recipe fails verification, it is marked by the system as obsoleted. Verification can fail due to modifications to the recipe's underlying area model, or if the recipe references other missing or obsoleted recipes. An obsoleted recipe cannot be revised in order to pass verification. A WIP copy of an obsoleted version can be created and then modified to pass verification.

See also

- How recipe versions are named on page 142
- Restrictions for recipe versions on page 143
- Enable recipe versioning option on page 144
- Recipe version control on page 146

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.
Recipe versioning overview

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

See also

Recipe versioning overview on page 141

How recipe versions are named

Recipe versions are maintained through a naming convention enforced by the FactoryTalk Batch Recipe Editor.

Recipe version naming has three components:

- **Basename** is the unique name of a recipe. The recipe author gives the recipe this name. The basename is identifies all related versions and work-in-progress (WIP) copies.

- **Version name** is the name assigned by the FactoryTalk Batch Recipe Editor to a versioned recipe created from a new, existing, or WIP recipe. The versioned recipe's name includes the basename and an appended version number (for example ~V1).

- **WIP name** is the name assigned by the FactoryTalk Batch Recipe Editor to a WIP copy of a recipe. The name includes the basename, the version number of the copied versioned recipe, and an appended _WIP, indicating it is a work-in-progress recipe. The FactoryTalk Batch Recipe Editor increments the version number for a WIP copy to the next unassigned version number.

This illustration includes two examples of how recipe version names are derived. In both examples, the recipe basename is *Make_Sauce*. The first version saved is *Make_Sauce~V1*, and the first work-in-progress copy of that version is *Make_Sauce~V2_WIP*. 


Tip: When a WIP copy is made using the Check Out command, FactoryTalk Batch Recipe Editor looks ahead to when the WIP is saved as a version. For example, the WIP copy of Make_Sauce~V1 is saved as the second version Make_Sauce~V2.

The example on the left shows a straightforward, linear progression from V1 to V4 of the Make_Sauce recipe. The right shows the WIP numbering method, where earlier versions are used as the basis for creating the next version. FactoryTalk Batch Recipe Editor manages version naming to avoid duplicates of version numbers and WIP copy names.

See also

Recipe versioning overview on page 141

Restrictions for recipe versions on page 143
Restrictions for recipe versions

When using recipe versioning, observe these restrictions:

- A versioned recipe can only reference other versioned recipes or steps.
- Once a recipe version, or work-in-progress recipe (WIP), is created, its basename cannot be changed afterwards, even if recipe version control is disabled.
- A new recipe’s basename must be unique; it cannot be the same as an existing recipe basename or existing recipe name (in the current working set).
- A recipe's basename cannot contain any of the naming conventions used by the system, namely ~V or _WIP. These strings are reserved for use by the FactoryTalk Batch Recipe Editor.
- Prior to creating a versioned recipe, the recipe is automatically verified by the FactoryTalk Batch Recipe Editor; it must pass this verification without any errors.
- If a recipe makes reference to missing or obsoleted recipes, it will not pass verification, and is marked by the system as obsoleted.
- Import of versioned recipes can result in conflicts, both with version numbering and version naming.

Tip: Recipe versioning and the recipe approval process are independent of each other. Recipe steps can be approved or reverted without being considered as edits to a versioned recipe.

Tip: Recipe version control works with all recipe storage formats (BINARY, XML, and RDB).

See also

Import conflicts on page 185.
Enable recipe versioning option

Configure Recipe Versioning within the FactoryTalk Batch Equipment Editor. Area Model authors must have FactoryTalk Full Edit access to FactoryTalk Batch Equipment Editor to enable and disable versioning.

The FactoryTalk Batch Recipe Editor handles enforcement of versioning naming. This flow chart shows the sequence of tasks to enable and use recipe versioning.

![Flow chart showing the sequence of tasks to enable and use recipe versioning.]

See also

- Disable recipe versioning option on page 145

Disable recipe versioning option

When recipe versioning is disabled, previously versioned recipes remain read-only and have all header boxes disabled. It is still prohibited to name a new recipe in conflict with any existing basename, in order to avoid name collisions if recipe versioning is re-enabled.

Rename a previously versioned recipe by using the Save As command to create a new recipe. Edit and save a previous work-in-progress (WIP) recipe--use Save As and provide a new recipe name to disassociate the recipe from its former basename.

See also

- Copy recipe options on page 75
Prior to starting this procedure, ensure that recipe versioning is enabled in the **Server Options** dialog box in the FactoryTalk Batch Equipment Editor.

**To create a recipe version:**

1. Create a new recipe using any of these options:
   - Use the **File** menu **New** command.
   - Use the **Save As** command.
   - Create a new recipe as part of adding a step to an open recipe.
   Be sure to specify a unique basename when creating the new recipe. Observe naming restrictions.

2. Edit the recipe, and then **Save** the recipe.

3. Use the **Check In** command to create the first version.
   - To edit this version of the recipe, use the **Check Out** command to create an editable copy and make the necessary changes.
   - When finished editing, use the **Check In** command again to create the next version of the recipe.

4. Use the **Check Out** command to create an editable copy of this version.

This process can be repeated as many times as required (the only restriction is that the length of the recipe name cannot exceed 50 valid characters).

**See also**

[Enable recipe versioning option](#) on page 144

[Restrictions for recipe versions](#) on page 143

**Recipe version control**

With recipe version control **enabled**, these menu items (and their corresponding toolbar icons) are available:

- **Check In** (under the **File** menu) - Creates a read-only recipe version from a new or Checked Out work-in-progress (WIP) recipe. Enabled if a new or Checked Out recipe is open in the FactoryTalk Batch Recipe Editor. A Checked In recipe cannot be edited, but recipe step approvals can be signed off.

- **Check Out** (under the **File** menu) - Creates an editable WIP recipe from a recipe version. Enabled if a versioned or obsolete recipe is open in the FactoryTalk Batch Recipe Editor.
- **Version History** (under the Recipe menu) - Opens a dialog box displaying information about the Previous recipe version and the current recipe. Enabled when any recipe is open in the FactoryTalk Batch Recipe Editor.

With recipe version control enabled, the File menu commands in the FactoryTalk Batch Recipe Editor operate as follows, depending on the type of recipe (unversioned, WIP, or versioned) that is open:

<table>
<thead>
<tr>
<th>Command</th>
<th>Unversioned recipe or new recipe</th>
<th>Checked Out (WIP) recipe</th>
<th>Checked In (versioned) recipe</th>
</tr>
</thead>
</table>
| New       | • Creates a new recipe.  
           | • Prompts to provide a unique name for the recipe. 
           | Observe the naming restrictions. | Not allowed. |
| Save      | Saves changes in the open file to the working set. | Saves the changes in the WIP file to the working set. | • Versioned recipes cannot be edited—they cannot be saved.  
           | | | • If Recipe Approvals are enabled, the recipe with its approval state is saved automatically with every approval or revert signoff. |
| Save As   | • Saves the current recipe as a new recipe with a new name.  
           | • The original recipe is closed and unchanged.  
           | • Any approval steps are removed. | • Saves the current recipe as a new recipe with a unique, user-specified basename. Observe the naming restrictions.  
           | | | • The original recipe is closed and unchanged.  
           | | | • Version history information is removed.  
           | | | • Any approval steps are removed. |
| Modify    | • If Recipe Approvals are enabled and an approval process is not active:  
           | | | • Any user with editing rights is allowed to edit all aspects of the recipe.  
           | | | • See Rename command below. | • Versioned recipes cannot be edited.  
           | | | • If Recipe Approvals are enabled, approval steps can be signed and reverted. |
| Export    | • Save an exact copy of a specified recipe to the specified working set of recipes.  
           | • User is prompted before overwriting an existing versioned file.  
           | • Existing recipe approval steps are maintained. |
## Chapter 12  Recipe versioning overview

<table>
<thead>
<tr>
<th>Command</th>
<th>Unversioned recipe or new recipe</th>
<th>Checked Out (WIP) recipe</th>
<th>Checked In (versioned) recipe</th>
</tr>
</thead>
</table>
| Import                 | • Import an exact copy of a recipe from another working set of recipes into the current working set of recipes.  
• Existing recipe approval steps are maintained. | • If a version conflict exists, choose which version to import.  
• If a recipe basename conflict exists with one or more recipes in the working directory, import the recipe that causes the conflict.  
• Existing recipe approval steps are maintained. | • Confirm to overwrite an existing versioned file.  
• If no name conflict exists, this imports an exact copy of a recipe from another working set of recipes into the current working set of recipes.  
• If a version conflict exists, prompts to resolve the conflict.  
• If a recipe basename conflict exists with one or more recipes in the working directory, the recipe that causes the conflict cannot be imported.  
• Existing recipe approval steps are maintained. |
| Remove Recipe          | • Deletes the recipe from the working set (any referencing recipe will then have a missing element). |                        |                               |
| Rebuild Recipe Directory | • If a recipe is added or deleted from the recipe directory without using the FactoryTalk Batch Recipe Editor, this command rebuilds the recipe directory and provides an option to verify all recipes within it. | • If a recipe basename or version numbering conflict exists with two recipes in the recipe directory, choose which recipe to retain in the directory. |                               |
| Rename (Change Header data & Save) | • Changes the name of the file and automatically propagates to all referencing files.  
• Updates all the transition expressions and parameters with the new name.  
• Name cannot change if an approval process is active.  
• The recipe remains a New Recipe. | Not allowed. | |
| Print                  | Print the recipe.                 |                          |                               |
| Verify                 | Standard verification process.     |                          | Additional verification rules apply. If the recipe refers to a missing or obsolete unit procedure or operation, the recipe is marked as obsolete. |
Recipe versioning overview

<table>
<thead>
<tr>
<th>Command</th>
<th>Unversioned recipe or new recipe</th>
<th>Checked Out (WIP) recipe</th>
<th>Checked In (versioned) recipe</th>
</tr>
</thead>
</table>
| Check In        | • If the recipe verifies, create a new file (basename~V1)  
• Replace all references to the basename with basename~V1.  
• Recipe basename is deleted.  
• Existing recipe approval steps are maintained.  | • If the recipe verifies, create a new file (basename~Vn)  
• Replace all references to the WIP recipe with basename~Vn.  
• WIP recipe is deleted.  
• Existing recipe approval steps are maintained.  | Not Applicable. |
| Check Out       | Not Applicable.                  |                          | • Creates the next WIP recipe for the current recipe (basename~Vn_WIP).  
• Existing recipe approval steps are removed. |

See also

- Restrictions for recipe versions on page 143
- Recipe versioning overview on page 141
- Import conflicts on page 185

Recipe version history

Select Version History to view recipe version information.

The version history consists of:

- Recipe name
- Version description
- Version creation date
- Version verification date
- Area model date (when last saved)
- Area model name against which the version was verified

All boxes are read-only with the exception of the Version Description box under Current Recipe. This box can be edited if:

- The recipe itself is editable (editable recipes are both unversioned and work-in-progress (WIP) recipes that have not started a recipe approval process).
The FactoryTalk user has full access rights.

Obsoleted recipe versions

During verification, the area model configuration stored in the recipe is compared against the current area model. If the recipe configuration and area model do not match, the recipe fails verification and is subsequently marked by FactoryTalk Batch as Obsoleted.

Obsoleted recipes:

- Are still versions of the basename recipe.
- Can be imported and exported, and removed from the working recipe set (using the Remove Recipe command).
- Cannot be opened, modified, renamed, checked in, printed, checked in, or saved.
- Do not allow a containing recipe to verify—an obsoleted recipe is treated as a missing recipe element. See Verification issues with a versioned recipe in the section on Troubleshooting for more information.
- Any step in a visible recipe that references an obsoleted recipe is outlined in red.

To work with an obsoleted recipe:

- Use the Save As command to create a new recipe. The new recipe is not marked as obsoleted.
- Use the Check Out command to create a new, editable, work-in-progress (WIP) recipe. The FactoryTalk Batch Recipe Editor will make all required area model configuration changes to the WIP as it is created, indicating the modifications in the verification progress dialog box. The original obsoleted recipe remains unchanged.

Once the WIP recipe is checked back in:
Recipe versioning overview

Chapter 12

- It is given the latest version number.
- The original obsoleted recipe remains in the working directory.

Alternatively, to reinstate an obsoleted recipe (clear its obsoleted status), modify the area model in the FactoryTalk Batch Equipment Editor to match the recipe configuration. When the recipe next undergoes verification by the FactoryTalk Batch Recipe Editor, it is reinstated as a current versioned recipe.

**Tip:** If a Recipe Approval Process is associated with a recipe when it is marked as obsoleted, its approval properties remain in effect and can be approved or reverted as needed.

**See also**

[Verification results for a versioned recipe](on page 198)

Use these examples to help determine whether to enable recipe versioning or recipe approvals:

Recipe versioning can be used in conjunction with the Recipe Approval Process—playing a complementary role in recipe development.

**Tip:** When using recipe versioning and approvals together, consider applying expedited approvals to a checked-out (WIP) recipe, and the formal approval process to a checked-in (Versioned) recipe.

This section details the four scenarios when enabling or disabling Recipe Version Control and the Recipe Approvals Process:

<table>
<thead>
<tr>
<th>Recipe Approvals Process</th>
<th>Recipe Versioning Disabled</th>
<th>Recipe Versioning Enabled</th>
</tr>
</thead>
<tbody>
<tr>
<td>Process Disabled</td>
<td>Scenario 1</td>
<td>Scenario 2</td>
</tr>
<tr>
<td>Process Enabled</td>
<td>Scenario 3</td>
<td>Scenario 4</td>
</tr>
</tbody>
</table>

- **Scenario 1:** Recipe Versioning and Recipe Approvals are **disabled**. Commands in the FactoryTalk Batch Recipe Editor operate in their normal fashion. No enhanced version control or formal approval process is involved in authoring and readying a recipe for production use. The **Release Recipe as Step** and **Release Recipe to Production** properties function in this way:
  - To use a recipe or operation in another recipe, select the **Release Recipe as Step** check box in the **Recipe Header Data** dialog box.
  - To add a recipe to the Recipe List in FactoryTalk Batch View, FactoryTalk eProcedure, or FactoryTalk Batch ActiveX controls, select...
the Release Recipe to Production check box in the Recipe Header Data dialog box.

- **Scenario 2:** Recipe Versioning is disabled and Recipe Approvals are enabled. The approval process governs if the Release Recipe as Step and Release Recipe to Production steps are approved or not.

- **Scenario 3:** Recipe Versioning is enabled and Recipe Approvals are disabled: These additional commands are available in the FactoryTalk Batch Recipe Editor (enabled or disabled depending on what versioning state the current open recipe is in): Check In, Check Out, and Parent information.
  - When a version of a recipe is created using Check In, the Release Recipe as Step and Release Recipe to Production properties are carried over unchanged.
  - When a work-in-progress (WIP) recipe copy is created using Check Out, the Release Recipe as Step and Release Recipe to Production properties are always cleared (set to false).
  - When a new recipe is created using Save As, the Release Recipe as Step and Release Recipe to Production properties are always cleared (set to false). The original recipe is unchanged.
  - To use a versioned or WIP recipe in another recipe, select the Release Recipe as Step check box in the Recipe Header Data dialog box.
  - To add a versioned or WIP recipe to the Recipe List in FactoryTalk Batch View, FactoryTalk eProcedure, or the FactoryTalk Batch ActiveX controls, select the Release Recipe to Production check box in the Recipe Header Data dialog box. All sub-recipes and procedures within a recipe must also have their Release Recipe to Production check boxes selected.

- **Scenario 4:** Recipe Versioning and Recipe Approvals are enabled: Versioning commands in the FactoryTalk Batch Recipe Editor are available (some may be disabled depending on the state of the open recipe), as are menu commands for Approvals Process and Expedited Approval.

  **Tip:** Changing the Release Recipe as Step and the Release Recipe to Production approval states is not considered an edit or change to a recipe itself.

  - A new recipe (created using Save As or New) is set to Approval process not started and can go through either the formal approval process or expedited process.
  - A newly-created version or WIP recipe keeps any completed or in-progress approval steps from the original recipe.
  - When a version of a recipe is created, the Release Recipe as Step and Release Recipe to Production properties (Recipe Approvals disabled)
or approval steps (Recipe Approvals enabled) are carried over unchanged.

- Approve and revert steps in a versioned recipe--there is no need to create a WIP recipe to do this.

See also

Recipe versioning overview on page 141
Recipe approval process overview on page 133
Chapter 13

Security authority overview

FactoryTalk Batch Security Authority, when enabled, protects intellectual property (as contained in recipes) and ensures it is only used within its intended scope. Security Authority helps secure recipes from unauthorized copying, editing, import, export, and use. Binary is the only recipe format that is allowed for the Security Authority setting.

The FactoryTalk Batch Recipe Editor, when directed by an authenticated user to secure a recipe, associates the Security Authority Identifier (SAI) stored in the FactoryTalk Network Directory with the recipe. When the FactoryTalk Batch Recipe Editor is later directed to open the secured recipe, the SAI in the recipe is compared to the current SAI in the current FactoryTalk Network Directory. If they match, the FactoryTalk Batch Recipe Editor opens the recipe. If they do not match, for example when running under another instance of FactoryTalk or if the SAI has been regenerated, the editing session does not have the authority to open the recipe and stops the process.

Important: If the SAI in a FactoryTalk Network Directory is changed or lost, access to any recipe that is bound to it may be lost. Rockwell Automation recommends that, before securing a recipe, back up the FactoryTalk Network Directory, and store unsecured versions of the recipe files in binary (.OUP, .UPC or .BPC), XML (.oxml, .uxml, or .bxml), or RDB formats in a secure location.

See also

Security authority configuration on page 155

Secure recipe on page 157
Security authority configuration

This flow chart shows the sequence of tasks in the configuration and use of security authority.

Authority to secure area models and recipes is assigned in the FactoryTalk Administration Console. Security Authority is disabled by default.

Area models are secured within FactoryTalk Batch Equipment Editor using the Security Authority command. The FactoryTalk Network Directory Security Authority Identifier (SAI) is written into the area model schema. To subsequently open and edit the area model, the SAI in the area model must match that of the current FactoryTalk Network Directory. No match prevents the opening and editing of an area model and its associated recipes in FactoryTalk Batch Recipe Editor.

Recipes are secured within FactoryTalk Batch Recipe Editor using the Security Authority command. The FactoryTalk Network Directory SAI is written into the recipe header data. To subsequently open and edit the recipe in FactoryTalk Batch Recipe Editor, the SAI in the recipe must match the one in the current FactoryTalk Network Directory.

Import and export operations with secured recipes are restricted.

See also

Import and export restrictions for secured recipes on page 188

Security authority overview on page 155
Secure recipe

Recipes are secured using the Security Authority command. Use these instructions to secure the recipe.

To secure a recipe:

1. Open the recipe to secure in the FactoryTalk Batch Recipe Editor.

   **Important:** Before securing a recipe to a specific FactoryTalk Network Directory, Rockwell Automation recommends backing up the FactoryTalk Network Directory and store unsecured versions of the recipe files in binary (.OUP, .UPC or .BPC), XML (.oxml, .uxml, or .bxml), or RDB formats in a secure location.

   For backup details, see FactoryTalk Help: Select Start, point to All Programs > Rockwell Software > FactoryTalk Tools, and then select FactoryTalk Help.

   Once a recipe is secured to a particular FactoryTalk Network Directory, it cannot be opened if the security authority identifier associated with that directory no longer exists.

2. Select Recipe > Security Authority. The Security Authority dialog box opens.

3. Select the check box to secure the recipe. If the check box is not enabled, the user account is not authorized to use this feature.

4. Select Save.

See also

Security authority configuration on page 155
Chapter 14

Complete and maintain recipe

Once the recipe is saved, verify the recipe and release to production for batch operator access.

See also

Verify recipe on page 159

SFC validation on page 161

Release Recipe as Step option on page 170

Release a recipe to production option on page 170

Rebuild the recipe directory on page 170

Verify recipe

Verify recipe is the process that checks the recipe for completion and accuracy. A recipe must be verified to ensure that all connections and references are properly made. The verification process examines the selected recipe and all lower-level recipes. If there are errors in the recipes being verified, messages explaining the nature of the error are displayed.

Tip: To verify all recipes, from the File menu select Verify All Recipes or Verify All Recipes and Validate All SFCs.

To verify a recipe:

1. Open the recipe to verify.

2. Select Verify or select Verify Recipes from the Recipe menu. The FactoryTalk Batch Recipe Editor reads the current recipe and displays a list of the recipe’s outstanding problems, if any, and saves the recipe in the process.

Important: FactoryTalk users with ViewOnly permissions to the FactoryTalk Batch Recipe Editor cannot open the FactoryTalk Batch Recipe Editor to verify recipes. The FactoryTalk Batch Recipe Editor will close.

3. If there are errors, double-click an error message to open the recipe to the recipe level where the error was detected.
Tip: Copy the error text and paste it to a text document or spreadsheet if desired. Hold down Ctrl and select on the errors to copy. Use the Ctrl+C command to copy to the Clipboard.

4. Fix the problem and run the verification again. Continue this process until the recipe verifies successfully.

5. When the recipes are successfully verified, the SFC structure can be validated. From the File menu select Verify All Recipes and Validate All SFCs.

See also

SFC validation on page 161

What gets verified?

When a recipe is opened in FactoryTalk Batch Recipe Editor, FactoryTalk Batch Recipe Editor checks that all the phases, operation sequences, parameters, and reports in the recipe also exist in the area model. If any parameters or reports have changed in the area model since the recipe was originally built, FactoryTalk Batch Recipe Editor automatically adds or removes parameters or reports from the recipe accordingly. Material references are also checked when the recipe is opened. If the material is not present in the material database, the material parameter is reset to NULL_MATERIAL.

When a recipe is verified, these items are also checked:

- **Basic Structure** is verified to ensure there is an initial step, a terminal step, other steps, and that they are all linked together.

- **Unit Requirements** are verified against the area model. For example, does the Unit Class or the Unit Instance exist in the area model? Is there a flow path in the area model that satisfies the Procedure Unit Requirement specified flow path?

- **Deferred Parameters** are within the min/max and limits of the parameters that are deferred to them.

- **Enumeration** set names and elements found in the area model.

- **Phase links** referential integrity (have partners), along with the message partner information found in the area model.

- **Phase class names** found in the area model.

- **Header Identifier, version, timestamp and author** are not empty.

- **Recipe Approval Steps** that are incomplete use valid signoffs (signature templates are assigned to valid FactoryTalk users and/or groups).

- **Release Recipe as Step property** is enabled on recipes.
- Recipe Versions are compatible with other parent or offspring recipes with the same basename. The verification results include information on Recipes to be Obsoleted and Obsoleted Recipes to be Repaired.

- All step recipe paths are valid (only if Verify All Recipes and Validate All SFCs is selected). For Procedures, this means that each referenced Unit Procedure recipe exists. For Unit Procedures, it means that each referenced Operation exists. For Operations, it means the Phase is in the area model.

- Referenced recipes existence (through the step name).

  Important: If a working recipe is changed and fails verification, the Release Recipe as Step and the Release Recipe to Production properties in the recipe header are preserved. Due to verification errors, the recipe may be unavailable for use in production. Fix all verification errors to make the recipe available for use.

See also

Verify recipe on page 159

SFC validation

The verification feature ensures that basic SFC legality rules, such as step-transition-step sequencing, are enforced, but the recipe verification check does not detect logic errors within the SFC structure. Beginning with FactoryTalk Batch v. 10, the FactoryTalk Batch Recipe Editor provides SFC validation which is an error check that looks for logic errors in the SFC structure defined within a recipe.

For example, this SFC is structurally invalid:

While the above SFC structure would pass the current verification checks done by the FactoryTalk Batch Recipe Editor, this structure would never run to completion for this reason: The final transition condition following the AND Convergence cannot execute unless all prior steps above it are concurrently active. The OR Divergence ensures that only one of the steps prior to the AND Convergence can be active at one time. While the error in the SFC shown above is
relatively easy to identify, this type of SFC flaw and others can be much more difficult to identify in more complex recipe structures.

The SFC Validation tool can detect a variety of programming errors that result in invalid or illegal SFC programs. SFC structures identified as invalid by the tool results in recipe verification warnings. Note that the **Release Recipe to Production** approval or check box is not cleared by the verification checks.

See also

- [FactoryTalk Batch Recipe Editor toolbar on page 15](#)
- [Run SFC validation on page 162](#)
- [Set allowable SFC permutations on page 163](#)
- [SFC validation error types on page 163](#)

### Run SFC validation

While SFC validation is optional, it is good practice to get in the habit of validating the SFC structures.

**Tip:** To validate all recipes, from the File menu, select **Verify all Recipes and Validate all SFCs.**

**To run SFC validation:**

1. Open the recipe containing the SFC to validate.
2. From the Recipe menu, select **Verify all Recipes and Validate all SFCs,** or select **Verify and Validate.**

   The FactoryTalk Batch Recipe Editor reads the current recipe structure, determines its validity, displays a list of the recipe’s outstanding problems, and saves the recipe.

   Errors display in the **Recipe Verification** dialog box.

3. If there are errors, double-click an error message to open the recipe containing the error.
4. Fix the problem and run the validation again. Continue this process until the recipe validates successfully.

See also

- [Set allowable SFC permutations on page 163](#)
- [SFC validation error types on page 163](#)
Set allowable SFC permutations

The SFC Validation tool deals with OR Divergences by examining each possible leg of the divergence as if it were a separate SFC. This leads to a combinatorial number of possible SFCs when encountering multiple divergences within an SFC. For example, if an SFC contained three OR Divergences, one with three legs, one with two legs, and the third with four legs, the number of SFC permutations that would need to be examined would be 24 (3x2x4 = 24).

If configured a recipe with an extremely large number of OR Divergences, validation may take several minutes to complete. For this reason, define how many OR divergences the validation checks.

Tip: If the recipe contains more OR divergences than the upper limit allows, the recipe is considered too complex and validation stops. Configure the upper limit in the FactoryTalk Batch Recipe Editor Options dialog.

To set allowable SFC permutations:

1. To set the maximum number of SFC permutations, select Options from the Recipe menu.

2. Change the value in the SFC Validation: Max Number of OR Permutations box to the desired number.

   The default setting is 65535. The allowable range for the configuration parameter is from 1024 to 2,097,152, inclusive.

3. Select OK to save the changes.

See also

SFC validation error types on page 163
SFC validation error types

Five basic errors can occur within an SFC structure. These examples show how to create and detect these errors.

The SFC Validation function identifies the linear segment of an error. Note that this may not be the location of the actual structural issue. The error message generated by SFC Validation when it detects an SFC structure error may include a linear segment specifier that provides the name of the first step in the linear segment of the error. If the first step in the linear segment is an Initial Step or Terminal Step of the SFC, then the step name, as currently used by the FactoryTalk Batch Server, is INITIALSTEP:1 or TERMINALSTEP:1.

Example:

WARNING: INITIALSTEP:1 >> Parallel Activation of Linear Segment

If there are no steps present in the linear segment, the string STEP NOT PRESENT IN LINEAR SEGMENT is used.

Example:

WARNING: STEP NOT PRESENT IN LINEAR SEGMENT >> Parallel Activation of Linear Segment

See also

Parallel activation of a linear segment on page 164
Unreachable terminal step on page 165
Parallelism with terminal step on page 166
Linear segment cannot reach terminal step on page 167
Unreachable linear segment on page 169
Parallel activation of a linear segment

The loopback from the right leg of the OR Divergence back into the Linear Segment prior to the AND Convergence is the source of the error. Two execution tokens exists in the same linear segment, which is the segment containing the step marked with an X:

The actual error message text is similar to: **WARNING: SFC Validation Tool detects dual activation of Linear Segment beginning with step:%1 (where %1 is the actual segment identifier).** The reported error location may not be close to the SFC structural issue. Refer to the product documentation for examples of SFC structures that can report this error. Other errors may or may not be present.

See also

- Unreachable terminal step on page 165
- Parallelism with terminal step on page 166
- Linear segment cannot reach terminal step on page 167
- Unreachable linear segment on page 169
**Unreachable terminal step**  

SFC Validation can detect when an SFC structure makes it impossible for a recipe execution to reach the SFC terminal step.

In this SFC, the OR Divergence activates only one of the two steps below it. This means that the transition below the AND Convergence never activates, since it is not permitted to fire unless all prior steps are active. This error makes it impossible for an execution token to reach the terminal step of the SFC.

![Diagram of SFC structure](image)

The actual error message text is similar to: **WARNING: SFC Validation Tool detects unreachable Terminal Step.** Refer to the product documentation for examples of SFC structures that report this error. Other errors may or may not be present.

**See also**

- [SFC validation error types](#) on page 163
- [Parallelism with terminal step](#) on page 166
- [Linear segment cannot reach terminal step](#) on page 167
- [Unreachable linear segment](#) on page 169
Parallelism with terminal step

SFC Validation can detect when an SFC structure makes it possible to reach the terminal step of the SFC while other linear segments within the structure still have active execution tokens.

For example, in the SFC below, the AND Divergence activates both steps below it. Then, whichever leg of the parallelism completes first has its execution token reach the terminal step. This results in an execution token reaching the terminal step of the SFC while another execution token is active within the structure. This is defined as an illegal behavior — when an execution token reaches the terminal step of the SFC, it should be the only execution token present in the SFC structure.

![Diagram of SFC structure](image)

The error message text is similar to: **WARNING: SFC Validation Tool detects parallelism with Terminal Step of Linear Segment beginning with step:%1 (where %1 is the segment identifier).** The reported error location may not be close to the SFC structural issue. Refer to the product documentation for examples of SFC structures that can report this error. Other errors may or may not be present.

See also

- [Linear segment cannot reach terminal step](#) on page 167
- [Unreachable linear segment](#) on page 169
**Linear segment cannot reach terminal step**

SFC Validation Tool detects when an SFC structure makes it impossible for an active execution token contained within a linear segment to reach the SFC terminal step. For example, in the SFC example below, if the active token goes down the right branch of the first OR Divergence, then the execution token enters an infinite loop through which it can never reach the terminal step of the SFC.

![SFC Diagram](image)

A second type of SFC structure can also generate this error. In the SFC structure below, if the recipe execution goes through the right-most transition, indicated by the arrow, then the execution token reaches a dead end in the second step of the right-most leg of the AND Divergence and recipe execution is unable to reach the terminal step.

![SFC Diagram](image)

In this SFC example, the linear segment reported by the Validation Tool as the source of the error actually contains no steps.
The actual error message is similar to: **WARNING: SFC Validation Tool detects Linear Segment beginning with step:%1 (where %1 is the segment identifier) has no path to Terminal Step.** Refer to the product documentation for examples of SFC structures that can report this error. Other errors may or may not be present.

See also

[SFC validation error types on page 163](#)

**Unreachable linear segment**

SFC Validation also detects when an SFC structure makes it impossible for an active execution token to reach one or more linear segments within the SFC. For example, in this SFC, the linear segment under the AND Convergence is unreachable because it is impossible for both prior steps to have active execution tokens simultaneously due to the initial OR divergence. This means that if recipe execution takes the left branch of the first OR Divergence, or the left branch of the second OR Divergence, then recipe execution becomes hung and unable to proceed.

The error message is similar to: **WARNING: SFC Validation Tool detects unreachable Linear Segment beginning with step:%1 (where %1 is the segment identifier).** Refer to the product documentation for examples of SFC structures that report this error. Other errors may or may not be present.

See also

[SFC validation error types on page 163](#)
Release Recipe as Step option

To use a recipe within another recipe, set the **Release Recipe as Step** property to true. When no Recipe Approval process is in use, select the **Release Recipe as Step** check box in the **Recipe Header** dialog box. When enabled Recipe Approval process, set this property by completing the signoff for Release Recipe as Step during the recipe approval process.

**Tip:** A recipe is valid for release to production only when all steps and procedures within it are approved as **Release Recipe as Step**.

See also

[Release a recipe to production option on page 170](#)

Release a recipe to production option

Only recipes created in the FactoryTalk Batch Recipe Editor have access to FactoryTalk Batch View, FactoryTalk mProcedure, FactoryTalk Batch ActiveX controls, or other client applications. If a recipe is to appear as a choice in a client application **Recipe List**, release the recipe to production. When no Recipe Approval process is in use, a recipe is released to production by checking the **Release Recipe to Production** check box in the **Recipe Header** dialog box. When a Recipe Approval process is enabled, the release to production property is set by completing all signoffs in the recipe approval process.

A recipe is valid for release to production only when all procedures within it are respectively approved as **Release Recipe as Step**.

See also

[Release Recipe as Step option on page 170](#)

Rebuild the recipe directory

The working set of recipes is stored in the location specified on the **Project Settings** tab of the FactoryTalk Batch Equipment Editor **Server Options** dialog box. If a recipe has been added to or deleted from this recipe storage location without using the FactoryTalk Batch Recipe Editor, rebuild the recipe directory.

To rebuild the recipe directory:

1. There are two options:
• If the FactoryTalk Batch Recipe Editor is closed, open it to rebuild the recipe directory. Any recipe file additions or deletions are shown when using the Open, Remove, Import, Export, or Generate Reports menus in the FactoryTalk Batch Recipe Editor.

• If the FactoryTalk Batch Recipe Editor is already open, select File > Rebuild Recipe Directory. The FactoryTalk Batch Recipe Editor reads all of the currently stored recipes and updates the recipe directory file.

2. If a recipe version conflict occurs when rebuilding the directory, the Resolve Version Conflict for Rebuild Recipe Directory Request dialog box opens. A prompt opens to retain one of the two recipes that conflict.

Tip: If there are unverified recipes in the recipe directory, the option to verify all recipes or verify and validate all recipes displays.

See also

Import conflicts on page 185

SFC validation on page 161

Recipe maintenance

Recipe maintenance includes find recipe references, page setup, generating reports, remove recipe, and translate recipes.

See also

Find Recipe References overview on page 171

Page setup on page 173

Generate reports option on page 174

Remove a recipe on page 177

Translation on page 178

Find Recipe References overview

Use Find Recipe References to locate recipes that contain a specified operation or unit procedure. The reference recipe is the operation or unit procedure referenced in container recipes searched. A container recipe contains one or more steps assigned to a specified operation-level recipe or unit procedure-level recipe retrieved. A step reference is a step within a container recipe configured to reference an operation or unit procedure.

For example, an ice cream factory wants to use a new SWEETMILK_ORGANIC operation recipe in place of their SWEETMILK operation recipe. The recipe author uses Find Recipe References to search for all recipes containing steps with
SWEETMILK (reference recipe) and manually updates them with SWEETMILK_ORGANIC.

The reference list of recipes that contain steps configured with the specified operation or unit procedure displays in a table. Every entry in the resulting list is the full recipe path and corresponding recipe levels to that step. When a container recipe opens, the highest level recipe opens. In this example, CLS_SWEETMILK_UP / CLS_SWEETMILK_OP:1 is selected. The recipe that opens is CLS_SWEETMILK_UP. For an explanation of recipe levels, see the FactoryTalk Batch Recipe Editor introduction.

See also

FactoryTalk Batch Recipe Editor introduction on page 9

Find recipe references on page 172

Find recipe references

Search recipes for a reference to an operation or unit procedure. Recipes can be stored as a binary file, XML file, or SQL Server database.

Before you begin:

- Obtain at least view-only security privileges to view the results.
- Close all recipes.
- Verify all recipes (full edit privileges required).

To search for recipe references:

2. (optional) Set the Recipe Filter to narrow the list of displayed recipes.
3. Select one operation or unit procedure to find. The selected recipe is the reference recipe.
4. Select Find. The Recipe References dialog box opens. A reference list of recipes having steps matching the reference recipe displays. The recipes are container recipes and the steps are step references. Total indicates the number of step references listed.
Tip: The search includes checked-in recipes, checked-out recipes, recipes that are not versioned (versioning is disabled), obsolete recipes, and recipes having invalid Security Authority Identifiers.

5. (optional) Select Copy All to copy the entire list to the clipboard and paste in a document for future reference.

6. Select one step reference and then select Open to view and modify a container recipe in the list. Edit privileges and recipe status determines if the recipe can be modified. This table describes possible outcomes (assuming edit privileges):

<table>
<thead>
<tr>
<th>Container Recipe Status</th>
<th>Open Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>Invalid Security Authority</td>
<td>Container recipe cannot be opened.</td>
</tr>
<tr>
<td>Checked-in</td>
<td>Container recipe cannot be edited.</td>
</tr>
<tr>
<td>Checked-out</td>
<td>Container recipe can be edited.</td>
</tr>
<tr>
<td>Obsolete</td>
<td>Prompts to update the recipe to be consistent with area model changes.</td>
</tr>
<tr>
<td>Versioning disabled</td>
<td>Container recipe can be edited.</td>
</tr>
<tr>
<td>Previously versioned recipes with versioning disabled</td>
<td>Container recipe cannot be edited.</td>
</tr>
</tbody>
</table>

See also

Find Recipe Reference overview on page 171

Security authority overview on page 155

Page setup

Page Setup is accessed through the File menu or Ctrl+P, and allows adjustment of:

- Margins (must be greater than 0.25 inches)
- Orientation
- Paper size and source
- Preview (view only)
- Printer and associated properties

Tip: The print area cannot be smaller than 4 inches (10 cm) wide or high.

See also

Complete and maintain recipe on page 159
Generate reports option

Filter and print a complete working set of recipes, or if desired, a single recipe using the Generate Reports option in the FactoryTalk Batch Recipe Editor. The report format consists of these sections: the Overview Page, the SFC, and Recipe Descriptive Information.

When generating reports, the entire recipe procedure appears in the SFC rectangle grid. An SFC rectangle represents a portion of the SFC that prints on one page at 100% scale as defined in the Page Setup. Activating the Show/Hide Page Boundaries button displays the SFC rectangles in the SFC rectangle grid.

If a procedure description is in the Header Data dialog box for a recipe, it displays under the report header information.

If enabled, the approval process data displays in the recipe descriptive information. If approval process data is defined for the recipe, but disabled, nothing related to the approval process displays.

See also

Print a working set of recipes on page 174

Print a single recipe on page 176

Print a working set of recipes

The Print function in the FactoryTalk Batch Recipe Editor uses the configured default printer. If there is no default printer specified, a prompted to establish one opens. Set or change the default printer without having to close the FactoryTalk Batch Recipe Editor application.
To print a working set of recipes:

1. Close all open recipes. Select File > Close All.

2. Select Generate Reports. The Generate Reports dialog box opens.

3. The Recipe Procedures list displays the current working set of recipes. Use Select All to include all displayed recipes in the report. To customize the recipe list for printing, use the filters in the Recipe Filter tab to refine which recipes are displayed. To select nonadjacent files, select the name of any file, then hold down the Ctrl key and select the name of each additional file.

   Tip: Improperly secured recipes in the current working set cannot be printed—a notification window opens and lists the improperly secured recipes. Recipe names listed can be selected and copied to the clipboard.

4. The Report Selection area contains these options:

   - **Overview Page**: Displays the entire SFC grid scaled on a single report page with its page or SFC rectangle boundaries. A cross reference grid provides numeric vertical and alphabetic horizontal coordinates. The SFC rectangles are numbered from left to right, top to bottom. Numbers only appear in the upper right corner of rectangles containing a portion of the SFC.

   - **SFC**: Displays the SFC at 100% scale and includes a cross reference grid with numeric vertical and alphabetic horizontal coordinates. Off-page references indicate the SFCs continued direction.

   - **Descriptive Information**: Displays the recipe header information (with security authority and version history information), unit requirements, phase link groups, recipe formula parameters, steps, step parameters, reports, binding preferences, requirements, report expressions, transitions, and text boxes. Steps and transitions print in the SFC report order, left to right, top to bottom. If a group,
requirement, parameter value, report limit, or text box does not exist, the report notes that status also.

- **Report Description**: Enter a name or descriptive phrase to display in the reports footer.

5. Select **Page Setup** to finalize any page printing adjustments.

6. When all report options have been determined, select **Print**.

**Tip:** Print Preview can only view a single procedure and is not an option when printing an entire recipe set.

See also

Improperly secured recipe ineligible for selection on page 197

### Print a single recipe

Rather than print a set of recipes, print a single recipe.

**To print a single recipe:**

1. Open the recipe to print.

2. Select **Generate Reports**. The **Generate Reports** dialog box opens.

![Generate Reports dialog box]

3. The **Recipe Procedures** list displays the selected (root) recipe and its sub-recipes if any.

4. In the **Report Selection** area, choose the report elements to print.

   - **Overview Page**: Displays the entire SFC scaled on a single report page with its page boundaries. A cross reference grid provides numeric vertical and alphabetic horizontal coordinates. The SFC rectangles are numbered from left to right, top to bottom. Numbers only display in the upper right corner of rectangles containing a portion of the SFC.

   - **SFC**: Displays the SFC at 100% scale and includes a cross reference grid with numeric vertical and alphabetic horizontal coordinates. Off-page references indicate the SFC’s continued direction.

   - **Descriptive Information**: Displays in text form the recipe header information (with security authority and version history information),
unit requirements, phase link groups, recipe formula values, steps, step parameters/reports, binding preferences/requirements, report expressions, transitions, and text boxes. Steps and transitions print in the SFC report order, left to right, top to bottom. If a group, requirement, parameter value, report limit, or text box does not exist, the report notes that status also.

- **Report Description:** Enter a name or descriptive phrase to display in the reports footer.

5. Select **Page Setup** to finalize any page printing adjustments.

6. To view the finished report, select **Print Preview**.

7. When all report options have been determined, select **Print**.

See also

*Generate reports option* on page 174

**Remove a recipe**

When a recipe file is no longer needed, remove it from the current working set of recipes.

**To remove a recipe:**

1. Select **File > Remove Recipe**. The **Remove Recipe** dialog box opens.

2. (optional) Set filters with the **Recipe filter** to narrow the number of recipe names displayed.

   **Tip:** The filter options are the same options as the **Open Recipe** dialog box.

3. From the **Recipe Name** list, select the recipe or recipes to remove, and then select **Remove**.
See also

Open recipe on page 55

Recipe maintenance on page 171

Translation

Recipes created in FactoryTalk Batch 11.x automatically translate as long as the area model translated correctly, and the recipes are not stored in XML, or SQL.

For recipes created in versions older than FactoryTalk Batch version 11.x, contact the Rockwell Customer Support Representative.

Tip: If enabled Recipe Approvals, when 11.x recipes are translated in the FactoryTalk Batch Recipe Editor, they are assigned a default Expedited Approvals process with two steps:

Release Recipe as Step and Release Recipe to Production

These steps initially have the $System signoff following verification. Before verification, these steps are in a $System signoff pending state.

Tip: The RDB Recipes function no longer supports the use of Oracle — use Microsoft SQL Server.

See also

Automatic system signoff ($System) on page 139
Chapter 15

Import and export recipes

The recipe Import and Export commands move exact copies of recipes in and out of the current working set. Recipes can be imported and exported in binary, XML, or RDB format.

When Recipe Version Control is enabled, the FactoryTalk Batch Recipe Editor prompts if version conflicts are encountered during import and export.

- Recipe Approval states are exported as part of a recipe, and are brought in unchanged when a recipe is imported.
- Security Authority supports only the binary file format.

See also

Import recipes on page 179
Export recipes on page 183
Import conflicts on page 185
Import and export restrictions for secured recipes on page 188

Import recipes

Import recipes from any format into the current working set of recipes (binary only for secured recipes).

Important: Importing or exporting FactoryTalk Batch recipes to or from a computer containing more than one instance of Microsoft SQL Server is not supported. The SQL Server instance name cannot be defined in the Import/Export Recipe dialog box.

Before you begin:

- Make sure the recipe file format (binary, RDB, or XML) for the working set of recipes is set to the correct format.
- Create the recipe database on the computer running SQL Server. FactoryTalk Batch uses a database called MasterRecipes.
Tip: If the recipe file format in the FactoryTalk Batch Equipment Editor was changed, restart the FactoryTalk Batch Server if it is running, and re-open the FactoryTalk Batch Recipe Editor for the change to take effect.

To import recipe:

1. Select File > Import Recipe Into Working Set option. The Import Recipe dialog box opens. The title bar of the dialog box indicates the file storage type selected for the current working recipe set in the FactoryTalk Batch Equipment Editor. Imported recipes convert to this format.

2. (optional) In the Recipe directory from which recipes will be imported area, change the storage format for importing recipes.

3. Select the recipe directory or database that contains the files to import.
   - Binary Files or XML Files
   - Microsoft SQL Server Database

4. (optional) Set the Recipe Filter to narrow the list of displayed recipes.

   Tip: The filter options are the same options as the Open Recipe dialog box.

5. From the Recipe Name list, select the recipes to be imported.
• To select nonadjacent files in the **Open** dialog box, select the name of any file. Hold down the **Ctrl** key and select the name of each additional file.

• To select adjacent files in the **Open** dialog box, select the name of the first file in the sequence. Hold down the **Shift** key and select the name of the last file.

• To clear a file selection, hold down the **Ctrl** key and select the file name again.

6. Select **Import**. The recipes selected in the **Recipe Name** list are imported into the current working set.

**Tip:** If one of the imported recipes has the same name as a recipe in the working set, a notification dialog box opens. If a versioned recipe would conflict, when imported, with a version in the working set, a notification dialog box opens. Improperly secured recipes in the current working set cannot be imported--a notification window opens and lists the improperly secured recipes. Recipe names listed can be selected and copied to the clipboard.

**See also**

- [Set recipe file format](#) on page 195
- [Open recipe](#) on page 55
- [Create the MasterRecipes database](#) on page 192
- [Select a directory](#) on page 181
- [Select a SQL Server database](#) on page 182
Select a directory

If Binary Files or XML Files are imported, select the directory that contains the binary or XML files to import.

To select a directory:

1. Select browse next to the Recipe Directory box of the selected storage format. The Select Directory dialog box opens. The default directory, BATCHCTL, displays unless another directory was previously selected.

   ![Select Directory dialog box]

   Tip: If selected the same directory used by the working set of recipes, an error message displays. Choose another directory.

2. Select the directory, and then select Open. The Recipe Name list in the Import Recipe dialog box updates with the recipe list from the specified directory.

See also

Import recipes on page 179

Select a SQL Server database on page 182
Select a SQL Server database

If RDB files or export files to RDB format are imported, select the Microsoft SQL Server database that contains the files to import or the database to export recipes.

**To select a SQL Server database:**

1. Select browse next to the **Computer** box to select the computer on which the SQL Server database resides. The **Select Computer** dialog box opens.

2. In the **Enter the object name to select** box, type the name of the computer where the SQL Server database resides, and select **OK**.

3. Select browse next to the **Database** box to select the database to import recipes or export recipes.

4. Select the database, and then select **OK**. The Recipe Name list on the **Import Recipe** dialog box updates with the list of recipes from the specified database.

**See also**

[Import and export recipes on page 179](#)

Export recipes

Export recipes from the current working set of recipes into any supported storage format, including RDB, XML, or binary (secured recipes only maintain security when exported to binary).

**To export recipes:**

1. Open the FactoryTalk Batch Recipe Editor.
2. From the File menu, select Export Recipe from Working Set. The Export Recipe dialog box opens. The title bar of the dialog box indicates the file storage type selected for the current working set of recipes (set in the FactoryTalk Batch Equipment Editor). The recipes export from this format.

3. In the Recipe directory to which recipes will be exported area, change the storage format for exporting files.
   - Binary Files or XML Files
   - Microsoft SQL Server Database

   **Important:** Importing or exporting FactoryTalk Batch recipes to or from a computer containing more than one instance of Microsoft SQL Server is not supported. The Import/Export Recipe dialog box only allows you to enter the name of the computer hosting the Microsoft SQL Server and an existing database name. The SQL Server instance name cannot be defined in the Import/Export Recipe dialog box.

4. (optional) Set the Recipe Filter to narrow the list of recipes displayed for export.

   **Tip:** The filter options are the same options as the Open Recipe dialog box.

5. From the Recipe Name list, select the recipes to export.
• To select nonadjacent files in the **Open** dialog box, select the name of one file. Hold down the **Ctrl** key and select the name of each additional file.

• To select adjacent files in the **Open** dialog box, select the name of the first file in the sequence. Hold down the **Shift** key and select the name of the last file. To clear a selected file, hold down the **Ctrl** key and select the file name again.

**Tip:** Improperly secured recipes in the current working set cannot be printed—a notification window opens and lists the improperly secured recipes. Recipe names listed can be selected and copied to the clipboard.

6. Select **Export** to initiate the export process. If recipes with the same name exist in the target format, indicate whether or not to overwrite them with the new recipe file.

**See also**

- *Open recipe* on page 55
- *Select a directory* on page 181
- *Select a SQL Server database* on page 182
- *Improperly secured recipe ineligible for selection* on page 197
- *Resolve duplicate name conflict* on page 186

**Import conflicts**

Recipe imports can create naming and versioning conflict. This can take several forms:

• Recipe name conflict
• Versioned recipe basename conflict
• Recipe version number conflict
  • During import
  • When rebuilding the current working recipe set

**See also**

- *Resolve duplicate name conflict* on page 186
- *Resolve recipe basename conflict* on page 186
- *Resolve recipe version conflict* on page 187
- *Resolve recipe conflict when rebuilding the recipe directory* on page 188
Chapter 15  Import and export recipes

Resolve duplicate name conflict

If one of the imported recipes has the same name as a recipe in the working set, the Duplicate Recipe Name dialog box opens.

To resolve duplicate name conflict:

1. Select one of the listed options.

Tip: If the versioned recipe basename exists in the working recipe set, importing versioned recipes can result in a Recipe basename conflict.

Tip: If Recipe Approvals are enabled, the approval states of the imported recipe overwrite those of the existing recipe.

See also

Resolve recipe basename conflict on page 186

Import conflicts on page 185

Resolve recipe basename conflict

If a versioned recipe with the same basename as one or more versioned recipes is imported a warning message opens.

To resolve recipe basename conflict:

1. Select OK to close the warning.
2. Either rename the recipe to import, or remove the existing versioned recipes from the current working set that conflict with the recipe to import.

See also

Import conflicts on page 185

Resolve recipe version conflict

Use these instructions to resolve recipe version conflict.

To resolve recipe version conflict:

1. If a recipe version conflict occurs when importing a versioned recipe into the current working set, the Resolve Version Conflict for Import Request dialog box opens.

2. Select one of these options:

   - Select Import with Delete, which continues the import operation. This deletes the conflicting recipe in the working directory and replaces it with the imported recipe.
   - Select Cancel Import. This does not overwrite the recipe in the working set.

See also

Resolve recipe conflict when rebuilding the recipe directory on page 188
Chapter 15  Import and export recipes

Resolve recipe conflict when rebuilding the recipe directory

When rebuilding the recipe directory (File > Rebuild Recipe Directory), and a recipe version conflict occurs, the Resolve Version Conflict for Rebuild Recipe Directory Request dialog box opens:

To resolve recipe conflict when rebuilding the recipe directory:

1. Choose which recipe to retain in the recipe directory.
2. Select the corresponding Retain button.

See also

Import conflicts on page 185

Import and export restrictions for secured recipes

When recipes are secured with a security authority identifier (SAI), secure import and export can only occur when:

- The recipe to import or export is BINARY.
- The SAI in the recipe matches the SAI in the current FactoryTalk Network Directory.

These tables summarize recipe import and export operation restrictions:

<table>
<thead>
<tr>
<th>Import or Export of binary recipes</th>
<th>SAI match</th>
<th>SAI mismatch</th>
<th>No SAI</th>
</tr>
</thead>
<tbody>
<tr>
<td>To binary</td>
<td>Allowed</td>
<td>Not allowed</td>
<td>Allowed</td>
</tr>
<tr>
<td>To RDB</td>
<td>Not allowed</td>
<td>Not allowed</td>
<td>Allowed</td>
</tr>
</tbody>
</table>
Import or Export of XML or RDB recipes

<table>
<thead>
<tr>
<th></th>
<th>SAI import</th>
<th>SAI export</th>
<th>No SAI</th>
</tr>
</thead>
<tbody>
<tr>
<td>To binary</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Allowed, SAI ignored</td>
<td>Not applicable</td>
<td>Allowed</td>
</tr>
<tr>
<td>To RDB</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Allowed, SAI ignored</td>
<td>Not applicable</td>
<td>Allowed</td>
</tr>
<tr>
<td>To XML</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Allowed, SAI ignored</td>
<td>Not applicable</td>
<td>Allowed</td>
</tr>
</tbody>
</table>

Operations marked as Not allowed open a warning message.

See also

Import and export recipes on page 179
Recipe formats

FactoryTalk Batch recipes can be stored in the proprietary FactoryTalk Batch binary format, XML, or a relational database (RDB) format. The three formats are mutually exclusive. Converting recipes from one format to the other is allowed. Select one of these formats in the FactoryTalk Batch Equipment Editor, to store all recipes created or imported into the working set of recipes in that format.

The binary format is the default for FactoryTalk Batch recipes.

**Important:** Only the binary format is secure from outside modification—it can be viewed and edited only in the FactoryTalk Batch Recipe Editor. Recipes stored in RDB or XML can be edited in external or third-party application software. To protect RDB and XML recipe files that have completed the Recipe Approvals process, save them in a secured location.

See also

- [RDB format on page 191](#)
- [XML format on page 193](#)
- [Set recipe file format on page 195](#)

**RDB format**

The RDB recipe format gives greater flexibility when generating reports. A default Microsoft SQL Server database, MasterRecipes, is created during FactoryTalk Batch installation and selected **Yes, I want to store recipes in SQL Server.**

**Tip:** The MS SQL Server must be installed before installing FactoryTalk Batch. Check software compatibility in the FactoryTalk Batch Components Upgrade and Installation Guide.

Open Database Connectivity (ODBC) is used as the database connectivity protocol between the FactoryTalk Batch Recipe Editor and the Microsoft SQL Server database.
This diagram describes the flow paths and functional relationships of FactoryTalk Batch using a relational database:

See also

Create the MasterRecipes database on page 192

Create the MasterRecipes database

Before importing recipes into the RDB format, first create the recipe database on the computer running SQL Server. FactoryTalk Batch uses a database called MasterRecipes.

Tip: Follow these instructions if the Yes, I want to store recipes in SQL Server option was not selected during FactoryTalk Batch installation, and MS SQL Server is installed.

Before you begin:

- Install and run SQL Server.

To create the MasterRecipes database:

1. If they do not exist, create the MasterRecipeAuthor and MasterRecipeViewer local user groups on the computer to install.

2. If workgroup security is used, create a local user account for the FactoryTalk Batch Server.

3. Add the FactoryTalk Batch Server user account, or the domain account used by the FactoryTalk Batch Server, to the MasterRecipeViewer user group. This allows the FactoryTalk Batch Server to read the MasterRecipes database.

4. Open the Scripts folder (in the Batch directory).
5. Double-click `createmasterreceptedb.bat`. The batch file runs and creates the MasterRecipes database.

6. After the batch file runs, configure FactoryTalk Batch to use the database. This is done in the FactoryTalk Batch Equipment Editor.

**See also**

Set recipe file format on page 195

---

**Add a Windows user to the MasterRecipeAuthor and MasterRecipeViewer local groups**

Add a Windows user to the MasterRecipeAuthor and MasterRecipeViewer according to the function needed. Add recipe authors to the MasterRecipeAuthor local group and recipe viewers to the MasterRecipeViewer local group. The MasterRecipeAuthor user is allowed to edit recipes, and the MasterRecipeViewer user is only allowed to view recipes. FactoryTalk Batch uses a database called MasterRecipes.

To add a Windows user to the MasterRecipeAuthor and MasterRecipeViewer local groups

1. Add the FactoryTalk Batch Server user account, or the domain account used by the FactoryTalk Batch Server, to the MasterRecipeViewer user group. This allows the FactoryTalk Batch Server to read the MasterRecipes database.

2. Add the Windows user to the MasterRecipeAuthor or MasterRecipeViewer local group.

**See also**

Set recipe file format on page 195

---

**XML format**

XML schemas have become an industry standard for providing implementation-neutral information. An XML schema defines what a textual file should look like to represent structured data, for instance a master recipe. Recipes are an instance document of an XML schema and can easily exchange between different companies.

The XML recipe format gives greater flexibility when generating recipes that must be usable across multiple operating systems and platforms. The FactoryTalk Batch Recipe Editor is capable of editing and creating XML recipes, and the FactoryTalk Batch Server can execute these recipes.

**See also**

RDB format on page 191
XML recipe schema on page 195
XML recipe schema

The FactoryTalk Batch Server and FactoryTalk Batch Recipe Editor can only read a valid XML recipe. An invalid recipe, one that does not conform to the MasterRecipe schema (MasterRecipe.xsd), is rejected, thus protecting the FactoryTalk Batch Recipe Editor and Server from faulty input. A valid recipe does not imply that the recipe has been verified by the FactoryTalk Batch Recipe Editor.

In addition to the information that is verified by FactoryTalk Batch Recipe Editor and FactoryTalk Batch Server, these items in XML recipes are also validated by the FactoryTalk Batch Recipe Editor and the MasterRecipe schema:

- **Referential Integrity**: An element referenced by another element exists.
- **Uniqueness**: Element names within a collection are unique.
- **Name Length**: Element names are restricted to a certain length. The recipe name cannot exceed 50 valid characters. The maximum length of an expression is 1023 characters. The maximum number of characters contained in a text box is 1024.
- **Name Pattern**: Names conform to a certain syntax. Sequential function charts must follow a step-transition-step pattern.

See also

Transition expressions on page 25

Set recipe file format

Store the working set of recipes in one file format. All recipes that imported into the working set must be in the format designated for the working set of recipes.

**Important**: Shut down and restart the FactoryTalk Batch Recipe Editor when changing between recipe storage types.

To set or change the recipe file format:

1. Select Start, point to All Programs > Rockwell Software > FactoryTalk Batch Suite > FactoryTalk Batch, and then select Equipment Editor. The FactoryTalk Batch Equipment Editor application opens.
2. From the Options menu, select Server Options. The Server Options dialog box opens with the Project Settings tab displayed.
3. In the Store Recipes Using area, select the recipe file format to store recipes.

If the Microsoft SQL Server Database was selected:
Appendix A  Recipe formats

- Locate the SQL Server by selecting browse next to Node. The Select Computer dialog box displays.
- Select the SQL Server computer and select OK to return to the Server Options dialog box.
- Locate the appropriate database by selecting browse next to Database. A list of databases display.
- Select the SQL Server database and select OK to return to the Server Options dialog box. The selected database name is inserted into the Database box.

Tip: If the SQL Server is not installed on the same computer as the FactoryTalk Batch Client, to view a list of available SQL Server databases install the SQL Server Client Tools Connectivity option. Do a custom installation of SQL Server to install the Client Tools Connectivity option.

If Binary Files or XML Files was selected:

- Locate the recipe directory by selecting browse next to Recipe Directory. The Select Directory dialog box displays.
- Select the appropriate directory and select OK to return to the Server Options dialog box. The selected directory path name is inserted into the Recipe Directory box.

4. Select OK and then exit the FactoryTalk Batch Equipment Editor.

See also

- RDB format on page 191
- XML format on page 193
Appendix B

Import/Export error issues

Errors may occur during recipe import or export due to security issues or verification issues.

See also

- Improperly secured recipe ineligible for selection on page 197
- Verification results for a versioned recipe on page 198
- Invalid recipe folder or directory path warning on page 201

Improperly secured recipes in the current working set cannot be selected when using the Import, Export, or Generate Reports command in the FactoryTalk Batch Recipe Editor. An improperly secured recipe is one in which the security authority identifier (SAI) in the recipe does not match that in the current FactoryTalk Network Directory.

This notification window opens to list any improperly secured recipes.

The recipe name listed in the notification window can be selected and copied to the clipboard for reference.
See also

Import/Export error issues on page 197

Verification results for a versioned recipe

When checking in a recipe or procedure as part of the versioning process, the recipe or procedure must pass verification. During the verification process, the FactoryTalk Batch Recipe Editor opens the **Verification Results** dialog box, listing any warnings or errors found. Warnings allow creation of a versioned recipe. Errors prevent successful verification and the creation of a versioned recipe.

See also

Solutions for common verification warnings and errors on page 198

Solutions for common verification warnings and errors

When a lower level operation or unit procedure prevents verification of its parent recipe or procedure, these are the likely causes:

- A lower level operation or unit procedure does not have its **Release Recipe as Step** property enabled.
  Solution: In the **Verification Results** dialog box, select **Recipe > Header Data** command to check each operation or procedure identified with this warning, and enable its **Release Recipe as Step** property. If enabled Primary or Expedited Approvals, sign off on its **Release Recipe as Step** approval step.

- All lower level operations or unit procedures must be versioned before the parent can be versioned.
  Solution: In the Verification Results dialog box use the **Check in** command to apply versioning to each operation or procedure identified with this error.

- A lower level operation or unit procedure has been marked as obsolete, because of conflicts with the area model.
  Solution: Refer to the two methods described in these examples.

As a best practice, apply either of these methods starting with the lowest level of the procedure's hierarchy and work up to the top level.
• Method 1: modify the area model to reinstate an obsolete procedure

• Method 2: use Check Out and Redefine Step to remove an area model conflict

See also

Modify the area model to reinstate an obsolete procedure example on page 199

Use Check Out and Redefine Step to remove an area model conflict example on page 200

Modify the area model to reinstate an obsolete procedure example

A Unit Procedure, CLS_FRENCHVANILLA_UP, is verified when opened in the FactoryTalk Batch Recipe Editor. An operation within the unit procedure, CLS_FRENCHVANILLA_OP~V1, is marked as Obsoleted during verification, due to a conflict detected with the area model. The conflict is because parameter INGREDIENT_A was added to a phase after the recipe CLS_FRENCHVANILLA_OP~V1 was versioned.

The formula engineer determines that the parameter, INGREDIENT_A, is not needed in the area model.

To modifying the area model to reinstate an obsolete procedure:

1. Close the FactoryTalk Batch Recipe Editor.

2. Open the FactoryTalk Batch Equipment Editor, delete the unnecessary parameter INGREDIENT_A, then save the area model.

3. Open the FactoryTalk Batch Recipe Editor.

4. Open the unit procedure CLS_FRENCHVANILLA_UP. When it opens, verification is performed and the operation within the unit procedure, CLS_FRENCHVANILLA_OP~V1, is marked as Reinstated.

No further action is required.

See also

Verification results for a versioned recipe on page 198

Solutions for common verification warnings and errors on page 198
Use Check Out and Redefine Step to remove an area model conflict example

A unit procedure, CLS_FRENCHVANILLA_UP, is verified when opened in the FactoryTalk Batch Recipe Editor. An operation within the unit procedure, CLS_FRENCHVANILLA_OP~V1, is marked as Obsoleted during the verification, because of a conflict detected with the area model. The conflict is because parameter INGREDIENT_A was added to a phase after the recipe CLS_FRENCHVANILLA_OP~V1 was versioned.

The formula engineer determines that the parameter INGREDIENT_A is needed in the area model and for the operation which is currently marked Obsoleted.

To use Check Out and Redefine Step to remove an area model conflict:

1. Select File > Open Top Level. In the open recipe dialog box, select the Recipe Filter tab, then select Obsoleted from the Versioning Data group.

   CLS_FRENCHVANILLA_OP~V1 appears in the Recipe List.

2. Double-click CLS_FRENCHVANILLA_OP~V1 to open and verify it. The Verification Results dialog box shows that CLS_FRENCHVANILLA_OP~V1 has a Conflict and is Not Updated.

3. In the Verification Results dialog box, select Check Out.

4. In the Check Out dialog box, select OK. Verification is performed and the Verification Results dialog box shows that CLS_FRENCHVANILLA_OP~V2_WIP is created, and updated to include the parameter INGREDIENT_A.

5. Select Close to dismiss the Verification Results dialog box.

6. Select File > Check In. The Verification Results dialog box opens.

7. Select Create Version. The Check In dialog box opens.

8. Select OK.

9. In the Search for recipes has completed dialog box, select Proceed. CLS_FRENCHVANILLA_OP~V2 appears in the left pane view.

10. Double-click CLS_FRENCHVANILLA_UP to open it.

    Verification is performed, and the Verification Results dialog box shows that CLS_FRENCHVANILLA_OP~V1 is Obsoleted.

11. In the Verification Results dialog box, select Close.

12. In the left pane tree view of the FactoryTalk Batch Recipe Editor, select CLS_FRENCHVANILLA_OP~V1:1.
13. Select the **Step > Redefine Step** menu command (toolbar icon).

   The **Redefine Step** command allows reference to an alternative copy or version of the operation or unit procedure.

14. From the **Operation Select** dialog box, select CLS_FRENCHVANILLA_OP~V2.

15. Select **OK**. Unit procedure CLS_FRENCHVANILLA_UP can be checked in as a versioned recipe.

**See also**

- [*Verification results for a versioned recipe* on page 198](#)
- [*Solutions for common verification warnings and errors* on page 198](#)

### Invalid recipe folder or directory path warning

Invalid recipe folder or directory path warning may occur when performing recipe file operations such as **Export** in the FactoryTalk Batch Recipe Editor:

![Error selecting folder, try again.](image)

This error can result if:

- The selected folder is not valid.
- The path to the folder is not valid.
- The selection is not a folder.

Confirm the folder is valid in Windows Explorer, and then retry the operation.

**See also**

- [*Import/Export error issues* on page 197](#)
Copyright notice

Copyright © 2018 Rockwell Automation Technologies, Inc. All Rights Reserved. Printed in USA.

This document and any accompanying Rockwell Software products are copyrighted by Rockwell Automation Technologies, Inc. Any reproduction and/or distribution without prior written consent from Rockwell Automation Technologies, Inc. is strictly prohibited. Please refer to the license agreement for details.

End User License Agreement (EULA)

You can view the Rockwell Automation End-User License Agreement (“EULA”) by opening the License.rtf file located in your product's install folder on your hard drive.

Trademark Notices

Allen-Bradley, CompactLogix, ControlLogix, Connected Components Workbench, eProcedure, FactoryTalk, GuardLogix, GuardPLC, Micro800, MicroLogix, MobileView, MobileView Guard, PanelBuilder, PanelView, PhaseManager, PlantLink, PlantPAx, PLC-2, PLC-3, PLC-5, PowerMonitor, RSBizWare, RSBizWare BatchHistorian, RSBizWare Historian, RSBizWare MaterialTrack, RSLinx, Rockwell Automation, Rockwell Software, SLC, SoftLogix, Studio 5000, and XMare trademarks of Rockwell Automation, Inc.

Any Rockwell Automation software or hardware not mentioned here is also a trademark, registered or otherwise, of Rockwell Automation, Inc.

For a complete list of products and their respective trademarks, go to: http://www.rockwellautomation.com/rockwellautomation/legal-notices/overview.page?#tab4

Warranty

This product is warranted in accordance with the product license. The product’s performance may be affected by system configuration, the application being performed, operator control, maintenance, and other related factors. Rockwell Automation is not responsible for these intervening factors. The instructions in this document do not cover all the details or variations in the equipment, procedure, or process described, nor do they provide directions for meeting every possible contingency during installation, operation, or maintenance. This product’s implementation may vary among users.
This document is current as of the time of release of the product; however, the accompanying software may have changed since the release. Rockwell Automation, Inc. reserves the right to change any information contained in this document or the software at any time without prior notice. It is your responsibility to obtain the most current information available from Rockwell when installing or using this product.

**Environmental compliance**


**Contact Rockwell Automation**

Customer Support Telephone — 1.440.646.3434

Index

...)

A
adding
  branch structures to an SFC  84
  operations to a recipe  58
  parallel steps to a table  96
  parallel steps to an SFC  82
  phase link groups  130
  procedure  60
  recipe header data  70
  steps before parallel steps  97
  steps to an SFC  82
  transitions to an SFC  85
  unit procedures  58
Amount parameter  107
AND convergence  35, 83
AND divergence  35, 83
AND path, example  30
AND structure  23
arbitration
  null phase  34
assigning
  recipe formula parameters  116, 117
  report limits  112
  step formula values  112
  Timer step formula values  113
  transition expressions  88
associating comments  91
associating recipe comments  91
associating text boxes  91

B
begin parallel step  42
binary format
  exporting recipes to  181
binary recipe format  189
binding expressions

C
changing
  recipe storage format  192
  unit requirements  60, 64
command buttons  17
comments
  associating  91
  disassociating  91
  editing  90
configuring
  binding preferences  68
  binding requirements  66
  Recipe Editor options  12
container binding  108
Container parameter  107
control strategies
  formula values  106, 112
  phase link groups  127
copying a recipe  75
creating
  operations  58
  phase link groups  130
  procedure  60
  unit procedures  58

D
data types  29
  binding expressions  100
  enumeration  100
  examples  100
  integer  100
  real  100
string 100
decimal places
  setting 12
decision logic 30
default printer 172
defer
  parameters 108
defining
  steps 86
deleting
  equipment requirements 64
  phase link groups 131
  recipes 174
descriptive information 173
descriptive information report 172
disassociating recipe comments 91
disassociating text boxes 91
  display 108
dynamic unit allocation 12, 53

E
Edit menu 15
  editing comments 90
  editing text boxes 90
  end parallel step 42
enum/EU 108
enumeration
  data type 29
equipment requirements
  deleting 64
EU 108
examples
  AND path 30
  complex parallel structure 44
data types 29
decision logic 30
  loop logic 30
OR path 30
parallel logic 30
REPEAT path 30
SFC execution 32
  simple parallel structure 43
  transitions 25
export
  RDB files 180
  recipes 181
exporting
  recipes to binary 181
  recipes to SQL 181
  with same name 183
XML recipes 181
expression builder
  unit binding 99
validation-Equipment Editor 101
expression length
  maximum 101
expressions
  operands-Recipe Editor 101
  operators-Recipe Editor 124
  validating-Recipe Editor 101
expressions, transition 25

F
FactoryTalk Batch
  upgrading older versions 175
FactoryTalk Batch Equipment Editor
  log on to the Recipe Editor 10
FactoryTalk Batch Recipe Editor
  menu bar 15
  recipe construction toolbox 17
  toolbar 15
feed type 108
File menu 15
file types 74
final step 23
footer
  report description 173
Format menu 15
formula values
  assigning 112, 113
  control strategies 106, 112
  material-enabled phase parameters 106

G
generate reports 171

H
H 108
header data 70
Help menu 15
HH 108
HHH 108
hiding
  recipe construction toolbox  17

I
ID  108
import
  RDB files  180
importing
  binary recipes  177
  RDB recipes  177
  recipes  177
  with same name  183
  XML recipes  177
initial step  23
inserting
  steps into a table  95
  steps into an SFC  79
integer
  data type  29

L
L  108
Label parameter  107
limit calculation  108
limitations
  SFC  31
Link menu  15
linking
  SFC components  87
  SFC elements  30
LL  108
LLL  108
loop
  material  89
  rebinding  89
loop logic  30
Lot parameter  107

M
maintaining recipes  157
Material Class parameter  107
material class-based recipes  50
material loop
  creating  89
  examples  89
Material parameter  107
material report parameters  108
material-based recipes  49
material-enabled phase
  feed type  108
  parameters  106
max  108
maximum expression length  101
menu bar  15
min  108
modifying
  unit requirements  60, 64
moving text boxes  90

N
name  108
naming a recipe  70
null
  procedure  33
null phase
  arbitration  34
NULL_CLASS  108
NULL_CONTAINER  108
NULL_MATERIAL  108

O
Open Database Connectivity  189
opening
  a recipe  54
  Equipment Editor  10
operands
  binding expression-Recipe Editor  101
operation  47
  changing unit requirements  60
  creating  58
  file types  74
  S88.01 definition  9
operators, expressions-Recipe Editor  124
operators, transitions  28
Options dialog box  12
OR convergence  89
OR divergence  89
OR path, example  30
OR structure  23
origin  108
overview page report  172, 173
Index

P
Page Boundaries 171
parallel logic 30
parallel steps-recipe structure 42
  adding step before 97
  adding to a table 96
  adding to an SFC 82
parallel structure
  complex 44
  simple 43
  table view 43
parameter values
  assigning 112, 113
parameters
  defer 108
  recipe formula 116, 117
  setting decimal places 12
phase link groups
  control strategies 127
  creating 130
  deleting 131
phase report limits
  assigning 112
phase, Procedure levels 47
  S88.01 definition 9
printing
  binding preferences 172, 173
  binding requirements 172, 173
  recipes 171
  working set 172
printing report expressions 172, 173
procedure 47
  changing unit requirements 64
  creating 60
  file types 74
  S88.01 definition 9
procedure filtering 172
procedure level
  operation 47
  phase 47
  procedure 47
  unit procedure 47
procedure levels 47
procedure levels 47
process levels 47
process-connected device 13

R
RDB files
  export 180
  import 180
RDB format 189
  exporting recipes to 181
  process flow diagram 189
RDB recipe format 189
real
  data type 29
  rebinding loop
  creating 89
  rebuilding
    recipe directory 168
recipe
  assigning recipe formula parameters 116, 117
  building overview 18
  copying 75
  exporting to binary 181
  exporting to SQL 181
  exporting XML 181
  file types 74
  header data 70
  importing binary 177
  importing RDB 177
  importing XML 177
  material class-based 50
  material-based 49
  opening 54
  printing 171
    multiple recipes 172
  procedure levels 47
  process levels 47
  rebuilding directory 168
  releasing to production 168
  removing 174
  saving 74
  SFC view 23
  translating 175
  unit-based 49
  upgrading
    FactoryTalk Batch 175
    verifying 157
Recipe menu 15
recipes
  printing a working set 172
rectangle
SFC 171
rectangle grid 171
releasing
   recipe to production 168
removing
   recipes 174
   SFC link 87
steps
   from a table 98
   from an SFC 88
REPEAT path, example 30
report
   descriptive information 172, 173
   header 171
report description 172, 173
report expressions
   printing 172, 173
report footer
   report description 173
report information 171
report limits
   assigning 112
report parameters
   material 108
report selection 172, 173
reports
   generate 171
   print 171
   procedure filtering 172
resource
   arbitration 34
assigning transition expressions 88
branch structures 84
components
   AND structure 23
   final step 23
   initial step 23
   OR structure 23
   step 23
   transition 23
   defining steps 86
element linking 30
examples 89
   AND path 30
   execution 32
   OR path 30
   REPEAT path 30
inserting steps 79
limitations 31
linking components 87
null procedures 33
removing
   links 87
   steps 88
SFC rectangle 171
SFC rectangle grid 171
SFC report 172, 173
SFC validation 159
SFC validation errors
   Linear Segment Cannot Reach Terminal Step 165
   Parallel Activation of Linear Segment 162
   Parallelism with Terminal Step 164
   Unreachable Linear Segment 166
   Unreachable Terminal Step 163
SFC view, overview of SFC 23
Show/Hide Page Boundaries 171
showing
   recipe construction toolbox 17
simple recipes 42
simple step 42
smart binding 65
starting
   Equipment Editor 10
   step formula values
      assigning 112
      assigning to Timers 113
Step menu 15
steps 23, 24
adding
  before parallel steps  97
parallel steps to a table  96
parallel steps to an SFC  82
to an SFC  82
defining  86
inserting
  into a table  95
  into an SFC  79
removing
  from SFC  88
  removing from table  98
string
  data type  29
symbols  42
  SFC recipe structure  42

T

table
  adding parallel steps  96
  adding step before parallel steps  97
defining steps  86
inserting steps  95
removing steps  98
table, viewing table-based recipes
  parallel structures  43
  recipe structure  42
transition expressions  43
target parameter  108
text boxes
  associating  91
disassociating  91
  editing  90
  moving  90
timer steps  38
inserting  80
toolbar-Recipe Editor  15
  toolbox  17
transition name  85
transitions-SFC components  23
  adding to an SFC  85
assigning conditions  88
data types-Recipe Editor  29, 100
elements  25
expressions  25, 43
expressions in table view  43
identifiers  25
operators  28
  translating
  recipes  175
type-Parameter Value/Report Limit definition  108

U
unit binding expression builder-Recipe Editor  99
unit procedure  47
  changing unit requirements  60
creating  58
file types  74
S88.01 definition  9
unit requirements  48
  changing  60, 64
unit-based recipe  49
upgrading
  older versions of FactoryTalk Batch  175
  recipes  175

V
validating
  binding expressions-Recipe Editor  101
validating SFCs  159
value  108
verification method  108
verifying recipes  157
View menu  15
viewing
  SFC recipes  23

X
XML recipe format  191
XML recipes
  exporting  181
  importing  177
Rockwell Automation support

Rockwell Automation provides technical information on the web to assist you in using its products. At http://www.rockwellautomation.com/support you can find technical and application notes, sample code, and links to software service packs. You can also visit our Support Center at https://rockwellautomation.custhelp.com for software updates, support chats and forums, technical information, FAQs, and to sign up for product notification updates.

In addition, we offer multiple support programs for installation, configuration, and troubleshooting. For more information, contact your local distributor or Rockwell Automation representative, or visit http://www.rockwellautomation.com/services/online-phone.

Installation assistance

If you experience a problem within the first 24 hours of installation, review the information that is contained in this manual. You can contact Customer Support for initial help in getting your product up and running.

<table>
<thead>
<tr>
<th>United States or Canada</th>
<th>1.440.646.3434</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outside United States or Canada</td>
<td>Use the Worldwide Locator available at <a href="http://www.rockwellautomation.com/locations">http://www.rockwellautomation.com/locations</a>, or contact your local Rockwell Automation representative.</td>
</tr>
</tbody>
</table>

New product satisfaction return

Rockwell Automation tests all of its products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning and needs to be returned, follow these procedures.

<table>
<thead>
<tr>
<th>United States</th>
<th>Contact your distributor. You must provide a Customer Support case number (call the phone number above to obtain one) to your distributor to complete the return process.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outside United States</td>
<td>Please contact your local Rockwell Automation representative for the return procedure.</td>
</tr>
</tbody>
</table>

Documentation feedback

Your comments will help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete the feedback form, publication RA-DU002.

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

www.rockwellautomation.com
# Important User Information

Solid-state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation, and Maintenance of Solid State Controls (publication SGI-1.1 available from your local Rockwell Automation sales office or online at [http://www.rockwellautomation.com/literature/](http://www.rockwellautomation.com/literature/)) describes some important differences between solid-state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid-state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

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.

<table>
<thead>
<tr>
<th>Icon</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>⚠️</td>
<td><strong>WARNING:</strong> 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.</td>
</tr>
<tr>
<td>⚠️</td>
<td><strong>ATTENTION:</strong> 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</td>
</tr>
<tr>
<td>⚠️</td>
<td><strong>SHOCK HAZARD:</strong> Labels may be on or inside the equipment, for example, a drive or motor, to alert people that dangerous voltage may be present.</td>
</tr>
<tr>
<td>⚠️</td>
<td><strong>BURN HAZARD:</strong> Labels may be on or inside the equipment, for example, a drive or motor, to alert people that surfaces may reach dangerous temperatures.</td>
</tr>
<tr>
<td>⚠️</td>
<td><strong>Important:</strong> Identifies information that is critical for successful application and understanding of the product.</td>
</tr>
</tbody>
</table>

Allen-Bradley, Rockwell Software, and Rockwell Automation ControlLogix, sProcedure, FactoryTalk, RSView, RSView Batch Historian, RSView Historian, RSView Material Track, and RSLogix are trademarks of Rockwell Automation, Inc.

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

Chapter 1

FactoryTalk Batch View Overview

- FactoryTalk Batch View security .............................................................. 7
- Configure security for the FactoryTalk Batch View windows .................... 8
- Security for FactoryTalk Batch commands .............................................. 9
- Configure security for FactoryTalk Batch commands .............................. 9
- Enable the FactoryTalk Batch View confirm settings .............................. 11
- Set the automatic logoff time period ......................................................... 12
- Change the logged-on user with your HMI .............................................. 12
- External security devices ......................................................................... 14
- Enable an external security device .......................................................... 15
- Open FactoryTalk Batch View ................................................................. 16
- Log into FactoryTalk Batch View ............................................................. 18
- FactoryTalk Batch View interface ............................................................ 19
- FactoryTalk Batch View toolbar ............................................................... 20
- FactoryTalk Batch View command buttons ............................................ 21
- FactoryTalk Batch View status bar .......................................................... 22
- Communication status for the FactoryTalk Batch Server and data servers 23
  - View the Conversation Status of Data Servers dialog box .................. 24
- Communication status for the FactoryTalk Batch Server and Material Server 24
  - Process batches without the Material Server ....................................... 25
    - Batch creation when the Material server is not communicating .......... 26
    - Prompt to bind unit requirement ...................................................... 27
    - Manual unit binding ...................................................................... 28
    - Prompt to bind a material phase step .......................................... 28
    - Look up material information ....................................................... 29
    - Phase control ............................................................................. 29
  - Reestablish communications with the Material server ......................... 30
  - Held batches .................................................................................. 30
    - Reestablish communication with the Material Server on manual batches 31
- System Configuration and Defaults dialog box ....................................... 31
  - Modify the configuration of the FactoryTalk Batch View ..................... 32
    - System Configuration and Defaults dialog box - General tab ............ 32
    - System Configuration and Defaults dialog box - Signatures tab ........ 34
    - System Configuration and Defaults dialog box - Commands tab .......... 35
    - System Configuration and Defaults dialog box - Batch List tab .......... 36
    - System Configuration and Defaults dialog box - Hierarchical Step List tab ................................................................. 38
    - System Configuration and Defaults dialog box - SFC View tab ........... 39
    - System Configuration and Defaults dialog box - Procedure Prompts tab ............................................................. 40
    - System Configuration and Defaults dialog box - Procedure Parameters tab .......................................................... 41

Rockwell Automation Publication FactoryTalk Batch View User Guide - February 2017
# Table of contents

- System Configuration and Defaults dialog box - Procedure Reports tab 43
- System Configuration and Defaults dialog box - Procedure Binding tab 44
- System Configuration and Defaults dialog box - Procedures Table tab 46
- System Configuration and Defaults dialog box - Unit Procedures Table tab 47
- System Configuration and Defaults dialog box - Operations Table tab 48
- System Configuration and Defaults dialog box - Journal tab 49
- System Configuration and Defaults dialog box - Unacknowledged Prompts tab 50
- System Configuration and Defaults dialog box - Alarm Summary tab 52
- System Configuration and Defaults dialog box - Phase Control tab 53
- System Configuration and Defaults dialog box - Phase Summary tab 55
- Column Display area 56
- Change the selected font in the status bar 57
- SFC Display area 58
- Batch List view 59
- Add a batch to the Batch List 60
- Remove a batch from the Batch List 62
- Add a comment to an electronic batch record 63

## Chapter 2

### FactoryTalk Batch View windows

- Procedure as SFC window 65
  - Commands available in the Procedure as SFC window 66
  - Procedural Hierarchy area 67
  - SFC area 67
  - Auxiliary Index view 68
    - Binding Requirements and Preferences view 70
    - Unit requirements 70
    - Binding requirements 71
    - Binding requirement expression values 72
    - Binding preferences 73
    - Binding preferences expression values 74
  - Procedure as Table window 75
    - Commands available in the Procedure as Table window 75
    - Recipe table area 76
  - Event Journal window 76
    - View the event journal after batch execution 77
    - Filter entries in an event journal 77
  - Unacknowledged Prompts window 77
    - Acknowledge a prompt 78
    - Acknowledge a material phase step prompt 78
  - Signature List window 79
    - Acknowledge a signature 79
    - Set parameter signature overview 80
    - Command signature overview 81
    - Additional signature request types 82
## Table of contents

- Signature request cancellation ................................................................. 83
- Signoff failures ......................................................................................... 83
- Phase Control window ............................................................................... 83
  - Process Cell or Unit Display area ......................................................... 85
  - Phase List Display area ........................................................................ 86
  - Phase Specific Information Display area ............................................. 86
- Command phases in the Phase Control window ........................................ 87
  - Acquire a phase .................................................................................... 88
  - Acquire a material phase ..................................................................... 89
  - Start a phase ........................................................................................ 90
  - Acknowledge a prompt for a phase .................................................... 90
  - Reset a phase ...................................................................................... 91
  - Release a phase .................................................................................. 91
  - Command a phase using AUTO mode ................................................. 92
- Semi-auto processing ................................................................................ 93
  - Perform semi-auto processing ............................................................ 94
  - Semi-auto phase processing ............................................................... 95
  - Perform semi-auto phase processing ................................................. 95
- Arbitration window ................................................................................ 96
  - Acquire and release resources ............................................................ 96
- Alarm Summary window .......................................................................... 97
  - View alarm messages ........................................................................ 97
  - Clear failure messages ........................................................................ 97
- Phase Summary window .......................................................................... 98

### Chapter 3

#### Command a batch

- Start a batch ........................................................................................... 99
- Hold a batch ............................................................................................. 100
- Restart a batch ......................................................................................... 101
- Abort a batch .......................................................................................... 101
- Stop a batch ............................................................................................. 102
- Override an expression value ................................................................. 102
- Force a transition to fire ......................................................................... 103
- Recipe modes .......................................................................................... 104
  - AUTO recipe mode ............................................................................. 104
  - SEMI-AUTO recipe mode ................................................................. 105
  - MANUAL recipe mode ........................................................................ 105
- Timer procedure modes .......................................................................... 106
- Place a batch in AUTO mode ................................................................. 107
- Semi-auto processing ............................................................................ 108
  - Perform semi-auto processing ............................................................ 108
- Place a batch in MANUAL mode ............................................................. 109
- Command timer steps ........................................................................... 110
  - Reset a timer step ............................................................................... 110
  - Force a timer step to complete ........................................................... 110
  - Change the setpoint of a timer step .................................................. 111
Table of contents

Perform an active step change................................................................. 111
Manual binding....................................................................................... 113
  Perform manual binding................................................................. 114
Material binding.................................................................................. 115
  Perform material binding............................................................... 115
Prompted binding.................................................................................. 116
  Perform prompted binding............................................................. 116
Disconnect a phase............................................................................... 116
Cross invocation.................................................................................... 118
  Perform cross invocation............................................................... 119

Chapter 4

Troubleshoot batch issues

Index

Legal Notices

Legal Notices....................................................................................... 127
Chapter 1

FactoryTalk Batch View Overview

The FactoryTalk Batch View is used to initiate and execute FactoryTalk Batch automation processing. The FactoryTalk Batch View secured objects are located in the FactoryTalk Diagnostics and are modified using the FactoryTalk Administration Console. A system administrator can customize the FactoryTalk Batch View security to meet the needs of the facility. The FactoryTalk Batch View is used in conjunction with a Human-Machine Interface (HMI).

See also

FactoryTalk Batch View security on page 7
Open FactoryTalk Batch View on page 16
FactoryTalk Batch View interface on page 19
System Configuration and Defaults dialog box on page 31
Batch List view on page 58

FactoryTalk Batch View security


Any FactoryTalk Security user with the appropriate permissions can log on or off of the FactoryTalk Batch View. The FactoryTalk Batch View can be configured to require user authentication each time a command is issued. Users can log on or off of the FactoryTalk Batch View without stopping or exiting ANY application. The FactoryTalk Batch View security can automatically log off the current user if the FactoryTalk Batch View is inactive for a period of time. Commands sent to the FactoryTalk Batch Server from the FactoryTalk Batch View are logged in the electronic batch record along with the name of the user who initiated the command.

See also

Enable the FactoryTalk Batch View confirm settings on page 11
Configure security for the FactoryTalk Batch View windows

To tighten security for a FactoryTalk Batch View window, remove the All Users group from the corresponding product policy and then add the necessary user accounts.

To configure security for the FactoryTalk Batch View windows:

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

2. Expand System > Policies > Product Policies > Batch > BatchView & ActiveX.

3. Right-click View and then select Properties.

4. Select the policy setting you want to configure and then click the corresponding browse button.

5. In the Configure Securable Action dialog box:
   - Select a user or group and then click Remove.
   - Allow/deny a user or group access to the feature by selecting or clearing the corresponding check box and then clicking OK.
     Tip: Product policies do not inherit security settings. When specifying permissions for product policies, clearing both the Allow and Deny check boxes denies access to the product feature.
   - 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 and Computer dialog box you can:
   - Select a user, user group or computer 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 View Properties dialog box.

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

See also

FactoryTalk Batch View security on page 7
Configure FactoryTalk Security to require user confirmation for batch commands and phase commands. A command that has the confirm feature enabled does not execute until the user account is validated in the Log on to Confirm User dialog box.

To enable the Confirm feature, configure the FactoryTalk Batch product policy for the command and then set the associated <Command> Confirm policy setting to True. These policy settings are defined in FactoryTalk Directory for each command button within FactoryTalk Batch View and every phase command issued from the Phase Control window.

Tip: When the Confirm feature is enabled for a command, only active user accounts assigned to the Command are allowed to execute the command. Users validated in the Log on to Confirm User dialog box are not logged on to FactoryTalk Batch View.

The <Command> Confirm policy settings are found in the following locations:

- FactoryTalk Directory\Local | Network\System\Policies\Product Policies\Batch\BatchView and ActiveX\Commands
- FactoryTalk Directory\Local | Network\System\Policies\Product Policies\Batch\BatchView and ActiveX\Phase Commands

See also

FactoryTalk Batch View security on page 7
Configure security for FactoryTalk Batch commands on page 9

To allow only specified users to issue commands against a batch or a phase, add the appropriate users or user groups to the Command or Phase Command policy setting. Then remove or restrict the All Users group.

To configure security for FactoryTalk Batch commands:

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

2. Expand System > Policies > Product Policies > Batch > BatchView & ActiveX.
3. Right-click **Commands** and then select **Properties**.

4. Select the policy setting to configure and then select the corresponding browse button.

5. (optional) In the **Configure Securable Action** dialog box:
   - Select a user or group and then select **Remove**.
   - Allow or deny a user or group access to the feature by selecting or clearing the corresponding check box and then selecting **OK**.
   - Add a user account or user group by selecting **Add**. This option displays the **Select User and Computer** dialog box.

6. (optional) In the **Select User or Group** dialog box:
   - Select a user, user group, computer, or computer group and then select **OK**.
   - Select **Create New** to create and then add a new user, user group, computer, or computer group.

7. Select **OK** to close the **View Properties** dialog box.

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

**See also**

- FactoryTalk Batch View security on page 7
- Security for FactoryTalk Batch commands on page 8
Enable the FactoryTalk Batch View confirm settings

If specific users and/or groups are assigned to a batch command or phase command and the corresponding Confirm policy setting is enabled, only the specified users are allowed to issue the command. All commands in FactoryTalk Batch View have a command confirmation option. If confirm is enabled for a command, that command is not issued until a user with the appropriate permissions is validated using the Log on to Confirm User dialog box. Users validated in the Log on to Confirm User dialog box are not logged in to the FactoryTalk Batch View.

To enable the FactoryTalk Batch View confirm settings:

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

2. Expand System > Policies > Product Policies > Batch > BatchView & ActiveX.

3. Right-click Phase Commands (or Commands) and then select Properties.

4. Scroll to the BatchView and ActiveX - Phase Commands (or Commands) - Confirmations policy settings. Select the policy setting and then click the corresponding drop-down list box.

5. Select True to require a user log on each time the command button is clicked. Select False if no log on is required.

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

See also

Security for FactoryTalk Batch commands on page 8
Set the automatic logoff time period on page 12
Change the logged-on user with your HMI on page 12
External security devices on page 14

Set the automatic logoff time period

Configure a specified period of inactivity before logging off a user using FactoryTalk Batch View security. FactoryTalk Batch View uses the Windows screen savior activation to send the a log off message. If the Batch List window of FactoryTalk Batch View is the top window in your system when the screen saver activates, FactoryTalk Batch View logs off the current user, the status bar displays No authenticated User, and FactoryTalk Batch View continues to run.

To set the automatic logoff time period:

1. Right-click the Windows desktop.
2. From the shortcut menu, select Properties.
3. From the Display Properties dialog box, click the Screen Saver tab.
4. Select a screen savior from the list, and then enter the automatic log off time period in the Wait box.
5. Click OK to return to the desktop.

See also

Security for FactoryTalk Batch commands on page 8
Change the logged-on user with your HMI on page 12
External security devices on page 14

Change the logged-on user with your HMI

Configure your human-machine interface (HMI) to notify FactoryTalk Batch View to change the logged on user. The HMI sends a user name and password so the FactoryTalk Batch View can attempt to log on the specified user.

A Component Object Model (COM) component was created to log on and off of FactoryTalk Batch either programatically, or by using an HMI. The following example, created in Visual Basic, is one way to implement external log-ons using the COM component.
To change the logged-on user with your HMI:

1. In Visual Basic, create a new project and add a reference to the BatchViewUser 1.0 Type Library.

2. Create a Visual Basic form that allows a user to enter a user name and a password.

3. Add two Command buttons, one for user log-ons and another to log off. Here is an example of a Visual Basic form:

4. Create a new COM object and add code to support the log on functionality.
The following code sample demonstrates how to create a COM object in Visual Basic that is used for logging on and off of FactoryTalk Batch View.

```
VB
Dim objView As BatchView.CBVUser
Private Sub Command1_Click()
    objView.Login txtUserName.Text, txtPassword.Text
End Sub

Private Sub Command2_Click()
    objView.Logout
End Sub

Private Sub Form_Load()
    Set objView = New BatchView.CBVUser
End Sub
```

5. Save the project and make an executable file.

6. Run the executable to log users on and off of FactoryTalk Batch View.

**Tip:** DDE is not a supported communication protocol with the FactoryTalk Batch suite of products.

**See also**
- Security for FactoryTalk Batch commands on page 8
- Set the automatic logoff time period on page 12
- External security devices on page 14

### External security devices

Use an external security device, such as a retinal scanner, with the FactoryTalk Batch View. You must use a \*.dll with a function that collects the user name and password. When the FactoryTalk Batch View security prompts the user for user name and password, your \*.dll is called. Your function is expected to copy the user name to <USERNAME> and the password to <PASSWORD>. The size limit on each of these is 1023 bytes.

The following is an example of a function that would return the user name and password to the FactoryTalk Batch View:

```
int <func>(char username, char password)

return code of 0 == SUCCESS
```
Any value other than 0 is an error and an entry is recorded in the Batchview.log file. If the .dll fails to load, the function is not found in the .dll, or the function causes an exception, the logon mechanism is disabled. The standard logon dialog box must then be used.

The FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor also support an external security device. The product policies used to configure an external security device for the FactoryTalk Batch Equipment Editor and FactoryTalk Batch Recipe Editor are created in the following FactoryTalk Local Directory locations.

- **System > Policies > Product Policies > Batch > Equipment Editor > Configuration > Options - External Login**
- **System > Policies > Product Policies > Batch > Recipe Editor > Configuration > Options - External Login**

See also

- [Enable an external security device](#) on page 15
- [Security for FactoryTalk Batch commands](#) on page 8
- [Change the logged-on user with your HMI](#) on page 12

### Enable an external security device

Use the FactoryTalk Administration Console to modify the required security policy settings for enabling an external security device. These policy settings are located in the FactoryTalk Local Directory.

To enable an external security device:

1. Run the FactoryTalk Administration Console and log on to the Local Directory.
2. Expand **System > Policies > Product Policies > Batch > BatchView & ActiveX > Configuration**.
3. Right-click **Options** and then select **Properties**.
4. Edit the following policy settings:
   - Enabled = True
   - Name of DLL = `<YourDLLNameWithPath>`
   - Name of Function = `<FunctionNameInDLL>`
Tip: Restart all FactoryTalk Batch components to update changes made in the FactoryTalk Directory.

See also

- External security devices on page 14
- Security for FactoryTalk Batch commands on page 8
- Change the logged-on user with your HMI on page 12

Open FactoryTalk Batch View

The FactoryTalk Batch View Single-SignOn product policy setting created in the FactoryTalk Local Directory is used in conjunction with the FactoryTalk Use single sign-on System policy setting to enable or disable the FactoryTalk single sign-on feature. Use the FactoryTalk single sign-on feature to log on just once, per directory, on a given computer.

Once logged on to the FactoryTalk Directory, all participating FactoryTalk-enabled products that run in that directory on that computer automatically use the security credentials of the FactoryTalk logged on user.

Opening the FactoryTalk Batch View when Single-SignOn is Enabled and FactoryTalk Use single sign-on is True

Opening the FactoryTalk Batch View when the Batch View Single-SignOn product policy setting is Enabled and the FactoryTalk Use single sign-on system policy setting is True, the following scenarios are possible:

Tip: FactoryTalk Windows-linked user: A user account created in Windows and then added to the FactoryTalk Directory.

- If a user is logged on to the FactoryTalk Directory, the FactoryTalk Batch View opens using the security permissions defined for that user.
- If there is no user logged on to the FactoryTalk Directory but the current Windows user is a FactoryTalk Windows-linked user, the FactoryTalk Batch View opens using the security permissions defined for the current FactoryTalk Windows-linked user. The FactoryTalk Windows-linked user is then logged on to the FactoryTalk Directory.
- If a user is not found, the Log on to FactoryTalk dialog box opens and remains open until a FactoryTalk user is authenticated or the Log on to FactoryTalk dialog box Cancel button is clicked. Canceling the Log on to FactoryTalk dialog box has the following affect:
• Only the currently displayed View window is accessible.

• The Login button is the only enabled option.

• The FactoryTalk Batch View status bar displays a message in the following format: <No authenticated user>.

Opening the FactoryTalk Batch View when Single-SignOn is disabled or FactoryTalk Use single sign-on is false

Opening the FactoryTalk Batch View when the Batch View Single-SignOn product policy setting is Disabled or when the FactoryTalk Use single sign-on system policy setting is False, the following scenarios are possible.

Tip: If the FactoryTalk Batch View Single-SignOn product policy setting is disabled, users logged on to the FactoryTalk Directory are ignored when opening the FactoryTalk Batch View.

• If the current Windows user is a FactoryTalk Windows-linked user, the application opens using the security permissions defined for the current FactoryTalk Windows-linked user.

• If the current Windows user is not a FactoryTalk Windows-linked user, then the Log on to FactoryTalk dialog box opens and remains open until a FactoryTalk user is authenticated or the Cancel button is clicked. Canceling the Log on to FactoryTalk dialog box has the following affect:

  • Only the currently displayed View window is accessible.
  
  • The Login button is the only enabled option.
  
  • The status bar displays a message in the following format: <No authenticated user>.

See also

Log into FactoryTalk Batch View on page 17

FactoryTalk Batch View security on page 7
Log into FactoryTalk Batch View

Log into the FactoryTalk Batch View using the following steps.

Tip: If the Display Configuration dialog box opens, it means that the Batch Server has been configured to allow multiple instances of the Batch View. Select the desired Batch View configuration file, and then click Start. (See Running Multiple Instances of FactoryTalk Batch View in the FactoryTalk Batch Administrator Guide for information.)

To log into FactoryTalk Batch View:

1. With the FactoryTalk Batch Server active (and the FactoryTalk Batch Phase Simulator, if running in Demo mode), click Start>All Programs>Rockwell Software>FactoryTalk Batch Suite>FactoryTalk Batch, and select View.

Tip: When the FactoryTalk Batch View connects to a running FactoryTalk Batch Server that is an incompatible version, the FactoryTalk Batch View displays the message: "The Batch View is incompatible with the current Batch Server. It is strongly recommended that you upgrade your system before continuing."

Click OK to continue using the FactoryTalk Batch View. When the message is acknowledged, the FactoryTalk Batch View remains active.

2. If the Log on to FactoryTalk dialog box opens, type your user name in the Name box and your password in the Password box. Click OK. The authenticated user displays in the FactoryTalk Batch View status bar.

3. If you enter a user not found in the FactoryTalk Directory, a Logon Message dialog box opens. Click OK to close the dialog box.

4. Click the Login button to open the Log on to FactoryTalk dialog box.

5. To close the Log on to FactoryTalk dialog box without logging on, click the Cancel button.

See also

Open FactoryTalk Batch View on page 16
FactoryTalk Batch View interface

Use the FactoryTalk Batch View to initiate and control the batch process. View a graphical representation of a running batch and its associated data. The FactoryTalk Batch View can be resized or minimized. To disable the resize and minimize features, use the System Configuration and Defaults dialog box.

See also

FactoryTalk Batch View toolbar on page 19

FactoryTalk Batch View command buttons on page 21

FactoryTalk Batch View status bar on page 22

System Configuration and Defaults dialog box on page 31
FactoryTalk Batch View toolbar

The FactoryTalk Batch View has ten different windows for viewing batch process and data. These windows can be accessed using the first ten buttons on the toolbar. As you move your cursor over any of these buttons, an information tip displays the button name.

<table>
<thead>
<tr>
<th>Button</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>📊</td>
<td>Batch List</td>
<td>View and interact with a table of the existing batches.</td>
</tr>
<tr>
<td>🌐</td>
<td>Procedure as SFC</td>
<td>View and interact with each component of the currently selected batch, in an SFC format.</td>
</tr>
<tr>
<td>📊</td>
<td>Procedure as Table</td>
<td>View and interact with each component of the currently selected batch, in a table format.</td>
</tr>
<tr>
<td>📚</td>
<td>Event Journal</td>
<td>Open and view event data for batches. All commands sent to the FactoryTalk Batch Server are recorded in the electronic batch record. This includes the user name who initiated the command.</td>
</tr>
<tr>
<td>📋</td>
<td>Unacknowledged Prompts</td>
<td>View and respond to a list of prompts that require operator input.</td>
</tr>
<tr>
<td>🗯️</td>
<td>Signature List</td>
<td>View and acknowledge the list of signatures that require signoffs.</td>
</tr>
<tr>
<td>🕵️‍♂️</td>
<td>Phase Control</td>
<td>Access all procedures and equipment for the currently loaded area model.</td>
</tr>
<tr>
<td>📚</td>
<td>Arbitration</td>
<td>View and interact with current resource allocations.</td>
</tr>
<tr>
<td>📚</td>
<td>Alarm Summary</td>
<td>Displays a list of alarm and failure messages for units and phases of the current batches.</td>
</tr>
<tr>
<td>📚</td>
<td>Phase Summary</td>
<td>Lists the status of all phases in the currently loaded area model.</td>
</tr>
</tbody>
</table>

The FactoryTalk Batch View toolbar also contains buttons that access configuration information, online help, HMI, security, and that exits the FactoryTalk Batch View.

<table>
<thead>
<tr>
<th>Button</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🕵️‍♂️</td>
<td>System Configuration and Defaults</td>
<td>Configure the display of each View window.</td>
</tr>
<tr>
<td>📚</td>
<td>Help</td>
<td>Opens online Help for the FactoryTalk Batch View.</td>
</tr>
<tr>
<td>🌐</td>
<td>Go to HMI</td>
<td>Opens a user-defined HMI (Human-Machine Interface). If the Cross Invocation function is enabled, this button allows context information regarding the selected component to be passed to an OLE Automation Server.</td>
</tr>
<tr>
<td>🎯</td>
<td>Login</td>
<td>Opens the Log on to FactoryTalk dialog box.</td>
</tr>
</tbody>
</table>
### FactoryTalk Batch View command buttons

The buttons along the right side of the FactoryTalk Batch View windows are command buttons. They represent the actions that can be performed. When changing windows in FactoryTalk Batch View, the available command buttons change based on the selected view.

The FactoryTalk Batch View command buttons are as follows:

<table>
<thead>
<tr>
<th>Button</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>🎁</td>
<td>Exit</td>
<td>Closes the FactoryTalk Batch View and returns to the Windows Desktop.</td>
</tr>
<tr>
<td>📅</td>
<td>Start Batch</td>
<td>Begin a batch that is currently on the batch list.</td>
</tr>
<tr>
<td>🕳️</td>
<td>Hold Batch</td>
<td>Hold a batch that is currently running.</td>
</tr>
<tr>
<td>⏪</td>
<td>Stop Batch</td>
<td>Permanently terminate the execution of a batch at the current step. Generally used in non-emergency situations.</td>
</tr>
<tr>
<td>✖️</td>
<td>Abort Batch</td>
<td>Permanently terminate the execution of a batch at the current step. Generally used in emergency situations.</td>
</tr>
<tr>
<td>⌛️</td>
<td>Timer Reset</td>
<td>Reset the ELAPSED_TIME in a timer step to 0 (if the timer step is configured as a COUNT_DOWN timer the REMAINING_TIME is reset to the setpoint value).</td>
</tr>
<tr>
<td>⌛️</td>
<td>Timer Complete</td>
<td>Force a timer step to complete.</td>
</tr>
<tr>
<td>🤖</td>
<td>Manual</td>
<td>Place a batch in this mode to control the execution of the remaining steps in a batch.</td>
</tr>
<tr>
<td>🕵️</td>
<td>Semi-Auto</td>
<td>Place a batch in this mode to resume the execution of all lower level steps in a batch.</td>
</tr>
<tr>
<td>📚</td>
<td>Auto</td>
<td>Give control of the running batch to the FactoryTalk Batch Server.</td>
</tr>
<tr>
<td>🕒</td>
<td>Comment</td>
<td>Add a comment to the electronic batch record for the current batch.</td>
</tr>
<tr>
<td>📜</td>
<td>Add Batch</td>
<td>Add a batch to the batch list.</td>
</tr>
<tr>
<td>🕵️</td>
<td>Bind</td>
<td>Binds a unit procedure or operation to a particular unit. The Bind button is only available in the <strong>Procedure as SFC</strong> and <strong>Procedure as Table</strong> windows.</td>
</tr>
<tr>
<td>🚫</td>
<td>Remove Batch</td>
<td>Remove a batch from the batch list.</td>
</tr>
</tbody>
</table>

See also

- FactoryTalk Batch View interface on page 19
- FactoryTalk Batch View command buttons on page 21
- FactoryTalk Batch View status bar on page 22
### Button Name Description

<table>
<thead>
<tr>
<th>Button</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Active Step Change</td>
<td>Specify the next step to run in the recipe when the current step is complete. The batch must be in Manual mode. The <strong>Active Step Change</strong> button is only available in the <strong>Procedure as SFC</strong> and <strong>Procedure as Table</strong> windows.</td>
</tr>
<tr>
<td></td>
<td>Clear All Failures</td>
<td>Clear all failures and system-generated messages listed in the <strong>Alarm Summary</strong> window.</td>
</tr>
<tr>
<td></td>
<td>Disconnect</td>
<td>Disconnect an unusable phase from the FactoryTalk Batch Server. The <strong>Disconnect</strong> button is only available in the <strong>Procedure as SFC</strong>, <strong>Procedure as Table</strong> and <strong>Manual Phase Control</strong> windows.</td>
</tr>
</tbody>
</table>

**See also**

- [FactoryTalk Batch View interface](#) on page 19
- [FactoryTalk Batch View toolbar](#) on page 19
- [FactoryTalk Batch View status bar](#) on page 22

**FactoryTalk Batch View status bar**

The bottom line of the FactoryTalk Batch View displays status information regarding the data server and the Material Server. The ID of the currently selected batch is displayed to the right of the Material Server status. The current user name and system time are also located on the status bar.

**Tip:** Some windows within the FactoryTalk Batch View consist of resizable panes. To resize a pane, click the vertical or horizontal splitter bar, which is used to separate the panes, and drag it to a new position.

**See also**

- [FactoryTalk Batch View interface](#) on page 19
- [FactoryTalk Batch View command buttons](#) on page 21
- [FactoryTalk Batch View toolbar](#) on page 19
Communication status for the FactoryTalk Batch Server and data servers

The Data Server Status indicator that is located at the bottom of each FactoryTalk Batch View window to the right of the time display indicates the communication status of the FactoryTalk Batch Server with the FactoryTalk Batch View, its data server(s), and the control system. The various status indicators and their meanings are as follows:

<table>
<thead>
<tr>
<th>Data Server Status Indicator Letter</th>
<th>Color</th>
<th>Meaning</th>
</tr>
</thead>
<tbody>
<tr>
<td>G</td>
<td>Green</td>
<td>Communication is good between the FactoryTalk Batch Server and following components:</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- FactoryTalk Batch View</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Data server(s)</td>
</tr>
<tr>
<td></td>
<td></td>
<td>- Control system</td>
</tr>
<tr>
<td>S</td>
<td>Yellow</td>
<td>Communication is suspect between the FactoryTalk Batch server and the control system. The Server and the control system handshake with each other via the setting and resetting of a tag in the control system called the Watchdog (which is usually a bit or a flag). The server sets the Watchdog and the control system resets the Watchdog. A Suspect indication means that communication between the server and the control system do not reflect that the Watchdog was reset by the control system at least once (but less than the maximum number of allowable times) within the communication timeout period. The communication timeout period defaults to ten seconds and maximum number of allowable failures defaults to five times, but these values are configurable.</td>
</tr>
<tr>
<td>B</td>
<td>Red</td>
<td>Communication is bad between the FactoryTalk Batch server and the control system. A Bad indication means that communication between the server and the control system do not reflect that the Watchdog was reset by the control system within the maximum number of allowable failures.</td>
</tr>
<tr>
<td>LOST</td>
<td>Red</td>
<td>Communication between the FactoryTalk Batch server and the FactoryTalk Batch View are lost.</td>
</tr>
<tr>
<td>L</td>
<td>Dark Red</td>
<td>Communication is completely lost between the FactoryTalk Batch server and one or more of its data servers.</td>
</tr>
</tbody>
</table>

The Data Server Status indicator displays the most severe conversation status of all the data servers.

See also

View the conversation status of all data servers on page 24.
View the Conversation Status of Data Servers dialog box

The Conversation Status of Data Servers displays an overview of the current conversation status between all OPC data servers communicating with the FactoryTalk Batch Server. The dialog box cannot be opened if conversation status between the FactoryTalk Batch View and the server has been lost. If the FactoryTalk Batch View loses conversation with the server while the Conversation Status of Data Servers dialog box is open, the dialog box closes.

<table>
<thead>
<tr>
<th>Data Server</th>
<th>Status</th>
<th>Application</th>
<th>Topic</th>
<th>Watchdog</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPC SIM</td>
<td>GOOD</td>
<td>PHASES</td>
<td></td>
<td>WATCHDOG</td>
</tr>
<tr>
<td>INSTRUCTIONBASEDSERVER</td>
<td>LOST</td>
<td>INSTRUCTIONS</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

To view the Conversation Status of Data Servers dialog box:

1. Open FactoryTalk Batch View.
2. Double-click the Data Server Status indicator.

See also

Communication status for the FactoryTalk Batch Server and data servers on page 22

Communication status for the FactoryTalk Batch Server and Material Server

In order to run material-based recipes, the FactoryTalk Batch Server and the FactoryTalk Batch Material Server must communicate with each other. In the FactoryTalk Batch Equipment Editor, set up how the system handles batches if the Batch Server loses communication with the Material Server. You can either set the Batch Server to fail and hold execution of material-based recipes. Or, you can switch to manual processing, which prompts the operator to make binding decisions until communication with the Material Server is restored. See the FactoryTalk Batch Equipment Editor User Guide for more information on setting the Material Server communication options.

The FactoryTalk Batch View displays the communication status between the Material Server and the FactoryTalk Batch Server. The Material Server Status indicator is located at the bottom of each View window to the right of the Data Server Status indicator.
The following status indicators may be displayed:

<table>
<thead>
<tr>
<th>Status</th>
<th>Color</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>MT-G</td>
<td>GOOD</td>
<td>Material Server communication in use.</td>
</tr>
<tr>
<td>MT-S</td>
<td>SUSPECT</td>
<td>Material Server communication available, but not restored.</td>
</tr>
<tr>
<td>MT-B</td>
<td>BAD</td>
<td>Material Server communication not available.</td>
</tr>
</tbody>
</table>

See also

- Process batches without the Material server on page 25
- Reestablish communications with the Material server on page 30
- Communication status for the FactoryTalk Batch server and data servers on page 22

Process batches without the Material Server

If the FactoryTalk Batch Server is configured to **Failure and Hold**, you cannot continue running batches or add new batches to the Batch List view until communication with the Material server is restored.

If the FactoryTalk Batch Server is configured to **Switch to Manual**, you can continue processing batches and add batches to the Batch List view without input from the Material server. All material-based decisions or selections that are made by the operator without communications with the Material server are reported in the batch event journal as Loss of Material server and Loss of Material Tracking events. When the material database becomes available, you must manually reconcile the event journal with the material journal and update the material database with the Amount, Lot, and Label information for material inventory.

Exercise extreme caution when operating without the Material server. Loss of communication with the Material server causes the FactoryTalk Batch Server to switch to prompt binding and presents an unfiltered list of container/phase pairs from which to select binding candidates. You must select the container/phase pair that can supply the correct material to continue processing the batch. Failure to select the correct container/phase pair could result in a hazardous situation, depending on the material.

When processing material-based batches manually, you are required to make selections without information from the material database.

See also

- Reestablish communications with the Material server on page 30
- Batch creation when the Material server is not communicating on page 26
- Prompt to bind unit requirement on page 26
Batch creation when the Material server is not communicating

When there is no communication with the Material server, the Material Server Not in Use warning is displayed in the Batch Creation dialog box for all recipes. You can create the batch or cancel and create the batch at another time when the Material server is available.

If the failure to the Material server occurs when the Batch Creation dialog box is open, the warning message does not show.

If you add a material class-based recipe to the batch list when the Material server is unavailable, the materials will not be filtered by class when the recipe is run. Instead, all materials in the material enumeration are shown.

If the failure to the Material server occurs when the Material Value Selection dialog box is open, the warning message does not show.

See also

- Prompt to bind unit requirement on page 26
- Manual unit binding on page 27
- Prompt to bind a material phase step on page 28
- Phase control on page 29
Prompt to bind unit requirement

The **Data Not Supplied by Material Server** warning displays in the **Prompt to Bind Unit Requirement** dialog box when there is no communication with the Material server.

The dialog box displays data supplied by the area model in place of the data normally supplied by the Material server. When the Material server is not available, the first available binding is serviced as though it was prompt binding.

If the failure to the Material server occurs when the **Prompt to Bind Unit Requirement** dialog box is open, the warning message does not show.

**See also**

- Batch creation when the Material server is not communicating on page 26
- Manual unit binding on page 27
- Prompt to bind a material phase step on page 28
Manual unit binding

The **Material Server Not in Use** warning displays in the **Manual Bind of Step** dialog box when there is no communication with the Material server.

![Manual Bind of Step dialog box](image)

The dialog box displays data supplied by the area model in place of the data normally supplied by the Material server.

If the failure to the Material Server occurs when the **Manual Bind of Step** dialog box is open, the warning message does not show.

See also

- Batch creation when the Material server is not communicating on page 26
- Prompt to bind unit requirement on page 26
- Prompt to bind a material phase step on page 28

Prompt to bind a material phase step

For prompt binding, the list of container/phase pairs generated from the area model is substituted for the list that would have been produced if the Material server were available. When the Material server is unavailable, material additions may not be filtered by lot or label specification.

![Prompt to Bind a Material Phase Step](image)
If you do not know which phase to select, you can look up some material information on the computer where the Material server is installed and running.

If the failure to the Material server occurs when the **Prompt to Bind a Material Phase Step** dialog box is open, the warning message does not show.

**Look up material information**

If you do not know which phase to select when there is no communication with the Material server, you can look up material information on the computer where the Material server is installed and running.

**To look up material information:**

1. Open the **Material Editor**.
2. Navigate to **Independent Containers or Materials**.
3. Double-click the desired container or material to see its properties.
4. Check the **Materials**, **Containers**, and **Inventory** tabs and compare the values to the **Unit**, **Material**, and **Amount to Charge** to determine which phase can best accommodate the material phase step.

If the failure to the Material server occurs when the **Prompt to Bind a Material Phase Step** dialog box is open, the warning message does not show.

**Phase control**

The **Material Server Not in Use** warning is displayed when there is no communication with the Material server.

For material binding in manual phase control, the list of container-phase pairs generated from the area model is substituted for the list that would have been produced if the Material server were available. The list is filtered by the container enumeration. All the data displayed is contained in the area model and the recipe. The **Label** and **Lot** fields can only be supplied by the Material server and therefore is blank when there is no communication with the Material server.
If the failure to the Material server occurs when the **Phase Control** dialog box is open, the warning message does not show.

---

**See also**

- [System Configuration and Defaults dialog box - Phase Control tab](#) on page 53
- [Phase Control window](#) on page 83

---

**Reestablish communications with the Material server**

Communication with the Material server is reestablished depending upon the FactoryTalk Batch Server configuration.

After communication with the Material server has been restored, any material-based steps that were in a HELD state must finish executing to the COMPLETE state before further processing can occur. You must acknowledge any unacknowledged prompts before the batch will continue to run.

All Loss of Material Tracking and Loss of Material Server events indicate material steps that took place without Material server communication. Use this information to manually update the Amount, Lot, and Label information for the material inventory in the material database. (See the *FactoryTalk Batch Material Editor User Guide* for more information.)

**See also**

- [Held batches](#) on page 30
- [Reestablish communication with the Material server on manual batches](#) on page 31
- [Unacknowledged Prompts window](#) on page 77

---

**Held batches**

If the FactoryTalk Batch Server is configured to **Failure and Hold**, the server automatically resumes communication with the material server when it becomes available. The material server status indicator on the right of the status bar changes to green and displays MT-G. Batches that were HELD continue to run.

**See also**

- [Reestablish communications with the Material server](#) on page 30
- [Reestablish communication with the Material server on manual batches](#) on page 31

If the FactoryTalk Batch Server is configured to **Switch to Manual**, you must reestablish communication with the Material server when it becomes available.
Reestablish communication with the Material Server on manual batches

again. When the Material Status indicator changes to Suspect (MT-S), the Material server is available and you can attempt reconnection.

To reestablish communication with the Material Server on manual batches:

1. Double-click the **Material Server Status** indicator.
2. Click the **Material Server Control** button.
3. Manually reconcile the batch event journal with the material database.
4. Select the **Reestablish Communication with Material Server** check box, and then click **OK**.

See also

[Acknowledge a prompt on page 78](#)

The **System Configuration and Defaults** dialog box opens to the tab that corresponds to the active FactoryTalk Batch View window. On each tab you can customize a specific portion of the FactoryTalk Batch View.

Each FactoryTalk Batch View window can be customized to suit the needs of a specific computer. You can specify the columns to be displayed, the column size, and the row size. You can sort and filter on one field in certain windows, and set the font for each View window. A system administrator can use the password protection option on this dialog box to limit the information that is shown. Password protection is defined in the FactoryTalk Directory using the FactoryTalk Administration Console.

See also

[Column Display area on page 56](#)

[Change the selected font in the status bar on page 57](#)

[SFC Display area on page 58](#)

[Modify the configuration of the FactoryTalk Batch View on page 31](#)
Modify the configuration of the FactoryTalk Batch View

Use the following steps as a general guideline for modifying the configuration of the FactoryTalk Batch View. Not all sections are available for all tabs.

To modify the configuration of the FactoryTalk Batch View:

1. Click the Configuration and Defaults button.
2. (optional) If Log on to Confirm User dialog box opens, enter the appropriate user information and click OK to open the System Configuration and Defaults dialog box.
3. Select the tab for the portion of the FactoryTalk Batch View that you want to customize.
4. Click OK to save any changes and close the System Configuration and Defaults dialog box. Or, click Cancel to exit without saving any changes.

See also

System Configuration and Defaults dialog box on page 31
Change the selected font in the status bar on page 57

System Configuration and Defaults dialog box - General tab

In the System Configuration and Defaults dialog box, use the General tab to customize the configuration of the FactoryTalk Batch View.

You can specify the following:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Human-Machine Interface (HMI) and Cross Invocation</td>
<td>Type the Title to be used for the HMI window. If you want to use a custom Go To HMI toolbar button, you must specify the path and file name of the bitmap. The bitmap should be 42 x 42 pixels. If you select Enable Cross Invocation, in the Invocation ProgID box, enter the identity of the Automation Server. All information concerning the HMI is disabled.</td>
</tr>
<tr>
<td>Equipment Bitmap Paths</td>
<td>Type the location of the process cell and unit bitmaps.</td>
</tr>
<tr>
<td>Server Communications</td>
<td>The FactoryTalk Batch Server communication time-out, in seconds.</td>
</tr>
<tr>
<td>BatchID Editing</td>
<td>Select to allow editing of the Batch ID. If you have written a custom program to automatically generate Batch IDs, you should disable BatchID Editing. Important: Batch IDs must be unique or the Arbitration view will not update correctly.</td>
</tr>
<tr>
<td>Split Bar Size</td>
<td>The size of the splitter bars used in the Procedure as SFC view, Procedure as Table view, Event Journal view and Phase Control view. Type the thickness, in pixels, for the vertical and horizontal splitter bars.</td>
</tr>
</tbody>
</table>
Select the **Allow Resize** check box to allow the FactoryTalk Batch View window to be resized. Select the **Allow Minimize** check box to allow the FactoryTalk Batch View window to be minimized. Type the horizontal and vertical percentage of the screen that is used by the FactoryTalk Batch View. The FactoryTalk Batch View window occupies the specified percentage of the screen, with its origin being in the upper-left corner of the screen. You are able to move the FactoryTalk Batch View window around on the desktop.

<table>
<thead>
<tr>
<th>Screen Size (%)</th>
<th>Status Bar Font</th>
</tr>
</thead>
<tbody>
<tr>
<td>Select the <strong>Allow Resize</strong> check box to allow the FactoryTalk Batch View window to be resized. Select the <strong>Allow Minimize</strong> check box to allow the FactoryTalk Batch View window to be minimized. Type the horizontal and vertical percentage of the screen that is used by the FactoryTalk Batch View. The FactoryTalk Batch View window occupies the specified percentage of the screen, with its origin being in the upper-left corner of the screen. You are able to move the FactoryTalk Batch View window around on the desktop.</td>
<td>Select the font, size, and style to use for the status bar.</td>
</tr>
</tbody>
</table>

**Tip:** To connect the FactoryTalk Batch View to a server computer other than that displayed in the **Server Communications** section, you must use the Registry Editor to edit the .NET registry key. (See the *FactoryTalk Batch Administrator Guide* for information on editing the .NET key.)

**See also**

- [System Configuration and Defaults dialog box](#) on page 31
- [Configure the General tab](#) on page 33

**Configure the General tab**

Customize the configuration of FactoryTalk Batch View using the **General** tab.

**To configure the General tab:**

1. Type the path and file name for the HMI application or select the **Enable Cross Invocation** box.
2. Type the location of the process cell and unit bitmaps.
3. Type the thickness, in pixels, for the vertical and horizontal splitter bars.
4. Select the **Allow Resize** check box to allow the FactoryTalk Batch View window to be resized.
5. Select the **Allow Minimize** check box to allow the FactoryTalk Batch View window to be minimized.
6. Type the horizontal and vertical percentage of the screen that is used by the FactoryTalk Batch View. The FactoryTalk Batch View window occupies the specified percentage of the screen, with its origin being in the upper-left corner of the screen. You are able to move the FactoryTalk Batch View window around on the desktop.

7. Click **Apply** to save the changes made on any tab.

8. Click **OK** to save the changes. (Some of the changes may not be applied until you exit and reopen the FactoryTalk Batch View.) Or, click **Cancel** to abort the changes.

See also

- System Configuration and Defaults dialog box - General tab on page 32
- System Configuration and Defaults dialog box on page 31

**System Configuration and Defaults dialog box - Signatures tab**

The **Signatures** tab in the System Configuration and Defaults dialog box corresponds to the **Signature List** view. In this tab, specify the columns to be displayed, the column header text, the column width, filter specifications for a specific column, the field and order in which the signatures are sorted, and the font to use.

If **Display Complete, Canceled, and System Canceled Signatures** is selected, the **Signature List** window displays all signatures generated against a batch until the batch is removed from the batch list. The default setting displays only the incomplete signatures in the **Signature List** window. If **Display Parameter Limits** is cleared, parameter limits are not displayed in signature dialog boxes.

See also

- Signature List window on page 79
- Configure the Signatures tab on page 34

**Configure the Signatures tab**

Customize the configuration of the **Signature List** window using the **Signatures** tab.

To configure the Signatures tab:

1. Open FactoryTalk Batch View.

2. Click the **Configuration and Defaults** button.

3. (optional) If **Log on to Confirm User** dialog box opens, enter the appropriate user information and click **OK** to open the System Configuration and Defaults dialog box.
4. Click the **Signatures** tab.

5. (optional) In the **Column Display** area, select the specific columns to display in the **Signature List** window.
   a. For each column selected, enter the column header text.
   b. For each column selected, enter the column width in pixels.

6. In the **Filtering** area, select a specific column and the column filter.

7. In the **Sorting** area, select a specific column to sort the **Signature List** window.
   a. Select **Ascending** or **Descending** for the sort order.

8. In the **Table Font** area, click **Select Font** to customize the font, font size, and font style.

9. In the **Display Options** area, select one of the display options.

10. Click **Apply** to save the changes made on any tab.

11. Click **OK** to save the changes. (Some of the changes may not be applied until you exit and reopen the FactoryTalk Batch View.) Or, click **Cancel** to abort the changes.

**See also**

   [System Configuration and Defaults dialog box - Signatures tab on page 34](#)

   [System Configuration and Defaults dialog box on page 31](#)

   [Signature List window on page 79](#)

---

**System Configuration and Defaults dialog box - Commands tab**

In the **System Configuration and Defaults** dialog box, use the **Commands** tab to customize the configuration of the commands that need confirmation and the font used for the status bar.

You can specify the following:

- The commands that have a confirmation prompt.
- The font used for the status bar.

**See also**

   [Configure the Commands tab on page 36](#)
Configure the Commands tab

Customize which commands require confirmation using the Commands tab.

To configure the Commands tab:

1. Open FactoryTalk Batch View.
2. Click the Configuration and Defaults button.
3. (optional) If Log on to Confirm User dialog box opens, enter the appropriate user information and click OK to open the System Configuration and Defaults dialog box.
4. Click the Commands tab.
5. Select the check box for each of the commands that require a confirmation prompt.
   
   When a command is enabled, you are prompted to confirm that this is the correct action to take.
6. (optional) Click the Select Font button to change the font used in the status bar.
7. Click Apply to save the changes made on any tab.
8. Click OK to save the changes, or click Cancel to abort the changes.

See also

System Configuration and Defaults dialog box - Commands tab on page 35

System Configuration and Defaults dialog box on page 31

Change the selected font in the status bar on page 57

System Configuration and Defaults dialog box - Batch List tab

The Batch List tab in the System Configuration and Defaults dialog box corresponds to the Batch List view. In this dialog box, you specify which columns to view, the column header text, the column width, on which column to filter and sort, the row height, and which font to use.

See also

Configure the Batch List tab on page 37

System Configuration and Defaults dialog box on page 31

Batch List view on page 58
Configure the Batch List tab

Customize the configuration of the Batch List window using the Batch List tab.

To configure the Batch List tab:

1. Open FactoryTalk Batch View.
2. Click the Configuration and Defaults  button.
3. (optional) If Log on to Confirm User dialog box opens, enter the appropriate user information and click OK to open the System Configuration and Defaults dialog box.
4. Click the Batch List tab.
5. (optional) In the Column Display area, select the specific columns to display in the Batch List window.
   a. For each column selected, enter the column header text.
   b. For each column selected, enter the column width in pixels.
6. In the Filtering area, select a specific column and the column filter.
7. In the Sorting area, select a specific column to sort the Batch List window.
   a. Select Ascending or Descending for the sort order.
8. In the Table Row Height box, enter the height of the table rows displayed in the Batch List window.
9. In the Table Font area, click Select Font to customize the font, font size, and font style.
10. Click Apply to save the changes.
11. Click OK.

See also

- System Configuration and Defaults dialog box - Batch List tab on page 36
- System Configuration and Defaults dialog box on page 31
- Batch List view on page 58
System Configuration and Defaults dialog box - Hierarchical Step List tab

The Hierarchical Step List tab in the System Configuration and Defaults dialog box corresponds to the Procedural Hierarchy area in the Procedure as SFC and Procedure as Table views. In this dialog box, you specify which columns to view, the column header text, the column width, the row height, and the font to use.

See also

- System Configuration and Defaults dialog box on page 31
- Configure the Hierarchical Step List tab on page 38
- Recipe Table area on page 76

Configure the Hierarchical Step List tab

Customize the configuration of the Procedural Hierarchy area in the Procedure as SFC and Procedure as Table windows using the Hierarchical Step List tab.

To configure the Hierarchical Step List tab:

1. Open FactoryTalk Batch View.
2. Click the Configuration and Defaults button.
3. (optional) If Log on to Confirm User dialog box opens, enter the appropriate user information and click OK to open the System Configuration and Defaults dialog box.
4. Click the Hierarchical Step List tab.
5. (optional) In the Column Display area, select the specific columns to display in the Procedure as SFC and Procedure as Table windows.
   a. For each column selected, enter the column header text.
   b. For each column selected, enter the column width in pixels.
6. In the Step List Font area, click Select Font to customize the font, font size, and font style.
7. In the Table Row Height box, enter the height of the table rows displayed in the Procedure as SFC and Procedure as Table windows.
8. Click Apply to save the changes.
9. Click OK.
See also

System Configuration and Defaults dialog box - Hierarchical Step List tab on page 37

System Configuration and Defaults dialog box on page 31

Procedure as SFC window on page 65

Procedure as Table window on page 74

System Configuration and Defaults dialog box - SFC View tab

The SFC View tab in the System Configuration and Defaults dialog box corresponds to the SFC area in the Procedure as SFC view. In this dialog box, you specify if the transition expressions show, the width of the lines used in the SFC, and the font to use.

See also

System Configuration and Defaults dialog box on page 31

Procedure as SFC window on page 65

SFC area on page 67

Configure the SFC View tab on page 39

Configure the SFC View tab

Customize the configuration of the Procedure as SFC window using the SFC View tab.

To configure the SFC View tab:

1. Open FactoryTalk Batch View.

2. Click the Configuration and Defaults button.

3. (optional) If Log on to Confirm User dialog box opens, enter the appropriate user information and click OK to open the System Configuration and Defaults dialog box.

4. Click the SFC View tab.

5. In the SFC Font area, click Select Font to customize the font, font size, and font style.

6. In the SFC Display area, do the following:
   a. Select to show transition expressions in the Procedure as SFC window.
   b. Enter a value for the width of the lines used in the SFC.
7. Click **Apply** to save the changes.

8. Click **OK**.

**See also**

- [System Configuration and Defaults dialog box - SFC View tab](#)
- [System Configuration and Defaults dialog box on page 31](#)

**System Configuration and Defaults dialog box - Procedure Prompts tab**

The **Procedure Prompts** tab in the **System Configuration and Defaults** dialog box corresponds to the **Prompts** tab in the Auxiliary Index view in the **Procedure as SFC** view and **Procedure as Table** view. In this dialog box, you specify which columns to view, the column header text, the column width, the priority and order in which the prompts are sorted, the row height, and the font to use.

**See also**

- [System Configuration and Defaults dialog box on page 31](#)
- [Configure the Procedure Prompts tab on page 40](#)
- [Auxiliary Index view on page 68](#)

**Configure the Procedure Prompts tab**

Customize the **Prompts** tab in the **Auxiliary Index** view of the **Procedure as SFC and Procedure as Table** windows using the **Procedure Prompts** tab.

**To configure the Procedure Prompts tab:**

1. Open **FactoryTalk Batch View**.

2. Click the **Configuration and Defaults** button.

3. (optional) If **Log on to Confirm User** dialog box opens, enter the appropriate user information and click **OK** to open the **System Configuration and Defaults** dialog box.

4. Click the **Procedure Prompts** tab.

5. (optional) In the **Column Display** area, select the specific columns to display in the **Procedure as SFC** window.

   a. For each column selected, enter the column header text.

   b. For each column selected, enter the column width in pixels.

6. In the **Sorting Priority** area, move the columns up and down based on preferred priority order.
7. In the **Sorting Order** area, select **Ascending** or **Descending** for the sort order.

8. In the **Row Height** box, enter the height of the rows displayed in the **Prompts** tab in the **Auxiliary Index** view of the **Procedure as SFC** and **Procedure as Table** windows.

9. In the **Table Font** area, click **Select Font** to customize the font, font size, and font style.

10. Click **Apply** to save the changes made on any tab.

11. Click **OK** to save the changes. (Some of the changes may not be applied until you exit and reopen the FactoryTalk Batch View.) Or, click **Cancel** to abort the changes.

**See also**

- [System Configuration and Defaults dialog box - Procedure Prompts tab on page 40](#)
- [System Configuration and Defaults dialog box on page 31](#)
- [Procedure as SFC window on page 65](#)
- [Procedure as Table window on page 74](#)
- [Auxiliary Index view on page 68](#)

**System Configuration and Defaults dialog box - Procedure Parameters tab**

The **Procedure Parameters** tab in the **System Configuration and Defaults** dialog box corresponds to the **Parameters** tab in the **Auxiliary Index** view in the **Procedure as SFC** view and **Procedure as Table** view. In this dialog box, you specify which columns to view, the column header text, the column width, the priority and order in which the parameters are sorted, the row height, and the font to use.

**See also**

- [System Configuration and Defaults dialog box on page 31](#)
- [Configure the Procedure Parameters tab on page 42](#)
- [Procedure as SFC window on page 65](#)
- [Procedure as Table window on page 74](#)
- [Auxiliary Index view on page 68](#)
### Configure the Procedure Parameters tab

Customize the **Parameters** tab in the **Auxiliary Index** view of the **Procedure as SFC and Procedure as Table** views using the **Procedure Parameters** tab.

**To configure the Procedure Parameters tab:**

1. Open **FactoryTalk Batch View**.
2. Click the **Configuration and Defaults** button.
3. (optional) If **Log on to Confirm User** dialog box opens, enter the appropriate user information and click **OK** to open the **System Configuration and Defaults** dialog box.
4. Click the **Procedure Parameters** tab.
5. (optional) In the **Column Display** area, select the specific columns to display.
   a. For each column selected, enter the column header text.
   b. For each column selected, enter the column width in pixels.
6. In the **Sorting Priority** area, move the columns up and down based on preferred priority order.
7. In the **Sorting Order** area, select **Ascending** or **Descending** for the sort order.
8. In the **Table Row Height** box, enter the height of the table rows displayed in the **Parameters** tab in the **Auxiliary Index** view of the **Procedure as SFC and Procedure as Table** windows.
9. In the **Table Font** area, click **Select Font** to customize the font, font size, and font style.
10. Click **Apply** to save the changes made on any tab.
11. Click **OK** to save the changes. (Some of the changes may not be applied until you exit and reopen the FactoryTalk Batch View.) Or, click **Cancel** to abort the changes.

**See also**

- **System Configuration and Defaults dialog box - Procedure Parameters tab** on page 41
- **System Configuration and Defaults dialog box** on page 31
System Configuration and Defaults dialog box - Procedure Reports tab

The Procedure Reports tab in the System Configuration and Defaults dialog box corresponds to the Reports tab in the Auxiliary Index view in the Procedure as SFC view and Procedure as Table view. In this dialog box, you specify which columns to view, the column header text, the column width, the priority and order in which the reports are sorted, the row height, and the font to use.

See also

System Configuration and Defaults dialog box on page 31
Configure the Procedure Reports tab on page 43
Procedure as SFC window on page 65
Procedure as Table window on page 74
Auxiliary Index view on page 68

Configure the Procedure Reports tab

Customize the Reports tab in the Auxiliary Index view of the Procedure as SFC and Procedure as Table views using the Procedure Reports tab.

To configure the Procedure Reports tab:

1. Open FactoryTalk Batch View.
2. Click the Configuration and Defaults button.
3. (optional) If Log on to Confirm User dialog box opens, enter the appropriate user information and click OK to open the System Configuration and Defaults dialog box.
4. Click the Procedure Reports tab.
5. (optional) In the Column Display area, select the specific columns to display.
   a. For each column selected, enter the column header text.
   b. For each column selected, enter the column width in pixels.
6. In the Sorting Priority area, move the columns up and down based on preferred priority order.
7. In the **Sorting Order** area, select **Ascending** or **Descending** for the sort order.

8. In the **Table Row Height** box, enter the height of the table rows displayed in the **Reports** tab in the **Auxiliary Index** view of the **Procedure as SFC and Procedure as Table** windows.

9. In the **Table Font** area, click **Select Font** to customize the font, font size, and font style.

10. Click **Apply** to save the changes made on any tab.

11. Click **OK** to save the changes. (Some of the changes may not be applied until you exit and reopen the FactoryTalk Batch View.) Or, click **Cancel** to abort the changes.

See also

- [System Configuration and Defaults dialog box - Procedure Reports tab on page 43](#)
- [System Configuration and Defaults dialog box on page 31](#)
- [Procedure as SFC window on page 65](#)
- [Procedure as Table window on page 74](#)
- [Auxiliary Index view on page 68](#)

**System Configuration and Defaults dialog box - Procedure Binding tab**

The **Procedure Binding** tab in the **System Configuration and Defaults dialog box** corresponds to the **Binding** tab in the Auxiliary Index view in the **Procedure as SFC** view and **Procedure as Table** view. In this dialog box, you specify the column header text, the column width, the sort order for the **Unit Requirements**, **Binding Requirements**, and **Binding Preferences** areas, and the font to use. You can also turn off the **Binding Requirements** and **Binding Preferences** areas if dynamic unit allocation is not in use.

See also

- [System Configuration and Defaults dialog box on page 31](#)
- [Binding Requirements and Preferences view on page 70](#)
- [Procedure as SFC window on page 65](#)
- [Procedure as Table window on page 74](#)
- [Configure the Procedure Binding tab on page 45](#)
Configure the Procedure Binding tab

Customize the Binding tab in the Auxiliary Index view of the Procedure as SFC and Procedure as Table views using the Procedure Binding tab.

To configure the Procedure Binding tab:

1. Open FactoryTalk Batch View.
2. Click the Configuration and Defaults \( \text{button} \).
3. (optional) If Log on to Confirm User dialog box opens, enter the appropriate user information and click OK to open the System Configuration and Defaults dialog box.
4. Click the Procedure Binding tab.
5. In the Column Display area, enter the column header text and column width in pixels.
6. In the Sorting Order area, select Ascending or Descending for the sort order.
7. In the Row Height box, enter the height of the rows displayed in the Binding tab in the Auxiliary Index view of the Procedure as SFC and Procedure as Table windows.
8. (optional) Select Disable Binding Req and Pref screens to disable the binding requirements and preferences screens if dynamic unit allocation is not in use.
9. In the Table Font area, click Select Font to customize the font, font size, and font style.
10. Click Apply to save the changes made on any tab.
11. Click OK to save the changes. (Some of the changes may not be applied until you exit and reopen the FactoryTalk Batch View.) Or, click Cancel to abort the changes.

See also

System Configuration and Defaults dialog box - Procedure Binding tab on page 44

Procedure as SFC window on page 65

Procedure as Table window on page 74

Auxiliary Index view on page 68
The Procedures Table tab in the System Configuration and Defaults dialog box corresponds to the Procedures Table displayed in the Recipe Table area in the Procedure as Table view. In this dialog box, you specify which columns to view, the column header text, the column width, the number of parameters to view, the row height, and the font to use.

See also

System Configuration and Defaults dialog box on page 31

Configure the Procedures Table tab on page 46

Procedure as Table window on page 74

Configure the Procedures Table tab

Customize the Procedures Table in the Recipe Table area of the Procedure as Table view using the Procedures Table tab.

To configure the Procedures Table tab:

1. Open FactoryTalk Batch View.
2. Click the Configuration and Defaults button.
3. (optional) If Log on to Confirm User dialog box opens, enter the appropriate user information and click OK to open the System Configuration and Defaults dialog box.
4. Click the Procedures Table tab.
5. (optional) In the Column Display area, select the specific columns to display.
   a. For each column selected, enter the column header text.
   b. For each column selected, enter the column width in pixels.
6. In the Table Row Height box, enter the height of the table rows displayed.
7. In the Table Font area, click Select Font to customize the font, font size, and font style.
8. Click Apply to save the changes made on any tab.
9. Click OK to save the changes. (Some of the changes may not be applied until you exit and reopen the FactoryTalk Batch View.) Or, click Cancel to abort the changes.
The **Unit Procedures Table** tab in the **System Configuration and Defaults** dialog box corresponds to the Unit Procedures Table displayed in the **Recipe Table** area in the **Procedure as Table** view. In this dialog box, you specify which columns to view, the column header text, the column width, the number of parameters to view, the row height, and the font to use.

**System Configuration and Defaults dialog box - Unit Procedures Table tab**

**Configure the Unit Procedures Table tab**

Customize the **Unit Procedures Table** in the **Recipe Table** area of the **Procedure as Table** view using the **Unit Procedures Table** tab.

**To configure the Unit Procedures Table tab:**

1. Open **FactoryTalk Batch View**.
2. Click the **Configuration and Defaults** button.
3. (optional) If **Log on to Confirm User** dialog box opens, enter the appropriate user information and click **OK** to open the **System Configuration and Defaults** dialog box.
4. Click the **Unit Procedures Table** tab.
5. (optional) In the **Column Display** area, select the specific columns to display.
   a. For each column selected, enter the column header text.
   b. For each column selected, enter the column width in pixels.
6. In the **Table Row Height** box, enter the height of the table rows displayed.

**See also**

- **System Configuration and Defaults dialog box** on page 31
- **Procedure as Table window** on page 74
- **Recipe table area** on page 76
7. In the Table Font area, click Select Font to customize the font, font size, and font style.

8. Click Apply to save the changes made on any tab.

9. Click OK to save the changes. (Some of the changes may not be applied until you exit and reopen the FactoryTalk Batch View.) Or, click Cancel to abort the changes.

See also

- System Configuration and Defaults dialog box - Unit Procedures Table tab on page 47
- System Configuration and Defaults dialog box on page 31
- Procedure as Table window on page 74
- Recipe table area on page 76

**System Configuration and Defaults dialog box - Operations Table tab**

The Operations Table tab in the System Configuration and Defaults dialog box corresponds to the Operations Table displayed in the Recipe Table area in the Procedure as Table view. In this dialog box, you specify which columns to view, the column header text, the column width, the number of parameters and reports to view, along with the height and width of the parameters and reports columns, the table row height, and the font to use.

See also

- System Configuration and Defaults dialog box on page 31
- Configure the Operations Table tab on page 48
- Procedure as Table window on page 74

**Configure the Operations Table tab**

Customize the Operations Table in the Recipe Table area of the Procedure as Table view using the Operations Table tab.

To configure the Operations Table tab:

1. Open FactoryTalk Batch View.

2. Click the Configuration and Defaults button.

3. (optional) If Log on to Confirm User dialog box opens, enter the appropriate user information and click OK to open the System Configuration and Defaults dialog box.

4. Click the Operations Table tab.
5. In the **Column Display** area, select the specific columns to display.
   
a. For each column selected, enter the column header text.

b. For each column selected, enter the column width in pixels.

6. In the **Column Display and Counts** area, select the specific columns to display.
   
a. For each column selected, enter the column header text.

b. For each column selected, enter the column width in pixels.

c. For each column selected, enter the number to view.

7. In the **Table Row Height** box, enter the height of the table rows displayed.

8. In the **Table Font** area, click **Select Font** to customize the font, font size, and font style.

9. Click **Apply** to save the changes made on any tab.

10. Click **OK** to save the changes. (Some of the changes may not be applied until you exit and reopen the FactoryTalk Batch View.) Or, click **Cancel** to abort the changes.

See also

- [System Configuration and Defaults dialog box - Procedures Table tab](#) on page 46
- [System Configuration and Defaults dialog box](#) on page 31
- [Procedure as Table window](#) on page 74
- [Recipe table area](#) on page 76

**System Configuration and Defaults dialog box - Journal tab**

The **Journal** tab in the **System Configuration and Defaults** dialog box corresponds to the **Event Journal** view. In this dialog box, you specify which columns to view, the column header text, the column width, filter specifications for up to three columns, the row height, and the font to use.

See also

- [System Configuration and Defaults dialog box](#) on page 31
- [Configure the Journal tab](#) on page 50
- [Event Journal window](#) on page 76
Configure the Journal tab

Customize the configuration of the Event Journal view using the Journal tab.

To configure the Journal tab:

1. Open FactoryTalk Batch View.
2. Click the Configuration and Defaults button.
3. (optional) If Log on to Confirm User dialog box opens, enter the appropriate user information and click OK to open the System Configuration and Defaults dialog box.
4. Click the Journal tab.
5. In the Column Display area, select the specific columns to show in the Event Journal view.
   a. For each column selected, enter the column header text.
   b. For each column selected, enter the column width in pixels.
6. In the Filtering area, select a specific column and the column filter for three columns.
7. In the Table Font area, click Select Font to customize the font, font size, and font style.
8. Click Apply to save the changes.
9. Click OK.

See also

System Configuration and Defaults dialog box - Journal tab on page 49

System Configuration and Defaults dialog box on page 31

The Unacknowledged Prompts tab in the System Configuration and Defaults dialog box corresponds to the Unacknowledged Prompts view. In this dialog box, you specify which columns to view, the column header text, the column width, filter specifications for a specific column, the field and order in which the prompts are sorted, the row height, and the font to use.

See also

System Configuration and Defaults dialog box on page 31

Unacknowledged Prompts window on page 77
Configure the Unacknowledged Prompts tab

Customize the configuration of the Unacknowledged Prompts view using the Unacknowledged Prompts tab.

To configure the Unacknowledged Prompts tab:

1. Open FactoryTalk Batch View.
2. Click the Configuration and Defaults button.
3. (optional) If Log on to Confirm User dialog box opens, enter the appropriate user information and click OK to open the System Configuration and Defaults dialog box.
4. Click the Unacknowledged Prompts tab.
5. In the Column Display area, select the specific columns to display in the Unacknowledged Prompts view.
   a. For each column selected, enter the column header text.
   b. For each column selected, enter the column width in pixels.
6. In the Filtering area, select a specific column and the column filter for three columns.
7. In the Sorting area, select a specific column to sort on and select Ascending or Descending for the sort order.
8. In the Table Row Height box, enter the height of the table rows displayed.
9. In the Table Font area, click Select Font to customize the font, font size, and font style.
10. Click Apply to save the changes.
11. Click OK.

See also

System Configuration and Defaults dialog box on page 31
System Configuration and Defaults dialog box - Unacknowledged Prompts tab on page 50
The Alarm Summary tab in the System Configuration and Defaults dialog box corresponds to the Alarm Summary view. In this dialog box, you specify which columns to view, the column header text, the column width, filter specifications for a specific column, the field and order in which the alarms are sorted, the row height, and the font to use.

See also

System Configuration and Defaults dialog box on page 31
Configure the Alarm Summary tab on page 52
Alarm Summary window on page 97

Configure the Alarm Summary tab

Customize the configuration of the Alarm Summary view using the Alarm Summary tab.

To configure the Alarm Summary tab:

1. Open FactoryTalk Batch View.
2. Click the Configuration and Defaults button.
3. (optional) If Log on to Confirm User dialog box opens, enter the appropriate user information and click OK to open the System Configuration and Defaults dialog box.
4. Click the Alarm Summary tab.
5. In the Column Display area, select the specific columns to display in the Unacknowledged Prompts view.
   a. For each column selected, enter the column header text.
   b. For each column selected, enter the column width in pixels.
6. In the Filtering area, select a specific column and the column filter for three columns.
7. In the Sorting area, select a specific column to sort on and select Ascending or Descending for the sort order.
8. In the Table Row Height box, enter the height of the table rows displayed.
9. In the Table Font area, click Select Font to customize the font, font size, and font style.
10. Click Apply to save the changes.
11. Click OK.

See also

- [System Configuration and Defaults dialog box - Alarm Summary tab](#) on page 51
- [System Configuration and Defaults dialog box](#) on page 31
- [Alarm Summary window](#) on page 97

**System Configuration and Defaults dialog box - Phase Control tab**

The **Phase Control** tab in the [System Configuration and Defaults](#) dialog box corresponds to the **Phase Control** view. In this dialog box, you specify which font to use for the following:

- Equipment View
- Phase State
- Phase Messages
- Phase Parameters Listbox
- Phase Name

Also use this dialog box to determine whether to enable or disable the **Obtain [Batch ID] From Unit** and **Allow [Batch ID] Override** options (not to be confused with Auto Batch ID options).

See also

- [System Configuration and Defaults dialog box](#) on page 31
- Configure the Phase Control tab on page 53
- Phase Control window on page 83

**Configure the Phase Control tab**

Customize the configuration of the **Phase Control** view using the **Phase Control** tab.

**To configure the Phase Control tab:**

1. Open **FactoryTalk Batch View**.
2. Click the **Configuration and Defaults** button.
3. (optional) If **Log on to Confirm User** dialog box opens, enter the appropriate user information and click **OK** to open the **System Configuration and Defaults** dialog box.
4. Click the **Phase Control** tab.

5. In the **Equipment View Font** area, click **Select Font** to customize the font, font size, and font style.

6. In the **Phase State Font** area, click **Select Font** to customize the font, font size, and font style.

7. In the **Phase Messages Font** area, click **Select Font** to customize the font, font size, and font style.

8. In the **Phase Parameters Listbox Font** area, click **Select Font** to customize the font, font size, and font style.

9. In the **Phase Name Font** area, click **Select Font** to customize the font, font size, and font style.

10. In the **Batch ID options** area, enable or disable the following options:
   a. Select **Obtain From Unit** to obtain the Batch ID from the unit.
   b. Select **Allow Override** to allow overrides of the Batch ID.

11. Click **Apply** to save the changes.

12. Click **OK**.

**See also**

- *System Configuration and Defaults dialog box - Phase Control tab* on page 53
- *System Configuration and Defaults dialog box* on page 31
- *Phase Control window* on page 83
System Configuration and Defaults dialog box - Phase Summary tab

The Phase Summary tab in the System Configuration and Defaults dialog box corresponds to the Phase Summary view. In this dialog box, you specify which columns to view, the column header text, the column width, filter specifications for a specific column, the field and order in which the phases are sorted, the row height, and the font to use.

See also

System Configuration and Defaults dialog box on page 31

Configure the Phase Summary tab on page 55

Configure the Phase Summary tab

Customize the configuration of the Phase Summary view using the Phase Summary tab.

To configure the Phase Summary tab:

1. Open FactoryTalk Batch View.
2. Click the Configuration and Defaults button.
3. (optional) If Log on to Confirm User dialog box opens, enter the appropriate user information and click OK to open the System Configuration and Defaults dialog box.
4. Click the Phase Summary tab.
5. In the Column Display area, select the specific columns to display in the Unacknowledged Prompts view.
   a. For each column selected, enter the column header text.
   b. For each column selected, enter the column width in pixels.
6. In the Filtering area, select a specific column and the column filter.
7. In the Sorting area, select a specific column to sort on and select Ascending or Descending for the sort order.
8. In the Table Row Height box, enter the height of the table rows displayed.
9. In the Table Font area, click Select Font to customize the font, font size, and font style.
10. Click Apply to save the changes.
11. Click OK.
See also

[Link to System Configuration and Defaults dialog box - Phase Summary tab on page 54]

[Link to System Configuration and Defaults dialog box on page 31]

[Link to Phase Summary window on page 98]

### Column Display area

The **Column Display** area is included in the following tabs:

- Batch List
- Procedures Table
- Phase Summary
- Operations Table
- Procedure Parameters
- Procedure Binding
- Signatures List
- Hierarchical Step List
- Unit Procedures Table
- Alarm Summary
- Procedure Prompts
- Procedure Reports
- Journal
- Unacknowledged Prompts

Several of the tabs also have filtering and sorting capabilities in the **Column Display** area.

See also

[Link to Configure column displays in the System Configuration and Defaults dialog box tabs on page 31]
Configure column displays in the System Configuration and Defaults dialog box tabs

Specify how many columns display, the column header text, the column width, filter specifications, sorting order, row height, and the font to use in the Column Display area. Options are not available in all tabs.

To configure column displays in the System Configuration and Default dialog box tabs:

1. Select the Visible? check boxes to display the columns. Clear the check boxes for those you do not want to be shown.

For each column, you can change the text for the column header and the width of the column.

2. Type the appropriate filter information using wildcard characters. A wildcard character is an asterisk (*) and can be used in conjunction with other characters. The wildcard character is placed before, after, or before and after the other characters, but it cannot be embedded within the characters.

3. Rows of data are sorted according to the field(s) selected in the Sorting area. Sorting does not change the order of the columns. Specify sort priority by selecting an item from the Column list.

4. In the Procedure Prompts, Procedure Parameters and Procedure Reports tabs, you can sort on multiple fields. Select a field and click the Up or Down button to move the field. The data is sorted according to the order of the fields in the Sorting Priority area, from top to bottom.

5. Specify the sort order by clicking Ascending or Descending.

6. Make any required row height changes by modifying the Table Row Height value.

See also

Column Display area on page 56

System Configuration and Defaults dialog box on page 31

Change the selected font in the status bar

You can change the font used in the status bar. The Select Font button that opens the Font dialog box is included on all tabs in the System Configuration and Defaults dialog box.

To change the selected font in the status bar:

1. Click Select Font from any tab.

2. Select the desired font options and click OK to close the Font dialog box.
See also

System Configuration and Defaults dialog box on page 31

Column Display area on page 56

SFC Display area

In the System Configuration and Defaults dialog box SFC View tab, you select the Show Expressions check box to indicate that you want the transition expressions to display in the SFC area of the Procedure as SFC view.

See also

System Configuration and Defaults dialog box on page 31

Procedure as SFC window on page 65

Configure the SFC View tab on page 39
Batch List view

You can use the Batch List view to create and command a batch. Once a batch is created it remains in the Batch List view until you remove it, or the FactoryTalk Batch Server is re-started with a cold boot.

You can configure the information displayed in the Batch List view.

The following data can be displayed in the Batch List:

- Batch ID
- Recipe
- Description
- Start Time
- Elapsed Time
- State
- Mode
- Process Cell
- Unit
Add a batch to the Batch List

The Batch List view provides you with an overview of batches that are complete, running and ready to run. Batches can be commanded from this window.

Tip: When a recipe is set to Release Recipe to Production, but makes reference to one or more sub-recipes or embedded procedures that are not set to Release Recipe as Step, that recipe cannot be used to create a batch. Ensure that all steps in the recipe are set to Release Recipe as Step.

To add a batch to the Batch List:

1. Click the Add Batch button. The Recipe List dialog box contains all recipes that have been approved as Release Recipe to Production. The dialog box allows you to specify the field on which to filter and sort.

2. Click Options.

3. Select a recipe in the Recipe List dialog box. Click OK.

4. If the recipe contains a material class-based phase, the Material Value Selection dialog box displays.

5. From the Material list, select the material you want to use in this particular batch, and then click Create.
6. If a batch ID is not automatically generated, type a unique **Batch ID**. The batch ID can be comprised of any character except the following:

- The defined list separator
- Spaces
- Single or double quotes
- Brackets
- Parentheses
- Percent sign (%)
- Tab character (\t)
- Carriage return character (\r)
- New line character (\n)

The maximum length of the **Batch ID** is 255 characters.

**Tip:** If the **Batch ID** box appears dimmed, you cannot change it. The option to make the **Batch ID** editable is set by the system administrator in the **System Configuration and Defaults** dialog box on the **General** tab.

7. In the **Batch Scale** box, type the percentage that you want to scale the recipe, if the batch is to be run at a percentage other than 100%.

- All phase parameters that were marked as scalable in the FactoryTalk Batch Equipment Editor and all operation or unit parameters that were marked as scalable in the FactoryTalk Batch Recipe Editor are adjusted proportionally by the percentage you enter. Any phase, unit, or operation parameters that are not marked as scalable are not adjusted; their values remain the same as the original value specified no matter how the batch is scaled.

8. (optional) Select the **Value** cell in the **Formula Values** area and type a new value if there are formula values that need to be assigned or modified.
• If this is a class-based recipe, and the recipe has a unit allocation method of **At Batch Creation** or **Operator Choice**, you must specify the unit allocation requirements.

• If the unit allocation method is **At Batch Creation**, you must choose a specific unit from a list of the available units.

• If the unit allocation method is **Operator Choice**, then you can choose from a list of available units, or you can select the **Prompt** or **First Available** unit allocation methods.

9. Click the **Bound Unit** cell in the **Unit Binding** area to activate the **Bound Unit** list. Make an appropriate selection.

**Tip:** If the recipe does not require that any formula values be assigned, the **Formula Values** area contains a single blank row. Likewise, if the recipe does not require any unit binding, the **Unit Binding** area contains a single blank row.

10. Click **Create** to add the batch to the **Batch List** window. The **Start Batch** command button is enabled.

**Important:** If control strategies are enabled for a recipe, changing the selected control strategy after the recipe is added to the batch list could corrupt that recipe.

**See also**

- **Batch List view** on page 58
- **Process batches without the Material Server** on page 25
- **Remove a batch from the Batch List** on page 62
- **Add a comment to an electronic batch record** on page 63

### Remove a batch from the Batch List

To remove a batch from the Batch List view, select the batch that is to be removed. The batch:

- Cannot be in MANUAL mode.
- Must be inactive. Inactive states are: STOPPED, COMPLETE, ABORTED or READY.

**To remove a batch from the Batch List:**

1. Select the batch to remove.

2. Click the **Remove Batch** button.
Add a comment to an electronic batch record

Comments can be added to an electronic batch record for a specific batch or all batches that are currently in the batch list. An event type of Comment is placed in the batch record of the specified batch(es). Comments are added from the Batch List window.

To add a comment to an electronic batch record:

1. In the Batch List view, select the batch that you want to add a comment.
2. Click the Comment button.
   
   The Batch Comment dialog box opens.
3. Type the appropriate comment.
4. (optional) If you want to add the comment to all batches that are currently in the batch list, select the All Batches check box.
5. Click OK to return to the Batch List view.

See also

Add a batch to the Batch List on page 60
Batch List view on page 58
Add a comment to an electronic batch record on page 63
Chapter 2

FactoryTalk Batch View windows

The FactoryTalk Batch View consists of ten different windows. The main View interface and the Batch List view are described in the section titled FactoryTalk Batch View Interface.

The remaining View windows are described in this section.

See also

- Procedure as SFC window on page 65
- Procedure as Table window on page 74
- Event Journal window on page 76
- Unacknowledged Prompts window on page 77
- FactoryTalk Batch View interface on page 19

Procedure as SFC window

The Procedure as SFC window displays the sequential function charts (SFCs) of the currently selected batch. You can watch a batch execute its steps and transitions as well as review applicable recipe comments. You can also command a batch from within the Procedure as SFC window by using the command buttons.

The Procedure as SFC window consists of three distinct sections that are separated by split bars: Procedural Hierarchy area, SFC area, and Auxiliary Index View. You can click and drag any of the splitter bars to resize the individual sections. If you drag the split bar that is between the SFC area and the command buttons to the left, the Recipe Table area is revealed.

See also

- Commands available in the Procedure as SFC window on page 66
- Command a batch on page 99
- Procedural Hierarchy area on page 67
- Auxiliary Index view on page 68
Command a batch from within the Procedure as SFC window by using the command buttons.

The following commands can be issued:

- **Start Step** (if a batch is PAUSED, then the Start Step button becomes the Resume Step button; if a batch is HELD, then the Start Step button becomes the Restart Step button)
- Hold Step
- Stop Step
- Abort Step
- Timer Reset
- Timer Complete
- Manual
- Semi-Auto
- Auto
- Bind
- Active Step Change
- Clear All Failures
Procedural Hierarchy area

The **Procedural Hierarchy** area displays the details of the entire batch. Each procedural element can be displayed with each of the following:

- Current State
- Current Unit
- Current Mode
- Key Parameters

See also

- Procedure as SFC window on page 65
- Command a batch on page 99
- Procedure as SFC window on page 65
- SFC area on page 67
- Auxiliary Index view on page 68
- Batch List view on page 58

SFC area

The **SFC** area provides information regarding the running batch. At the top of the area is the **SFC** area header. The header contains the view sizing buttons and displays the level information.

The **Recipe Level**, **Step Name**, **Step State**, and **Mode** displays for the selected step. The **Binding Info or PAUSED** text box also displays for the selected step. The information varies depending on the current step state. If the selected step is in the **PAUSED** mode, then **PAUSED** shows. Otherwise, the binding information for the selected step shows. If the step is bound, the name of the unit to which it is bound shows. If the step is not bound to a unit, then the unit requirement name shows. Steps waiting for an operator prompt, or unit availability, display **BINDING** in place of the unit name or unit requirement name.

The **SFC** structure is color coded, which allows you to easily recognize the state of a specific step. The **SFC** structure displays the step name on the first line of the step, regardless of the level being reviewed. Steps that represent unit procedures or operations display their binding information on the second line in the step. If the step is bound, then the name of the unit to which it is bound shows. If the step is
not bound, then the unit requirement name shows. The step’s current state displays in the lower left corner of the step, and the current mode displays in the lower right corner of the step.

**Tip:** Transition conditions do not update dynamically in the FactoryTalk Batch View. You must reopen the Transition Expression dialog box to update the transition data.

When a transition is paused, the SFC area displays PAUSED to the left of the transition. When a step is acquiring resources, the SFC area displays ACQUIRING to the left of the transition. To view the results of the transition expression evaluation, double-click the transition to open the Transition Expression dialog box. Transition expressions display to the right of the transition.

See also

- Procedure as SFC window on page 65
- Batch List view on page 58

**Auxiliary Index view**

The Auxiliary Index view displays information regarding the recipe, prompts, parameters, reports, arbitration, and unit binding for the step that is selected in the Procedural Hierarchy area or the Procedure as SFC view.

The Recipe Information tab displays general information about the selected control recipe such as: name, product code, version, and procedure information.

The Prompts tab displays all prompts for the selected step of the control recipe. You can also acknowledge prompts from this tab by double-clicking the prompt item.

The Parameters tab displays the parameters for the selected step of the batch. If a control strategy is enabled for the selected step, the control strategy displays at the top of the list and all parameters assigned to that control strategy are displayed below it. Parameters can be changed from this tab by double-clicking the parameter item and typing a new value in the Update Parameter Value dialog box. Deferred parameters can only be viewed. (See the FactoryTalk Batch Recipe Editor User’s Guide for more information about deferred parameters.)

If a parameter expression was used to determine the value of the parameter, you can display that information by double-clicking that parameter item to open the View Expression Parameter Value dialog box.

The expression value cannot be changed. However, there may be times when you must override the expression and assign a new value when responding to extraordinary circumstances.
The **Reports** tab displays the reports for the selected step of the batch. If a control strategy is enabled for the selected step, only reports associated with that control strategy show.

If an expression was used to determine the value of a report, you can display that information by double-clicking a report item to open the **View Expression Report Value** dialog box.

Report parameters cannot be changed by the operator because that would be considered corrupting the record of the execution of a procedure. However, there may be circumstances where the value of a report causes needed operator interaction, for example, a force transition or a recipe parameter change:

- A transition expression references the report parameter and its value is preventing the transition from firing. Use the **Force Transition** option to force the transition to fire.
- A recipe parameter is dependent on a report parameter to provide its value. If the report parameter is incorrect or undesirable, you can override the recipe parameter so that the recipe continues to run.

The **Arbitration** tab displays resource arbitration information in the context of the selected recipe element.

**Tip:** Once a batch is added to the **Batch List** view, control strategies associated with the batch cannot be reassigned. However, parameter and report values assigned to control strategies can be altered within the pre-defined ranges.

The **Binding** tab displays the unit requirements for the step that is selected in the **Procedural Hierarchy** area or the **Procedure as SFC** view. The information on the **Binding** tab includes:

- **Unit Requirements** for the Batch Procedure or the unit requirement associated with the step that is selected in the **Procedural Hierarchy** area or the **Procedure as SFC** view.
- **Binding Requirements** associated with the step selected in the **Procedure as SFC** view or **Procedure as Table** view.
- **Binding Preferences** associated with the step selected in the **Procedure as SFC** view or **Procedure as Table** view. At the unit procedure or unit operation level, there is only one unit requirement.

**See also**

- [Procedure as SFC window](#) on page 65
- [SFC area](#) on page 67
The Binding tab of the Auxiliary Index view is divided into 3 sections:

- Unit requirements
- Binding requirements
- Binding preferences

A binding requirement is an object that can be evaluated against instances of a unit class to determine which instances of the class are legal bind candidates for a unit requirement. When attempting to bind a unit requirement during recipe execution, the entire set of binding requirements defined on the unit requirement is evaluated against each potential binding candidate. Units that are unable to meet every binding requirement defined on the unit requirement are eliminated as potential binding candidates for the unit requirement.

A binding preference is an object that can be evaluated against an instance of a unit class in order to sort the legal bind targets for a unit requirement into a most preferred order.

See also

- Unit requirements on page 70
- Binding requirements on page 71
- Binding preferences on page 72

Unit requirements

Unit Requirements displays all the unit requirements for the batch procedure or the unit requirement associated with the step that is selected in the Procedural Hierarchy area or the Procedure as SFC view. At the unit procedure or operation levels, there is only one unit requirement.
See also

- Binding Requirements and Preferences view on page 70
- Procedure as SFC window on page 65
- Procedural Hierarchy area on page 67

### Binding requirements

**Binding Requirements** is in the middle section of the **Binding** tab of the Auxiliary Index view. When you select a step in the **Procedure as SFC** view or **Procedure as Table** view for a batch procedure, the associated binding requirements are displayed in the **Binding Requirements** area. If the recipe is a unit procedure or operation, the binding requirements are displayed independent of a step being selected.

**Tip:** When the **Binding** tab of the Auxiliary Index view is first shown, the Binding Requirements associated with the step selected in the **Procedure as SFC** view or **Procedure as Table** view are shown. If no step is selected, the Binding Requirements list is blank unless the recipe is a Unit Procedure or Operation.

![Binding Requirements Table](image)

The following columns are available in the **Binding Requirements** area:

- **Requirement Type:** Displays the user-configurable bind requirement type. The possible types are: Require Phase, Reject Phase, Require Attribute, Reject Attribute, and Expression.

- **Requirement Description:** Displays a text description of the requirement and depends on the type of requirement selected. The description can be the Phase Name, Unit Attribute Name, or an Expression. The contents of the Description column depends on the Type selected. The table below summarizes the contents of the Description field for each requirement Type:

<table>
<thead>
<tr>
<th>Bind Requirement Types</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Require Phase</td>
<td>Phase name</td>
</tr>
<tr>
<td>Require Attribute</td>
<td>Unit Attribute name</td>
</tr>
<tr>
<td>Reject Phase</td>
<td>Phase name</td>
</tr>
<tr>
<td>Reject Attribute</td>
<td>Unit Attribute name</td>
</tr>
<tr>
<td>Expression</td>
<td>Boolean expression</td>
</tr>
</tbody>
</table>
Binding requirement expression values

Double-clicking a binding requirement expression in the Auxiliary Index view opens the Binding Requirement Expression Values dialog box. In the dialog box, you find:

- List of units that are potential legal binding candidates for the unit requirement.
- Value indicating whether those candidates currently meet the selected binding requirement.
- List of the inputs used to evaluate the expression.

The content of the Expression is Boolean; it evaluates to Yes or No. This is the only expression allowed for a binding requirement. The values dynamically update as the inputs to the expression change.

See also

- Binding requirements on page 71
- Binding Requirements and Preferences view on page 70
- Auxiliary Index view on page 68
**Binding preferences**

On the **Binding** tab of the Auxiliary Index view, when you select a step in the **Procedure as SFC** view or **Procedure as Table** view for a batch procedure, the associated binding preferences are displayed in the **Binding Preferences** area. If the recipe is a unit procedure or operation, the binding preferences are displayed independent of a step being selected.

**Tip:** When the **Binding** tab of the Auxiliary Index view is first shown, the Binding Preferences associated with the step selected in the **Procedure as SFC** view or **Procedure as Table** view are shown. If no step is selected, the Binding Preferences list is blank unless the recipe is a Unit Procedure or Operation.

The following columns are available in the **Binding Preferences** area:

- **Preference Type:** Displays the bind preference type as configured by the user. The possible types are: Prefer Phase, Avoid Phase, Prefer Attribute, Avoid Attribute, Minimize Expression, Maximize Expression, and Boolean Expression.

- **Preference Description:** Displays a text description of the preference. The description depends on the type of preference selected. The description can be the Phase Name, Unit Attribute Name, or an Expression. The contents of the Description column depends on the Type selected. The table below lists the contents of the Description field for each preference Type:

<table>
<thead>
<tr>
<th>Bind Preference Types</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Prefer Phase</td>
<td>Phase name</td>
</tr>
<tr>
<td>Prefer Attribute</td>
<td>Unit Attribute name</td>
</tr>
<tr>
<td>Avoid Phase</td>
<td>Phase name</td>
</tr>
<tr>
<td>Avoid Attribute</td>
<td>Unit Attribute name</td>
</tr>
<tr>
<td>Expression</td>
<td>Expression that must evaluate to Yes or No</td>
</tr>
<tr>
<td>Minimize Expression</td>
<td>Expression to be minimized</td>
</tr>
<tr>
<td>Maximize Expression</td>
<td>Expression to be maximized</td>
</tr>
</tbody>
</table>

See also

- **Binding preferences expression values** on page 74
- **Binding Requirements and Preferences view** on page 70
- **Procedure as SFC window** on page 65
Binding preferences expression values

The Binding Preference Expression Values dialog box displays the result of evaluating the binding preference expression for each of the units that are potential legal binding candidates for the selected unit requirement. Double-click a binding preference in the Auxiliary Index view opens the Binding Preference Expression Values dialog box. In this dialog box, you find a:

- List of units that are potential legal binding candidates for the unit requirement.
- Value indicating the evaluation of the selected binding preference expression for that unit.
- List of the inputs used to evaluate the expression.

The content of the Expression depends on the Type selected. The following lists the contents of the Expression field for each Type:

<table>
<thead>
<tr>
<th>Bind Preference Types</th>
<th>Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boolean Expression</td>
<td>Expression that must evaluate True</td>
</tr>
<tr>
<td>Minimize Expression</td>
<td>Expression to be minimized</td>
</tr>
<tr>
<td>Maximize Expression</td>
<td>Expression to be maximized</td>
</tr>
</tbody>
</table>

See also

Binding preferences on page 72

Binding Requirements and Preferences view on page 70

Auxiliary Index view on page 68
**Procedure as Table window**

The **Procedure as Table** window displays batches in table format. You can see components of every procedural element in the currently selected batch. You can command a batch from within the **Procedure as Table** window using the command buttons.

The **Procedure as Table** window consists of three distinct sections separated by splitter bars: the **Procedural Hierarchy** view, the **Auxiliary Index** view, and the **Recipe Table** area. You can click and drag the splitter bars to resize the individual sections. If you drag the splitter bar between the **Procedural Hierarchy** view and the **Recipe Table** area to the right, the SFC area is revealed.

See also

- [Batch List view on page 58](#)
- [Procedural Hierarchy area on page 67](#)
- [Recipe table area on page 76](#)
- [SFC area on page 67](#)

**Commands available in the Procedure as Table window**

Command a batch from within the **Procedure as Table** window using the command buttons.

The following commands can be issued from within the **Procedure as Table** window:

- Start Step | Resume Step | Restart Step
- Hold Step
- Stop Step
- Abort Step
- Reset Timer Step
- Force Timer Step to Complete

![Procedure as Table window](image)
Recipe table area

The **Recipe Table** area displays batch information in a table format. One of the following tables displays based on the recipe level selected in the **Procedural Hierarchy** area:

- Procedure table
- Unit Procedure table
- Operation table

See also

- Procedure as Table window on page 74
- Auxiliary Index view on page 68
- SFC area on page 67

Event Journal window

The **Event Journal** window displays the electronic batch record for a specific batch. The electronic batch record contains information about the events that occur during batch execution or phases executed in the **Phase Control** window. FactoryTalk Batch automatically creates the electronic batch record. You can view the electronic batch record while the batch or phase is running or after processing is complete. To specify the event types included in the electronic batch record, use the FactoryTalk Batch Equipment Editor **Server Options** dialog box. (See the **FactoryTalk Batch Equipment Editor User Guide** for more information.)

See also

- View the event journal after batch execution on page 77
- Filter entries in an event journal on page 77

View information about the events that occur during batch execution in the electronic batch record.
To view the event journal after batch execution:

1. Open the Event Journal window.
   If the Filtering pane is hidden, drag the split bar to the right.
2. Click the Journal button to open the Event Data Files dialog box.
3. Select the appropriate batch and click OK.

See also

Event Journal window on page 76
Filter entries in an event journal on page 77

Filter entries in an event journal

Specify the type of entries to display on the Filtering pane in an event journal, up to three fields at a time. You can view specific event types, for example, reports, parameters, or step changes.

The wild card is an asterisk (*) character and can be used in conjunction with other characters. For example, type A* in the Filter column for State. This limits the batches that display in the Batch List view to only those batches that have been ABORTED. Or, type *ED in the Filter column for State to display only those batches whose states end in ED, such as ABORTED, STOPPED. The wild card can begin, or end, or both start and end the Filter column.

To filter entries in an event journal:

1. With the appropriate event journal opened, from the Column list in the Filtering pane, select the first field on which to filter.
2. In Filter 1, type the appropriate filter information using wild cards.
3. If required, repeat steps 1 and 2 for Column 2 and Column 3.
4. Click Refresh to display the event data that meets the specified filter criteria.

See also

View the event journal after batch execution on page 77
Event Journal window on page 76

Unacknowledged Prompts window

The Unacknowledged Prompts window displays prompts for the currently running batches. Prompts are used to alert operators when requests for phase parameter values are issued. Batch execution cannot complete until the prompt(s)
are acknowledged. When the **Unacknowledged Prompts** button flashes, click the button to open the **Unacknowledged Prompts** window.

When cross invocation is invoked from the **Unacknowledged Prompts** window, the PATH or CREATEID information (escape sequence %P) for the selected prompt is only correct for the first menu selection. You must leave the **Unacknowledged Prompts** window and return to display the information correctly. (See the *FactoryTalk Batch Administrator Guide* for more information.)

**See also**

- [Acknowledge a prompt on page 78](#)
- [Acknowledge a material phase step prompt on page 78](#)

### Acknowledge a prompt

Batch execution cannot complete until prompts are acknowledged. The prompt is removed when the acknowledgment is complete.

**To acknowledge a prompt:**

1. Select **Unacknowledged Prompts**.
2. Double-click the prompt to be acknowledged.
3. Type the appropriate value for the parameter.
4. Select **Acknowledge**.

**See also**

- [Acknowledge a material phase step prompt on page 78](#)

### Acknowledge a material phase step prompt

Prompts alert operators when requests for parameter values are issued. Control recipe execution cannot complete until the prompt(s) are acknowledged.

**To acknowledge a material phase step prompt:**

1. Open the **Prompt to Bind a Material Phase Step** dialog box and select a container and phase to use.
2. Click the **Unacknowledged Prompts** button to open the **Unacknowledged Prompts** window.
3. Double-click the prompt. The **Prompt to Bind a Material Phase Step** dialog box opens with the current prompt data shown.
4. Select a lot or storage container to bind the material to and click **OK**. The prompt is acknowledged and removed from the **Prompts List**.
See also

Acknowledging a prompt on page 78

Signature List window

The **Signature List** window displays signatures waiting to be acknowledged. Signatures are requests for signoffs that require user permissions and optional comments. When a signature is generated against a batch, you acknowledge the signature by completing the associated signoffs.

The requested user information in a signature is determined by the signature template defined in the area model.

See also

Acknowledging a signature on page 79

Additional signature request types on page 82

Signature request cancellation on page 82

Signoff failures on page 83

Acknowledge a signature

To acknowledge a signature, enter the signoff information that is required for validation. Selecting the **Close** button without completing a signature closes the dialog box and leaves the signature on the signature list.

**To acknowledge a signature:**

1. Open the **Signature** dialog box.
2. Click the flashing **Signature List** button to display the **Signature List** window.
3. Double-click the appropriate signature to open the **Signature** dialog box.
4. Enter the **User ID** and **Password** for the selected signoff.
5. Enter a comment for the selected signoff (if required).
6. Click the **Sign** button.

**Tip:** The Signer must be listed under **Security Requirements** or be a member of one of the groups in the list to complete a signoff.

7. Select the next incomplete signoff and repeat steps 3 through 6.
8. When all signoffs are Complete, click the **Close** button.
Tip: Signatures contain: Signoffs (1 to 3), Comments (Required|Optional|None), Last Signoff (Optional - marked with an asterisk).

See also

Set parameter signature overview on page 80
Command signature overview on page 80
Additional signature request types on page 82
Signature request cancellation on page 82
Signoff failures on page 83

Set parameter signature overview

If you change a parameter value in the FactoryTalk Batch View and a Param Change verification policy is configured on the recipe parameter, the Signature prompt begins flashing. Batch execution is suspended until the signature is complete or canceled. The parameter value changes when the signature is completed.

If a phase parameter has a Param Change verification policy and Deviation Limits defined, then changing the parameter to a value outside the defined limits generates two signature requests. The requests are displayed in the Generated Signatures dialog box.

- If you cancel one of the signatures in a Generated Signatures dialog box, then all other signatures within the Generated Signatures dialog box are System Canceled.
- If you close the Generated Signatures dialog box, you can use the Signature List window to complete the necessary signatures.

See also

Acknowledge a signature on page 79
Signature List window on page 79
Signature request cancellation on page 82
Signoff failures on page 83
Command signature overview

When a Command verification policy is configured in the area model, and that command is issued against a batch, a signature request is generated and the Command Signature dialog box opens.

The action associated with the command is not executed until the signature request is complete. If a command signature request is canceled, then the associated command is canceled. Command verification policies are configured in the FactoryTalk Batch Equipment Editor.

You can configure a signature request for the following commands:

- START
- RESTART
- HOLD
- MANUAL
- RESUME
- PAUSE
- SEMI-AUTO
- CLEAR FAILURES
- AUTO
- ABORT
- DISCONNECT
- TIMER RESET
- STOP
- TIMER COMPLETE

Important: ABORT is the exception. When an ABORT command is issued, the command executes. Signature completion is required to remove the ABORTED batch from the batch list. The ABORT command cannot be canceled by the user.

See also

 Acknowledge a signature on page 79

 Signature List window on page 79
Additional signature request types

The area model supports several types of command verification policies that can generate signature requests during batch execution, prior to batch execution, or when batch execution is complete. If enabled, the following Signature requests must be complete before the requested action executes:

- Remove/Reset Batch
- Active Step Change
- Force Transition
- Override Parameter Expression Value
- Clear Override
- Parameter Change
- Parameter Prompt Acknowledge
- Phase Bind
- Phase Bind Prompt Acknowledge
- Set Lots
- Set Labels
- Unit Bind
- Unit Bind Prompt Acknowledge

If an area model is configured to support parameter deviations and a recipe parameter value exceeds the defined limits, Parameter Deviation Signatures and Report Parameter Deviation Signatures are generated. If a general usage phase logic request is received during phase execution, General Usage Signature requests are generated. (See the PCD Programmer Technical Reference Guide for more information on General Usage Phase Logic Requests.)

See also

- Acknowledge a signature on page 79
- Signature List window on page 79
- Set parameter signature overview on page 80
- Command signature overview on page 80
**Signature request cancellation**

Signatures can be system canceled or user canceled. Selecting the **Cancel Signature** button on a **Signature** dialog box displays the following dialog box.

![Signature Cancellation Warning](image)

Selecting **Yes** cancels the signature and the associated action. System cancelled signatures are cancelled along with their associated actions.

See also

- [Acknowledge a signature](#) on page 79
- [Signoff failures](#) on page 83
- [Signature List window](#) on page 79

**Signoff failures**

Electronic signatures are configured and authenticated using FactoryTalk Security. Only FactoryTalk users with assigned permissions are authorized to complete signature signoffs. If a user enters an invalid user name or password, an entry is recorded in the Event Journal and the **Signoff Failure** dialog box opens. (For more information on Electronic Signature security, see the *FactoryTalk Batch Administrator Guide*.)

See also

- [Signature List window](#) on page 79
- [Acknowledge a signature](#) on page 79
- [Signature request cancelation](#) on page 82

**Phase Control window**

The **Phase Control** window shows phases relative to their physical units rather than to recipes. From this window, you can view phase-specific information and run phases manually, independently of recipes. You can manually force the phase logic to execute. For this reason, this window is very useful for testing purposes.

The **Phase Control** window displays process equipment and associated equipment phases. Process cells or units are displayed on the left side of the window. Clicking a process cell displays its units in the left side of the window. Clicking a unit displays status information for its corresponding equipment phase(s) on the right side of the window.
You can issue the following commands for a selected phase from within the **Phase Control** window:

- Start
- Pause
- Hold
- Stop
- Abort
- Reset
- Acquire
- Release
- Clear All Failures
- Disconnect

These commands affect the selected phase.

**See also**

- [Process Cell or Unit Display area on page 84](#)
- [Phase List Display area on page 86](#)
- [Phase Specific Information Display area on page 86](#)
- [Command phases in the Phase Control window on page 87](#)
Process Cell or Unit Display area

The Process Cell and Unit Display area displays unit icons and flow path lines. The name of the selected Unit or Process Cell displays at the top of this section. The button in the top right corner of this area is used as a toggle. The button is only active if a process cell or unit is selected. If the unit(s) show, the button is labeled Display Process Cells. If the process cell(s) show, the button is labeled Display Units.

Tip: If Process Cell icons are changed, or units are added in the FactoryTalk Batch Equipment Editor, you must close and restart the FactoryTalk Batch View to display the changes in the Phase Control window.

There are three view sizing buttons:

- Decrease display size
- Increase display size
- Fit display to window size

If the area model exceeds the viewable area of the Unit Display area, a scroll bar is shown that allows you to scroll through the area model. You can also click and drag the vertical split bar between this area and the Phase List Display area to enlarge or decrease the size of each area.

When a unit is selected in the Manual Phase Control window, cross invocation does not pass the PATH or CREATEID information for the escape sequence, %P. (See the FactoryTalk Batch Administrator Guide for more information.)

See also

Phase Control window on page 83
Phase List Display area on page 86
Phase Specific Information Display area on page 86
Phase List Display area

The Phase List Display area includes a button for each of the phases that belong to the unit selected in the Unit Display area. Each phase button displays the phase name, phase state, step index (when running), and phase mode. The possible modes are:

- **A** (Operator Controlled Auto)
- **PA** (Program Controlled Auto)
- **S** (Operator Controlled Semi-Auto)

The State text and Step Index box are color coded to match the state of the active phase. There is a series of three lights to the right of each phase button that indicates the current owner of the phase. If the phase is owned, the appropriate light is illuminated (Pr for Program, Opr for Operator, or Ext for External). If a phase fails, **FAILURE!!!** displays in red text below the owner lights for the failed phase.

The Next button is enabled if the unit contains more phases than can be displayed in the area. Click the Next button to display the next group of phase buttons. The Previous button is enabled after clicking the Next button. In the area below these buttons, messages and failures related to the selected phase are shown.

See also

- Phase Control window on page 83
- Process Cell or Unit Display area on page 84
- Phase Specific Information Display area on page 86

Phase Specific Information Display area

Information regarding the selected phase is displayed in the Phase Specific Information Display area, including the phase name, phase state, batch ID, and batch state. There is also an Unacknowledged Prompts area, where the prompts for the selected phase are displayed and can be acknowledged.

Use the Acknowledge button to open the Acknowledge dialog box for the selected prompt. Requests for the selected phase are displayed below the prompts.
The Mode options are used to specify the mode for the selected phase. The default mode is Auto. These options are disabled until the phase is started, or on any subsequent executions of this phase.

**AUTO mode**

You can run a phase using the AUTO mode in the Phase Control window. When a phase is in the AUTO mode, an A is displayed next to the step index on the phase button. While the phase is running, RUNNING displays on the phase button. RUNNING also displays in the Phase Specific Information Display area.

**SEMI-AUTO mode**

You can manually step through a phase using the SEMI-AUTO mode in the Phase Control window, as long as the phase logic is programmed with break points. Semi-auto processing allows you to test the functionality of the phase logic prior to releasing the phase for general use. When a phase is in SEMI-AUTO mode, an S displays next to the step index on the phase button. PAUSED displays on the phase button, in dark blue text, on the line between the phase name and its...
current state. PAUSED also displays, along with the current state, in the **Phase Specific Information Display** area.

**Manual phase control**

The phase can be run from within the **Phase Control** view, in a process called manual phase control. You must release the phase when it is no longer needed in order to make it available to other batches.

**See also**

- [Perform semi-auto phase processing](#) on page 95
- [Command a phase using AUTO mode](#) on page 91

**Acquire a phase**

To directly command a phase, you must first acquire it. A phase can only be acquired when it is not currently owned.

**To acquire a phase:**

1. Open the **Phase Control** window.
2. Click the button of the phase that you want to acquire.
3. Click the **Acquire** button, and then confirm the command.

   The **Opr** light for the selected phase turns green. The green indicates that the operator has control of the phase. The mode changes to **A**.

   **Tip:** If the phase you are attempting to acquire is running, the phase is acquired upon completion of that phase.

**See also**

- [Command phases in the Phase Control window](#) on page 87
- [Phase Control window](#) on page 83
- [Acquire a material phase](#) on page 88
Acquire a material phase

To directly manipulate a phase, you must first acquire it. A phase is only acquired when it is not currently owned.

If communication with the Material Server fails while you have the Phase Control dialog box open, all containers associated with the phase are shown. The list is not filtered by the selected material. Use extreme caution when operating without the Material Server. Failure to select the correct container or phase pair could result in a hazardous situation, depending on the material.

To acquire a material phase:

1. Open the Phase Control window.

2. Select the material-enabled phase to be acquired by clicking the corresponding Phases button.

3. Click the Acquire button, and then confirm the command.

   The Opr light for the selected phase turns green. The green indicates that the operator has control of the phase. The mode changes to A.

   Tip: If the phase you are attempting to acquire is running, the phase is acquired upon completion of that phase.

4. Click Start and then confirm the command.

5. If the phase is configured as both a material distribution and addition, from the Feed Type list, select the type you need for this phase.

   If not, the configured Feed Type for the material phase is listed and cannot be changed.

6. Select a material to be acquired from the Material list, and click OK.

   The Phase Control dialog box opens in which you bind the phase.

   Any configured containers associated with the selected material are listed in the Make a Selection to Bind Phase area.

7. Select a container that determines the requirements of the phase and the material to bind the MATERIAL step, and click OK.

   You are returned to the Phase Control window. The batch state displays DOWNLOADING.
Tip: If the phase has **Automatic Upload/Download** configured, the **Unacknowledged Prompts** area displays a description of the prompt to be acknowledged.

See also

- [Acknowledge a prompt for a phase](#) on page 90
- [Command phases in the Phase Control window](#) on page 87
- [Process batches without the Material Server](#) on page 25

### Start a phase

Start a phase in the **Phase Control** window.

**To start a phase:**

1. Open the **Phase Control** window.
2. In the **Phases** area, select the appropriate phase and acquire the phase.
3. Click **Start** to initiate the phase and then confirm the command.
4. Type a unique batch ID for the manual phase control batch.
5. (optional) If control strategies are enabled for the phase, select the appropriate control strategy from the **Control Strategy** list.
6. Click **OK**.

**Important:** Batch IDs must be unique or the **Arbitration** window will not update correctly.

See also

- [Command phases in the Phase Control window](#) on page 87
- [Acquire a phase](#) on page 88

### Acknowledge a prompt for a phase

Prompts alert operators when requests for parameter values are issued. Control recipe execution cannot complete until the prompt(s) are acknowledged.

**To acknowledge a prompt for a phase:**

1. Acquire the appropriate phase, and then start the phase.
2. In the **Phase Specific Display** area, select the prompt to be acknowledged.
3. Click **Acknowledge**.
The **Acknowledge** dialog box opens.

4. Type the appropriate value for the prompt.

5. Click **Acknowledge**.

The prompt is removed from the list when the acknowledgment completes.

**See also**

- [Command phases in the Phase Control window](#) on page 87
- [Phase Control window](#) on page 83

### Reset a phase

Reset a phase in the **Phase Control** window.

**To reset a phase:**

1. In the **Phases** area, select the phase button for the phase to be reset.

2. Click the **Reset** button.

   If the phase has been run, it must be reset prior to being released.

**See also**

- [Command phases in the Phase Control window](#) on page 87
- [Phase Control window](#) on page 83

### Release a phase

Release the phase when it is no longer needed to make it available to other batches.

**To release a phase:**

1. Select the phase button for the phase that is to be released.

2. Click **Release**. The **Opr** light for the selected phase turns gray, indicating that you no longer own the phase.

**See also**

- [Command phases in the Phase Control window](#) on page 87
- [Acquire a phase](#) on page 88
Command a phase using AUTO mode

You can run a phase using the AUTO mode in the Phase Control window. When a phase is in the AUTO mode, an A is displayed next to the step index on the phase button. While the phase is running, RUNNING displays on the phase button. RUNNING also displays in the Phase Specific Information Display area.

In the System Configuration and Defaults dialog box Phase Control tab, if the Obtain From Unit option is enabled, a default batch ID is automatically created based on the control recipe that currently has ownership of the unit in which the phase is started. If no control recipe exists, the batch ID is assigned the default BATCH_ID.

**Important:** Batch IDs must be unique or the Arbitration window will not update correctly.

In the System Configuration and Defaults dialog box Phase Control tab, if the Allow Override option is enabled, you can type a Batch ID other than the one that is automatically created. If Allow Override is not enabled, the automatically created Batch ID cannot be changed. The Allow Override option is only applicable if the Obtain From Unit option is enabled.

**To command a phase using AUTO mode:**

1. Open the Phase Control window.
2. Select the unit in the Unit Display area on which the desired phase resides. The Phases List Display area displays the status and ownership of all phases in the unit. Phases that are not running are in IDLE status.
3. In the Phases List Display area, select the desired phase.
4. Click the Acquire button and the Opr button turns green when the phase acquisition is complete.
5. Click the Start button to initiate the phase.

A confirmation dialog box displays that prompts you to confirm that you want to start the phase.
6. Click Yes to start the phase, or No to cancel the phase initiation.
7. Type a unique batch ID when prompted and click OK to continue, or Cancel to cancel the phase initiation.

The Phase button displays the current step index and the active state, both in the appropriate color. The mode displays next to the step index box as an A for operator-controlled AUTO mode. The phase transitions to the
RUNNING state and then to the COMPLETED state after all steps are executed.

**Important:** Batch IDs must be unique or the Arbitration window will not update correctly.

8. Once the phase is complete, it must be reset. Click the appropriate Phase button, and then click the Reset button.

9. Click the Release button.

The Opr button is no longer illuminated, which indicates that there is no current owner of the phase.

**See also**

Phase Control window on page 83

**Semi-auto processing**

Semi-auto processing allows you to control the execution of steps in a specific recipe level. SEMI-AUTO mode can be helpful when attempting to troubleshoot a recipe. A recipe level can be placed in SEMI-AUTO mode when the batch is in MANUAL or AUTO mode. When a batch or step is placed in SEMI-AUTO mode, all higher level steps are placed in MANUAL mode, and all lower level steps are placed in AUTO mode. You should select the lowest appropriate recipe level and then place it in SEMI-AUTO mode. The step that is active at the time the recipe level is placed in SEMI-AUTO mode completes, including its subordinate steps. Batch execution then pauses at the next transition as the conditional expression becomes TRUE. You must issue the RESUME command to initiate the next step of the selected recipe level.

For example, assume you have a recipe with two unit procedures in succession. The first unit procedure consists of three operations. The first two operations run in parallel and the third follows the two parallel operations. Once you start the batch, you place the first unit procedure within this recipe in SEMI-AUTO mode, which places the procedure in MANUAL mode. The two parallel operations execute normally, and execution pauses prior to starting the third operation. You then issue the Resume command to continue the execution of the third operation. However, execution of the batch halts prior to the second unit procedure because the procedure is in MANUAL mode. To continue execution of the second unit procedure, you must place the procedure in AUTO mode. The batch then continues to execute normally and runs to completion.
A batch or step can be placed in SEMI-AUTO mode when it is in MANUAL or AUTO mode. A batch can be placed in SEMI-AUTO mode from the following FactoryTalk Batch View windows:

- Batch List view
- Procedure as SFC view
- Procedure as Table view

See also

Perform semi-auto processing on page 94

Place a batch or step in SEMI-AUTO mode, to control the execution of steps in a specific recipe level.

To perform semi-auto processing:

1. In the Batch List view, select the batch that is to be placed in SEMI-AUTO mode.
2. Open the appropriate FactoryTalk Batch View window.
3. Select the recipe level that is to be placed in SEMI-AUTO mode.
4. Click the Semi-auto button.
5. Once you are ready to continue execution, select the step that is to be resumed.
6. Click the Resume Batch button or Resume Step button and the step transitions to the RUNNING state.
7. Repeat steps 5 and 6 until each of the subordinate steps have been run.
8. You can now place the batch in AUTO mode and continue normal execution, perform an active step change, or perform semi-auto processing at another recipe level.

See also

Semi-auto processing on page 93

SEMI-AUTO recipe mode on page 104
Semi-auto phase processing

You can manually step through a phase using the SEMI-AUTO mode in the Phase Control window, as long as the phase logic is programmed with break points. Semi-auto processing allows you to test the functionality of the phase logic prior to releasing the phase for general use. When a phase is in SEMI-AUTO mode, an S displays next to the step index on the phase button. PAUSED displays on the phase button, in dark blue text, on the line between the phase name and its current state. PAUSED also displays, along with the current state, in the Phase Specific Information Display area.

See also

Perform semi-auto phase processing on page 95

Perform semi-auto phase processing

Manually step through a phase using the SEMI-AUTO mode in the Phase Control window.

Tip: The phase logic must set the Paused bit where appropriate. These programmed break points determine where the phase pauses while in semi-auto processing.

To perform semi-auto phase processing:

1. Select the Unit in the Unit Display area on which the desired phase resides.
2. Select the desired phase button in the Phases List Display area.
3. Click Acquire.
4. Click Start to initiate the phase.
5. Click Yes to start the phase, or click No to cancel the phase initiation.
6. Type the batch ID when prompted, and click OK to continue.
7. Select the Semi-Auto option to initiate processing.
8. Respond to any unacknowledged prompts.
9. Click the Resume button to restart execution of the phase.
10. With the appropriate phase button selected, click Auto to return the phase to the AUTO mode.
11. Click the Reset button.
12. Click the Release button.
In the Arbitration window, you can view current resource allocation information, acquire available resources, and release operator-owned resources.

The Arbitration window provides a method for displaying arbitration information from different perspectives. Resources have a limited number of owners based on their definition in the area model. You can acquire a resource when it is not currently owned, or the number of current owners is less than the defined maximum number of owners. Only the current owner can release a resource.

The Arbitration window consists of the following areas:

- All Devices
- Devices List
- Command buttons
- Current Profile
- Priority List

You can acquire a resource when it is not currently owned, or the number of current owners is less than the defined maximum number of owners. Only the current owner can release a resource.

To acquire and release resources:

1. Click the appropriate List Devices option to specify the device focus.
2. Select a resource from the Devices list.
   
   Its Current Profile and Priority List shows.
3. Click the Acquire button to gain ownership of a resource.
   
   If the resource is available, you gain ownership of the resource. Otherwise, Operator is added to the resource’s Needed By list.
4. Click the Release button to release ownership of a resource.
The **Release** button is only enabled if you currently own the resource.

**See also**

[Arbitration window on page 96](#)

### Alarm Summary window

The **Alarm Summary** window displays all phase failure and phase error messages for the current batches. Each message listed includes the following columns:

- Phase
- State
- Failure
- Unit
- Batch ID

The **Alarm Summary** button flashes when an alarm is being generated and stops flashing when the **Alarm Summary** window becomes the active FactoryTalk Batch View window.

**See also**

[View alarm messages on page 97](#)

[Clear failure messages on page 97](#)

### View alarm messages

View any phase failure or error messages that have not been cleared in the **Alarm Summary** window.

**To view alarm messages:**

1. Open the **Alarm Summary** window.

**See also**

[Clear failure messages on page 97](#)

[Alarm Summary window on page 97](#)

### Clear failure messages

You can clear all failure and error messages by clicking **Clear All Failures** from any of the following windows:

- Batch List
- Procedure as SFC
- Procedure as Table
• Manual Phase Control

See also

Alarm Summary window on page 97

Phase Summary window

The Phase Summary window displays all equipment phases and their associated status information in table format. The following information displays for each phase:

• Phase Name
• State
• Mode
• Unit
• Step Index
• Owner
• Batch ID
• Message
• Failure

When a phase is selected in the Phase Summary window, cross invocation does not pass the CREATEID information for the escape sequence, %P. (See the FactoryTalk Batch Administrator Guide for more information.)

Tip: If the Phase Summary window does not display data, it is possible that the FactoryTalk Batch Server is not configured to support the Server API item used in this window. (For additional information, see the PhaseDataList item in the Server API Communication Language Reference Guide.)

See also

System Configuration and Defaults dialog box - Phase Summary tab on page 54

Configure the Phase Summary tab on page 55
Chapter 3

Command a batch

Using FactoryTalk View, you can issue batch commands from the Batch List view, the Procedure as SFC view, or the Procedure as Table view. The command buttons are only enabled for commands that are valid for the batch’s current conditions. The command buttons on the Batch List view affect the entire batch. The same command buttons on the Procedure as SFC and Procedure as Table views affect any currently selected step that is in O-AUTO mode.

Tip: A step is any recipe level within the batch:
- Procedure
- Unit Procedure
- Operation
- Phase

The defaults for the instructions given in this chapter for commanding a batch are Batch and Step Levels unless otherwise noted.

While commanding a batch, you may receive prompts that require operator input in order to continue processing the batch. These prompts display in the Unacknowledged Prompts window of the FactoryTalk Batch View.

See also

- Start a batch on page 99
- Hold a batch on page 100
- Restart a batch on page 101
- Abort a batch on page 101
- Stop a batch on page 102

Start a batch

Start a batch from the following windows.

- Batch List view
- Procedure as SFC view
- Procedure as Table view
To start a batch:

1. In the Batch List view, select the batch.

2. (optional) In the toolbar, click the Procedure as SFC button or Procedure as Table button to select the view that you want.
   a. Select the highest level of the procedure.
   b. The mode must be O-AUTO.

3. Click Start Batch or Start Step.

See also

Command a batch on page 99

Hold a batch

This command temporarily stops execution of the batch at a step. A HELD batch can be RESTARTED, STOPPED, or ABORTED. A batch can be held from the following FactoryTalk Batch View windows:

- Batch List view
- Procedure as SFC view
- Procedure as Table view

To hold a batch:

1. In the Batch List view, select the batch.

2. (optional) Click Procedure as SFC or Procedure as Table to select the view you want.
   a. Select the step you want to hold.
   b. The mode must be O-AUTO.

3. Click Hold Batch or Hold Step.

See also

Command a batch on page 99

Start a batch on page 99
Restart a batch

Restarting a batch begins the execution of the batch that is in the HELD state. Unless there is an active step change, the batch restarts with the step that was the active step at the time the batch was HELD.

To restart a batch:

1. In the Batch List view, select the batch.
2. (optional) Choose the SFC View or Table View.
   a. Select the step you want to restart.
   b. The mode must be O-AUTO.
3. Click Restart Batch or Restart Step.

See also

Command a batch on page 99
Hold a batch on page 100

Abort a batch

The ABORT command is used to permanently stop the execution of a batch at the current step. It is generally used in emergency situations, as you cannot restart an ABORTED batch.

Abort a batch from the following windows:

- Batch List view
- Procedure as SFC view
- Procedure as Table view

Tip: If a batch contains a phase that is in the UPLOAD_HELD state, the ABORT command may not abort that phase (it does abort all other phases, however). If the phase is uploading data to the FactoryTalk Batch Server, it will not abort unless the phase has transitioned to a different state. This verifies that all report values are captured. If you want to abort a phase that is in UPLOAD_HELD, you must first disconnect the phase and then abort it.

To abort a batch:

1. In the Batch List view, select the batch.
2. (optional) Click Procedure as SFC or Procedure as Table to select the view you want.
   a. Select the step that you want to abort.
b. The mode must be O-AUTO.

3. Click **Abort Batch** or **Abort Step**.

See also

[Command a batch on page 99](#)

**Stop a batch**

The STOP command is used to permanently stop the execution of a batch at the current step in a non-emergency situation. A batch can be stopped from the following FactoryTalk Batch View windows:

- Batch List view
- Procedure as SFC view
- Procedure as Table view

**Important:** You **cannot** restart a STOPPED batch.

**To stop a batch:**

1. In the **Batch List** view, select the batch.

2. (optional) In the toolbar, click **Procedure as SFC** or **Procedure as Table** to select the view you want.
   
   a. Select the step you want to stop.
   
   b. The mode must be O-AUTO.

3. Click **Stop Batch** or **Stop Step**.

See also

[Command a batch on page 99](#)

**Override an expression value**

Like deferred parameter values, the expression value cannot be changed. However, there may be times when it is imperative to override the expression and assign a new value when responding to extraordinary circumstances.

Once entered, the new expression value remains valid until you click the **Clear Override** button, which commands the expression to evaluate again. The override information is captured in the event journal for the batch.
Important: It is strongly recommended that this command be used in conjunction with a command signature or FactoryTalk Security to require authorization before an expression value can be overridden. See the FactoryTalk Batch Administrator Guide for information on configuring FactoryTalk Security on Batch View commands, or the FactoryTalk Batch Equipment Editor User Guide for information on configuring Electronic Signature Templates and Command Policies.

To override an expression value:

1. Double-click the recipe parameter expression containing the value that you want to override.
2. Click the Override button.
3. Type the override value and click OK.

See also

Command a batch on page 99

Force a transition to fire

The Force Transition command allows a transition to be fired immediately without the necessity of performing an active step change. If a transition expression references the report parameter and its value is preventing the transition from firing, use the Force Transition option to force the transition to fire.

Important: It is strongly recommended that this command be used in conjunction with a command signature or that FactoryTalk Security require authorization before a transition can be forced to fire. See the FactoryTalk Batch Administrator Guide for information on configuring FactoryTalk Security on Batch View commands, or the FactoryTalk Batch Equipment Editor User Guide for information on configuring Electronic Signature Templates and Command Policies.

To force a transition to fire:

1. Double-click the transition.
2. Click the Force Transition button and then click OK to confirm that you want the transition to fire.
3. If configured, complete the confirm or electronic signature.

See also

Override an expression value on page 102
Recipe modes

Understanding recipe modes is critical to understanding how to command a batch. The mode of a recipe defines how the recipe sequences from one step to the next when transition statements become true. There are three modes of procedural (recipe) control:

- **AUTO**: The recipe automatically progresses from step to step when its transition conditions become true.
- **SEMI-AUTO**: The recipe does not progress to the next step when its transition conditions become true until the operator commands the recipe to resume.
- **MANUAL**: The operator has complete control over the execution of steps within a recipe.

**See also**

- [AUTO recipe mode](#)
- [SEMI-AUTO recipe mode](#)
- [MANUAL recipe mode](#)

**AUTO recipe mode**

In AUTO mode, the recipe automatically progresses from step to step when its transition conditions become true. There are two types of AUTO mode:

- **O-AUTO (Operator-Auto)**: A recipe step in O-AUTO mode can be commanded by an operator.
- **P-AUTO (Procedure-Auto)**: A recipe step in P-AUTO mode can only be commanded by the higher-level recipe step to which it belongs.

The highest recipe level is usually in O-AUTO mode while all subordinate recipe steps are in P-AUTO mode. Once the operator commands the highest level of the recipe (which is in O-AUTO mode) to start, the subordinate steps further down in the recipe hierarchy are automatically put into P-AUTO mode.

**See also**

- [Recipe modes](#)
- [Place a batch in AUTO mode](#)
**SEMI-AUTO recipe mode**

In SEMI-AUTO mode, the recipe does not progress to the next step when its transition conditions become true until the operator commands the recipe to resume. In SEMI-AUTO mode, the FactoryTalk Batch Server stops normal sequencing. Once you place a recipe step in SEMI-AUTO mode, all higher-level recipe steps that contain the SEMI-AUTO step are automatically placed in MANUAL mode. When a lower-level recipe step has stopped normal sequencing, all higher-level recipe steps that contain it also stop normal sequencing.

**See also**

Recipe modes on page 103

Perform semi-auto processing on page 94

**MANUAL recipe mode**

In MANUAL mode, the operator has complete control over the execution of steps in a recipe. The FactoryTalk Batch Server stops normal sequencing. Once a recipe step is placed in MANUAL mode, the lower level steps it contains are placed in O-AUTO mode. As a result, if a step is placed in MANUAL mode, the operator can command any lower level steps it contains. Furthermore, when a step is placed in MANUAL mode, the higher-level steps that contain it are also automatically placed in MANUAL mode. When a lower-level recipe step has stopped normal sequencing, all higher-level recipe steps that contain it also stop normal sequencing.

**Steps in MANUAL mode**

When a step is placed in MANUAL mode, the steps which contain it further up in the recipe hierarchy are also automatically placed in MANUAL mode.

For example, if an operation within a procedure-level recipe is placed in MANUAL mode, the phases that the operation would normally command are automatically put into O-AUTO mode so they can be directly controlled by the operator. The unit procedure that contains the operation, which has been placed in MANUAL mode, is automatically put into MANUAL mode. The procedure that contains the unit procedure is also automatically put into MANUAL mode.

**Recipe levels in MANUAL mode**

When you place a recipe level in MANUAL mode, two things happen:

- Any higher-level recipes that contain the step you put into MANUAL mode, also go into MANUAL mode.
- Any lower-level steps contained by the recipe step in MANUAL mode go into O-AUTO mode.
A batch or step that is placed in MANUAL mode allows any lower-level active step(s) to complete, but the lower-level steps will not automatically transition to the next step.

You can place a batch or step in MANUAL mode from the following FactoryTalk Batch View windows:

- Batch List view
- Procedure as SFC view
- Procedure as Table view

For example, assume you have a recipe consisting of two unit procedures in succession. The first unit procedure consists of three operations: the first two operations run in parallel and the third follows the two parallel operations.

Once you start the batch, you place the first unit procedure in this recipe into MANUAL mode. This causes the FactoryTalk Batch software to place the procedure into MANUAL mode. Since the unit procedure that contains the two parallel operations is in MANUAL mode, the mode of the two parallel operations changes to O-AUTO and they execute normally. Execution halts prior to starting the next operation.

You must now either perform an Active Step Change to specify the next operation to execute (which you can do since the mode of the operation is O_AUTO) or return the unit procedure to AUTO mode. If you return the unit procedure to AUTO mode, it continues to execute normally, but execution stops prior to starting the second unit procedure because the procedure is still in MANUAL mode. To start execution of the second unit procedure, you must place the procedure in AUTO mode to resume normal sequencing.

See also

- Recipe modes on page 103
- Place a batch in MANUAL mode on page 109

Timer procedure modes

Timer procedures use modes the same way as Null steps. A timer step must be in O-AUTO before it can be reset or forced to complete. However, a timer step itself cannot be placed into MANUAL or O-AUTO mode. The procedure, operation, or unit procedure containing the timer step must be placed in MANUAL mode in order to change the timer step to O-AUTO.

Procedures containing timer steps observe the following rules:

- If a higher-level Procedure is placed into O-AUTO or P-AUTO, the mode of the timer step is set to P-AUTO.
• If the level immediately above the procedure containing the timer step is placed into MANUAL, the mode of the timer step changes to O-AUTO.

• If the procedure containing the timer step is placed into MANUAL, all recipe levels above the timer step procedure are placed into MANUAL.

• If the procedure containing the timer step is placed into AUTO (only possible when current mode is MANUAL), the timer step transitions to O-AUTO mode.

Tip: While the mode of a timer step has no effect on its operation, the timer step must be in O-AUTO before it can be reset or forced to complete.

See also

Recipe modes on page 103

Place a batch in MANUAL mode on page 109

MANUAL recipe mode on page 105

Place a batch in AUTO mode

A batch or step that is in SEMI-AUTO or MANUAL mode must be placed in AUTO mode to resume normal execution. A batch or step can be placed in the AUTO mode from the following FactoryTalk Batch View windows:

• Batch List view

• Procedure as SFC view

• Procedure as Table view

To place a batch in AUTO mode:

1. In the Batch List view, select the batch.

   To place the entire batch in AUTO mode, select the highest recipe level and place it in AUTO mode (for example, operation, unit procedure, or procedure).

2. Open the appropriate FactoryTalk Batch View window.

3. Select the recipe level that you want to place in AUTO mode.

4. Click Auto and the batch continues to sequence automatically.

See also

Recipe modes on page 103

AUTO recipe mode on page 104
Semi-auto processing

Semi-auto processing allows you to control the execution of steps in a specific recipe level. SEMI-AUTO mode can be helpful when attempting to troubleshoot a recipe. A recipe level can be placed in SEMI-AUTO mode when the batch is in MANUAL or AUTO mode. When a batch or step is placed in SEMI-AUTO mode, all higher level steps are placed in MANUAL mode, and all lower level steps are placed in AUTO mode. You should select the lowest appropriate recipe level and then place it in SEMI-AUTO mode. The step that is active at the time the recipe level is placed in SEMI-AUTO mode completes, including its subordinate steps. Batch execution then pauses at the next transition as the conditional expression becomes TRUE. You must issue the RESUME command to initiate the next step of the selected recipe level.

For example, assume you have a recipe with two unit procedures in succession. The first unit procedure consists of three operations. The first two operations run in parallel and the third follows the two parallel operations. Once you start the batch, you place the first unit procedure within this recipe in SEMI-AUTO mode, which places the procedure in MANUAL mode. The two parallel operations execute normally, and execution pauses prior to starting the third operation. You then issue the Resume command to continue the execution of the third operation. However, execution of the batch halts prior to the second unit procedure because the procedure is in MANUAL mode. To continue execution of the second unit procedure, you must place the procedure in AUTO mode. The batch then continues to execute normally and runs to completion.

A batch or step can be placed in SEMI-AUTO mode when it is in MANUAL or AUTO mode. A batch can be placed in SEMI-AUTO mode from the following FactoryTalk Batch View windows:

- Batch List view
- Procedure as SFC view
- Procedure as Table view

See also

Perform semi-auto processing on page 94

Perform semi-auto processing

Place a batch or step in SEMI-AUTO mode, to control the execution of steps in a specific recipe level.

To perform semi-auto processing:

1. In the Batch List view, select the batch that is to be placed in SEMI-AUTO mode.
2. Open the appropriate FactoryTalk Batch View window.
3. Select the recipe level that is to be placed in SEMI-AUTO mode.
4. Click the **Semi-auto** button.

5. Once you are ready to continue execution, select the step that is to be resumed.

6. Click the **Resume Batch** button or **Resume Step** button and the step transitions to the **RUNNING** state.

7. Repeat steps 5 and 6 until each of the subordinate steps have been run.

8. You can now place the batch in **AUTO** mode and continue normal execution, perform an active step change, or perform semi-auto processing at another recipe level.

**See also**

- [Semi-auto processing](#)
- [SEMI-AUTO recipe mode](#)

If you need to determine which step to execute next through a procedure called an Active Step Change, place a batch or step in **MANUAL** mode.

**Place a batch in **MANUAL** mode**

**To place a batch in **MANUAL** mode:**

1. In the **Batch List** view, select the batch that contains the step.

2. Open the appropriate View window (**Procedure as SFC view** or **Procedure as Table** view).

3. Select the lowest appropriate recipe level containing the steps that you want to command.

4. Click the **Manual** button.

   The batch then completes the active step and the step transitions to the **COMPLETE** state. The batch must be placed in **AUTO** mode to resume automatic execution.

**See also**

- [MANUAL recipe mode](#)
- [Recipe modes](#)
Command timer steps

If the recipe on the batch list contains one or more timer steps, you can reset the timer step or force it to complete. The batch must be in MANUAL mode, which places the timer step into O-AUTO mode.

See also

Reset a timer step on page 110
Force a timer step to complete on page 110

Reset a timer step

When a timer step in the RUNNING or HELD state receives the Timer-Reset command, the ELAPSED_TIME report will be reset to 0. If the timer step is configured as a COUNT_DOWN timer, the REMAINING_TIME report will be reset to the setpoint value. If the Timer-Reset command is issued and the command is no longer valid, for example, due to a change of state, the command is ignored.

To reset a timer step:

1. In the Batch List view, select the batch.
2. In the Procedure as SFC view or Procedure as Table view, select the timer step.
3. Click Timer-Reset.

See also

Command timer steps on page 109
Force a timer step to complete on page 110
Auxiliary Index view on page 68

Force a timer step to complete

When a timer step is in the RUNNING state and receives a Timer-Complete command, the timer step will transition to the COMPLETE state. If a Timer-Complete command is issued and the command is no longer valid, for example, due to a change of state, the command is ignored.

To force a timer step to complete:

1. In the Batch List view, select the batch.
2. In the Procedure as SFC view or Procedure as Table view, select the timer step.
3. Click Timer-Complete.
See also

Command timer steps on page 109

Reset a timer step on page 110

Change the setpoint of a timer step

You can change the setpoint of a timer step whether or not it is currently active and regardless of its state.

To change the setpoint of a timer step:

1. Select the Timer step in the Procedure as SFC view or Procedure as Table view, and then click the Details button.
   
The Properties dialog box opens.

2. Select the Parameters tab to view the Hold Behavior, Setpoint (COUNT_DOWN timers only), and Timer Type for the selected Timer step.


4. Type a new setpoint value in the Value box, and then click OK.
   
Tip: The value entered must be within the displayed minimum and maximum values for the parameter.

5. Click OK to close the Properties dialog box and apply the change.

See also

Command timer steps on page 109

Force a timer step to complete on page 110

Reset a timer step on page 110

Perform an active step change

Active step changes cannot be performed at the batch level. They can only be done on the following:

- Step level on procedures
- Unit procedures
- Operations
- Phases
You can change the active step in a batch if the recipe level that contains the steps requiring the active step change is in MANUAL mode.

For example, if you want to command phase steps, the operation must be in MANUAL mode. Steps that are in the HELD state cannot be removed as the active step. An active step change can be performed in the **Procedure as SFC** view or **Procedure as Table** view, although it is recommended that it is performed within the **Procedure as SFC** view.

**To perform an active step change:**

1. In the **Batch List** view, select the batch.
2. Open the **Procedure as SFC** view or **Procedure as Table** view.
3. Select the procedure in the **Procedural Hierarchy** view section.
4. Select the recipe level that contains the active step and place this recipe level in MANUAL mode.
5. If the **Procedure as SFC** view is the active window, select the recipe level that contains the active step to be changed. Place this recipe level in MANUAL mode.

   **Tip:** Double-click the selected step in the SFC. This displays the steps within this recipe level.

6. If the **Procedure as Table** view is the active window, select the level that requires the active step to be changed. Place this recipe level in MANUAL mode.
   a. Switch to the **Procedure as SFC** view.
   b. Double-click the selected step in the SFC. This displays the steps within this recipe level.
   c. Switch back to the **Procedure as Table** view.
7. If a step to activate is a material phase step, you must first bind the step to a phase before doing the active step change. Select each material step that you want to activate and perform a manual bind.
8. Once the current active steps are in a quiescent state (COMPLETE, STOPPED, or ABORTED), click the **Active Step Change** button.
9. A new set of command buttons displays: **Execute ASC** and **Cancel ASC**.

10. Position the hand cursor over the quiescent step(s) and click to deactivate them.

11. Using the Hand cursor, click the new step(s) to activate.

12. Do one of the following:

   - Select **Execute ASC** to accept the active step change.
   - Select **Cancel ASC** to cancel the active step change.

13. Click **Yes** on the confirmation message.

   **Tip:** If you did not bind a material phase step to a phase, you are prompted to bind the step, and then you must begin the active step change again.

14. Select the new active step in the **Procedural Hierarchy** view section; it is in the READY state and in O_AUTO mode.

15. Click **Start Step** to start the step.

16. Place the batch in AUTO mode to complete normal execution.

**See also**

- Perform manual binding on page 114
- Place a batch in MANUAL mode on page 109
- Place a batch in AUTO mode on page 107

### Manual binding

Manual binding is the process of binding a unit, or rebinding a previously bound unit, to a step within a control recipe without being prompted. Manual binding is performed from within the **Procedure as SFC** view or **Procedure as Table** view or from the **ProcedureView** ActiveX control.

For units, the operator can select **First Available**, **Prompt** or one of a list of units for binding. For material phases, the operator can select **Automatic**, **Prompt** or any one of a list of container/phase/lot entries.
Tip: All other steps within the recipe that are associated with the same unit requirement are bound to the same unit.

The following criteria are required to perform manual binding:

- The step is associated with a unit class in the recipe equipment requirements.
- The step represents the highest recipe level wholly contained within the unit. (For example, one step in the recipe represents a unit procedure that is associated with the Premixer unit class, and another step that is associated with the same Premixer unit class is an operation. As long as the balance of the manual binding requirements are met, then you can manually bind the unit procedure, but not the operation, because the unit procedure is the highest recipe level associated with the unit.)
- The step is inactive or has a binding status of BINDING.
- More than one legal binding choice exists for the unit that is associated with the step.

See also

Perform manual binding on page 114

Perform manual binding

Manual binding is the process of binding a unit, or rebinding a previously bound unit, to a step within a control recipe without being prompted to do so.

Tip: This procedure applies to steps (not batches) and can only be performed at the step level.

To perform manual binding:

1. In the Batch List view, select the batch that requires binding.
2. Open the Procedure as SFC or Procedure as Table view.
3. To bind a unit to a step, you must select the unit procedure in either the SFC area, Recipe Table area, or the Procedural Hierarchy area.
4. Click Bind.
5. Select the appropriate binding option and click OK.

See also

Manual binding on page 113
Material binding

Material binding is the process of binding activated material-enabled phase steps to material phases. Or, the process of rebinding a previously bound material to a step within a control recipe without being prompted. Material phase binding must be performed from within the Procedure as SFC or Procedure as Table view or the ProcedureView ActiveX control.

Tip: This procedure applies to steps (not batches) and can only be performed at the Step level.

The following criteria are required to perform manual binding:

- The step is associated with a material in the recipe.
- The step is inactive or has a binding status of BINDING.

See also

Perform material binding on page 115

Perform material binding

Bind active material-enabled phase steps to material phases.

Tip: This procedure applies to steps (not batches) and can only be performed at the step level.

To perform material binding:

1. From the Batch List view, select the batch that requires binding.
2. Open the Procedure as SFC or Procedure as Table view.
3. To bind a material to a step, you must select the material step in either the SFC area, Recipe Table area, or the Procedural Hierarchy. If the manual binding criteria have been met, the Bind command button is enabled.
4. Click the Bind button, and the Prompt to Bind a Material Phase Step dialog box displays with a list of legal binding options.
5. Select the material specification that determines the lot or storage container to bind the material step. Then click OK.

See also

Material binding on page 114
Prompted binding

Prompted binding is the process of binding a unit to a step after a prompt is generated. Only unit class requirements that are defined with the Prompted Binding method generate a prompt, including those that are defined as such at the time the batch is created.

The execution of the batch pauses when the transition above the unbound step becomes True. A prompt displays in the Unacknowledged Prompts view and on the Prompts tab of the Auxiliary Index view in the Procedure as SFC view or Procedure as Table view.

**Tip:** This procedure applies to unit steps and can only be performed at the Unit Step level.

See also

Perform prompted binding on page 116

Perform prompted binding

Bind a unit to a step after a prompt is generated.

**Tip:** All other steps within the procedure that are associated with the same unit requirement are bound to the same unit.

**To perform prompted binding:**

1. In the Batch List view, select the batch that requires binding.
2. Locate the prompt in the Unacknowledged Prompts view, or in the Prompts tab of the Auxiliary Index view in the Procedure as SFC view.
3. Double-click the prompt.
4. Select the unit to bind to the step, and click OK.

See also

Prompted binding on page 115

Unacknowledged Prompts window on page 77

Disconnect a phase

When a phase becomes unusable, the Disconnect command can be used to disconnect the FactoryTalk Batch Server from the unusable phase. The Disconnect command can be executed on any step in O_AUTO or S_AUTO mode that is connected to a phase and that is in any state other than NOT_CONNECTED (READY). This applies to a phase that is under external control, has a failure, is PAUSED, or is HELD.

The Disconnect button is only available in the Procedure as SFC view, Procedure as Table view, and Phase Control view. When disconnecting from a
phase while in the **Manual Phase Control** view, the affected batch is removed from the **Batch List** view.

**To disconnect a phase:**

1. In the **Procedure as SFC**, **Procedure as Table** or **Phase Control** view, select the phase associated with the phase to be disconnected.

2. Click **Disconnect**, and then confirm the disconnect.

| Important | Reset the phase to an IDLE state before using it again with the FactoryTalk Batch Server. |

See also

[Reset a phase on page 91](#)
Cross invocation

Cross invocation is the process of passing pre-configured context information to an automation server. This information is relative to the element selected in the FactoryTalk Batch View application. When you customize the FactoryTalk Batch View in the System Configuration and Defaults option to enable cross invocation and identify the automation server by the Invocation ProgID, a shortcut menu becomes available by clicking the Go To HMI button.

The items in the shortcut menu are formatted using cross invocation strings associated with the equipment resource to which the current selection in the FactoryTalk Batch View corresponds. These cross invocation strings also contain the context data that is passed to the automation server. (See the FactoryTalk Batch Administrator Guide for more information on cross invocation strings and the context data that can be configured.)

The following are the elements that can be selected for cross invocation from each of the FactoryTalk Batch View windows.

<table>
<thead>
<tr>
<th>Batch View Window</th>
<th>Available Elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch List</td>
<td>• Batch</td>
</tr>
</tbody>
</table>
| Procedure as SFC or Procedure as Table | • Batch  
|                                   | • Recipe Phase                                         |
|                                   | • Operation                                             |
|                                   | • Unit Procedure                                        |
| Event Journal                     | • Batch                                                |
| Unacknowledged Prompts            | • Unacknowledged Prompts (not binding requests)        |
| Phase Control                     | • Process Cell                                          |
|                                   | • Unit                                                  |
|                                   | • Equipment Phase                                       |
| Arbitration                       | • Batch                                                |
|                                   | • Process Cell                                          |
|                                   | • Unit                                                  |
|                                   | • Equipment Phase                                       |
|                                   | • Operation                                             |
|                                   | • Unit Procedure                                        |
| Alarm Summary                     | • N/A                                                   |
| Signature List                    | • N/A                                                   |
| Phase Summary                     | • Equipment Phase                                       |

See also

Perform cross invocation on page 118
Perform cross invocation

Customize the FactoryTalk Batch View in the System Configuration and Defaults option to enable cross invocation and identify the automation server by the Invocation ProgID.

**Tip:** Check with your system administrator to determine what context data is configured for each of the shortcut menu items displayed on your system.

**To perform cross invocation:**

1. In the FactoryTalk Batch View window, select the available element, and click the Go To HMI button.

2. Select a menu item or sub-menu item to send context data to the Automation Server.

**See also**

- Cross invocation on page 118
- Command a batch on page 99
# Troubleshoot batch issues

When running batches, you may occasionally encounter situations that are difficult to resolve. This section provides solutions to some of the problems you may encounter.

<table>
<thead>
<tr>
<th>Issue</th>
<th>Solution</th>
</tr>
</thead>
</table>
| The following error displays when running a batch: **Recipe SFC Requires Illegal Use of Dedicated Resources.** As a result, the batch goes into a HELD state. | Parallel steps requesting the same phase cannot occur just after an AND divergence.  
• Abort or Stop the batch. Remove the batch from the batch list.  
• Revise the recipe to include a null procedure before each of the parallel steps after the AND divergence.  
• Create and run a batch using the new recipe. (See the FactoryTalk Batch Recipe Editor User Guide for information on adding null phases to a recipe.) |
| A batch seems to be running, but it is not progressing as it should. No errors are generated. The status is RUNNING. | Parallel steps requesting the same phase cannot occur just before an AND convergence.  
• Abort or Stop the batch. Remove the batch from the batch list.  
• Revise the recipe to include a null procedure after each of the parallel steps before the AND convergence.  
• Create and run a batch using the new recipe. (See the FactoryTalk Batch Recipe Editor User Guide for information on adding null phases to a recipe.) |
| A failure is generated and the batch is held because the FactoryTalk Batch Server selected an equipment phase under EXTERNAL control to bind to during batch execution. | Phases put into an external state can still be viewed as binding candidates by the FactoryTalk Batch Server. (The external state is a tag in the controller that prevents the Server from controlling the phase.) To remove a phase from being viewed as a binding candidate, you must take ownership of the phase. (For more information on the **External** attribute, see the FactoryTalk Batch PCD Programmer Technical Reference Guide.) |

See also

- [FactoryTalk Batch View windows](#) on page 65
- [Command a batch](#) on page 99
- [Acquire a phase](#) on page 88
Index

A
abort a batch 97
acknowledge a prompt 75
material phase step 76
acknowledge a signature 77
acquire
  a material phase 85
  a phase 85
  resources 92
active step change 106
add a batch 58
alarm summary
  window 93
AND convergence-troubleshoot 115
AND divergence-troubleshoot 115
arbitration
  window 92
auto
  mode 102
  phase processing 88

B
batch
  comment 61
  mode 57
batch creation
  without material server 25
batch scale 58
Batch Server
  incompatible version 17
  batch state 57

C
cancel a signature
  system canceled 80
clear failure messages 93
clear override 98
command
  phases 84
command buttons 21
  abort 21
  add 21
  auto 21
  clear all failures 21

comment 21
disconnect 21
hold 21
manual 21
remove 21
semi-auto 21
start 21
stop 21
timer-complete 21
timer-reset 21
command verification policies 79
comment 61
communication status 22
  indicators 22
configuration and defaults
  setting 31
create a batch 58
cross invocation 112
  enable 32
  manual phase control window 81
  phase summary window 94
  unacknowledged prompts window 75
current profile 92

D
Data Servers
  conversion status 23
device focus 92
disconnect phase 111

E
error messages 93
event journal
  filter criteria 75
  open 74
  sort criteria 75
exit from View 19

F
failure messages 93
filters 55, 75

G
generated signatures 77
H

HMI
  Go To 32
  hold a batch 96

I

incompatible
  Batch Server 17

L

list devices 92
log on to Batch View 19
login 17

M

manual
  mode 104
  phase control 84
  phase control, material-enabled phase 85
manual unit binding
  without Material Server 27
material phase
  acquire 85
Material Server
  communication status 24
  outage 24
  reestablish communications 30
material value selection
  without Material Server 25
messages
  clear failure 93
  view 93
minimize the View 32
mode 57
  auto 102
  manual 104
  semi-auto processing 90
mode column 57
multiple instances
  Batch View 17

O

open the View 17
override 124

P

parallel steps 115
password 17
phase control
  window 81
  without Material Server 29
phase list display area 83
phase specific information display area 83
phase summary
  window 94
phases
  acknowledge phase prompt 87
  acquire 85, 92
  command 84
  control 84
  release 88
  reset 87
  start 86
priority list 92
prompt
  acknowledge 75
  prompt binding
    without Material Server 26, 28
  prompted binding 111

R

recipe list dialog box 58
release
  phase 88
  resources 92
remove a batch 61
reset phase 87
restart the View 32

S

scale 58
security 7
  commands 7
  log on 7
semi-auto phase processing 91
semi-auto processing 90
setpoint
change 106
SFC area
  work in the 65
SFC View
  window 63
signature list
  window 76
signature requests
  active step change 79
  clear override 79
  force transition 79
  general usage signature 79
  override expression value 79
  parameter change 79
  parameter deviation signatures 79
  parameter prompt acknowledge 79
  phase bind 79
  phase bind prompt acknowledge 79
  remove batch 79
  report parameter deviation signatures 79
  reset batch 79
  set all label 79
  set all lot 79
  timer complete 78
  timer reset 78
  unit bind 79
  unit bind prompt acknowledge 79
signature template 76
Single-SignOn View 16
SQL wildcards 55, 75
start
  a batch 95
  a phase 86
stop
  batch 21

T
Table
  window 72
timer step
  change the setpoint 106
  complete 105
  reset 105
toolbar-View 19

U
unacknowledged prompts
  window 75

V
View
  configure 31
    alarm summary 51
    Batch List view 36
    Cross Invocation 32
    event journal 49
    Hierarchical Step List 37
    HMI 32
    operations table 47
    phase control 52
    phase summary 53
    procedure binding 44
    procedure parameters 41
    procedure prompts 39
    procedure reports 42
    procedures table 45
    SFC view 38
    Signature List 34
    unacknowledged prompts 50
    unit procedures table 46
login 17
View interface 19
View security 7
  FactoryTalk Security 7
  Single-SignOn 16
View windows
  alarm summary 93
  arbitration 92
  Batch List 63
  Batch List view 57
  phase control 81
  phase summary 94
  SFC View 63
  signatures 76
  Table View 72
  unacknowledged prompts 75

W
wildcard character 55
Legal Notices

Copyright notice

© 2017 Rockwell Automation Technologies, Inc. All rights reserved. Printed in USA.

This document and any accompanying Rockwell Software products are copyrighted by Rockwell Automation Technologies, Inc. Any reproduction and/or distribution without prior written consent from Rockwell Automation Technologies, Inc. is strictly prohibited. Please refer to the license agreement for details.

End User License Agreement (EULA)

You can view the Rockwell Automation End-User License Agreement ("EULA") by opening the License.rtf file located in your product's install folder on your hard drive.

Trademark Notices

Allen-Bradley, CompactLogix, ControlLogix, Connected Components Workbench, eProcedure, FactoryTalk, GuardLogix, GuardPLC, Micro800, MicroLogix, MobileView, MobileView Guard, PanelBuilder, PanelView, PhaseManager, PlantLink, PlantPAX, PLC-2, PLC-3, PLC-5, PowerMonitor, RSBizWare, RSBizWare BatchHistorian, RSBizWare Historian, RSBizWare MaterialTrack, RSLinx, Rockwell Automation, Rockwell Software, SLC, SoftLogix, Studio 5000, and XMare trademarks of Rockwell Automation, Inc.

Any Rockwell Automation software or hardware not mentioned here is also a trademark, registered or otherwise, of Rockwell Automation, Inc.

For a complete list of products and their respective trademarks, go to:
http://www.rockwellautomation.com/rockwellautomation/legal-notices/overview.page?#tab4
http://www.rockwellautomation.com/rockwellautomation/legal-notices/overview.page?#tab4

Warranty

This product is warranted in accordance with the product license. The product's performance may be affected by system configuration, the application being performed, operator control, maintenance, and other related factors. Rockwell
Automation is not responsible for these intervening factors. The instructions in this document do not cover all the details or variations in the equipment, procedure, or process described, nor do they provide directions for meeting every possible contingency during installation, operation, or maintenance. This product’s implementation may vary among users.

This document is current as of the time of release of the product; however, the accompanying software may have changed since the release. Rockwell Automation, Inc. reserves the right to change any information contained in this document or the software at any time without prior notice. It is your responsibility to obtain the most current information available from Rockwell when installing or using this product.

**Environmental compliance**


**Contact Rockwell Automation**

Customer Support Telephone — 1.440.646.3434

Rockwell Automation support

Rockwell Automation provides technical information on the web to assist you in using its products. At http://www.rockwellautomation.com/support you can find technical and application notes, sample code, and links to software service packs. You can also visit our Support Center at https://rockwellautomation.custhelp.com for software updates, support chats and forums, technical information, FAQs, and to sign up for product notification updates.

In addition, we offer multiple support programs for installation, configuration, and troubleshooting. For more information, contact your local distributor or Rockwell Automation representative, or visit http://www.rockwellautomation.com/services/online-phone.

Installation assistance

If you experience a problem within the first 24 hours of installation, review the information that is contained in this manual. You can contact Customer Support for initial help in getting your product up and running.

<table>
<thead>
<tr>
<th>United States or Canada</th>
<th>1.440.646.3434</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outside United States or Canada</td>
<td>Use the Worldwide Locator available at <a href="http://www.rockwellautomation.com/locations">http://www.rockwellautomation.com/locations</a>, or contact your local Rockwell Automation representative.</td>
</tr>
</tbody>
</table>

New product satisfaction return

Rockwell Automation tests all of its products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning and needs to be returned, follow these procedures.

<table>
<thead>
<tr>
<th>United States</th>
<th>Contact your distributor. You must provide a Customer Support case number (call the phone number above to obtain one) to your distributor to complete the return process.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outside United States</td>
<td>Please contact your local Rockwell Automation representative for the return procedure.</td>
</tr>
</tbody>
</table>

Documentation feedback

Your comments will help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete the feedback form, publication RA-DU002.
Important User Information

Solid-state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation, and Maintenance of Solid State Controls (publication SGI-1.1 available from your local Rockwell Automation sales office or online at http://www.rockwellautomation.com/literature/) describes some important differences between solid-state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid-state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

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.

---

**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.

---

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

---

Allen-Bradley, Rockwell Software, and Rockwell Automation, ControlLogix, iProcedure, FactoryTalk, RSNetWorx, RSNetWorx BatchHistorian, RSNetWorx Historian, RSNetWorx MaterialTrack, and RSLogix are trademarks of Rockwell Automation, Inc.

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

## Chapter 1

### FactoryTalk Event Archiver introduction
- System architecture............................................................................................................... 8
- Additional resources............................................................................................................. 9

## Chapter 2

### Work with electronic batch records
- Electronic batch record interpretation examples................................................................. 11

## Chapter 3

### Data source overview

## Chapter 4

### Implement FactoryTalk Event Archiver
- Configure the reporting option............................................................................................ 17
- Configure SQL Server to enable FactoryTalk Event Archiver with Reporting Services ................................................................. 19
- Configure SQL Server 2014 or SQL Server 2016 to enable FactoryTalk Event Archiver with Reporting Services ................................................................. 20
- Configure FactoryTalk Event Archiver for End of Batch archiving.................................... 21
- Configure FactoryTalk Event Archiver for optimal record transfer .................................... 22
- Run FactoryTalk Event Archiver............................................................................................ 22
  - Stop or pause FactoryTalk Event Archiver ........................................................................ 24
- FactoryTalk Event Archiver Service.................................................................................... 24
  - Assign a user account to the FactoryTalk Event Archiver Service..................................... 25
  - Enable the FactoryTalk Event Archiver Service............................................................... 25

## Chapter 5

### Custom archiver
- Custom archiver executable file........................................................................................... 27
  - Arguments from the FactoryTalk Batch Server................................................................. 27
  - Archque.txt file ................................................................................................................ 28
    - UNC line format example............................................................................................... 29
    - Local Path Name line format example.......................................................................... 29
- Output table.......................................................................................................................... 29
  - Display all instruction HTML data from an SQL database.............................................. 32
- Configure a custom archiver................................................................................................. 35
- Windows event log entries.................................................................................................... 36
# Chapter 6

FactoryTalk Event Archiver reports................................................................. 39
  Report parameters .................................................................................... 40
  Print reports ............................................................................................... 41
  Report caching ........................................................................................... 42
  Batch Listing report .................................................................................. 42
  Batch Detail report ..................................................................................... 43
  Batch Summary report ................................................................................ 45
  Material Usage report ............................................................................... 47
  Forward Tracking report .......................................................................... 48
  Backward Tracing report .......................................................................... 50
  Duration Comparison report ..................................................................... 51
  Batch Execution report ............................................................................ 52
  Batch Exceptions report .......................................................................... 53
  Operation Sequence report ...................................................................... 55
  Event Summary report ............................................................................. 58

# Chapter 7

Report files Report Definition Language (RDL) ................................................. 61
  Datasets ..................................................................................................... 61
  Parameters ................................................................................................ 62
  Code .......................................................................................................... 62
  Batch listing ............................................................................................... 63
  Batch summary ......................................................................................... 64
  Forward tracking ....................................................................................... 65
  Backward tracing ...................................................................................... 66
  Batch detail ............................................................................................... 67
  Material usage .......................................................................................... 69
  Duration comparison ................................................................................ 70
  Batch execution ........................................................................................ 71
  Batch exceptions ...................................................................................... 72
  Operation sequence ................................................................................ 74
  Event Summary ......................................................................................... 75

# Chapter 8

Backup types ............................................................................................... 77
Backup and delete database records ............................................................... 78
  Execution Settings window ...................................................................... 79
  Enter execution settings ......................................................................... 80
  Backup Settings window .......................................................................... 80
  Enter backup settings ............................................................................ 81
  Confirm Settings window ....................................................................... 82
  Execution dialog box for backup and delete ............................................. 82
Table of contents

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Restore database information from a backup</td>
<td>83</td>
</tr>
<tr>
<td>Restore settings window</td>
<td>83</td>
</tr>
<tr>
<td>Execution dialog box for restore</td>
<td>83</td>
</tr>
<tr>
<td>Appendix A</td>
<td></td>
</tr>
<tr>
<td>Material Usage Report</td>
<td>85</td>
</tr>
<tr>
<td>Forward/Backward Track and Trace Report</td>
<td>86</td>
</tr>
<tr>
<td>Setpoint vs. Actual</td>
<td>87</td>
</tr>
<tr>
<td>Batch Phase naming conventions</td>
<td>87</td>
</tr>
<tr>
<td>Single Specific Ingredient Addition Phase with no target_xx parameter</td>
<td>88</td>
</tr>
<tr>
<td>Single Specific Ingredient Addition Phase with target_xx parameter</td>
<td>88</td>
</tr>
<tr>
<td>Single Specific Ingredient Addition Phase</td>
<td>88</td>
</tr>
<tr>
<td>Single Specific Ingredient Addition Phase with Multiple Lot IDs</td>
<td>89</td>
</tr>
<tr>
<td>Multiple Specific Ingredient Addition Phase</td>
<td>89</td>
</tr>
<tr>
<td>Single Specific Ingredient Addition Phase with multiple parameter types</td>
<td>90</td>
</tr>
<tr>
<td>Generic Multiple Ingredient</td>
<td>90</td>
</tr>
<tr>
<td>Generic Multiple Ingredient Addition Phase with multiple lots per material</td>
<td>91</td>
</tr>
</tbody>
</table>

Special considerations – naming conventions

Legal Notices

Index

Legal Notices

93
Chapter 1

FactoryTalk Event Archiver

introduction

FactoryTalk Event Archiver collects data from the batch record files (.evt files) generated by the FactoryTalk Batch Server, and stores the data in a SQL Server database.

Features and capabilities

The FactoryTalk Batch Server creates an electronic batch record for every batch on the Batch List. View this batch record file with a text processor or spreadsheet application. FactoryTalk Event Archiver collects data from each electronic batch record file and writes it to the database. Each electronic batch record file contains individual lines, each representing an event that occurred in a batch. FactoryTalk Event Archiver inserts a unique identifier with date and time to each event into the database.

FactoryTalk Event Archiver writes data to the BatchHis SQL table in the BatchHistory database. FactoryTalk Event Archiver with Reporting Services writes data to the BHBatchHis SQL table in the BatchHistoryEx database.

Configure FactoryTalk Event Archiver to archive data to the database in either of two ways:

- End-of-Batch
- Incremental

With End-of-Batch archiving, FactoryTalk Event Archiver inserts records into the database after a batch is removed from the batch list. The FactoryTalk Batch Server initiates FactoryTalk Event Archiver when a batch is removed from the batch list to archive the removed batch.

With Incremental archiving, FactoryTalk Event Archiver inserts records into the database on a pre-defined schedule. Configure this archiving time interval in FactoryTalk Batch Equipment Editor.

When a control recipe on the FactoryTalk Batch Server executes, the server adds the name of the batch record file to the FactoryTalk Event Archiver work queue file (Archque.txt). After FactoryTalk Event Archiver inserts all of the electronic batch records into the database, and the file is marked for removal from the batch
list, FactoryTalk Event Archiver removes the name of the batch record file from the work queue file. FactoryTalk Event Archiver does not delete the batch record file unless it is specifically configured.

If FactoryTalk Event Archiver cannot successfully insert each item from the batch record file, it does not remove the name from the FactoryTalk Event Archiver work queue file. Each time FactoryTalk Event Archiver runs, it attempts to insert the records from each file listed in the queue into the database. In the event of a failure, FactoryTalk Event Archiver attempts to store the data later. This ensures that failures do not result in a loss of archived data.

**Important:** When FactoryTalk Event Archiver runs at the end of a batch, the FactoryTalk Batch Server runs FactoryTalk Event Archiver as a Windows IDLE_PRIORITY process. If the system is busy, higher-priority functions supersede FactoryTalk Event Archiver slowing the archiving process.

See also

Configure the reporting option on page 17

**System architecture**

FactoryTalk Event Archiver is a Windows service that archives events from the FactoryTalk Batch Server. Events are written to SQL database tables. View the events stored in the database as HTML reports with the Reporting Services.

FactoryTalk Event Archiver with Reporting Services system architecture
FactoryTalk Event Archiver can run incrementally while a batch is running, or after a batch completes. FactoryTalk Event Archiver executes in a continuous loop using these basic steps:

2. FactoryTalk Event Archiver reads event records from the event file and translates them into SQL statements.
3. FactoryTalk Event Archiver writes the data to the SQL database.
4. FactoryTalk Event Archiver logs all archiving activities to a log file.
5. When Reporting Services is installed, reports can be generated from event data stored in the database.

See also

FactoryTalk Event Archiver introduction on page 7

These documents contain additional information concerning related Rockwell Automation products.

<table>
<thead>
<tr>
<th>Resource</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>FactoryTalk Batch Components Upgrade and Installation Guide</td>
<td>Provides the technical information for installation of a FactoryTalk® Batch system. It can be used as reference information for implementation engineers and the system administrator.</td>
</tr>
<tr>
<td>FactoryTalk Batch System Files Reference Guide</td>
<td>Provides the technical information for working with electronic batch records. It can be used as reference information for implementation engineers and the system administrator.</td>
</tr>
</tbody>
</table>

View or download publications at [http://www.rockwellautomation.com/literature](http://www.rockwellautomation.com/literature). To order paper copies of technical documentation, contact the local Rockwell Automation distributor or sales representative.
Chapter 2

Work with electronic batch records

An electronic batch record (.evt) file is a record (history) of a batch. Use this record for regulatory and reporting purposes. The ISA S88.01 standard defines batch history as a collection of data describing or related to a batch.

For every batch execution, the FactoryTalk Batch Server creates and updates an electronic batch record (.evt) file during the execution of the batch procedure. Each electronic batch record entry contains fields entries based on event types. All data is in ASCII format, which may represent time, integer, real number, string, or enumerations.

As events occur during batch execution, the FactoryTalk Batch Server appends data to the batch record. Access the batch record from FactoryTalk Batch View. The batch data is in a simple ASCII format file. View batch data with a text editor, word processor, or spreadsheet program.

See the ISA S88.01 Standard for more information on modes, states, and commands.

See the FactoryTalk Batch System Files Reference Guide for more information on electronic batch records.

See also

Electronic batch record interpretation examples on page 11
Data source overview on page 15

The event file contains a record for each occurrence of a phase, operation, unit procedure, and procedure changing state (for example, STARTING, STOPPING, and HOLDING).

To look at values passed to a phase, look for the variable passed in the Description column. The PValue column indicates the parameter value passed for that variable during that phase.
Example:

<table>
<thead>
<tr>
<th>Recipe</th>
<th>Description</th>
<th>Event</th>
<th>PValue</th>
<th>EU</th>
</tr>
</thead>
<tbody>
<tr>
<td>12:TPROC\TUP:1\TOP:1\CHARGE:1-1</td>
<td>AMNT_TO_CHAR GE</td>
<td>Recipe Value</td>
<td>50</td>
<td>GALLONS</td>
</tr>
</tbody>
</table>

To see whether a level (for example, operation) became active, look in the **Description** column and find the term **Operation Started**, **Step Activated**, etc. The **EU** column displays **Phase** if the step activated is a phase.

Example:

<table>
<thead>
<tr>
<th>Recipe</th>
<th>Description</th>
<th>PValue</th>
<th>EU</th>
<th>Phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1-1</td>
<td>Step Activated</td>
<td>CHARGE:1</td>
<td>Phase</td>
<td>ICE_CREAM</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1\CHARGE:1-1</td>
<td>State Changed: RUNNING</td>
<td></td>
<td></td>
<td>CHARGEA</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1\CHARGE:1-1</td>
<td>State Changed: COMPLETE</td>
<td></td>
<td></td>
<td>CHARGEA</td>
</tr>
</tbody>
</table>

To evaluate a transfer of control, entries show the involved continuous phases becoming active and idle in turn. However, only the final continuous phase shows complete.

For example: If a transfer of control involves three instances of the same phase running in sequence, the first would start and become idle, then the second would start and become idle, and finally the third would start and become complete.

This example shows the state changes of an entire operation that contains a transfer of control operation in which a RECIRC phase occurs both before and after a transition:

<table>
<thead>
<tr>
<th>Recipe</th>
<th>Description</th>
<th>PValue</th>
<th>EU</th>
<th>Phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1-1</td>
<td>Operation Started</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1-1</td>
<td>State Changed: RUNNING</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1-1</td>
<td>Step Activated</td>
<td>RECIRC:2</td>
<td>Phase</td>
<td>RECIRC1</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1-1</td>
<td>Step Activated</td>
<td>CHARGE:1</td>
<td>Phase</td>
<td>CHARGEA</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1-1</td>
<td>Step Deactivated</td>
<td>Initial Step</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1\RECIRC:2-1</td>
<td>State Changed: IDLE</td>
<td></td>
<td>RECIRC1</td>
<td></td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1\CHARGE:1-1</td>
<td>State Changed: IDLE</td>
<td></td>
<td>CHARGEA</td>
<td></td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1\RECIRC:2-1</td>
<td>State Changed: RUNNING</td>
<td></td>
<td>RECIRC1</td>
<td></td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1\CHARGE:1-1</td>
<td>State Changed: RUNNING</td>
<td></td>
<td>CHARGEA</td>
<td></td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1\CHARGE:1-1</td>
<td>State Changed: COMPLETE</td>
<td></td>
<td>CHARGEA</td>
<td></td>
</tr>
</tbody>
</table>
### Recipe

<table>
<thead>
<tr>
<th>Recipe</th>
<th>Description</th>
<th>PValue</th>
<th>EU</th>
<th>Phase</th>
</tr>
</thead>
<tbody>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1-1</td>
<td>Operation Started</td>
<td>0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1-1</td>
<td>Step Activated</td>
<td>CHARGE_B: 1</td>
<td>Phase</td>
<td>CHARGE_B</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1-1</td>
<td>Step Activated</td>
<td>RECIRC:3</td>
<td>Phase</td>
<td>RECIRC1</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1\CHARGE:1-1</td>
<td>State Changed:</td>
<td>IDLE</td>
<td>Phase</td>
<td>CHARGE_A</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1-1</td>
<td>Step Deactivated</td>
<td>CHARGE:1</td>
<td>Phase</td>
<td>CHARGE_A</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1\RECIRC:3-1</td>
<td>State Changed:</td>
<td>RUNNING</td>
<td>Phase</td>
<td>RECIRC1</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1\RECIRC:3-1</td>
<td>Owner Changed:</td>
<td>PROGRAM</td>
<td>Phase</td>
<td>RECIRC1</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1\RECIRC:3-1</td>
<td>Attribute Change</td>
<td>PAUSED off</td>
<td>Phase</td>
<td>RECIRC1</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1-1</td>
<td>Step Deactivated</td>
<td>RECIRC:2</td>
<td>Phase</td>
<td>RECIRC1</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1\CHRG_B:1-1</td>
<td>State Changed:</td>
<td>IDLE</td>
<td>Phase</td>
<td>CHARGE_B</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1\CHRG_B:1-1</td>
<td>State Changed:</td>
<td>RUNNING</td>
<td>Phase</td>
<td>CHARGE_B</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1\CHRG_B:1-1</td>
<td>State Changed:</td>
<td>COMPLETE</td>
<td>Phase</td>
<td>CHARGE_B</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1\RECIRC:3-1</td>
<td>State Changed:</td>
<td>COMPLETE</td>
<td>Phase</td>
<td>RECIRC1</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1-1</td>
<td>Step Activated</td>
<td>Terminal Step</td>
<td>Phase</td>
<td>RECIRC1</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1\CHRG_B:1-1</td>
<td>State Changed:</td>
<td>IDLE</td>
<td>Phase</td>
<td>CHARGE_B</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1-1</td>
<td>Step Deactivated</td>
<td>CHARGE_B: 1</td>
<td>Phase</td>
<td>CHARGE_B</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1\RECIRC:3-1</td>
<td>State Changed:</td>
<td>IDLE</td>
<td>Phase</td>
<td>RECIRC1</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1-1</td>
<td>Step Deactivated</td>
<td>RECIRC:3</td>
<td>Phase</td>
<td>RECIRC1</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1-1</td>
<td>State Changed:</td>
<td>COMPLETE</td>
<td>Phase</td>
<td>RECIRC1</td>
</tr>
<tr>
<td>12:TRNPROC\TRANUP:1-1</td>
<td>Step Activated</td>
<td>Terminal Step</td>
<td>Phase</td>
<td>RECIRC1</td>
</tr>
<tr>
<td>12:TPROC\TUP:1\TRANCT:1-1</td>
<td>Operation Done</td>
<td>0</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

See also

[Work with electronic batch records](#) on page 11
Chapter 3

Data source overview

Installing the FactoryTalk Batch Server and Client software configures the Data Source name (DSN) and creates the ODBC Connect String. FactoryTalk Event Archiver uses the Connect String to locate the database. Once created, the Connect String is stored in the **BatchArc.ini** file. This Data Source name serves as an alias for the database.

After installation, configure FactoryTalk Event Archiver for Incremental or End-of-Batch archiving. This involves configuring FactoryTalk Batch Equipment Editor **Server Options** and the FactoryTalk Event Archiver service.

Make sure to configure user accounts according to the following:

- If using Incremental Archiving, make sure the user account assigned to the FactoryTalk Event Archiver service has login access to SQL Server and has permission to insert data into the BATCHHIS or BHBATCHHIS table.

- If using End of Batch Archiving, make sure the user account assigned to the FactoryTalk Batch Server has access to the BatchHistory database on the SQL Server and has permission to insert and execute data into the BATCHHIS or BHBATCHHIS table.

- Ensure the server’s user account is a member of the local Administrator’s group on the SQL Server computer.
Implement FactoryTalk Event Archiver

To implement FactoryTalk Event Archiver, configure the reporting option and the Archiver Server.

See also

- Configure the reporting option on page 17
- Configure FactoryTalk Event Archiver for End of Batch archiving on page 21
- Configure FactoryTalk Event Archiver for optimal record transfer on page 22
- Run FactoryTalk Event Archiver on page 22
- FactoryTalk Event Archiver service on page 24

Configure the reporting option

Enable FactoryTalk Event Archiver by configuring the FactoryTalk Batch Server to use FactoryTalk Event Archiver as the reporting program. The reporting option is required whether using reporting services or not.

**Tip:** If enabled, on the Event Journal Signatures option in the Batch Server tab, specify Archiver (End of Batch) archiving to ensure all event journal records properly archive.

**Tip:** Data added to the batch event journal files (.evt) by the ADDEVENT execute cannot be filtered out.
Implement FactoryTalk Event Archiver

Important: The date format used for the event journal file’s LclTime is YYYY.MM.DD HH:MM:SS and is not configurable.

To configure the reporting option:

1. Navigate to Rockwell Software, and then select Equipment Editor.

2. Select Options > Server Options.

3. On the Server Options dialog box, select the Batch Reporting tab.

4. In the Reporting Application area, select either Archiver (End of Batch) or Archiver (Incremental).
   - Archiver (End of Batch) – The FactoryTalk Batch Server initiates FactoryTalk Event Archiver when a batch is removed from the batch list. When using this option, the FactoryTalk Event Archiver Service must be disabled.
   - Archiver (Incremental) – FactoryTalk Event Archiver runs at periodic intervals.

5. To change the configuration set during product installation, perform the following steps:
   a. In Table Name, enter the name of the data table where the batch data archives. The default table name is BATCHHIS.
      - When configuring FactoryTalk Event Archiver without Reporting Services with the BatchHistory database, use BATCHHIS as the Table Name.
      - When configuring FactoryTalk Event Archiver with Reporting Services with the BatchHistoryEx database, use BHBATCHHIS as the Table Name.
   b. In the Database Connect String box, enter the database connect string.

6. If Archiver (Incremental) has been selected, in the Incremental Period box, enter the time interval between active archiving, and then select Minutes or Seconds. The minimum setting is 5 seconds. The maximum is 9999 minutes.
Tip: If the value is outside of this range, FactoryTalk Event Archiver runs in five-minute increments.

7. Select the Archiver Event Filters tab. Material Server Filters display only if Material Manager is installed. eProcedure Filters display only if eProcedure is installed.

8. Select the data to archive from the event journal files (.evt) for the batch, and then select OK.

See also

Configure FactoryTalk Event Archiver for End of Batch archiving on page 21

Configure SQL Server to enable FactoryTalk Event Archiver with Reporting Services

Configure SQL Server to enable FactoryTalk Event Archiver with Reporting Services

Tip: This procedure applies to versions of SQL Server prior to SQL Server 2014. See Configure SQL Server 2014 to enable FactoryTalk Event Archiver with Reporting Services.

Important: All users of the FactoryTalk Batch Reporting Services must have a user login and read and write access to the source databases, such as BatchHistoryEx or the FactoryTalk® Alarms and Events FTAE_Historian databases. Add any additional users or groups when adding the login account for the FactoryTalk Batch Server.

To configure SQL Server to enable FactoryTalk Event Archiver with Reporting Services:

1. Open the SQL Server Management Studio.

2. In the Security > Logins folder on the SQL Server instance, do the following:

   a. Add a FactoryTalk Batch Server user login to the Security > Logins folder on the SQL Server instance.

   b. Set the default database to BatchHistoryEx.

   c. Under User Mapping, select the BatchHistoryEX database.
d. Select `db_datareader`, `db_datawriter`, and select OK.

3. In Databases, right-click the BatchHistory or BatchHistoryEx database, and select Properties.
   a. Grant INSERT privilege for the FactoryTalk Batch Server user account on the BatchHistoryEX database.
   b. Under BatchHistoryEX\Programmability, add the EXECUTE permission for the FactoryTalk Batch Server user account to `dbo.BHInsert` properties.
   c. For BatchHistoryEX\Programmability, add the EXECUTE permission for the FactoryTalk Batch Server user account to `dbo.BHRecovery` properties.

See also

Configure SQL Server 2014 to enable FactoryTalk Event Archiver with Reporting Services on page 20

Configure SQL Server 2014 or SQL Server 2016 to enable FactoryTalk Event Archiver with Reporting Services

Configure SQL Server to enable FactoryTalk Event Archiver with Reporting Services to connect to and insert records into the FactoryTalk Batch database. Set up permissions for the BatchHistory or BatchHistoryEx database in SQL Server Management Studio. See the FactoryTalk Event Archiver User Guide to configure the reporting option for End of Batch or Incremental.

Tip: This procedure only applies to SQL Server 2014 and SQL Server 2016. See Configure SQL Server to enable FactoryTalk Event Archiver with Reporting Services to configure using versions prior to SQL Server 2014.

Important: All users of the FactoryTalk Batch Reporting Services must have a user login and read and write access to the source databases, such as BatchHistoryEx or the FactoryTalk® Alarms and Events FTAE_Historian databases. Add any additional users or groups when adding the login account for the FactoryTalk Batch Server.

To configure SQL Server 2014 or SQL Server 2016 to enable FactoryTalk Event Archiver with Reporting Services

1. From SQL Server Management Studio, add a FactoryTalk Batch Server user login to the Security > Logins folder on the SQL Server instance.
2. Set the default database to **BatchHistory** or **BatchHistoryEx**.

3. Under **User Mapping**, select the **BatchHistory** or **BatchHistoryEx** database.

4. Select **db_datareader, db_datawriter**, and select **OK**.

5. In **Databases**, right-click the **BatchHistory** or **BatchHistoryEx** database, and select **Properties**.

6. Select **Permissions**.

7. Select the Batch Server user in the list.

8. Grant **Execute, Insert**, and **Connect** permissions.

**See also**

[Configure SQL Server to enable FactoryTalk Event Archiver with Reporting Services](#) on page 19

To use End of Batch archiving, disable the Archiver Service so that FactoryTalk Event Archiver runs as a program triggered by the FactoryTalk Batch Server. Use these instructions to disable the Archiver Service.

**Tip:** When using End of Batch archiving, the FactoryTalk Batch Server's user account must have login access to the Batch History database on the SQL Server and the account must have permission to insert and execute data into the BatchHis table in the BatchHistory database or BHBatchHis table in the BatchHistoryEx database table. The server's user account must be a member of the local Administrator's group on the SQL Server computer.

To configure FactoryTalk Event Archiver for End of Batch archiving:

1. Navigate to **Services**.

2. From the **Services** list, double-click **FactoryTalk Event Archiver**. The **FactoryTalk Event Archiver Properties (Local Computer)** dialog box opens to the **General** tab.

3. From the **Startup Type** list, select **Disabled**.

4. Select **OK**, and then close the **Services** dialog box.

**See also**

[Configure FactoryTalk Event Archiver for optimal record transfer](#) on page 22
Configure FactoryTalk Event Archiver for optimal record transfer

FactoryTalk Event Archiver can be set to archive event data at either:

- End of Batch, after the batch operation has finished.
- Incremental, during the batch run at pre-set time increments.

The method used, either End of Batch or Incremental, does not have an impact on FactoryTalk Event Archiver performance. However, several variables can affect record transfer performance. These include the server hardware, processor speed, available memory, network configuration, database size, and the number of records generated.

Configure FactoryTalk Event Archiver for optimal record transfer on the server by trial and error. Start by initially setting the Maximum Records Per Transaction to 275.

To configure FactoryTalk Event Archiver for optimal record transfer:

1. Open FactoryTalk Batch Equipment Editor.
2. From the Options menu, select Server Options.
3. Select the Batch Reporting tab.
4. In the Reporting Application area, select Archiver (Incremental).

This sets the maximum number of records written to the database at each transaction. Use 275 as a starting point and adjust that number up or down based on the results.

Increasing the Maximum Records Per Transaction lowers the number of requests to the database, which improves performance to an extent.

See also

Run FactoryTalk Event Archiver on page 22

Run FactoryTalk Event Archiver

When using End of Batch Archiving, the FactoryTalk Batch Server runs FactoryTalk Event Archiver after a batch is removed from the batch list. When using Incremental Archiving, FactoryTalk Event Archiver runs continuously during batch execution and archives batch data at user-defined time increments. The FactoryTalk Batch Service Manager controls the Archiver Service.

The Batch Service Manager must communicate with the Windows Service Manager of the selected computer to determine available services. A noticeable delay may exist while communications are established. If the Batch Service
Manager cannot communicate with the Windows Services, an error message is displayed.

**Before you begin:**

- Ensure the computer where FactoryTalk Event Archiver is installed has administrative privileges.

**To run FactoryTalk Event Archiver:**

1. Navigate to **Rockwell Software**, and then select **Batch Service Manager**.

2. On the **FactoryTalk Batch Service Manager** dialog box, choose **Select Computer**.

3. On the **Select Computer** dialog box, select the name of the computer where the Archiver Service is installed, and then select **OK**. The **Select Computer** dialog box closes.

4. From the **Service** list, select **FactoryTalk Event Archiver**.

5. Select **Start** to start the FactoryTalk Event Archiver Service and then select **Close**.

![FactoryTalk Batch Service Manager](image)

**See also**

[Stop or pause FactoryTalk Event Archiver](#) on page 24
Stop or pause FactoryTalk Event Archiver

Stop or pause the Archiver Service at the FactoryTalk Batch Service Manager. **Pause** causes the Incremental Archiver to pause after processing the active .evt file. **Start** resumes Incremental Archiving.

**Before you begin:**

- Make sure there are administrative privileges on the computer where FactoryTalk Event Archiver is installed.

To stop or pause FactoryTalk Event Archiver:

1. Navigate to **Rockwell Software**, and then select **Batch Service Manager**.
2. On the **FactoryTalk Batch Service Manager** dialog box, choose **Select Computer**.
3. On the **Select Computer** dialog box, select the computer where the Archiver Service is installed, and then select **OK**. The **Select Computer** dialog box closes.
4. From the **Service** list, select **FactoryTalk Event Archiver**.
5. Select **Stop** or select **Pause**.
6. Select **Close**.

See also

- [Run FactoryTalk Event Archiver](#) on page 22

FactoryTalk Event Archiver Service

FactoryTalk Event Archiver runs as a Windows service. Assign a Windows user account to the Archiver Service to log on to the system. By default, the Archiver Service uses the FactoryTalk Batch Server user account specified during the installation. If archiving data to the local computer only, use the Local System account instead of the server user account.

**Important:** On Windows Server 2008 systems using Active Directory, change the default format of **user@domain.com** to **domain\user**. The FactoryTalk Event Archiver cannot access the database if the database account name has special characters.

See also

- Assign a [user account to the FactoryTalk Event Archiver Service](#) on page 25
- Enable the [FactoryTalk Event Archiver Service](#) on page 25
Assign a user account to the FactoryTalk Event Archiver Service

During installation, a user account is created for FactoryTalk Event Archiver. When an additional user account is needed, add the user account to the local Administrator's user group and assign the user name to the FactoryTalk Event Archiver Service.

**Tip:** The FactoryTalk Event Archiver Service and the FactoryTalk Batch Server service should use the same user account.

**To assign a user account to the FactoryTalk Event Archiver service:**

1. Navigate to Services.
2. In the Services dialog box, double-click FactoryTalk Event Archiver.
3. In the FactoryTalk Event Archiver Properties (Local Computer) dialog box, select the Log On tab.
4. Select browse (browse icon).
5. In the Select User dialog box, locate and select the FactoryTalk Event Archiver user account, and then select OK to close the Select User dialog box.
6. In Password, enter the password for the FactoryTalk Event Archiver user account.
7. In the Confirm password box, enter the password again and then select OK. A confirmation message box opens.
8. Select OK, and then close the Services dialog box.

**Tip:** If the FactoryTalk Event Archiver inserts data into a database on another computer and Trusted Security is used, the Batch Archiver Service must run in a domain account assigned with the appropriate privileges within the database program. Using Trusted Security creates one set of accounts.

**See also**

[Enable the FactoryTalk Event Archiver Service](#) on page 25

Enable the FactoryTalk Event Archiver Service

When FactoryTalk Batch installs, the FactoryTalk Event Archiver Service enables by default. In the event the Archiver Service disables, follow these steps to enable the service.

**To enable the FactoryTalk Event Archiver Service:**

1. Navigate to Services.
2. Double-click FactoryTalk Event Archiver, and then select the Log On tab.

3. Under Log on as, select the appropriate account.

4. From the General tab, from the Startup Type list, select Automatic.

5. Select OK, and then close the Services dialog box.

See also

Implement FactoryTalk Event Archiver on page 17
Chapter 5

Custom archiver

If requirements are not addressed by FactoryTalk Event Archiver, write a custom archiver. This custom archiver receives arguments from the FactoryTalk Batch Server, and reads and manipulates the ArchQue.txt, electronic batch record (.evt), and BatchArc.ini files.

**Important:** The archiver executable becomes a process that launches but does not display as an application within Windows Task Manager. It has no visibility on the desktop.

See also

- Custom archiver executable file on page 27
- Archque.txt file on page 28
- Output table on page 29

**Custom archiver executable file**

Create a custom archiver executable file as:

- C, C++, C#, or Visual Basic application.
- Windows batch (.bat) file.
- Windows command (.cmd) file.

See also

- Arguments from the FactoryTalk Batch Server on page 27
- Archque.txt file on page 28

**Arguments from the FactoryTalk Batch Server**

The FactoryTalk Batch Server passes these arguments to the FactoryTalk Event Archiver:

<path to custom executable file> <.ini file name> <path to archque.txt> <time delay in minutes>
Example

C:\Program Files\Rockwell Software\Batch\PROJECTS\CUSTOM\MYARCHVR.EXE
MYARCHVR.INI C:\Program Files\Rockwell Software\Batch\PROJECTS\JOURNALS\ARCHQUE.TXT

See also

ArchQue.txt file on page 28

ArchQue.txt file

The archive queue file, ArchQue.txt, contains a list of electronic batch record files. The line format within the list contains the path and file name of the batch record file. The path name can use either UNC names or local path names, depending on the definition of the journal project directory in the FactoryTalk Batch Equipment Editor Server Options dialog box.

If a batch is removed from the batch list, a tab character (\t) and the word REMOVED follows the file name. Each electronic batch record file is listed in a single record delimited with Carriage Return / Line Feed (CR/LF).

Tip: Keep the ArchQue.txt file locked while it is open and keep it open for only a few seconds. When the queue file is open for more than a few seconds, the FactoryTalk Batch Server cannot add entries to the queue as other control recipes are created.

Important: The ArchiverEnabled=Never setting in the BatchArc.ini file causes the Archiver queue file to not be written. To write to this queue file, use a setting other than Never, and manually maintain the size of the archive queue file.

See also

UNC line format example on page 28

Local Path Name line format example on page 29
UNC line format example

**Line Format (UNC)**

```
\computer_name\sharename\path_to_journal_file \t REMOVED{CR/LF}
```

**Example**

```
\station1\BATCHCTL\SAMPLEDEMO1\journals\123.evt \t REMOVED{CR/LF}
\station1\BATCHCTL\SAMPLEDEMO1\journals\124.evt \t REMOVED{CR/LF}
\station1\BATCHCTL\SAMPLEDEMO1\journals\125.evt \t REMOVED{CR/LF}
```

See also

[Local Path Name line format example on page 29](#)

Local Path Name line format example

**Line Format (Local Path Name)**

```
C:\path_to_journal_file \t REMOVED{CR/LF}
```

**Example**

```
C:\Program Files\Rockwell Software\Batch\SAMPLEDEMO1\journals\123.evt \t REMOVED{CR/LF}
C:\Program Files\Rockwell Software\Batch\SAMPLEDEMO1\journals\124.evt \t REMOVED{CR/LF}
C:\Program Files\Rockwell Software\Batch\SAMPLEDEMO1\journals\125.evt \t REMOVED{CR/LF}
```

See also

[Archque.txt file on page 28](#)

Output table

This table describes the schema of both the BATCHHIS table in the BatchHistory database and the BHBATCHHIS table in the BatchHistoryEx database. Use this schema as the basis for developing a custom output table.

**Tip:** SQL Server columns that have a box type of `ntext` only retrieve 256 characters at a time using a standard `SELECT` statement. To retrieve all the data, use the `READTEXT` function.
<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>GMT</td>
<td>datetime NULL</td>
<td>Time event is recorded in Greenwich Mean Time</td>
<td>2003-04-29 12:21:01.000</td>
</tr>
<tr>
<td>lclTime</td>
<td>datetime NULL</td>
<td>Local time of event</td>
<td>2003-04-29 14:52:07.000</td>
</tr>
<tr>
<td>UniqueID</td>
<td>varchar(12) NULL</td>
<td>Unique identifier assigned by the FactoryTalk Batch Server</td>
<td>31</td>
</tr>
<tr>
<td>BatchID</td>
<td>varchar(255) NULL</td>
<td>Alpha Batch ID</td>
<td>BATCH_ID-030102-3</td>
</tr>
<tr>
<td>Recipe</td>
<td>varchar(1000) NULL</td>
<td>The recipe element that was executing at the time of the electronic batch record entry. The box contents appear as: &lt;UniqueID&gt;:&lt;Procedure&gt;:&lt;Unit Procedure&gt;:Instance:Operation:Instance-Loop Counter. The data in the brackets is optional, depending on the recipe level.</td>
<td>13:Y_300_YELLOW_PAINT</td>
</tr>
<tr>
<td>Descript</td>
<td>varchar(255) NULL</td>
<td>Description of PValue data</td>
<td>Yellow Paint</td>
</tr>
<tr>
<td>Event</td>
<td>varchar(155) NULL</td>
<td>Event type triggering the record</td>
<td>Material Tracking / Material Bound / Material Unbound</td>
</tr>
<tr>
<td>PValue</td>
<td>varchar(255) NULL</td>
<td>Value reported to the electronic batch record</td>
<td>50</td>
</tr>
<tr>
<td>DescriptAPI</td>
<td>varchar(255) NULL</td>
<td>Description of PValue data</td>
<td>Engineering Units</td>
</tr>
<tr>
<td>EventAPI</td>
<td>varchar(155) NULL</td>
<td>Event type triggering the record</td>
<td>Report</td>
</tr>
<tr>
<td>PValueAPI</td>
<td>varchar(255) NULL</td>
<td>Value reported to the electronic batch record</td>
<td>50</td>
</tr>
<tr>
<td>EU</td>
<td>varchar(255) NULL</td>
<td>Engineering units associated with PValue</td>
<td>GALLONS</td>
</tr>
<tr>
<td>Area</td>
<td>varchar(155) NULL</td>
<td>Area model name in batch equipment configuration where batch was executed</td>
<td>THE_PAINT_COMPANY</td>
</tr>
<tr>
<td>ProcCell</td>
<td>varchar(155) NULL</td>
<td>Process cell in batch equipment configuration</td>
<td>YELLOWPAINT1</td>
</tr>
<tr>
<td>Unit</td>
<td>varchar(155) NULL</td>
<td>Unit name in batch equipment configuration</td>
<td>PREMIX_A</td>
</tr>
<tr>
<td>Phase</td>
<td>varchar(155) NULL</td>
<td>Name of the phase generating the record</td>
<td>PROC_ORDER_N</td>
</tr>
<tr>
<td>Printed</td>
<td>datetime NULL</td>
<td>Last date the batch report was printed</td>
<td>2003-04-29 14:52:07.000</td>
</tr>
<tr>
<td>UserID</td>
<td>varchar(85) NULL</td>
<td>Computer name/Windows user name</td>
<td>mjames1</td>
</tr>
<tr>
<td>PhaseDesc</td>
<td>varchar(50) NULL</td>
<td>Phase class description</td>
<td></td>
</tr>
<tr>
<td>Name</td>
<td>Type</td>
<td>Description</td>
<td>Example</td>
</tr>
<tr>
<td>----------------</td>
<td>---------------</td>
<td>----------------------------------------------------------------------------</td>
<td>--------------------------------</td>
</tr>
<tr>
<td>MaterialName</td>
<td>varchar(155)</td>
<td>Common name of the material</td>
<td>YELLOWPAINT</td>
</tr>
<tr>
<td>MaterialID</td>
<td>varchar(50)</td>
<td>Inventory control system material ID used to track the material</td>
<td>31</td>
</tr>
<tr>
<td>LotName</td>
<td>varchar(255)</td>
<td>Name assigned to a lot of material</td>
<td>TEST317_BRAND</td>
</tr>
<tr>
<td>Label</td>
<td>varchar(255)</td>
<td>ID assigned to a distributed portion of a lot</td>
<td>STANDARDBRANDYELLOW</td>
</tr>
<tr>
<td>Container</td>
<td>varchar(255)</td>
<td>Container's name where material and lot were distributed, or where material and lot are stored</td>
<td>WAREHOUSE2/PAINT _PALLET1</td>
</tr>
<tr>
<td>PromiseID</td>
<td>varchar(50)</td>
<td>Unique ID used as a key to allow grouping of separate event entries all resulting from a single material addition</td>
<td>100</td>
</tr>
<tr>
<td>Signature</td>
<td>varchar(255)</td>
<td>Used for event journal signatures by Batchsig program to detect changes in .evt file. Information stored here is a hexadecimal or a single spaced character.</td>
<td></td>
</tr>
<tr>
<td>ERP_Flag</td>
<td>varchar(1)</td>
<td>Tag indicating information reported to SAP</td>
<td></td>
</tr>
<tr>
<td>RecordNo</td>
<td>int NULL</td>
<td>Unique identifier assigned to the record when it is added to the database</td>
<td>31</td>
</tr>
<tr>
<td>ArchivedGMT</td>
<td>datetime NULL</td>
<td>Date and time</td>
<td>2003-04-29 14:52:07:000</td>
</tr>
<tr>
<td>Reactivation Number</td>
<td>int</td>
<td>Number of times an active eProcedure control step has been reactivated</td>
<td>1</td>
</tr>
<tr>
<td>InstructionHTML</td>
<td>ntext</td>
<td>HTML for completed eProcedure instruction steps</td>
<td>&lt;FORM&gt;Add a CIP batch to the Batch List. Select OK when done.&lt;/FORM&gt;</td>
</tr>
<tr>
<td>SignatureID</td>
<td>int</td>
<td>A 32-bit unsigned integer that uniquely identifies a Signature Request within the batch system.</td>
<td>1</td>
</tr>
</tbody>
</table>
### Display all instruction HTML data from an SQL database

Use the `READTEXT` function to retrieve all data in the Instruction HTML column for a specified record number. These steps represent a sample query and explain how to modify the query so that it will produce an HTML file.

**To display all instruction HTML data from an SQL database:**

1. In Query Analyzer go to the **Tools > Options > Results** tab and clear the **Print column headers (*)** option. Select **Apply** and then **OK** to close the dialog box.

2. From the **Query** menu, select **Results to File**.

3. Enter this `BHBATCHHIS` query into a blank query window. When running the query on `BATCHHIS`, change all instances of `bhbatchhis` to `batchhis`.

```sql
/*
 -- Displays the contents of a text column in a
255-character wide query window
*/
```

---

<table>
<thead>
<tr>
<th>Name</th>
<th>Type</th>
<th>Description</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>ActionID</td>
<td>int</td>
<td>A 32-bit unsigned integer that uniquely identifies an Action within the batch system. An Action is a queued Execute command to the FactoryTalk Batch Server that is pending one or more Signature completions prior to execution.</td>
<td>1</td>
</tr>
<tr>
<td>ContextsID</td>
<td>varchar(255)</td>
<td>Supports Context ID reporting. Lists the ContextID enumeration set member ordinals assigned to the input or output parameter for event generation.</td>
<td></td>
</tr>
<tr>
<td>Contexts</td>
<td>varchar(1024)</td>
<td>Supports Context ID reporting. Lists the ContextID enumeration set member strings assigned to the input or output parameter for event generation.</td>
<td></td>
</tr>
<tr>
<td>RecordID</td>
<td>PK, varchar(255) NOT NULL</td>
<td>The primary key.</td>
<td>IDEVSE110010Sep 3 2014881157</td>
</tr>
</tbody>
</table>

**See also**

[Display all instruction HTML data from an SQL Database](#) on page 32
-- @txtptrval is the text pointer value for the specified text column
-- @offsetval is the offset value and represents the starting position within a text column
-- @bufferval represents the amount of text to put in the row
-- in this case, it is set to 255 because isql/w only displays 255 characters in a row
-- @maxval is the full length of the entire text column
*/
BEGIN
SET NOCOUNT ON
DECLARE
@txtptrval VARBINARY(16),
@offsetval INT,
@bufferval INT,
@maxval INT
SELECT @txtptrval = TEXTPTR(bhbatchhis.InstructionHTML)
FROM bhbatchhis
WHERE recordno = '5817'
SELECT @offsetval = 0
SELECT @bufferval = 255
SELECT @maxval = DATALENGTH(bhbatchhis.InstructionHTML) / 2-1
FROM bhbatchhis
WHERE recordno = '5817'
--PRINT 'Total length of column: '
--PRINT '------'
--PRINT @maxval
--PRINT ''
-- Last chunk, reduce buffer size to the nChars remaining
IF (@offsetval + @bufferval) > @maxval
BEGIN
SELECT @bufferval = @maxval - @offsetval

--PRINT 'Last chuck... buffer size remaining is:'
--PRINT '------'
--PRINT @bufferval
END

WHILE @offsetval < @maxval
BEGIN
READTEXT bhbatchhis.InstructionHTML @txtptrval @offsetval @bufferval

--PRINT 'Data started at character position'
--PRINT @offsetval
SELECT @offsetval = @offsetval + @bufferval

--PRINT 'Data ended at character position'
--PRINT @offsetval
--PRINT ' ' 
-- Last chunk, reduce buffer size to the get the last nChars remaining
IF (@offsetval + @bufferval) > @maxval
SELECT @bufferval = @maxval - @offsetval + 1
END
SET NOCOUNT OFF
END

4. In the query, edit the WHERE clause to specify the recordno to view the InstructionHTML.

5. Run the query and save the results to a file name, such as Output.html.
Tip: Be sure to specify the .html extension, change the Save as type to All Files (*.*) and change the File Format to ANSI.

6. In Windows Explorer, double-click the .html file to view the page in the browser.

See also

Configure a custom archiver on page 35

Configure a custom archiver

Configure the custom archiver from the Equipment Editor.

To configure a custom archiver:

1. In the Equipment Editor, select Options > Server Options.

2. In the Server Options dialog box, select the Batch Reporting tab.

3. Select User-defined (End of Batch).

4. In Executable File, select browse (…) and locate the custom archiver's .exe file.

5. In Initialization File, select browse (…) and locate the custom archiver's initialization .ini file.

6. Select OK.

See also

Windows event log entries on page 35
Chapter 5  Custom archiver

Windows event log entries

FactoryTalk Event Archiver creates Windows event log entries during startup, run time, and shutdown.

Log entries contain:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Date</td>
<td>Date that the event occurred</td>
</tr>
<tr>
<td>Time</td>
<td>Time that the event occurred</td>
</tr>
<tr>
<td>Type</td>
<td>Type of event, such as Information or Error</td>
</tr>
<tr>
<td>User</td>
<td>Configured domain user</td>
</tr>
<tr>
<td>Computer</td>
<td>Computer on which FactoryTalk Event Archiver runs</td>
</tr>
<tr>
<td>Source</td>
<td>BATCHARC.log file</td>
</tr>
<tr>
<td>Category</td>
<td>Category of the event</td>
</tr>
<tr>
<td>Event ID</td>
<td>ID of the event</td>
</tr>
<tr>
<td>Description</td>
<td>Description of the event</td>
</tr>
<tr>
<td>Data</td>
<td>File name and line number from the FactoryTalk Event Archiver source code that generated the entry</td>
</tr>
</tbody>
</table>

These FactoryTalk Event Archiver events can occur and be entered in the Windows event log:

<table>
<thead>
<tr>
<th>Category</th>
<th>Event</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>1001</td>
<td>FactoryTalk Event Archiver Service started successfully in incremental mode</td>
</tr>
<tr>
<td>Start</td>
<td>1002</td>
<td>FactoryTalk Event Archiver Process started successfully in non-incremental mode</td>
</tr>
<tr>
<td>Start</td>
<td>1003</td>
<td>FactoryTalk Event Archiver failed to start; see the BATCHARC.LOG file for more information</td>
</tr>
<tr>
<td>Category</td>
<td>Event</td>
<td>Description</td>
</tr>
<tr>
<td>----------</td>
<td>-------</td>
<td>-------------</td>
</tr>
<tr>
<td>Start</td>
<td>1001</td>
<td>FactoryTalk Event Archiver Service started successfully in incremental mode</td>
</tr>
<tr>
<td>Start</td>
<td>1004</td>
<td>FactoryTalk Event Archiver failed to open the BATCHARC.LOG file</td>
</tr>
<tr>
<td>Start</td>
<td>1005</td>
<td>Service handler not installed</td>
</tr>
<tr>
<td>Start</td>
<td>1006</td>
<td>Bad Service Request</td>
</tr>
<tr>
<td>Start</td>
<td>3001</td>
<td>FactoryTalk Event Archiver Service paused</td>
</tr>
<tr>
<td>Start</td>
<td>3002</td>
<td>FactoryTalk Event Archiver Service continuing</td>
</tr>
<tr>
<td>Shutdown</td>
<td>4001</td>
<td>FactoryTalk Event Archiver Service shutting down</td>
</tr>
<tr>
<td>Setup</td>
<td>5001</td>
<td>FactoryTalk Event Archiver setup; new connect string = &lt;new string&gt;</td>
</tr>
</tbody>
</table>

See also

- [Output table on page 29](#)
- [Configure a custom archiver on page 35](#)
FactoryTalk Event Archiver with Reporting Services

Create simple to advanced batch reports using FactoryTalk Event Archiver with Reporting Services along with SQL Server and SQL Server Reporting Services (SSRS). FactoryTalk Event Archiver with Reporting Services can:

- Run the sample reports.
- Create reports using the samples as templates for data source, report datasets, or layouts.
- Create interactive and advanced custom batch reports that include selections, report detail drill-down, grouping, filtering, charts, and more, using the advanced sample reports.

Updates to the BatchHistoryEx database overwrite any customizations made to these reports.

See also

- Event Archiver reports on page 39
- Print reports on page 41
- Batch Listing report on page 42
- Batch Detail report on page 43
- Batch Summary report on page 45

FactoryTalk Event Archiver reports

These sample reports are available with FactoryTalk Event Archiver with Reporting Services.

Sample reports

<table>
<thead>
<tr>
<th>Report Name</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>010-Batch Listing</td>
<td>List of all batches within a specified time range.</td>
</tr>
<tr>
<td>020-Batch Summary</td>
<td>Summary of a batch.</td>
</tr>
<tr>
<td>030-Batch Detail</td>
<td>Details of a batch.</td>
</tr>
<tr>
<td>040-Material Usage</td>
<td>Summary of the usage of a specified material.</td>
</tr>
</tbody>
</table>
Chapter 6  FactoryTalk Event Archiver with Reporting Services

<table>
<thead>
<tr>
<th>Report Name</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>050-Forward Tracking</td>
<td>Lists batches for a lot or piece of equipment.</td>
</tr>
<tr>
<td>060-Backward Tracing</td>
<td>Displays the lot and the equipment used by a particular batch.</td>
</tr>
<tr>
<td>070-Batch Execution</td>
<td>Display the execution timeline of a specified batch.</td>
</tr>
<tr>
<td>080-Duration Comparison</td>
<td>Displays a comparison of batch and step durations.</td>
</tr>
<tr>
<td>090-Batch Exceptions</td>
<td>List of all batches with exceptions.</td>
</tr>
<tr>
<td>100-Operation Sequence</td>
<td>Displays the association and Operation Sequence data for FactoryTalk Batch and SequenceManager events.</td>
</tr>
<tr>
<td>110-Event Summary</td>
<td>Displays a custom event summary based on report parameter filter selection.</td>
</tr>
</tbody>
</table>

See also

Report parameters on page 40

Print reports on page 41

Report parameters

Enter report parameters to tailor the output of reports. Many reports require a start time and an end time, after which, other report parameters’ menus are populated.

Time and Date Range

Select time and date values from the report’s calendar controls, or type them. Leave either the start or the end time blank, but not both. When the start time is blank, the report queries all data back to the data’s origin. When the end time is blank, the report queries from the start time to the latest record inserted into the database. Enter time in the format hh:mm:ss AM or PM.

On an open report, the start date and time is the current time minus one day and the end date and time is the current date and time. The default time for the calendar control is 12:00:00 AM.

Filters

Filters are parameters that limit the data return. For example, a batch may run on three different units, but by selecting the filter parameter for specific units, the report only displays data for those units. There are filters for Recipes, Units, Process Cells, and other items, depending on the report.

Hyperlinks

Each report displays hyperlink icons that link to related reports. The hyperlinks pass specified parameters to the linked report. The linked report displays information based on the linking report’s data. Return to the original report with the browser’s back button.
Sorting

Some report columns allow the data to sort. Select the column arrows to perform the sort.

See also

Event Archiver reports on page 39

Print reports

Use these instructions to print a report.

To print reports:

1. From Select a format, select the print format.
2. Select the printer icon.

See also

Report caching on page 41
Report caching

Microsoft SQL Server Reporting Services (MSRS) offers report caching at the server level. This option improves reporting performance by storing a copy of the data and report in a local cache. Use the cache when navigating the reports using hyperlinks and drill-down features when the data parameters and underlying data sets do not change. The caching uses a time-out option to determine if the cache has expired. Manage caching and change the default settings through the MSRS report manager in Internet Explorer®.

See also

Print reports on page 41

Batch Listing report

The Batch Listing report lists all of the batches started within a specified time period. The report lists:

- Unique ID
- Batch ID
- Recipe name
- Start times
- Elapsed time
- State of each batch
The **Number of Rows** parameter, near the top of the page, defaults to the number of records fetched from the database. This parameter is editable.

![Batch Listing](image)

Hyperlinks

This report has hyperlinks to these reports:

<table>
<thead>
<tr>
<th>Icon</th>
<th>Destination Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>Yellow Tag</td>
<td>Batch Summary</td>
</tr>
<tr>
<td>Magnifying Glass</td>
<td>Batch Detail</td>
</tr>
<tr>
<td>White Arrow</td>
<td>Backward Tracing</td>
</tr>
<tr>
<td>Clock</td>
<td>Batch Execution</td>
</tr>
</tbody>
</table>

Filters

This report features filters for **Date** and **Time** range and **Recipe Name**. The **Batch ID** filter accepts search criteria in the form of a **Batch Name**. Use the asterisk (*) as a wildcard character.

See also

Event Archiver with Reporting Services on page 39

**Batch Detail report**

The Batch Detail report displays the details of a specified batch. This report includes a Phase table, header section, alarms and events, and state changes. The header includes the specifics of the batch, such as Batch ID and Recipe Name. The Phase table displays a list of the units used during a specified time period for a particular batch. The report lists:

- Unit Name
- Phase
- Start and End times
• Duration of each phase

Expand the Units in the Phase table to display the Equipment phase. Expand each Equipment phase to view the details, which include the recipe parameters and their values.

The Batch Detail report also displays Abnormal Exceptions that occurred during a batch, and the Setpoint versus Actual Summary report. The abnormal exception states are STOPPED, ABORTED, HELD, and FAULTED. It does not display any other states.

Hyperlinks

This report has hyperlinks to these reports:

<table>
<thead>
<tr>
<th>Icon</th>
<th>Destination Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>White Tag</td>
<td>Batch Listing</td>
</tr>
<tr>
<td>Yellow Tag</td>
<td>Batch Summary</td>
</tr>
<tr>
<td>White Arrow</td>
<td>Backward Tracing</td>
</tr>
<tr>
<td>Clock</td>
<td>Batch Execution</td>
</tr>
</tbody>
</table>

Filters

This report features filters for Recipe Name, Unique ID, Process Cell, and Unit Name. The Batch ID filter can filter data by inputting search criteria in the form of a Batch Name. Use the asterisk (*) as a wildcard character.

Setpoint vs. Actual Summary report
Batch Summary report

See also

Event Archiver with Reporting Services on page 39

The Batch Summary report includes two tables and a header section, the Phase, and Amount vs. Actual tables. The header displays the specifics of the batch, such as:

- Batch ID and Unique ID
- Recipe Name
- Area Name (can point to the FactoryTalk Batch Area or to the SequenceManager Controller ID)
- Process Cell (can point to the FTBatch Process Cell or to the SequenceManager Controller ID)
- Product Code
- Product Description
- Completion State
- Start Time
- End Time
- Duration

The Phase table displays a list of the units used for a specified batch. It lists the Unit Name, Phase, Start and End times, and the Duration of each phase, and includes Operation Sequences from both FactoryTalk Batch and SequenceManager events. The Amount vs. Actual table displays the Recipe Name along with the Recipe amount, the Actual amount used, and the Deviation from the recipe amount. If a measurement can be fractional, the decimal places round to three places.

Expand the Units in the Phase table to view the phases. Expand each phase to see the phase details.
Chapter 6  FactoryTalk Event Archiver with Reporting Services

The version of the FactoryTalk Batch report provides the following information:

- When the report was last modified.
- The version of the report that is currently installed.
- If a custom report is based on a particular version of a report, it shows which version of the report was used.

Hyperlinks

This report has hyperlinks to these reports:

<table>
<thead>
<tr>
<th>Icon</th>
<th>Destination Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>White Tag</td>
<td>Batch Listing</td>
</tr>
<tr>
<td>Magnifying Glass</td>
<td>Batch Detail</td>
</tr>
<tr>
<td>White Arrow</td>
<td>Backward Tracing</td>
</tr>
<tr>
<td>Clock</td>
<td>Batch Execution</td>
</tr>
</tbody>
</table>

Filters

This report features filters for the following:

<table>
<thead>
<tr>
<th>Filter name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Time</td>
<td>Filters based on start time. Select NULL to filter from the beginning of collected time.</td>
</tr>
</tbody>
</table>
## Material Usage report

The Material Usage report displays all of the batches that used a specified material during a specified time period. This report has a Material Usage table plus a header section. The header displays the specifics of the material and the amount consumed. The Material Usage table displays a list of batches that used the material. The report lists:

- Unique ID
- Batch ID
- Amount Consumed
- Lot ID
- Event Time

Expand the Batch to view associated Units used with the amount of material consumed.

<table>
<thead>
<tr>
<th>Filter name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>End Time</td>
<td>Filters based on end time. Select NULL to filter to the end of collected time.</td>
</tr>
<tr>
<td>Batch ID Filter</td>
<td>Accepts search criteria in the form of a Batch Name. Use the asterisk (*) as a wildcard character.</td>
</tr>
<tr>
<td>Process Cell</td>
<td>Filters based on the process cell name or controller serial number.</td>
</tr>
<tr>
<td>Recipe Name</td>
<td>Filters based on the recipe name or the sequence name.</td>
</tr>
<tr>
<td>Unit Name</td>
<td>Filters based on the unit name or the sequence unit ID.</td>
</tr>
<tr>
<td>Unique ID</td>
<td>Specifies which records display in the report.</td>
</tr>
</tbody>
</table>

See also

[FactoryTalk Event Archiver with Reporting Services](#) on page 39

The Material Usage report displays all of the batches that used a specified material during a specified time period. This report has a Material Usage table plus a header section. The header displays the specifics of the material and the amount consumed. The Material Usage table displays a list of batches that used the material. The report lists:

- Unique ID
- Batch ID
- Amount Consumed
- Lot ID
- Event Time

Expand the Batch to view associated Units used with the amount of material consumed.
Hyperlinks

This report has hyperlinks to these reports:

<table>
<thead>
<tr>
<th>Icon</th>
<th>Destination Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>White Tag</td>
<td>Batch Listing</td>
</tr>
<tr>
<td>Yellow Tag</td>
<td>Batch Summary</td>
</tr>
</tbody>
</table>

Filters

This report features filters for **Process Cell**, **Unit Name**, **Recipe Name**, and **Material Name**. The **Batch ID** filter can filter data by inputting search criteria in the form of a **Batch Name**. Use the asterisk (*) as a wildcard character.

The **Number of Rows** parameter, near the top of the page, defaults to the number of records fetched from the database. This parameter is editable.

See also

*Event Archiver with Reporting Services on page 39*

**Forward Tracking report**

The Forward Tracking report displays a list of the batches used during a specified lot, or a specified piece of equipment used for a batch.

The report includes a Lot table or an Equipment table depending on the Report Type selected from the parameter list.

The Lot table displays the batches associated with a specified lot. It lists:

- Unique ID
- Batch ID
- Amount Consumed
- Recipe Name
- Event Time

![Forward Tracking Report Image](image-url)
The Equipment table displays the batches associated with a specified piece of equipment. It lists:

- Unique ID
- Batch ID
- When the equipment was acquired and released

Hyperlinks

This report has hyperlinks to these reports:

<table>
<thead>
<tr>
<th>Icon</th>
<th>Destination Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>White Tag</td>
<td>Batch Listing</td>
</tr>
<tr>
<td>Yellow Tag</td>
<td>Batch Summary</td>
</tr>
</tbody>
</table>

Filters

This report features filters for Process Cell, Recipe Name, Report Type, and Equipment and Lot. The Batch ID filter filters data by inputting search criteria in the form of a Batch Name. Use the asterisk (*) as a wildcard character.

The Number of Rows parameter, near the top of the page, defaults to the number of records fetched from the database. This parameter is editable.

See also

Event Archiver with Reporting Services on page 39
Backward Tracing report

The Backward Tracing report displays the lot and equipment used for a particular batch. The Lot table displays the Lots used by a particular batch and the Equipment table displays the equipment used by the batch. The Lot table includes:

- Lot ID
- Material ID
- Amount Consumed
- Recipe Name
- Event Time

Expand the lot to display the units used by the batch. The Equipment table displays the equipment used by the batch and when it was acquired and released.

Hyperlinks

This report has hyperlinks to these reports:

<table>
<thead>
<tr>
<th>Icon</th>
<th>Destination Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>White Tag</td>
<td>Batch Listing</td>
</tr>
<tr>
<td>Yellow Tag</td>
<td>Batch Summary</td>
</tr>
<tr>
<td>Magnifying Glass</td>
<td>Batch Detail</td>
</tr>
<tr>
<td>Clock</td>
<td>Batch Execution</td>
</tr>
</tbody>
</table>

Filters

This report includes filters for Process Cell, Unit Name, Recipe Name, and Unique ID. Use the Batch ID filter to enter search criteria in the form of a Batch Name. Use the asterisk (*) as a wildcard character.

See also

Event Archiver with Reporting Services on page 39
Duration Comparison report

The Duration Comparison report displays a comparison of batch durations. Select multiple batches from the list to display them in the report's chart.

A table in the report outlines the properties of the displayed batches in more detail.

Hyperlinks

There is a hyperlink to return to the Batch Listing report. This hyperlink does not pass any parameters back to the Batch Listing report.

Filters

This report includes filters for Process Cell, Recipe Name, Unit Name, and Unique ID. The Batch ID filter accepts search criteria in the form of a Batch Name. Use the asterisk (*) as a wildcard character.

See also

Event Archiver with Reporting Services on page 39
Batch Execution report

The Batch Execution report displays an Execution Timeline of a batch. The Batch Execution report includes Operation Sequences from both FactoryTalk Batch and SequenceManager events. This report includes a header section and charts. The header displays the specifics of the batch, such as:

- Batch ID and Unique ID
- Recipe Name
- Area Name (can point to either the FactoryTalk Batch Area or to the SequenceManager Controller ID)
- Process Cell
- Product Code
- Product Description
- Completion State
- Start Time
- End Time
- Duration

The charts are Unit Procedures, Operations, and Phases. Each displays the duration in seconds.
Version

The version of the FactoryTalk Batch report provides the following information:

- When the report was last modified.
- The version of the report that is currently installed.
- If a custom report is based on a particular version of a report, it shows which version of the report was used.

Hyperlinks

There is a hyperlink to return to the Batch Listing report. This hyperlink does not pass any parameters back to the Batch Listing report.

Filters

This report features filters for the following:

<table>
<thead>
<tr>
<th>Filter name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Time</td>
<td>Filters based on start time. Select NULL to filter from the beginning of collected time.</td>
</tr>
<tr>
<td>End Time</td>
<td>Filters based on end time. Select NULL to filter to the end of collected time.</td>
</tr>
<tr>
<td>Batch ID Filter</td>
<td>Accepts search criteria in the form of a Batch Name. Use the asterisk (*) as a wildcard character.</td>
</tr>
<tr>
<td>Process Cell</td>
<td>Filters based on the process cell name or controller serial number.</td>
</tr>
<tr>
<td>Recipe Name</td>
<td>Filters based on the recipe name or the sequence name.</td>
</tr>
<tr>
<td>Unit Name</td>
<td>Filters based on the unit name or the sequence unit ID.</td>
</tr>
<tr>
<td>Unique ID</td>
<td>Specifies which records display in the report.</td>
</tr>
</tbody>
</table>

See also

Event Archiver with Reporting Services on page 39

Batch Exceptions report

The Batch Exceptions report displays Abnormal Exceptions that occurred during a batch. The abnormal exception states are STOPPED, ABORTED, HELD, and FAULTED. Other states are not included in this report.
This report includes a header section that displays the specifics of the batch, such as:

<table>
<thead>
<tr>
<th>Header name</th>
<th>For FactoryTalk Batch, displays</th>
<th>For SequenceManager, displays</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Time</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>End Time</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Batch Failures</td>
<td>Failure message.</td>
<td>The ordinal value of the failure message.</td>
</tr>
</tbody>
</table>

The Batch Exceptions report includes Operation Sequences from both FactoryTalk Batch and SequenceManager events, and lists:

- Abnormal State Changes
- Alarms and Events
- Batch Failures
- Out of Tolerance events

Out of Tolerance events
Version

The version of the FactoryTalk Batch report provides the following information:

- When the report was last modified.
- The version of the report that is currently installed.
- If a custom report is based on a particular version of a report, it shows which version of the report was used.

Hyperlinks

There is a hyperlink to return to the Batch Listing report. This hyperlink does not pass any parameters back to the Batch Listing report.

Filters

This report features filters for the following:

<table>
<thead>
<tr>
<th>Filter name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Time</td>
<td>Filters based on start time. Select NULL to filter from the beginning of collected time.</td>
</tr>
<tr>
<td>End Time</td>
<td>Filters based on end time. Select NULL to filter to the end of collected time.</td>
</tr>
<tr>
<td>Batch ID Filter</td>
<td>Accepts search criteria in the form of a Batch Name. Use the asterisk (*) as a wildcard character.</td>
</tr>
<tr>
<td>Process Cell</td>
<td>Filters based on the process cell name.</td>
</tr>
<tr>
<td>Recipe Name</td>
<td>Filters based on the recipe name.</td>
</tr>
<tr>
<td>Unit Name</td>
<td>Filters based on the unit name.</td>
</tr>
<tr>
<td>Unique ID</td>
<td>Specifies which records display in the report.</td>
</tr>
</tbody>
</table>

See also

FactoryTalk Event Archiver with Reporting Services on page 39

Operation Sequence report

The Operation Sequence report displays the association and Operation Sequence data for FactoryTalk Batch and SequenceManager events. This report includes a header section and the Associated Operation Sequences and Associated Operation Sequences - Details tables. The header displays the specifics of the batch, such as:

<table>
<thead>
<tr>
<th>Header name</th>
<th>For FactoryTalk Batch, displays</th>
<th>For SequenceManager, displays</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch ID</td>
<td>Batch ID</td>
<td>Sequence ID</td>
</tr>
<tr>
<td>Unique ID</td>
<td>FactoryTalk Batch Unique ID</td>
<td>SequenceManager Unique ID</td>
</tr>
<tr>
<td>Recipe Name</td>
<td>Recipe Name</td>
<td>Sequence Name</td>
</tr>
<tr>
<td>Header name</td>
<td>For FactoryTalk Batch, displays</td>
<td>For SequenceManager, displays</td>
</tr>
<tr>
<td>--------------------</td>
<td>---------------------------------</td>
<td>------------------------------</td>
</tr>
<tr>
<td>Area Name</td>
<td>Area</td>
<td>Controller Serial Number</td>
</tr>
<tr>
<td>Process Cell</td>
<td>Process Cell</td>
<td>Controller Serial Number</td>
</tr>
<tr>
<td>Product Code</td>
<td>Product Code</td>
<td>Blank for SequenceManager</td>
</tr>
<tr>
<td>Product Description</td>
<td>Procedure Description</td>
<td>Extended Text</td>
</tr>
<tr>
<td>Completion State</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Start Time</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>End Time</td>
<td>—</td>
<td>—</td>
</tr>
<tr>
<td>Duration</td>
<td>—</td>
<td>—</td>
</tr>
</tbody>
</table>

**Associated Operation Sequences**

For each associated Operation Sequence ID, select the Yellow Tag 🟢 to open the Batch Summary report. See the **Batch Summary report** for more information.

**Associated Operation Sequences - Details**
Version

The version of the FactoryTalk Batch report provides the following information:

- When the report was last modified.
- The version of the report that is currently installed.
- If a custom report is based on a particular version of a report, it shows which version of the report was used.

Hyperlinks

This report has hyperlinks to these reports:

<table>
<thead>
<tr>
<th>Icon</th>
<th>Destination Report</th>
</tr>
</thead>
<tbody>
<tr>
<td>White Tag</td>
<td>Batch Listing</td>
</tr>
<tr>
<td>Yellow Tag</td>
<td>Batch Summary</td>
</tr>
<tr>
<td>White Arrow</td>
<td>Backward Tracing</td>
</tr>
<tr>
<td>Clock</td>
<td>Batch Execution</td>
</tr>
</tbody>
</table>

Filters

This report features filters for the following:

<table>
<thead>
<tr>
<th>Filter name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Time</td>
<td>Filters based on start time. Select NULL to filter from the beginning of collected time.</td>
</tr>
<tr>
<td>End Time</td>
<td>Filters based on end time. Select NULL to filter to the end of collected time.</td>
</tr>
<tr>
<td>Batch ID Filter</td>
<td>Accepts search criteria in the form of a Batch Name. Use the asterisk (*) as a wildcard character.</td>
</tr>
<tr>
<td>Process Cell</td>
<td>Filters based on the process cell name or controller serial number.</td>
</tr>
<tr>
<td>Recipe Name</td>
<td>Filters based on the recipe name or the sequence name.</td>
</tr>
<tr>
<td>Unit Name</td>
<td>Filters based on the unit name or the sequence unit ID.</td>
</tr>
<tr>
<td>Unique ID</td>
<td>Specifies which records display in the report.</td>
</tr>
</tbody>
</table>

See also

- FactoryTalk Event Archiver with Reporting Services on page 39
- Batch Summary report on page 45
Event Summary report

The Event Summary report displays a custom event summary based on report parameter filter selection. The report displays event records from one event journal based on the Unique ID specified. This report includes a header section and an Event Record Data table. The header displays the specifics of the batch:

```
110-Event Summary
Event File Name: WNAUSPAZF4J7R52iBATCHCTLISampleDemo1UJOURNALS\3.ev
Unique ID: 3
Batch ID: TESTBATCH2
Recipe Name: CLS_TRIPLECHOCOLATE
Area Name: AREA1
Process Cell: WEST_PARLOR
Product Code: TCP-101
Product Description: Triple Chocolate Premium - class based
Report Status: READY
Report Generated By: RA-INTSMoon2
Report Time: 9/20/2017 1:50:04 PM
Row Count: 263
```

The header includes:

- Event file name
- Batch ID and Unique ID
- Recipe Name
- Area Name (can point to either the FactoryTalk Batch Area or to the SequenceManager Controller ID)
- Process Cell (can point to the FTBatch Process Cell or to the SequenceManager Controller ID)
- Product Code
- Product Description
- Report Status
- Report Generated By
- Report Time
- Row Count
Columns in the Event Record Data table can be sorted in ascending or descending order.

### Version

The version of the FactoryTalk Batch report provides the following information:

- When the report was last modified.
- The version of the report that is currently installed.
- If a custom report is based on a particular version of a report, it shows which version of the report was used.

### Filters

This report features filters for the following:

<table>
<thead>
<tr>
<th>Filter name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Time</td>
<td>Filters based on start time. Select NULL to filter from the beginning of collected time.</td>
</tr>
<tr>
<td>End Time</td>
<td>Filters based on end time. Select NULL to filter to the end of collected time.</td>
</tr>
<tr>
<td>Batch ID Filter</td>
<td>Accepts search criteria in the form of a Batch Name. Use the asterisk (*) as a wildcard character.</td>
</tr>
<tr>
<td>Process Cell</td>
<td>Filters based on the process cell or controller ID.</td>
</tr>
<tr>
<td>Recipe Name</td>
<td>Filters based on the recipe name.</td>
</tr>
<tr>
<td>Filter name</td>
<td>Description</td>
</tr>
<tr>
<td>-------------</td>
<td>-------------</td>
</tr>
<tr>
<td>Unit Name</td>
<td>Filters based on the unit name or unit ID.</td>
</tr>
<tr>
<td>Unique ID</td>
<td>Specifies which event records display in the report.</td>
</tr>
<tr>
<td>Event</td>
<td>Specifies which event records to include or exclude in the report based on event type.</td>
</tr>
<tr>
<td>Description</td>
<td>Specifies which event records to include or exclude in the report based on the event description.</td>
</tr>
<tr>
<td>Value</td>
<td>Specifies which event records to include or exclude in the report based on event value, such as recipe header information, scale factor, recipe value change, system message, state change, and so on.</td>
</tr>
<tr>
<td>Columns</td>
<td>Specifies which columns to display in the report.</td>
</tr>
</tbody>
</table>

See also

[FactoryTalk Event Archiver with Reporting Services](#) on page 39
Chapter 7

Report details

These sections define report details.

See also

- Report files Report Definition Language on page 61
- Datasets on page 61
- Parameters on page 62
- Code on page 62
- Batch listing on page 63

Report files Report Definition Language (RDL)

SQL Server reports are Extensible Markup Language (XML) files that describe the report’s layout and data requirements. The files are Report Definition Language (RDL) files and have the file extension .rdl.

Three high-level sections in a typical RDL file:

- **Page style**: This includes the objects to display, including fields, images, graphs, tables.
- **Field definitions**: The extended attributes of fields populated with formulas, dynamic data, or database derived data.
- **Parameters and Database connections**: Parameters can be user-defined or passed in from another application. The database connections and queries are for pulling data into the report.

See also

- Report details on page 61

Datasets

Reports display the results of queries executed against a SQL database. Each report has a dataset embedded.

Datasets contain data tables that temporarily store data used by the report. The dataset provides the report with a local, in-memory cache of the data to work with.
A dataset can be an embedded query or it can point to a stored procedure in the SQL database. Position the objects of a dataset in the report’s form to change the report’s layout.

See also

Report details on page 61

Parameters

In Reporting Services, use parameters to:

- Specify the data to use in a report.
- Connect related reports.
- Vary report presentation.

To design a report that uses parameters effectively, determine:

- How parameters and dataset queries work together.
- How parameters and expressions work together.
- How parameters are managed on the report server for a published report.
- What questions a report answers.

The parameter choices affect the report’s design and layout.

Add query parameters to a dataset query with the query designers or the Dataset Properties dialog box. After a query creates with parameters, Reporting Services automatically links query parameters to report parameters with the same name. Add and modify report parameters in the Report Data pane.

See also

Report details on page 61

Code

There are two custom functions used with the reports. GetRecipeName parses out and returns the Recipe name from the file path, and GetBatchListingURL returns the address of the Batch Listing report.

GetRecipeName

Public Function GetRecipeName(sPath as String) as String
Dim PathArray() as string
Dim sRecipeName as String
PathArray=Split(sPath, ",",-1,0)
sRecipeName = Left( PathArray(Ubound(PathArray)), len( PathArray(Ubound(PathArray))) - 4)
Return sRecipeName
End Function
GetBatchListingURL

Public Function GetBatchListingURL(sURL as String) as String
Dim sReturn as String
sURL= Replace(sURL, "http://",""
)sReturn = "http://" & Left(sURL, InStr(sURL, "/") ) & 
"Reports/Pages/Report.aspx?ItemPath=%2fBatch+Reports%2fBatch+Listing"
Return sReturn
End Function

See also

Report details on page 61

Batch listing

These tables outline the different components of the report.

Datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Stored Procedure</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetBatchListByDate</td>
<td>spGetBatchListByDate</td>
<td>Gets the Batches executed between two dates</td>
</tr>
<tr>
<td>GetRecipeNames</td>
<td>spGetRecipeNamesBYDate</td>
<td>Gets the Recipes used between two dates</td>
</tr>
<tr>
<td>GetUnitName</td>
<td>spGetUnitNameByDate</td>
<td>Gets the Units used between two dates</td>
</tr>
<tr>
<td>GetProcessCell</td>
<td>spGetProcessCellByDate</td>
<td>Gets the Process Cells used between two dates</td>
</tr>
</tbody>
</table>

Report Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Where used</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>StartTime</td>
<td>All procedures</td>
<td>Start Time</td>
</tr>
<tr>
<td>EndTime</td>
<td>All procedures</td>
<td>End Time</td>
</tr>
<tr>
<td>SearchWord</td>
<td>spGetBatchListByDate</td>
<td>User Defined Batch Filter</td>
</tr>
<tr>
<td>ProcessCell</td>
<td>NA</td>
<td>Process Cell Filter</td>
</tr>
<tr>
<td>RecipeName</td>
<td>NA</td>
<td>Recipe Name Filter</td>
</tr>
<tr>
<td>UnitName</td>
<td>NA</td>
<td>Unit Name Filter</td>
</tr>
</tbody>
</table>

Heading Data

<table>
<thead>
<tr>
<th>Object</th>
<th>Data Item</th>
<th>Data Source</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Time</td>
<td>StartTime</td>
<td>User input</td>
<td>Parameter</td>
</tr>
<tr>
<td>End Time</td>
<td>EndTime</td>
<td>User input</td>
<td>Parameter</td>
</tr>
</tbody>
</table>
Table Datasets

<table>
<thead>
<tr>
<th>Table Dataset</th>
<th>Dataset</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>table1_GetBatchListByDate</td>
<td>GetBatchListByDate</td>
<td>Batch List</td>
</tr>
</tbody>
</table>

See also

[Report details on page 61](#)

**Batch summary**

The following tables outline the different components of the report.

**Datasets**

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Stored Procedure</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>BHBatch</td>
<td>Embedded query</td>
<td>Gets the Batches executed between two dates</td>
</tr>
<tr>
<td>BHPhase</td>
<td>Embedded query</td>
<td>Gets the Recipes used between two dates</td>
</tr>
<tr>
<td>BHParamValue</td>
<td>Embedded query</td>
<td>Gets the Units used between two dates</td>
</tr>
<tr>
<td>GetUniqueID</td>
<td>spGetUniqueIDByDate</td>
<td>Gets the Unique IDs used between two dates</td>
</tr>
<tr>
<td>GetProcessCell</td>
<td>spGetProcessCellByDate</td>
<td>Gets the Process Cells used between two dates</td>
</tr>
<tr>
<td>GetRecipeNames</td>
<td>spGetRecipeNamesByDate</td>
<td>Gets the Recipes used between two dates</td>
</tr>
<tr>
<td>GetUnitName</td>
<td>spGetUnitNameByDate</td>
<td>Gets the Units used between two dates</td>
</tr>
</tbody>
</table>

**Report Parameters**

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Where used</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>StartTime</td>
<td>All Procedures</td>
<td>Start Time; can be set to Null</td>
</tr>
<tr>
<td>EndTime</td>
<td>All Procedures</td>
<td>End Time; can be set to Null</td>
</tr>
<tr>
<td>SearchWord</td>
<td>NA</td>
<td>User Defined Batch Filter</td>
</tr>
<tr>
<td>ProcessCell</td>
<td>NA</td>
<td>Process Cell Filter</td>
</tr>
<tr>
<td>RecipeName</td>
<td>NA</td>
<td>Recipe Name Filter</td>
</tr>
<tr>
<td>UniqueID</td>
<td>NA</td>
<td>Unique ID Filter</td>
</tr>
<tr>
<td>UnitName</td>
<td>NA</td>
<td>Unit Name Filter</td>
</tr>
</tbody>
</table>

**Heading Data**

<table>
<thead>
<tr>
<th>Object</th>
<th>Data Item</th>
<th>Data Source</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch ID</td>
<td>batchid</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Unique ID</td>
<td>uniqueid</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Recipe Name</td>
<td>recipefilename</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Area Name</td>
<td>area</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
</tbody>
</table>
Object | Data Item | Data Source | Type
---|---|---|---
Process Cell | processcell | BHPhase | Data
Product Code | productcode | BHBatch | Data
Product Description | productdescription | BHBatch | Data
Completion State | completionstate | BHBatch | Data
Start Time | starttimeexecution | BHBatch | Data
End Time | endtimeexecution | BHBatch | Data
Duration | endtimeexecution-starttimeexecution | BHBatch | Calculated

Table Datasets

<table>
<thead>
<tr>
<th>Table</th>
<th>Dataset</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>table_BHPhase</td>
<td>BHPhase</td>
<td>Phase Data</td>
</tr>
<tr>
<td>Amt vs Actual (subreport)</td>
<td>BHParamValue</td>
<td>Amount vs Actual Data</td>
</tr>
</tbody>
</table>

See also

[Batch detail on page 67](#)

[Report details on page 61](#)

**Forward tracking**

The following tables outline the different components of the report.

Datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Stored Procedure</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetLot_UnitUsage</td>
<td>spGetLotUsageDataByDate</td>
<td>Gets the Lots used between two dates</td>
</tr>
<tr>
<td>GetLot_UnitName</td>
<td>spGetLotID_UnitNameByDate</td>
<td>Gets the Lot Names used between two dates</td>
</tr>
<tr>
<td>GetEquipmentUsage</td>
<td>spGetEquipmentUsageDataByDate</td>
<td>Gets the equipment used between two dates</td>
</tr>
<tr>
<td>GetRecipeNames</td>
<td>spGetRecipeNamesBYDate</td>
<td>Gets the Recipes used between two dates</td>
</tr>
<tr>
<td>GetProcessCell</td>
<td>spGetProcessCellByDate</td>
<td>Gets the Process Cells used between two dates</td>
</tr>
</tbody>
</table>

Report Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Where used</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>StartTime</td>
<td>All Procedures</td>
<td>Start Time</td>
</tr>
<tr>
<td>EndTime</td>
<td>All Procedures</td>
<td>End Time</td>
</tr>
<tr>
<td>SearchWord</td>
<td>NA</td>
<td>User Defined Batch Filter</td>
</tr>
<tr>
<td>ProcessCell</td>
<td>NA</td>
<td>Process Cell Filter</td>
</tr>
<tr>
<td>RecipeName</td>
<td>NA</td>
<td>Recipe Name Filter</td>
</tr>
</tbody>
</table>
### Parameter

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Where used</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>ReportType</td>
<td>spGetLotUsageDataByDate, spGetEquipmentUsageDataByDate</td>
<td>Query Parameter</td>
</tr>
<tr>
<td>QueryParam</td>
<td>GetLot_UnitName</td>
<td>Lot or Equipment Parameter</td>
</tr>
</tbody>
</table>

### Heading Data

<table>
<thead>
<tr>
<th>Object</th>
<th>Data Item</th>
<th>Data Source</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Time</td>
<td>StartTime</td>
<td>User input</td>
<td>Parameter</td>
</tr>
<tr>
<td>End Time</td>
<td>EndTime</td>
<td>User input</td>
<td>Parameter</td>
</tr>
<tr>
<td>Report Type</td>
<td>QueryParam</td>
<td>User input</td>
<td>Parameter</td>
</tr>
</tbody>
</table>

### Table Datasets

<table>
<thead>
<tr>
<th>Table</th>
<th>Dataset</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>table_GetEquipmentUsage</td>
<td>GetEquipmentUsage</td>
<td>Equipment Usage</td>
</tr>
<tr>
<td>table_GetLot_UnitUsage</td>
<td>GetLot_UnitUsage</td>
<td>Lot Usage</td>
</tr>
</tbody>
</table>

### See also

- Backward tracing on page 66
- Report details on page 61

## Backward tracing

These tables outline the different components of the report.

### Datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Stored Procedure</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetRecipeNames</td>
<td>spGetRecipeNamesByDate</td>
<td>Gets the Recipes used between two dates</td>
</tr>
<tr>
<td>GetUnitName</td>
<td>spGetUnitNameByDate</td>
<td>Gets the Units used between two dates</td>
</tr>
<tr>
<td>GetProcessCell</td>
<td>spGetProcessCellByDate</td>
<td>Gets the Process Cells used between two dates</td>
</tr>
<tr>
<td>GetUniqueID</td>
<td>spGetUniqueIDByDate</td>
<td>Gets the Unique IDs used between two dates</td>
</tr>
<tr>
<td>GetLotID</td>
<td>spGetLotIDByDate</td>
<td>Gets the Lot Names used between two dates</td>
</tr>
<tr>
<td>GetBatchUsageData</td>
<td>spGetBatchUsageDataByDate</td>
<td>Gets the Lots used by batches</td>
</tr>
<tr>
<td>GetEquipmentNames</td>
<td>spGetLotID_UnitNameByDate</td>
<td>Gets the Equipment used between two dates</td>
</tr>
<tr>
<td>GetEquipmentUsed-ByBatch</td>
<td>spGetEquipmentUsedByBatchByDate</td>
<td>Gets the Equipment used by batches</td>
</tr>
</tbody>
</table>
Report Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Where used</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>StartTime</td>
<td>All Procedures</td>
<td>Start Time</td>
</tr>
<tr>
<td>EndTime</td>
<td>All Procedures</td>
<td>End Time</td>
</tr>
<tr>
<td>SearchWord</td>
<td>NA</td>
<td>User Defined Batch Filter</td>
</tr>
<tr>
<td>ProcessCell</td>
<td>NA</td>
<td>Process Cell Filter</td>
</tr>
<tr>
<td>RecipeName</td>
<td>NA</td>
<td>Recipe Name Filter</td>
</tr>
<tr>
<td>ReportType</td>
<td>NA</td>
<td>Query Parameter</td>
</tr>
<tr>
<td>QueryParam</td>
<td>GetUniqueID</td>
<td>Query Parameter</td>
</tr>
<tr>
<td>UnitName</td>
<td>GetUnitName</td>
<td>Unit Name Filter</td>
</tr>
</tbody>
</table>

Heading Data

<table>
<thead>
<tr>
<th>Object</th>
<th>Data Item</th>
<th>Data Source</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Time</td>
<td>StartTime</td>
<td>User input</td>
<td>Parameter</td>
</tr>
<tr>
<td>End Time</td>
<td>EndTime</td>
<td>User input</td>
<td>Parameter</td>
</tr>
<tr>
<td>Unique ID</td>
<td>QueryParam</td>
<td>User input</td>
<td>Parameter</td>
</tr>
<tr>
<td>Batch ID</td>
<td>BatchID</td>
<td>GetBatchUsageData</td>
<td>Data</td>
</tr>
<tr>
<td>Recipe Name</td>
<td>Recipefilename</td>
<td>GetBatchUsageData</td>
<td>Data</td>
</tr>
<tr>
<td>Area Name</td>
<td>area</td>
<td>GetBatchUsageData</td>
<td>Data</td>
</tr>
<tr>
<td>Process Cell</td>
<td>Processcell</td>
<td>GetBatchUsageData</td>
<td>Data</td>
</tr>
<tr>
<td>Product Description</td>
<td>productdescription</td>
<td>GetBatchUsageData</td>
<td>Data</td>
</tr>
<tr>
<td>Completion State</td>
<td>completionstate</td>
<td>GetBatchUsageData</td>
<td>Data</td>
</tr>
</tbody>
</table>

Table Datasets

<table>
<thead>
<tr>
<th>Table</th>
<th>Dataset</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>table_GetBatchUsageData</td>
<td>GetBatchUsageData</td>
<td>Batch Usage</td>
</tr>
<tr>
<td>table_GetEquipmentUsed-ByBatch</td>
<td>GetEquipmentUsed-ByBatch</td>
<td>Equipment Usage</td>
</tr>
</tbody>
</table>

See also

Forward tracking on page 65

These tables outline the different components of the report.

Datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Stored Procedure</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>BHBatch</td>
<td>Embedded query</td>
<td>Gets the Batches executed between two dates</td>
</tr>
</tbody>
</table>
### Dataset

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Stored Procedure</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>BHPhase</td>
<td>Embedded query</td>
<td>Gets the Recipes used between two dates</td>
</tr>
<tr>
<td>BHPParamValue</td>
<td>Embedded query</td>
<td>Gets the Units used between two dates</td>
</tr>
<tr>
<td>GetUniqueID</td>
<td>spGetUniqueIDByDate</td>
<td>Gets the Unique IDs used between two dates</td>
</tr>
<tr>
<td>GetProcessCell</td>
<td>spGetProcessCellByDate</td>
<td>Gets the Process Cells used between two dates</td>
</tr>
<tr>
<td>AbnormalStateChanges</td>
<td>Embedded query</td>
<td>Gets the Abnormal State Changes</td>
</tr>
<tr>
<td>BHPPhase_BHPParameters</td>
<td>Embedded query</td>
<td>Gets the Phase Data and Parameters</td>
</tr>
</tbody>
</table>

### Report Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Where used</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>StartTime</td>
<td>All Procedures</td>
<td>Start Time</td>
</tr>
<tr>
<td>EndTime</td>
<td>All Procedures</td>
<td>End Time</td>
</tr>
<tr>
<td>SearchWord</td>
<td>NA</td>
<td>User Defined Batch Filter</td>
</tr>
<tr>
<td>ProcessCell</td>
<td>NA</td>
<td>Process Cell Filter</td>
</tr>
<tr>
<td>RecipeName</td>
<td>NA</td>
<td>Recipe Filter</td>
</tr>
<tr>
<td>UniqueID</td>
<td>NA</td>
<td>Unique ID Filter</td>
</tr>
<tr>
<td>UnitName</td>
<td>NA</td>
<td>Unit Name Filter</td>
</tr>
</tbody>
</table>

### Heading Data

<table>
<thead>
<tr>
<th>Object</th>
<th>Data Item</th>
<th>Data Source</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Time</td>
<td>starttimeexecution</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>End Time</td>
<td>endtimeexecution</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Duration</td>
<td>End Time – Start Time</td>
<td>BHBatch</td>
<td>Calculated</td>
</tr>
<tr>
<td>Batch ID</td>
<td>batchid</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Unique ID</td>
<td>uniquoid</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Recipe Name</td>
<td>recipefilename</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Area Name</td>
<td>area</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Process Cell</td>
<td>processcell</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Product Description</td>
<td>productdescription</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Product Code</td>
<td>productcode</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Completion State</td>
<td>completionstate</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
</tbody>
</table>

### Table Datasets

<table>
<thead>
<tr>
<th>Table</th>
<th>Dataset</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>table_BHPhase</td>
<td>BHPParam_BHPhase_BHMaterial</td>
<td>Phase Data</td>
</tr>
</tbody>
</table>
## Material usage

These tables outline the different components of the report.

### Datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Stored Procedure</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetMaterialID</td>
<td>spGetMaterialIDByDate</td>
<td>Gets the Material IDs used between two dates</td>
</tr>
<tr>
<td>GetMaterialUsageData</td>
<td>spGetMaterialUsageDataByDate</td>
<td>Gets the Material amounts used between two dates</td>
</tr>
<tr>
<td>GetRecipeNames</td>
<td>spGetRecipeNamesBYDate</td>
<td>Gets the Recipes used between two dates</td>
</tr>
<tr>
<td>GetUnitName</td>
<td>spGetUnitNameByDate</td>
<td>Gets the Units used between two dates</td>
</tr>
<tr>
<td>GetProcessCell</td>
<td>spGetProcessCellByDate</td>
<td>Gets the Process Cells used between two dates</td>
</tr>
</tbody>
</table>

### Report Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Where used</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>StartTime</td>
<td>All Procedures</td>
<td>Start Time</td>
</tr>
<tr>
<td>EndTime</td>
<td>All Procedures</td>
<td>End Time</td>
</tr>
<tr>
<td>SearchWord</td>
<td>NA</td>
<td>User Defined Batch Filter</td>
</tr>
<tr>
<td>ProcessCell</td>
<td>NA</td>
<td>Process Cell Filter</td>
</tr>
<tr>
<td>RecipeName</td>
<td>NA</td>
<td>Recipe Name Filter</td>
</tr>
<tr>
<td>UnitName</td>
<td>NA</td>
<td>Unit Name Filter</td>
</tr>
<tr>
<td>MaterialName</td>
<td>GetMaterialUsageData</td>
<td>Material Parameter</td>
</tr>
</tbody>
</table>

### Heading Data

<table>
<thead>
<tr>
<th>Object</th>
<th>Data Item</th>
<th>Data Source</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Time</td>
<td>StartTime</td>
<td>User input</td>
<td>Parameter</td>
</tr>
<tr>
<td>End Time</td>
<td>EndTime</td>
<td>User input</td>
<td>Parameter</td>
</tr>
<tr>
<td>Material Name</td>
<td>materialname</td>
<td>GetmaterialUsageData</td>
<td>Data</td>
</tr>
</tbody>
</table>
Object      | Data Item   | Data Source                | Type    |
------------|-------------|-----------------------------|---------|
Total       | Consumed    | TotalUsed                  | GetmaterialUsageData | Data |

Table Datasets

<table>
<thead>
<tr>
<th>Table</th>
<th>Dataset</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetmaterialUsageData</td>
<td>GetmaterialUsageData</td>
<td>Material Usage</td>
</tr>
</tbody>
</table>

See also

Report details on page 61

These tables outline the different components of the report.

Datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Stored Procedure</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetRecipeNames</td>
<td>spGetRecipeNamesByDate</td>
<td>Gets the Recipes used between two dates</td>
</tr>
<tr>
<td>GetUnitNameByDate</td>
<td>spGetUnitNameByDate</td>
<td>Gets the Units used between two dates</td>
</tr>
<tr>
<td>GetBatchListByDate</td>
<td>spGetBatchListByDate</td>
<td>Gets the Batches executed between two dates</td>
</tr>
<tr>
<td>GetBatchListDuration</td>
<td>spGetBatchListDuration</td>
<td>Gets the Duration of Batches executed between two dates</td>
</tr>
<tr>
<td>GetUniqueID</td>
<td>spGetUniqueIDByDate</td>
<td>Gets ALL Unique IDs from Database</td>
</tr>
<tr>
<td>GetProcessCell</td>
<td>spGetProcessCellByDate</td>
<td>Gets ALL Process cells from Database</td>
</tr>
<tr>
<td>GetUnitName</td>
<td>spGetUnitNameByDate</td>
<td>Gets ALL Unit Names from Database</td>
</tr>
</tbody>
</table>

Report Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Where used</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>StartTime</td>
<td>All Procedures</td>
<td>Start Time</td>
</tr>
<tr>
<td>EndTime</td>
<td>All Procedures</td>
<td>End Time</td>
</tr>
<tr>
<td>SearchWord</td>
<td>NA</td>
<td>User Defined Batch Filter</td>
</tr>
<tr>
<td>ProcessCell</td>
<td>NA</td>
<td>Process Cell Filter</td>
</tr>
<tr>
<td>RecipeName</td>
<td>NA</td>
<td>Recipe Name Filter</td>
</tr>
<tr>
<td>UnitName</td>
<td>NA</td>
<td>Unit Name Filter</td>
</tr>
<tr>
<td>UniqueID</td>
<td>NA</td>
<td>Unique ID Filter</td>
</tr>
</tbody>
</table>
Heading Data

<table>
<thead>
<tr>
<th>Object</th>
<th>Data Item</th>
<th>Data Source</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Time</td>
<td>StartTime</td>
<td>User input</td>
<td>Parameter</td>
</tr>
<tr>
<td>End Time</td>
<td>EndTime</td>
<td>User input</td>
<td>Parameter</td>
</tr>
</tbody>
</table>

Table and Chart Datasets

<table>
<thead>
<tr>
<th>Table</th>
<th>Dataset</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>chart_</td>
<td>GetBatchListDuration</td>
<td>Batch Duration</td>
</tr>
<tr>
<td>GetBatchListDuration</td>
<td></td>
<td></td>
</tr>
<tr>
<td>table_</td>
<td>GetBatchListDuration</td>
<td>Batch Duration</td>
</tr>
<tr>
<td>GetBatchListDuration</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

See also

Report details on page 61

Batch execution

The following tables outline the different components of the report.

Datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Stored Procedure</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>BHPhase</td>
<td>Embedded Query</td>
<td>Gets Phase data for a given Unique ID</td>
</tr>
<tr>
<td>BHBatch</td>
<td>Embedded Query</td>
<td>Gets Batch data for a given Unique ID</td>
</tr>
<tr>
<td>BHOperation</td>
<td>Embedded Query</td>
<td>Gets ALL Operations from Database</td>
</tr>
<tr>
<td>GetUniqueID</td>
<td>spGetUniqueIDByDate</td>
<td>Gets ALL Unique IDs from Database</td>
</tr>
<tr>
<td>GetUnitName</td>
<td>spGetUnitNameByDate</td>
<td>Gets ALL Unit Names from Database</td>
</tr>
<tr>
<td>BHUnitProcedure</td>
<td>Embedded Query</td>
<td>Gets Unit Procedures for a given Unique ID</td>
</tr>
<tr>
<td>sGetRecipeNames</td>
<td>spGetRecipeNamesByDate</td>
<td>Gets ALL Recipes from Database</td>
</tr>
<tr>
<td>GetProcessCell</td>
<td>spGetProcessCellByDate</td>
<td>Gets ALL Process Cells from Database</td>
</tr>
</tbody>
</table>

Report Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Where used</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>StartTime</td>
<td>All Procedures</td>
<td>Start Time; can be set to Null</td>
</tr>
<tr>
<td>EndTime</td>
<td>All Procedures</td>
<td>End Time; can be set to Null</td>
</tr>
<tr>
<td>SearchWord</td>
<td>NA</td>
<td>User Defined Batch Filter</td>
</tr>
<tr>
<td>ProcessCell</td>
<td>NA</td>
<td>Process Cell Filter</td>
</tr>
<tr>
<td>RecipeName</td>
<td>NA</td>
<td>Recipe Name Filter</td>
</tr>
<tr>
<td>UnitName</td>
<td>NA</td>
<td>Unit Name Filter</td>
</tr>
</tbody>
</table>
### Parameter Where used Comment

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Where used</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>UniqueID</td>
<td>NA</td>
<td>Unique ID Filter</td>
</tr>
</tbody>
</table>

#### Heading Data

<table>
<thead>
<tr>
<th>Object</th>
<th>Data Item</th>
<th>Data Source</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Time</td>
<td>starttimeexecution</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>End Time</td>
<td>endtimeexecution</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Duration</td>
<td>End Time – Start Time</td>
<td>BHBatch</td>
<td>Calculated</td>
</tr>
<tr>
<td>Batch ID</td>
<td>batchid</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Unique ID</td>
<td>uniqueid</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Recipe Name</td>
<td>recipefilename</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Area Name</td>
<td>area</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Process Cell</td>
<td>processcell</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Product Description</td>
<td>productdescription</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Product Code</td>
<td>productcode</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Completion State</td>
<td>completionstate</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Start Time</td>
<td>starttimeexecution</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>End Time</td>
<td>endtimeexecution</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Duration</td>
<td>endtimeexecution - starttimeexecution</td>
<td>BHBatch</td>
<td>Calculated</td>
</tr>
</tbody>
</table>

#### Table and Chart Datasets

<table>
<thead>
<tr>
<th>Table</th>
<th>Dataset</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>chart_BHUnitProcedure</td>
<td>BHUnitProcedure</td>
<td>Unit Procedure</td>
</tr>
<tr>
<td>chart_BHOperation</td>
<td>BHOperation</td>
<td>Operations</td>
</tr>
<tr>
<td>Chart_BHPHase</td>
<td>BHPase</td>
<td>Phases</td>
</tr>
</tbody>
</table>

#### See also

[Report details on page 61](#)

### Batch exceptions

The following tables outline the different components of the report.

#### Datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Stored Procedure</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetUniqueID</td>
<td>spGetUniqueIDByDate</td>
<td>Gets the unique IDs used between two dates</td>
</tr>
<tr>
<td>GetRecipeNames</td>
<td>spGetRecipeNamesBYDate</td>
<td>Gets the recipes used between two dates</td>
</tr>
<tr>
<td>GetUnitName</td>
<td>spGetUnitNameByDate</td>
<td>Gets the units used between two dates</td>
</tr>
</tbody>
</table>
### Dataset

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Stored Procedure</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>GetProcessCell</td>
<td>spGetProcessCellByDate</td>
<td>Gets the process cells used between two dates</td>
</tr>
<tr>
<td>AbnormalState-Changes</td>
<td>Embedded Query</td>
<td>Get the abnormal state changes for a unique ID</td>
</tr>
<tr>
<td>GetUniqueID_BatchIDForAlarms</td>
<td>Embedded Query</td>
<td>Gets alarms</td>
</tr>
<tr>
<td>GetIntoleranceData</td>
<td>Embedded Query</td>
<td>Gets In tolerance data</td>
</tr>
<tr>
<td>GetBatchFailures</td>
<td>Embedded Query</td>
<td>Gets batch failures</td>
</tr>
</tbody>
</table>

### Report Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Where used</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>StartTime</td>
<td>All Procedures</td>
<td>Start Time; can be set to Null</td>
</tr>
<tr>
<td>EndTime</td>
<td>All Procedures</td>
<td>End Time; can be set to Null</td>
</tr>
<tr>
<td>SearchWord</td>
<td>NA</td>
<td>User Defined Batch filter</td>
</tr>
<tr>
<td>ProcessCell</td>
<td>NA</td>
<td>Process Cell filter</td>
</tr>
<tr>
<td>RecipeName</td>
<td>NA</td>
<td>Recipe Name filter</td>
</tr>
<tr>
<td>UnitName</td>
<td>NA</td>
<td>Unit Name filter</td>
</tr>
<tr>
<td>UniqueID</td>
<td>NA</td>
<td>Unique ID filter</td>
</tr>
</tbody>
</table>

### Heading Data

<table>
<thead>
<tr>
<th>Object</th>
<th>Data Item</th>
<th>Data Source</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Time</td>
<td>StartTime</td>
<td>User input</td>
<td>Parameter</td>
</tr>
<tr>
<td>End Time</td>
<td>EndTime</td>
<td>User input</td>
<td>Parameter</td>
</tr>
</tbody>
</table>

### Table Datasets

<table>
<thead>
<tr>
<th>Table</th>
<th>Dataset</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>table1_AbnormalState-Changes</td>
<td>AbnormalState-Changes</td>
<td>Abnormal State Changes</td>
</tr>
<tr>
<td>table_Alarms</td>
<td>GetUniqueID_BatchIDForAlarms</td>
<td>Alarms</td>
</tr>
<tr>
<td>table_BatchFailures</td>
<td>GetBatchFailures</td>
<td>Batch Failures</td>
</tr>
<tr>
<td>table_OutOfTolerance</td>
<td>GetIntoleranceData</td>
<td>Out of Tolerance data</td>
</tr>
</tbody>
</table>

### See also

Report details on page 61
### Operation sequence

The following tables outline the different components of the report.

#### Datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Stored Procedure</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>BHBatch</td>
<td>Embedded query</td>
<td>Gets the batch executed between two dates</td>
</tr>
<tr>
<td>Dataset1</td>
<td>Embedded query</td>
<td>Gets the operation sequences associated with the specified batches’ Unique IDs</td>
</tr>
<tr>
<td>Dataset3</td>
<td>Embedded query</td>
<td>Gets the associated operation sequence detailed data</td>
</tr>
</tbody>
</table>

#### Report Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Where used</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>StartTime</td>
<td>All Procedures</td>
<td>Start Time; can be set to Null</td>
</tr>
<tr>
<td>EndTime</td>
<td>All Procedures</td>
<td>End Time; can be set to Null</td>
</tr>
<tr>
<td>SearchWord</td>
<td>BatchID</td>
<td>Batch ID filter</td>
</tr>
<tr>
<td>ProcessCell</td>
<td>GetProcessCell</td>
<td>Process Cell filter</td>
</tr>
<tr>
<td>RecipeName</td>
<td>GetRecipeNames</td>
<td>Recipe filter</td>
</tr>
<tr>
<td>UnitName</td>
<td>GetUnitName</td>
<td>Unit Name filter</td>
</tr>
<tr>
<td>UniqueID</td>
<td>GetUniqueID</td>
<td>Unique ID filter</td>
</tr>
</tbody>
</table>

#### Heading Data

<table>
<thead>
<tr>
<th>Object</th>
<th>Data Item</th>
<th>Data Source</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch ID</td>
<td>batchid</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Unique ID</td>
<td>uniqueid</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Recipe Name</td>
<td>recipefilename</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Area Name</td>
<td>area</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Process Cell</td>
<td>processcell</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Product Code</td>
<td>productcode</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Product Description</td>
<td>productdescription</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Completion State</td>
<td>completionstate</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Start Time</td>
<td>starttimeexecution</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>End Time</td>
<td>endtimeexecution</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Duration</td>
<td>starttimeexecution – endtimeexecution</td>
<td>BHBatch</td>
<td>Calculated</td>
</tr>
</tbody>
</table>
Table Datasets

<table>
<thead>
<tr>
<th>Table</th>
<th>Dataset</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>table_Batch</td>
<td>BHBatch</td>
<td>Batch header data</td>
</tr>
<tr>
<td>table_AssociatedOperationSequences</td>
<td>Dataset1</td>
<td>Associated operation sequences</td>
</tr>
<tr>
<td>table_AssociatedOperationSequences_Detail</td>
<td>Dataset5</td>
<td>Associated operation sequences - details</td>
</tr>
</tbody>
</table>

See also

Report details on page 61

Event Summary

The following tables outline the different components of the report.

Datasets

<table>
<thead>
<tr>
<th>Dataset</th>
<th>Stored Procedure</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>BHBatch</td>
<td>Embedded query</td>
<td>Gets the batch executed between two dates</td>
</tr>
<tr>
<td>BHBatchHis</td>
<td>Embedded query</td>
<td>Gets the event records based on Unique ID, Event, Description, and Value specified</td>
</tr>
<tr>
<td>GetProcessCell</td>
<td>spGetProcessCellByDate</td>
<td>Gets the process cells for the dropdown list</td>
</tr>
<tr>
<td>GetUnitName</td>
<td>spGetUnitNameByDate</td>
<td>Gets the unit names for the dropdown list</td>
</tr>
<tr>
<td>GetRecipeNames</td>
<td>spGetRecipeNameByDate</td>
<td>Gets the recipe name for the dropdown list</td>
</tr>
<tr>
<td>GetUniqueID</td>
<td>spGetUniqueIDByDate</td>
<td>Gets the unique IDs for the dropdown list</td>
</tr>
<tr>
<td>Events</td>
<td>Embedded query</td>
<td>Gets the event types for the dropdown list</td>
</tr>
<tr>
<td>Descript</td>
<td>Embedded query</td>
<td>Gets the descriptions for the dropdown list</td>
</tr>
<tr>
<td>PValue</td>
<td>Embedded query</td>
<td>Gets the values for the dropdown list</td>
</tr>
<tr>
<td>Columns</td>
<td>Embedded query</td>
<td>Gets the columns for the dropdown list</td>
</tr>
</tbody>
</table>

Report Parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Where used</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start Time</td>
<td>All Procedures</td>
<td>Start Time; can be set to Null</td>
</tr>
<tr>
<td>End Time</td>
<td>All Procedures</td>
<td>End Time; can be set to Null</td>
</tr>
<tr>
<td>Search Word</td>
<td>BatchID</td>
<td>Batch ID keyword filter</td>
</tr>
<tr>
<td>Process Cell</td>
<td>GetProcessCell</td>
<td>Process Cell or Controller ID filter</td>
</tr>
<tr>
<td>Parameter</td>
<td>Where used</td>
<td>Comment</td>
</tr>
<tr>
<td>---------------</td>
<td>---------------------</td>
<td>--------------------------</td>
</tr>
<tr>
<td>Recipe Name</td>
<td>GetRecipeNames</td>
<td>Recipe filter</td>
</tr>
<tr>
<td>Unit Name</td>
<td>GetUnitName</td>
<td>Unit Name or Unit ID filter</td>
</tr>
<tr>
<td>Unique ID</td>
<td>GetUniqueID</td>
<td>Unique ID filter</td>
</tr>
<tr>
<td>Event</td>
<td>GetEvents</td>
<td>Event filter</td>
</tr>
<tr>
<td>Description</td>
<td>GetDescriptions</td>
<td>Description filter</td>
</tr>
<tr>
<td>Value</td>
<td>GetValues</td>
<td>Value filter</td>
</tr>
<tr>
<td>Columns</td>
<td>GetColumns</td>
<td>Column filter</td>
</tr>
</tbody>
</table>

**Heading Data**

<table>
<thead>
<tr>
<th>Object</th>
<th>Data Item</th>
<th>Data Source</th>
<th>Type</th>
</tr>
</thead>
<tbody>
<tr>
<td>Event File Name</td>
<td>eventfilename</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Batch ID</td>
<td>batchid</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Unique ID</td>
<td>uniqueid</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Recipe Name</td>
<td>recipefilename</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Area Name</td>
<td>area</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Process Cell</td>
<td>processcell</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Product Code</td>
<td>productcode</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Product Description</td>
<td>productdescription</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Report Status</td>
<td>reportstatus</td>
<td>BHBatch</td>
<td>Data</td>
</tr>
<tr>
<td>Report Generated By</td>
<td>UserID</td>
<td>User</td>
<td>Data</td>
</tr>
<tr>
<td>Report Time</td>
<td>report execution time</td>
<td>Globals</td>
<td>Data</td>
</tr>
<tr>
<td>Row Count</td>
<td>count of records</td>
<td>BHBatchHis</td>
<td>Calculated</td>
</tr>
</tbody>
</table>

**Table Datasets**

<table>
<thead>
<tr>
<th>Table</th>
<th>Dataset</th>
<th>Comment</th>
</tr>
</thead>
<tbody>
<tr>
<td>table_Batch</td>
<td>BHBatch</td>
<td>Event Header data</td>
</tr>
<tr>
<td>Tablix1</td>
<td>BHBatchHis</td>
<td>Event Record data</td>
</tr>
</tbody>
</table>

**See also**

[Report details](#) on page 61
Database Manager Tool

The FactoryTalk Batch Database Manager is designed to provide assistance in performing backup, delete, and restore operations on the FactoryTalk Batch Archiver (BatchHistoryEx) database, based on date. When the FactoryTalk Batch Database Manager is launched, choose to backup and delete database records or restore database information from a backup.

Backups are in the form of a Microsoft SQL Server backup (.bak) file or to a remote Microsoft SQL Server database. The .bak option is used to restore the original data to a database with a new name. The remote database option creates an identical database on another machine. The database has an identical schema to the FactoryTalk Batch Archiver database. All records deleted from the FactoryTalk Batch Archiver database are first backed up to the database or to a .bak file.

See also

- Backup and delete database records on page 78
- Restore database information from a backup on page 83

Backup types

All records deleted from the FactoryTalk Batch Archiver database are first backed up to a database or to a .bak file. Backups are to a Microsoft SQL Server backup (.bak) file or to a remote Microsoft SQL Server database. The .bak option is used to restore the original data to a database with a new name. The remote database option creates an identical database on another machine. The database has an identical schema to the FactoryTalk Batch Archiver database.

Remote server backup

Use the remote server option to backup Archiver data. By using a remote server, only data that is removed from the Archiver database will be stored. Data stored on a remote server is accessible instantly, instead of having to first restore the data from a backup file.
Important: A secondary machine running Microsoft SQL Server 2012 or later is required to use the remote server option.

Local backup file (.bak)

Use the local backup option to create a Microsoft SQL Server backup file containing all information currently in the database. The backup file takes the same amount of space as the database without the transaction log.

Use the local backup option only if instant access to backed up data is not needed, and there is sufficient disk space to store the backup file and allow for additional database growth. If a secondary machine running Microsoft SQL Server 2012 is not available, the remote backup option in the FactoryTalk Batch Database Manager is not available.

Important: The data in the backup file is not readily available for querying. Restore the backup file into a database and then query.

See also

- Restore database information from a backup on page 83
- Backup and delete database records on page 78

Backup and delete database records

Backup and delete specific FactoryTalk Batch records in the database based on date.

To backup and delete database records:

1. In Execution Settings, enter the starting connection data and execution parameters.

2. Select Next.

3. In Backup Settings, enter the information for how the FactoryTalk Batch Archiver database is backed up before deletion.

   a. For a remote backup destination, select Database Backup (remote), select the destination data source, and enter the database name.

   b. For a local backup destination, select File Backup (local) and click browse (Browse) or enter the directory path manually to select the target directory.

4. Select Next.
5. In **Confirm Settings**, verify all the information is correct.

6. Select **Execute**.

**See also**

- [Enter backup settings on page 81](#)
- [Enter execution settings on page 80](#)
- [Execution dialog box for backup and delete on page 82](#)

**Execution Settings window**

In the **Execution Settings** window, the starting connection data and execution parameters are entered to start the process for backing up and removing records in the Batch Archiver database.

The source DSN refers to an ODBC connection on the Batch Archiver Server. The DSN box shows a list with available DSNs or auto-populates using the Archiver .ini file, if it is available.

The ODBC connection is used to connect to the Batch Archiver database and perform necessary operations. Once a DSN has been selected from the list, or is auto-populated from the Archiver .ini file, the following occurs:

- The connection is tested.
- The database schema is verified.
- The database size is retrieved.
- The Microsoft SQL Server version is verified.
- The DSN is verified to be linked to a server that resides on the local machine that the tool is running on.

**Important:** Using an unsupported version of Microsoft SQL Server generates an error and the FactoryTalk Batch Database Manager closes once the error is acknowledged. See the *FactoryTalk Batch Components Upgrade and Installation Guide* (publication number BATCH-IN002) for system and software requirements.

Select a date and time to continue the backup and removal process. All records before the chosen time and date are backed up and then removed from the Batch Archiver database.

**See also**

- [Enter execution settings on page 80](#)
Enter execution settings

Enter the starting connection data and execution parameters to start the process for backing up and removing records in the FactoryTalk Batch Archiver database.

To enter execution settings:

1. In Source Data Source Name (DSN), select the appropriate data source name from the list.
   
   **Tip:** If the data that is being backed up and removed is on the same computer as the Batch Archiver Server, the Source Data Source Name (DSN) area is automatically populated and a list is not available.

2. In Backup and Delete Records Saved Before, select a date and time.

3. Select Next.

See also

- Execution Settings window on page 79
- Backup Settings window on page 80

Backup Settings window

In Backup Settings, select how the Batch Archiver database is to be backed up. Choose either the remote or local backup option and enter the required information. The remote option backs up the BatchHistoryEx database to another SQL Server database on a different machine, and the local option backs up to a .bak file to the location specified by the user.

**Database Backup (remote)**

Database Backup (remote) is the option to backup to another database on a different machine. This option is selected by default. Select the data source name linked to the backup server using a list of available DSNs in the Destination Data Source Name (DSN). Once a destination DSN has been selected, the connection is verified and the application checks to make sure the destination DSN is not linked to the Batch Archiver SQL Server.

**Important:** If the destination DSN is on a machine running a Microsoft SQL Server version older than Microsoft SQL Server 2012, or the DSN is linked to the Batch Archiver SQL Server, an error is generated.

The username used for the connection is the current logged in user, displayed in the status bar on the bottom left of the window.

The Destination Database Name is where the data deleted from the Batch Archiver server is backed up prior to deletion.
**File Backup (local)**

**File Backup (local)** is the option for backup to a .bak file. Enter a local path for the .bak file to be stored in the box. The file location must be on the Batch Archiver SQL Server machine. Alternately, select the browse ( ) button to pull up a file explorer window to select the target directory.

| Important: | If the target directory does not exist or the file already exists, an error is generated. Only one .bak file can be created in a specific directory per day. |

See also

*Enter backup settings* on page 81

**Enter backup settings**

Enter information for how the Batch Archiver database is backed up before deletion. Choose either the remote or local backup option and enter the required information.

**To enter backup settings for a remote backup:**

1. Select **Database Backup (remote)**.
2. In **Destination Data Source Name (DSN)**, select the destination data source.
3. In **Destination Database Name**, enter the database name.
4. Select **Next**.

**To enter backup settings for a local backup:**

1. Select **File Backup (local)**.
2. In **Local Backup Location**, click browse ( ).
3. Select the target directory.

| Tip: | The target directory location must be on the Batch Archiver SQL Server machine. |

4. Select **Next**.

See also

*Backup Settings window* on page 80

*Backup and delete database records* on page 78
Confirm Settings window

The Confirm Settings window shows all input information so it can be verified before execution. The contents of the table in the Confirm Settings window depend on which backup method was chosen.

Source Database shows all the details relating to the Archiver database. Destination Database shows all settings related to the backup method chosen. Execution Settings shows the settings related to the backup and deletion of records.

See also

Backup and delete database records on page 78

Execution dialog box for backup and delete

Once execution has started, the execution dialog box displays the operations being performed on the Batch Archiver database.

Operations shown in the dialog box vary depending on the backup method: remote or local.

Remote database backup

If the backup database does not yet exist, the first operation is Creating backup database. If this is the first time using the FactoryTalk Batch Database Manager, the next operation is Adding indexes to the BatchHistoryEx database. Once the database exists, the next operation is Performing the delete and backup operations now.

Local database backup to a .bak file

The first operation is Creating backup file to the specified backup directory location. If this is the first time using the FactoryTalk Batch Database Manager, the next operation is Adding indexes to the BatchHistoryEx database. Once the backup file is created, the next operation is Performing the backup and delete operations now.

Tip: The user running the Microsoft SQL Server service must have the permissions to save to the backup directory location, or this error message is generated: Backup not made, execution canceled. The user account running SQL Server does not have permission to access the selected backup directory.

See also

Backup and delete database records on page 78
The Restore option creates a new database of FactoryTalk Batch records from a backup previously generated by the FactoryTalk Batch Database Manager. This does not restore into the Batch Archiver database.

To restore database information from a backup:

1. In Restore Settings, enter the backup file and destination data source name.
2. Enter a name for the restore database.
3. Select Restore.

See also

- Restore settings window on page 83
- Execution dialog box for backup and delete on page 82

In Restore Settings, the backup file and the destination data source name are entered to restore records in the Batch Archiver database.

The backup file is located in the same place as the file that was created using the backup functionality of the FactoryTalk Batch Database Manager.

Once a destination data source name has been selected from the list, or is auto populated from the Archiver .ini file, the connection is tested and verified.

**Important:** If unsupported versions of Microsoft SQL Server are used, an error is generated and the FactoryTalk Batch Database Manager does not continue execution. See the FactoryTalk Batch Components Upgrade and Installation Guide (publication number BATCH-IN002) for system and software requirements.

See also

- Restore database information from a backup on page 83

Once execution has started, the execution dialog box displays the operations being performed on the local SQL Server.

The first operation is **Attempting to restore <filename>.bak to <database name>**. A progress bar is shown to indicate restoration is happening. The last operation is **Database has restored successfully**.
Tip: If the restore operation was cancelled, **Database restore cancelled** is shown. If there was an error restoring the database, an error message is generated: **Error restoring database.**

See also

[Restore database information from a backup](#) on page 83
Special considerations – naming conventions

Batch reporting uses naming conventions in the FactoryTalk Batch Area Model when Material Manager is not used.

Standard Material Manager supplies data for these reports so these naming conventions are not needed.

See also

Material Usage Report on page 85
Forward/Backward Track and Trace Report on page 86
Setpoint vs. Actual on page 87
Batch Phase naming conventions on page 87

Material Usage Report

Material Manager automatically creates these parameters:

- Formula Parameter: material
- Report Parameter: actual_amount

Without Material Manager, define these Formula and Report parameter names:

- Formula Parameter: material_xx (where xx is user defined)
- Report Parameter: actual_xx (where xx is user defined)

Add phase example:

- Formula Parameter: material_oil
- Report Parameter: actual_oil

-or-

Add phase example:

- Formula Parameter: material_oil
- Formula Parameter: material_water
- Report Parameter: actual_oil
Appendix A  Special considerations – naming conventions

Report Parameter: actual_water

See also

Forward/Backward Track and Trace Report on page 86
Setpoint vs. Actual on page 87
Batch Phase naming conventions on page 87

Material Manager automatically creates these parameters:

Formula Parameter: lot
Report Parameter: actual_amount

Without Material Manager, define these Formula and Report parameter names:

Report Parameter: lot_xx_yy (where xx is user defined and yy represents an incremental lot #)
Report Parameter: actual_lot_xx_yy (where xx is user defined and yy [optional] represents an incremental lot )

Single lot with single actual. Without yy value.

Add phase example:

Report Parameter: lot_oil
Report Parameter: actual_lot_oil
Single lot with single actual. With yy value.

Add phase example:

Report Parameter: lot_oil_01
Report Parameter: actual_lot_oil_01
Multiple lots with multiple actuals.

Add phase example:

Report Parameter: lot_oil_01
Report Parameter: lot_oil_02
Report Parameter: actual_lot_oil_01
Report Parameter: actual_lot_oil_02
See also

Special considerations – naming conventions on page 85

**Setpoint vs. Actual**

Material Manager automatically creates these parameters:

- Formula Parameter: amount
- Report Parameter: actual_amount

Without Material Manager, define these Formula and Report parameter names:

- Formula Parameter: setpoint_xx (where xx is user defined)
- Report Parameter: target_xx (where xx is user defined)
- Report Parameter: actual_xx (where xx is user defined)
- Report Parameter: in_tolerance_xx (where xx is user defined)
- Report Parameter: target_xx is optional
- Report Parameter: n_tolerance_xx is optional

**Add phase example:**

- Formula Parameter: setpoint_oil
- Report Parameter: actual_oil
- Report Parameter: in_tolerance_oil

**Add phase example:**

- Formula Parameter: setpoint_oil
- Report Parameter: target_oil
- Report Parameter: actual_oil
- Report Parameter: in_tolerance_oil

See also

Special considerations – naming conventions on page 85

**Batch Phase naming conventions**

See specific batch phase examples to understand how naming conventions are applied.

See also

- Single Specific Ingredient Addition Phase with no target_xx parameter on page 88
- Single Specific Ingredient Addition Phase with target_xx parameter on page 88
- Single Specific Ingredient Addition Phase on page 88
Appendix A  Special considerations – naming conventions

Single Specific Ingredient Addition Phase with no target_xx parameter

This is an example of Single Specific Ingredient Addition Phase with no target_xx parameter:

Formula Parameter: Material_Oil
Formula Parameter: Setpoint_Oil
Report Parameter: Lot_Oil
Report Parameter: Actual_Lot_Oil
Report Parameter: Actual_Oil
Report Parameter: In_Tolerance_Oil

See also

Single Specific Ingredient Addition Phase with target_xx parameter on page 89

Single Specific Ingredient Addition Phase on page 88

Single Specific Ingredient Addition Phase with Multiple Lot IDs on page 89

Single Specific Ingredient Addition Phase with target_xx parameter

This is an example of Single Specific Ingredient Addition Phase with target_xx parameter:

Formula Parameter: Material_Oil
Formula Parameter: Setpoint_Oil
Report Parameter: Lot_Oil
Report Parameter: Actual_Lot_Oil
Report Parameter: Target_Oil
Report Parameter: Actual_Oil
Report Parameter: In_Tolerance_Oil

See also

Single Specific Ingredient Addition Phase with target_xx parameter on page 89

Single Specific Ingredient Addition Phase on page 88

Single Specific Ingredient Addition Phase with Multiple Lot IDs on page 89

Single Specific Ingredient Addition Phase

This is an example of Single Specific Ingredient Addition Phase:

Formula Parameter: Material_Oil
Formula Parameter: Setpoint_Oil
Report Parameter: Lot_Oil
Report Parameter: Actual_Lot_Oil
Report Parameter: Target_Oil
Special considerations – naming conventions

Report Parameter: Actual_Oil
Report Parameter: In_Tolerance_Oil

See also

Single Specific Ingredient Addition Phase with Multiple Lot IDs on page 89

Single Specific Ingredient Addition Phase with Multiple Lot IDs

This is an example of Single Specific Ingredient Addition Phase with Multiple Lot IDs:

Formula Parameter: Material_Oil
Formula Parameter: Setpoint_Oil
Report Parameter: Lot_Oil_01
Report Parameter: Lot_Oil_02
Report Parameter: Actual_Lot_Oil_01
Report Parameter: Actual_Lot_Oil_02
Report Parameter: Target_Oil
Report Parameter: Actual_Oil
Report Parameter: In_Tolerance_Oil

See also

Multiple Specific Ingredient Addition Phase on page 89

Single Specific Ingredient Addition Phase with multiple parameter types on page 90

Multiple Specific Ingredient Addition Phase

This is an example of Multiple Specific Ingredient Addition Phase:

Formula Parameter: Material_Oil
Formula Parameter: Material_HFCS
Formula Parameter: Setpoint_Oil
Formula Parameter: Setpoint_HFCS
Report Parameter: Lot_Oil
Report Parameter: Lot_HFCS
Report Parameter: Actual_Lot_Oil
Report Parameter: Actual_Lot_HFCS
Report Parameter: Target_Oil
Report Parameter: Target_HFCS
Report Parameter: Actual_Oil
Report Parameter: Actual_HFCS
Report Parameter: In_Tolerance_Oil
Report Parameter: In_Tolerance_HFCS
Appendix A  Special considerations – naming conventions

See also

Single Specific Ingredient Addition Phase with multiple parameter types on page 90

Generic Multiple Ingredient on page 90

Generic Multiple Ingredient Addition Phase with multiple lots per material on page 91

**Single Specific Ingredient Addition Phase with multiple parameter types**

This is an example of Single Specific Ingredient Addition Phase with multiple parameter types:

Formula Parameter: Material_Oil
Formula Parameter: Setpoint_Oil
Formula Parameter: Setpoint_Oil_Temperature
Report Parameter: Lot_Oil
Report Parameter: Actual_Lot_Oil
Report Parameter: Target_Oil
Report Parameter: Target_Oil_Temperature
Report Parameter: Actual_Oil
Report Parameter: Actual_Oil_Temperature
Report Parameter: In_Tolerance_Oil
Report Parameter: In_Tolerance_Temperature

See also

Generic Multiple Ingredient on page 90

Generic Multiple Ingredient Addition Phase with multiple lots per material on page 91

**Generic Multiple Ingredient**

This is an example of Generic Multiple Ingredient:

Additional Phase: Add_Hand Adds
Formula Parameter: Material_01
Formula Parameter: Material_02
Formula Parameter: Setpoint_01
Formula Parameter: Setpoint_02
Report Parameter: Lot_01
Report Parameter: Lot_02
Report Parameter: Actual_Lot_01
Report Parameter: Actual_Lot_02
Report Parameter: Target_01
Report Parameter: Target_02
Report Parameter: Actual_01
Report Parameter: Actual_02
Special considerations – naming conventions

Report Parameter: In_Tolerance_01
Report Parameter: In_Tolerance_02

See also

**Generic Multiple Ingredient Addition Phase with multiple lots per material**

This is an example of Generic Multiple Ingredient Addition Phase with multiple lots per material:

Additional Phase: Add_Hand Adds
Formula Parameter: Material_01
Formula Parameter: Material_02
Formula Parameter: Setpoint_01
Formula Parameter: Setpoint_02
Report Parameter: Lot_01_01
Report Parameter: Lot_01_02
Report Parameter: Lot_02_01
Report Parameter: Lot_02_02
Report Parameter: Actual_Lot_01_01
Report Parameter: Actual_Lot_01_02
Report Parameter: Actual_Lot_02_01
Report Parameter: Actual_Lot_02_02
Report Parameter: Target_01
Report Parameter: Target_02
Report Parameter: Actual_01
Report Parameter: Actual_02
Report Parameter: In_Tolerance_01
Report Parameter: In_Tolerance_02

See also

**Generic Multiple Ingredient** on page 90
Legal Notices

Copyright notice

Copyright © 2018 Rockwell Automation Technologies, Inc. All Rights Reserved. Printed in USA.

This document and any accompanying Rockwell Software products are copyrighted by Rockwell Automation Technologies, Inc. Any reproduction and/or distribution without prior written consent from Rockwell Automation Technologies, Inc. is strictly prohibited. Please refer to the license agreement for details.

End User License Agreement (EULA)

You can view the Rockwell Automation End-User License Agreement ("EULA") by opening the License.rtf file located in your product's install folder on your hard drive.

Trademark Notices

Allen-Bradley, CompactLogix, ControlLogix, Connected Components Workbench, eProcedure, FactoryTalk, GuardLogix, GuardPLC, Micro800, MicroLogix, MobileView, MobileView Guard, PanelBuilder, PanelView, PhaseManager, PlantLink, PlantPAx, PLC-2, PLC-3, PLC-5, PowerMonitor, RSBizWare, RSBizWare BatchHistorian, RSBizWare Historian, RSBizWare MaterialTrack, RSLinx, Rockwell Automation, Rockwell Software, SLC, SoftLogix, Studio 5000, and XMare trademarks of Rockwell Automation, Inc.

Any Rockwell Automation software or hardware not mentioned here is also a trademark, registered or otherwise, of Rockwell Automation, Inc.

For a complete list of products and their respective trademarks, go to: http://www.rockwellautomation.com/rockwellautomation/legal-notices/overviewpage?#tab4

Warranty

This product is warranted in accordance with the product license. The product’s performance may be affected by system configuration, the application being performed, operator control, maintenance, and other related factors. Rockwell Automation is not responsible for these intervening factors. The instructions in this document do not cover all the details or variations in the equipment, procedure, or process described, nor do they provide directions for meeting every possible contingency during installation, operation, or maintenance. This product's implementation may vary among users.
This document is current as of the time of release of the product; however, the accompanying software may have changed since the release. Rockwell Automation, Inc. reserves the right to change any information contained in this document or the software at any time without prior notice. It is your responsibility to obtain the most current information available from Rockwell when installing or using this product.

**Environmental compliance**


**Contact Rockwell Automation**

Customer Support Telephone — 1.440.646.3434


Index

..

 evt files 7

A
ADDEVENT execute 17
archive
  end-of-batch 7
  incremental 7
Archiver
database schema 29
  reporting services 8, 37
  start 22
  system architecture 8
ArchQue.txt File 28

B
Batch Service Manager 22

C
configure 17, 21
  custom archiver 27
  server 17
create a data source 15
  data source name 15

D
data source 15
database
  schema 29
  types to use 7

E
electronic batch record 7
description 11
  interpretation examples 11
end-of-batch archive 7
Event Log entries 35
execute
  ADDEVENT 17

I
incremental archive 7

N
non-incremental archive 7

R
reports
  Backward Tracing 46, 64
  Batch Detail 41, 65
  Batch Exceptions 50, 70
  Batch Execution 48, 69
  Batch Listing 40, 61
  Batch Summary 42, 62
  Duration Comparison 47, 68
  Forward Tracking 45, 63
  Material Usage 44, 67

S
server configuration 17
service manager 22
  start
    Archiver 22
  system architecture 8
  system failure 7

T
troubleshoot
  Event Archiver is disabled 25

W
Windows Event Log 35
Rockwell Automation support

Rockwell Automation provides technical information on the web to assist you in using its products. At http://www.rockwellautomation.com/support you can find technical and application notes, sample code, and links to software service packs. You can also visit our Support Center at https://rockwellautomation.custhelp.com for software updates, support chats and forums, technical information, FAQs, and to sign up for product notification updates.

In addition, we offer multiple support programs for installation, configuration, and troubleshooting. For more information, contact your local distributor or Rockwell Automation representative, or visit http://www.rockwellautomation.com/services/online-phone.

Installation assistance

If you experience a problem within the first 24 hours of installation, review the information that is contained in this manual. You can contact Customer Support for initial help in getting your product up and running.

<table>
<thead>
<tr>
<th>United States or Canada</th>
<th>1.440.646.3434</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outside United States or Canada</td>
<td>Use the Worldwide Locator available at <a href="http://www.rockwellautomation.com/locations">http://www.rockwellautomation.com/locations</a>, or contact your local Rockwell Automation representative.</td>
</tr>
</tbody>
</table>

New product satisfaction return

Rockwell Automation tests all of its products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning and needs to be returned, follow these procedures.

<table>
<thead>
<tr>
<th>United States</th>
<th>Contact your distributor. You must provide a Customer Support case number (call the phone number above to obtain one) to your distributor to complete the return process.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outside United States</td>
<td>Please contact your local Rockwell Automation representative for the return procedure.</td>
</tr>
</tbody>
</table>

Documentation feedback

Your comments will help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete the feedback form, publication RA-DU002.
Important User Information

Solid-state equipment has operational characteristics differing from those of electromechanical equipment. Safety Guidelines for the Application, Installation, and Maintenance of Solid State Controls (publication SGI-1.1 available from your local Rockwell Automation sales office or online at http://www.rockwellautomation.com/literature/) describes some important differences between solid-state equipment and hard-wired electromechanical devices. Because of this difference, and also because of the wide variety of uses for solid-state equipment, all persons responsible for applying this equipment must satisfy themselves that each intended application of this equipment is acceptable.

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.

**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.

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

---

Allen-Bradley, Rockwell Software, and Rockwell Automation ControlLogix, eProcedure, FactoryTalk, RSView, RSView Batch, Batch Historian, RSView Historian, RSView Material Track, and RSView are trademarks of Rockwell Automation, Inc.

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

## Chapter 1

### ActiveX controls introduction
- Document Organization ................................................................. 7
- ActiveX Controls library introduction .................................................. 7
- ActiveX controls library installation .................................................... 8
  - DCOM for Security and Distributed Communications ......................... 8
  - File redistribution ........................................................................... 9
- ActiveX Controls .................................................................................. 9
  - Add ActiveX controls to the Visual Basic toolbox .............................. 9
  - Configure ActiveX controls in FactoryTalk View SE ......................... 10
    - VBA code example for RefreshSecurityToken .................................. 10
- Version compatibility with FactoryTalk Batch Server ............................ 11

## Chapter 2

### ControlRecipeList Control
- ControlRecipeList interface ............................................................. 13
- ControlRecipeList Shortcut Menu ...................................................... 15
  - Shortcut menu commands ............................................................. 15
- Log in to an ActiveX control ............................................................ 16
- Control recipe commands ............................................................... 16
  - Start a control recipe .................................................................. 17
  - Change the state of a control recipe ............................................. 17
  - Add a control recipe ..................................................................... 19
    - View Batch creation errors ...................................................... 20
  - Change the mode of a control recipe ........................................... 21
  - Remove a control recipe from the ControlRecipeList ..................... 22
  - Cross invocation ......................................................................... 22
    - Perform cross invocation ......................................................... 23
  - Work with batches without the Material Server ............................. 23
    - Material Server not in use warning .......................................... 25
    - Batch creation when the Material server is not communicating .... 25
    - Prompt to bind unit requirement ............................................. 26
    - Manual unit binding ................................................................ 27
    - Prompt to bind a material phase step ..................................... 27
    - Look up material information ................................................. 28
    - Reestablish communication with the Material Server on manual batches .... 29

## Chapter 3

### PromptsList Control
- PromptsList interface .................................................................... 31
- PromptsList shortcut menu ............................................................ 32
- Log in to an ActiveX control ............................................................ 32
- Acknowledge a prompt ................................................................... 33
<table>
<thead>
<tr>
<th>Acknowledge a material phase step prompt</th>
<th>33</th>
</tr>
</thead>
<tbody>
<tr>
<td>Cross invocation</td>
<td>34</td>
</tr>
<tr>
<td>Perform cross invocation</td>
<td>34</td>
</tr>
</tbody>
</table>

**Chapter 4**

**SignatureList Control**

<table>
<thead>
<tr>
<th>SignatureList Interface</th>
<th>37</th>
</tr>
</thead>
<tbody>
<tr>
<td>SignatureList control shortcut menu</td>
<td>38</td>
</tr>
<tr>
<td>Log in to an ActiveX control</td>
<td>38</td>
</tr>
<tr>
<td>Acknowledge a signature</td>
<td>39</td>
</tr>
<tr>
<td>Signature cancellation</td>
<td>39</td>
</tr>
<tr>
<td>Configure the SignatureList control in an external application</td>
<td>40</td>
</tr>
</tbody>
</table>

**Chapter 5**

**ProcedureView Control**

<table>
<thead>
<tr>
<th>Table View interface</th>
<th>44</th>
</tr>
</thead>
<tbody>
<tr>
<td>SFC view interface</td>
<td>45</td>
</tr>
<tr>
<td>ProcedureView shortcut menu</td>
<td>46</td>
</tr>
<tr>
<td>Log in to an ActiveX control</td>
<td>46</td>
</tr>
<tr>
<td>ProcedureView navigation</td>
<td>47</td>
</tr>
<tr>
<td>SFC View</td>
<td>47</td>
</tr>
<tr>
<td>Table View</td>
<td>48</td>
</tr>
<tr>
<td>Control recipe commands</td>
<td>48</td>
</tr>
<tr>
<td>Start a control recipe</td>
<td>48</td>
</tr>
<tr>
<td>Change the state of a control recipe</td>
<td>49</td>
</tr>
<tr>
<td>Change the mode of a control recipe</td>
<td>50</td>
</tr>
<tr>
<td>Timer step commands</td>
<td>51</td>
</tr>
<tr>
<td>ProcedureView control recipe properties</td>
<td>51</td>
</tr>
<tr>
<td>Parameter tab</td>
<td>51</td>
</tr>
<tr>
<td>Edit parameter values</td>
<td>52</td>
</tr>
<tr>
<td>Report tab</td>
<td>52</td>
</tr>
<tr>
<td>Recipe tab</td>
<td>52</td>
</tr>
<tr>
<td>Binding tab</td>
<td>52</td>
</tr>
<tr>
<td>Expression values</td>
<td>53</td>
</tr>
<tr>
<td>Perform an active step change</td>
<td>53</td>
</tr>
<tr>
<td>Manual binding</td>
<td>55</td>
</tr>
<tr>
<td>Perform manual binding</td>
<td>56</td>
</tr>
<tr>
<td>Bind a unit procedure to a unit</td>
<td>56</td>
</tr>
<tr>
<td>Material binding</td>
<td>57</td>
</tr>
<tr>
<td>Perform material binding</td>
<td>58</td>
</tr>
<tr>
<td>Cross invocation</td>
<td>59</td>
</tr>
<tr>
<td>Perform cross invocation</td>
<td>59</td>
</tr>
</tbody>
</table>
# Chapter 6

**TimerView Control**

- TimerView interface ................................................................. 61
- TimerView shortcut menu ......................................................... 63
- Log in to an ActiveX control ..................................................... 63
- Timer Step dialog box ............................................................. 64
- Setpoint values ............................................................................. 64
- Timer step filters ......................................................................... 65
- Timer step filter patterns ......................................................... 65

**Legal Notices**

- Legal Notices .............................................................................. 67

**Index**
ActiveX controls introduction

This document provides information and instructions required to use the FactoryTalk® Batch ActiveX® custom controls.

Use the information in this guide to minimize the effort required to create a custom look for batch automation system data.

This guide presents the following subjects:

- Installing the ActiveX Controls Library
- ControlRecipeList Interface
- PromptsList Interface
- SignatureList Interface
- ProcedureView Interface
- TimerView Interface

The FactoryTalk® Batch ActiveX Controls Library is a set of ActiveX® custom controls to create a dynamic picture of batch automation system data. Use the controls in the library to communicate with the FactoryTalk Batch Server, and to monitor and interact with a control recipe without having to run FactoryTalk Batch View. Place the controls within any ActiveX control/OLE container, such as your Human-Machine Interface (HMI) software, Microsoft Visual Basic, or web browsers. Use the ActiveX controls to create a custom look for your interface or to create a custom application specific to your needs.

Use the ActiveX controls to set their properties without writing any code.

The ActiveX Controls Library is distributed within the batchv01.ocx, batchv02.ocx, TimerStepOCX.ocx and SignatureListOCX.ocx files. These files contain five controls:

- ControlRecipeList control: Performs the functions of the Batch List window.
- PromptsList control: Performs the functions of the Unacknowledged Prompts window in FactoryTalk Batch View.
• ProcedureView control: Performs the functions of both the Procedure as Table and Procedure as SFC windows in FactoryTalk Batch View.

• Timer View control: Shows all the timer steps in batches on the Batch List using filter criteria.

• SignatureList control: Performs the functions of the Batch Signature List window.

These controls are used to create, command, and remove control recipes from the Batch List, acknowledge operator prompts, view running control recipes, change step formula values, acknowledge electronic signatures, and perform active step changes.

The Batch ActiveX controls vary in appearance through the use of the Properties and Methods that the controls support. The controls allow you to perform the same functions as their corresponding FactoryTalk Batch View windows.

**Important:**
In HTML pages that use the Batch ActiveX controls, the class IDs in the `<OBJECT>` tags must be updated. This is done by looking up the class ID in the registry, or by deleting and reinserting the control in an HTML editor. For HMI screens, the controls must be deleted and reinserted.

---

**ActiveX controls library installation**

When installing the ActiveX Controls Library on a computer without any other installed FactoryTalk Batch components, it is recommended that only the Client installation option be used.

**Important:**
FactoryTalk Batch 12.0 ActiveX Controls Library schema changes require you to rebuild applications built against earlier versions of the ActiveX Controls Library.

**See also**

[ActiveX Controls library introduction](#) on page 7

**DCOM for Security and Distributed Communications**

The DCOM (Distributed Component Object Model) protocol enables software components to communicate directly with each other across the Internet, intranet and other networks. The ActiveX Controls communicate with the FactoryTalk Batch Server through DCOM.

**See also**

[ActiveX controls library installation](#) on page 8

[File redistribution](#) on page 8
File redistribution

You may use and redistribute all the files with this product as long as you credit Rockwell Automation in the About box. Read and honor the license agreement.

See also

ActiveX controls library installation on page 8

ActiveX Controls

Use each control to interact with the FactoryTalk Batch Server once placed within an OLE container application (MS Visual Basic, MS Excel, MS PowerPoint, etc.). To use the ActiveX controls in an application other than Visual Basic, use the More Controls button on the Control Toolbox toolbar to retrieve the controls. They are listed with the following names:

- Batchview.ControlRecipeList
- Batchview.PromptsList
- Batchview2.ProcedureView
- SignatureListOCX.SignatureList
- TimerStepOCX.TimerStepList

See also

Configure ActiveX controls in FactoryTalk View SE on page 10
Add ActiveX controls to the Visual Basic toolbox on page 9

Add ActiveX controls to the Visual Basic toolbox

To use the ActiveX controls in Visual Basic, the controls must be added to the Visual Basic toolbox before using them in a project. The ProcedureView appears as an SFC/table combination, the PromptsList as a question mark, the ControlRecipeList as a card file, and the SignatureList as a tablet with a pen.

To manually add the ActiveX controls in Visual Basic:

2. From the Projects menu, select Components.
3. In the Components dialog box, select the Controls tab.
4. Scroll through the list, locate, and select the appropriate control:
   
   - The **BatchControl BatchView Controls** library to add the **ControlRecipeList** control and the **PromptsList** control.
   
   - The **BatchControl ProcedureView Control** library to add the **ProcedureView** control.
   
   - The **SignatureListOCX Controls** library to add the **SignatureList** control.

5. Click **OK** to close the dialog box.

See also

- [ActiveX Controls on page 9](#)
- [Configure ActiveX controls in FactoryTalk View SE on page 10](#)

**Configure ActiveX controls in FactoryTalk View SE**

If a FactoryTalk Batch ActiveX Control is hosted in FactoryTalk View SE, then the **RefreshSecurityToken** method can be used to synchronize the user logged on to the control with the user logged on to the FactoryTalk View SE application.

To configure a FactoryTalk Batch ActiveX Control hosted in FactoryTalk View SE:

1. Configure the **FactoryTalk Batch ActiveX Controls** and **FactoryTalk View SE** to use the same FactoryTalk Directory.

2. In **FactoryTalk View Studio**, set the **Expose ToVBA** property to **VBA Control** in the **Property Panel** dialog box.

   **Tip:** To open the **Property Panel** dialog box, right-click the display window containing the FactoryTalk Batch ActiveX Control, and then select **Property Panel** on the context menu.

3. In **FactoryTalk View Studio**, add VBA code for the **RefreshSecurityToken** to the display window containing the FactoryTalk Batch ActiveX control.

See also

- [ActiveX Controls on page 9](#)
- [VBA code example for RefreshSecurityToken on page 10](#)

**VBA code example for RefreshSecurityToken**

This code updates an ActiveX control (**batchviewControlRecipeList** in the example) with the user logon information when the user logged on to the
FactoryTalk View SE client application changes.

VBA code example for the RefreshSecurity Token

Option Explicit
Dim WithEvents MyApp As DisplayClient.Application
Private Sub Display_AnimationStart()
    Set MyApp = Application
End Sub
Private Sub Display_BeforeAnimationStop()
    Set MyApp = Nothing
End Sub
Private Sub MyApp_Login(ByVal UserName As String)
    batchviewControlRecipeList1.RefreshSecurityToken
End Sub
Private Sub MyApp_Logout(ByVal UserName As String)
    ' DO NOT ATTEMPT TO REFRESH THE SECURITY TOKEN HERE
End Sub

See also

Configure ActiveX controls in FactoryTalk View SE on page 10

Version compatibility with FactoryTalk Batch Server

The ProcedureView, InstructionView, and ControlListView ActiveX controls check the FactoryTalk Batch Server version on startup via the CPRVersion API item and determine compatibility with the Server. If a version incompatibility exists between the Server and the ActiveX control, a message displays along the bottom of the control: "The [Name] ActiveX control is incompatible with the current Server. It is strongly recommended that you upgrade your system before continuing."

Operation with an incompatible Server is untested and unsupported——upgrade your FactoryTalk Batch Server to the same version as the ActiveX controls currently installed.

The message is continually displayed for as long as the control is connected to the FactoryTalk Batch Server. If the Server is shut down and then restarted, the control once again checks for version compatibility.

See also

ActiveX Controls library introduction on page 7
Chapter 2

ControlRecipeList Control

The ControlRecipeList control is used to create control recipes, view current control recipes, and command a control recipe outside of the FactoryTalk Batch View. Once a control recipe is created it is displayed in the ControlRecipeList until it is removed by the operator (even if the control itself is closed), or the FactoryTalk Batch Server is restarted with a cold boot.

Similar to the Batch List window within FactoryTalk Batch View, the ControlRecipeList is used to create and command a control recipe.

To access the ControlRecipeList ActiveX control:

- Place the control within a web browser or other ActiveX container.
- Place the control on a Visual Basic form and open it as an executable.

Consult your system administrator for information on using these options.

See also

- ControlRecipeList interface on page 13
- ControlRecipeList Shortcut Menu on page 14
- Control recipe commands on page 16
- Add a control recipe on page 18

The appearance of the ControlRecipeList interface varies depending on the container and how the control is configured within that container. The ControlRecipeList interface functions the same regardless of the container and performs all of the standard functions of the Batch List window in the FactoryTalk Batch View without custom modifications. The information displayed in the ControlRecipeList window is user-configurable.
The following data can be displayed:

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch ID</td>
<td>User-entered or system-generated identification of the control recipe. Define each batch with a unique Batch ID.</td>
</tr>
<tr>
<td>Recipe</td>
<td>Name of the master recipe from which the control recipe was created.</td>
</tr>
<tr>
<td>Description</td>
<td>Description of the master recipe.</td>
</tr>
<tr>
<td>Start Time</td>
<td>Time the control recipe was started.</td>
</tr>
<tr>
<td>Elapsed Time</td>
<td>Elapsed time measured from the start time, updated after completion of a phase.</td>
</tr>
<tr>
<td>State</td>
<td>Current state of the control recipe.</td>
</tr>
<tr>
<td>Mode</td>
<td>Current mode of the control recipe.</td>
</tr>
<tr>
<td>Failure</td>
<td>Failure information for the control recipe.</td>
</tr>
<tr>
<td>Process Cell</td>
<td>Name of the process cell in which the control recipe is running.</td>
</tr>
<tr>
<td>Units in Use</td>
<td>Units currently in use by the control recipe.</td>
</tr>
<tr>
<td>Phases in Use</td>
<td>Phases the control recipe is currently running.</td>
</tr>
<tr>
<td>CreateID</td>
<td>Unique ID assigned by FactoryTalk Batch to the control recipe when it is created.</td>
</tr>
<tr>
<td>Overrides</td>
<td>Visual indication that an overridden input parameter expression exists within the recipe. Note that the assignment of a value by overriding prevents the parameter's configured expression from running until it is directed to evaluate and replace the overridden value.</td>
</tr>
</tbody>
</table>

See also

[ControlRecipeList Control](#) on page 13
You can access the ControlRecipeList control shortcut menu by right-clicking anywhere within the control. The availability of the commands on the shortcut menu depends on the item. Create and command control recipes, and remove a control recipe from the ControlRecipeList control. All of the ControlRecipeList commands are accessible from the shortcut menu, but only valid commands are enabled.

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Start</td>
<td>Begins execution of the selected control recipe in the ControlRecipeList.</td>
</tr>
<tr>
<td>Restart</td>
<td>Restarts the selected control recipe from the Held state.</td>
</tr>
<tr>
<td>Hold</td>
<td>Places the selected control recipe in the Held state.</td>
</tr>
<tr>
<td>Stop</td>
<td>Stops the selected control recipe.</td>
</tr>
<tr>
<td>Abort</td>
<td>Aborts the selected control recipe.</td>
</tr>
<tr>
<td>Manual</td>
<td>Places the selected control recipe in MANUAL operating mode.</td>
</tr>
<tr>
<td>Semi-Auto</td>
<td>Places the selected control recipe in SEMI-AUTO operating mode.</td>
</tr>
<tr>
<td>Auto</td>
<td>Places the selected control recipe in AUTOMATIC mode.</td>
</tr>
<tr>
<td>Resume</td>
<td>Resumes processing for the selected control recipe. The control recipe must be in the SEMI-AUTO mode for this option to be enabled.</td>
</tr>
<tr>
<td>Create Control Recipe</td>
<td>Opens the Master Recipe List dialog box allowing you to add a control recipe to the ControlRecipeList.</td>
</tr>
<tr>
<td>Remove</td>
<td>Removes the selected control recipe from the ControlRecipeList.</td>
</tr>
<tr>
<td>Comment</td>
<td>Allows you to enter a comment for the selected control recipe, or for all control recipes on the Batch List.</td>
</tr>
<tr>
<td>Clear Failures</td>
<td>Clears all failures and system-generated messages (3000 and 13000 requests) for the selected control recipe.</td>
</tr>
</tbody>
</table>
Log in to an ActiveX control

Log in to verify your permissions to use the functions of the ActiveX control, depending on the defined security settings. If the ActiveX control is invoked in a container which already has an established FactoryTalk User authenticated, then the established user is automatically authenticated.

Important: When the ProcedureView ActiveX control connects to a running FactoryTalk Batch Server which is an incompatible version, the control displays the following message:

"The ProcedureView ActiveX control is incompatible with the current Batch Server. It is strongly recommended that you upgrade your system before continuing."

THIS IS AN UNTESTED, UNSUPPORTED STATE. (See Version compatibility with FactoryTalk Batch Server for more information.)

To log in to an ActiveX control:

1. Right-click the ActiveX control and select Login from the shortcut menu.
2. Enter your user name in the User Name box.
3. Enter your password in the Password box.
4. Click OK to log in.

Tip: To log out of the ActiveX control, log off of the FactoryTalk Directory (if single-signon is enabled) or log off Windows.

See also

ActiveX Controls library introduction on page 7
Version compatibility with FactoryTalk Batch Server on page 11

Control recipe commands

Issue control recipe commands from the ControlRecipeList or ProcedureView.

Use the ControlRecipeList to create and command control recipes, and provide an overview of all control recipes that are running, ready to run, and/or complete. To see the shortcut menu, right-click anywhere in the ControlRecipeList.

Use the ProcedureView to command control recipes currently on the Batch List. Change control recipe states and modes, view and change step formula values,
control, and perform active step changes. Right-clicking anywhere within the ProcedureView control to open the shortcut menu.

See also

Start a control recipe on page 17
Change the state of a control recipe on page 17
Change the mode of a control recipe on page 21

**Start a control recipe**

Use the shortcut menu to start a control recipe.

**To start a control recipe:**

1. Select a control recipe.
   a. In the BatchList, select the control recipe.
   b. In the ProcedureView, select the procedure level of the control recipe.

   **Tip:** To activate the shortcut menu, if the SFC view is the active view, right-click outside of the SFC (do not click on a recipe step, comment, or transition), or if the Table view is the active view, right-click the table header. The ProcedureView shortcut menu opens.

2. Activate the shortcut menu.
   a. In the BatchList, right-click the control recipe.
   b. In the SFC view, right-click outside of the SFC (do not click on a recipe step, comment, or transition).
   c. In the Table view, right-click the table header.

3. Select **Start** from the shortcut menu.

See also

Control recipe commands on page 16
Change the state of a control recipe on page 17
Change the mode of a control recipe on page 21

**Change the state of a control recipe**

Use the shortcut menu to change the state of a control recipe to **Started**, **Restarted**, **Held**, **Stopped** or **Aborted**.
To change the state of a control recipe:

1. Select the control recipe.
   a. In the **BatchList**, select the control recipe.
   b. In the **ProcedureView**, select the procedure level of the control recipe.

2. Activate the shortcut menu.
   a. In the **BatchList**, right-click the control recipe.
   b. In the SFC view, right-click outside of the SFC (do not click on a recipe step, comment, or transition).
   c. In the Table view, right-click the table header.

3. Select the appropriate command from the shortcut menu.

See also

- [Control recipe commands](#) on page 16
- [Change the mode of a control recipe](#) on page 21
- [Start a control recipe](#) on page 17
Add a control recipe

Add a control recipe to the ControlRecipeList.

⚠️ If control strategies are enabled for a recipe, changing the selected control strategy after the recipe is added to the batch list could corrupt that recipe.

To add a control recipe:

1. Right-click the ControlRecipeList control and select Create Control Recipe.
   
   Tip: If your recipes do not display in the list, then close and reopen the Master Recipe List dialog box.

2. From the Filtering Column, select an option to filter the list of recipes, and then click Refresh.
   
   Tip: If the recipe list is empty, click Cancel to close the Master Recipe List dialog box. Click the Add button again. If the recipe list is still empty, please refer to the Administrator Guide for instructions on configuring Server Options to point to your recipe directory.

3. Select the appropriate master recipe and click OK.
   
   Tip: If the Batch Creation Errors dialog box opens, see the section View Batch Creation Errors for more information.

4. If the recipe contains a material class-based phase, the Material Value Selection dialog box opens. If not, skip to step 6.

5. From the Material list, select the material you want to use in this particular batch, and then click Create.

6. The Batch Creation dialog box opens.
   
   ⚠️ If you are creating a batch of a material-based recipe and the FactoryTalk Batch Server has lost communication with the Material Server and is set to continue processing in MANUAL mode, the following warning will display: "WARNING: MATERIAL SERVER NOT IN USE." You can continue to create and run the batch, but automatic binding will not occur. You are prompted to select the container/phase pair that can supply the correct material to continue processing the batch. Use extreme caution when operating without the Material Server. The container/phase list is not filtered by the Material Server when it is unavailable. Failure to select the correct container/phase pair could result in a hazardous situation, depending on the material.

7. If a Batch ID is not automatically generated, enter a Batch ID.
Chapter 2  ControlRecipeList Control

Tip: If the Batch ID box appears dimmed, you cannot change it. The option to make the Batch ID editable is set by the system administrator in the System Configuration and Defaults dialog box on the General tab.

8. In the Scale box, type a percentage by which you want to scale the recipe, if the batch is to be run at a percentage other than 100%.

9. (optional) If there are formula values that need to be assigned or modified, select the Value cell in the Formula Values area and enter a new value.

Tip: If this is a class-based recipe, and the recipe has a unit allocation method of At Batch Creation or Operator Choice, you must specify the unit allocation requirements. If the unit allocation method is At Batch Creation, you must choose a specific unit from a list of the available units. If the unit allocation method is Operator Choice, then you can choose from a list of available units, or you can select the Prompt or First Available unit allocation methods.

10. Click the Bound Unit cell in the Unit Binding area to activate the Bound Unit list. Make an appropriate selection.

Tip: If the recipe does not require that any formula values be assigned, the Formula Values area contains a single blank row. Likewise, if the recipe does not require any unit binding, the Unit Binding area contains a single blank row.

11. Click Create to add the batch to the Batch List window. The Start command button is enabled.

See also

ControlRecipeList Control on page 13

View Batch creation errors on page 20

**View Batch creation errors**

If any problems exist in the area model or recipe that will prevent a batch from running, either the No Legal Units Summary dialog box opens or a recipe approval error notification is shown. When a recipe is set to Release Recipe to Production, but makes reference to one or more sub-recipes or embedded procedures that are not set to Release Recipe as Step, that recipe cannot be used to create a batch. Ensure that all components contained in the recipe are set to Release Recipe as Step.

If dynamic unit allocation is enabled for the recipe, the No Legal Units Summary dialog box includes a summary error message, the reason for the error, the binding requirements and the binding requirement expression values for each unit requirement.

Tip: If dynamic unit allocation is not enabled, a message will display the error that prevents the batch from being created.
ControlRecipeList Control

### No Legal Units Summary Section

<table>
<thead>
<tr>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shows all the summary error messages returned by the failed execute and the associated unit requirement.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Summary Message</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shows the unit requirement(s) along with each unit that failed the unit requirement and the associated reason.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Unit Summary Information</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shows the unit requirement(s) along with each unit that failed the unit requirement and the associated reason.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Unit Binding Requirements</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shows the binding requirement information for those unit requirements whose reason for exclusion was a binding requirement, including the unit requirement, the binding requirement ID, and the description of the binding requirement.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Expression Data</th>
</tr>
</thead>
<tbody>
<tr>
<td>Shows any expression data information for the binding requirements displayed in the Unit Binding Requirements section that are of an expression type. Included is the unit requirement, the binding requirement ID, the unit name, the result and the expression values.</td>
</tr>
</tbody>
</table>

See also

- ControlRecipeList Control on page 13
- Add a control recipe on page 18

### Change the mode of a control recipe

Use the shortcut menu to change the mode of a control recipe from AUTO to MANUAL or SEMI-AUTO.

**To change the mode of a control recipe:**

1. Select the control recipe.
   - a. In the BatchList, select the control recipe.
   - b. In the ProcedureView, select the procedure level of the control recipe.

2. Activate the shortcut menu.
   - a. In the BatchList, right-click the control recipe.
   - b. In the SFC view, right-click outside of the SFC (do not click on a recipe step, comment, or transition).
   - c. In the Table view, right-click the table header.
3. Select the appropriate state from the shortcut menu.

   **Tip:** The ProcedureView does not indicate when a control recipe has paused while in SEMI-AUTO or MANUAL modes. To determine if a control recipe is PAUSED, open the ProcedureView shortcut menu. If a control recipe is PAUSED, the Resume command is enabled.

**See also**

- Control recipe commands on page 16
- Change the state of a control recipe on page 17
- Start a control recipe on page 17

### Remove a control recipe from the ControlRecipeList

Once a control recipe has completed or is no longer needed, remove it using the shortcut menu.

**To remove a control recipe from the ControlRecipeList:**

1. To remove a control recipe from the ControlRecipeList, the recipe state must be STOPPED, COMPLETE, ABORTED, READY or IDLE.

2. Right-click the appropriate control recipe, then select Remove on the ControlRecipeList shortcut menu.

**See also**

- ControlRecipeList Control on page 13

### Cross invocation

Cross invocation is the process of passing configured context information to an automation server. This information is relative to the current selection within the ActiveX controls. When cross invocation is enabled and an automation server is identified, the shortcut menu displays the Cross Invocation menu items.

   **Tip:** Cross invocation is enabled by setting two properties in the ActiveX Controls Library: InvocationProgID and ShowCustomMenuItems.

The items in the shortcut menu are formatted using cross invocation strings associated with the equipment resource. These cross invocation strings also indicate the context data to be passed to the automation server.
The FactoryTalk Batch selectable elements for cross invocation from the ActiveX controls are:

<table>
<thead>
<tr>
<th>Control</th>
<th>Selectable Elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>ControlRecipeList</td>
<td>Batch</td>
</tr>
<tr>
<td>ProcedureView</td>
<td>Batch, Recipe Phase, Operation, and Unit Procedure</td>
</tr>
<tr>
<td>PromptsList</td>
<td>Unacknowledged Prompts (not binding requests)</td>
</tr>
</tbody>
</table>

See also

Perform cross invocation on page 23

Perform cross invocation

When you enable cross invocation and identify the automation server, the shortcut menu displays the Cross Invocation menu items. The shortcut menu items are formatted using cross invocation strings associated with the equipment resource.

Tip: Cross invocation is enabled by setting two properties in the ActiveX Controls Library: InvocationProgID and ShowCustomMenuItems.

To perform cross invocation:

1. Right-click the appropriate selection in the ActiveX controls. The configured Cross Invocation menu items display.
2. Select a menu or sub-menu item. The context data is automatically passed to the configured automation server.

Tip: Check with your system administrator as to what context data is configured for each of the shortcut menu items displayed on your system.

See also

Cross invocation on page 22

Work with batches without the Material Server

When running material-based recipes, the FactoryTalk Batch Server and the Material Server communicate with each other. Use the FactoryTalk Batch Equipment Editor to configure how the system handles batches if communication between the servers is lost. You can set the FactoryTalk Batch Server to fail and hold execution of material-based recipes, or switch to manual processing, prompting the operator to make binding decisions until communication with the Material Server is restored. (See the FactoryTalk Batch Equipment Editor User Guide for more information on setting the Material Server communication options.)
If the FactoryTalk Batch Server is configured to **Failure and Hold**, you cannot continue running batches or add new batches to the **Batch List** view until communication with the Material Server is restored.

If the FactoryTalk Batch Server is configured to **Switch to Manual**, you can continue processing batches and add batches to the **Batch List** view without input from the Material Server. All material-based decisions or selections that are made by the operator without communications with the Material Server are reported in the batch event journal as Loss of Material Server and Loss of Material Tracking events. When the material database becomes available, you must manually reconcile the event journal with the material journal and update the material database with the Amount, Lot, and Label information for material inventory.

---

**Exercise extreme caution when operating without the Material Server.** Loss of communication with the Material Server causes the FactoryTalk Batch Server to switch to prompt binding and presents an unfiltered list of container/phase pairs from which to select binding candidates. You must select the container/phase pair that can supply the correct material to continue processing the batch. Failure to select the correct container/phase pair could result in a hazardous situation, depending on the material.

---

When processing material-based batches manually, you are required to make selections without information from the material database.

**See also**

- [Reestablish communication with the Material Server on manual batches](#) on page 29
- [Batch creation when the Material Server is not communicating](#) on page 25
- [Prompt to bind unit requirement](#) on page 26
- [Manual unit binding](#) on page 26
- [Prompt to bind a material phase step](#) on page 27
Material Server not in use warning

The Material Server Not in Use warning displays in the Batch Creation dialog box when communication with the Material Server is lost. Continue to create the batch or cancel and create the batch at another time when the Material Server is available.

![Batch Creation dialog box]

See also

Work with batches without the Material Server on page 23

Batch creation when the Material server is not communicating

When there is no communication with the Material server, the Material Server Not in Use warning is displayed in the Batch Creation dialog box for all recipes. You can create the batch or cancel and create the batch at another time when the Material server is available.

If the failure to the Material server occurs when the Batch Creation dialog box is open, the warning message does not show.

If you add a material class-based recipe to the batch list when the Material server is unavailable, the materials will not be filtered by class when the recipe is run. Instead, all materials in the material enumeration are shown.

![Material Value Selection dialog box]
If the failure to the Material server occurs when the Material Value Selection dialog box is open, the warning message does not show.

See also

Prompt to bind unit requirement on page 26

Manual unit binding on page 26

Prompt to bind a material phase step on page 27

Prompt to bind unit requirement

The Data Not Supplied by Material Server warning displays in the Prompt to Bind Unit Requirement dialog box when there is no communication with the Material server.

The dialog box displays data supplied by the area model in place of the data normally supplied by the Material server. When the Material server is not available, the first available binding is serviced as though it was prompt binding.

If the failure to the Material server occurs when the Prompt to Bind Unit Requirement dialog box is open, the warning message does not show.

See also

Batch creation when the Material server is not communicating on page 25

Manual unit binding on page 26

Prompt to bind a material phase step on page 27
Manual unit binding

The Material Server Not in Use warning displays in the Manual Bind of Step dialog box when there is no communication with the Material server.

The dialog box displays data supplied by the area model in place of the data normally supplied by the Material server.

If the failure to the Material Server occurs when the Manual Bind of Step dialog box is open, the warning message does not show.

See also

- Batch creation when the Material server is not communicating on page 25
- Prompt to bind unit requirement on page 26
- Prompt to bind a material phase step on page 27

Prompt to bind a material phase step
For prompt binding, the list of container/phase pairs generated from the area model is substituted for the list that would have been produced if the Material server were available. When the Material server is unavailable, material additions may not be filtered by lot or label specification.

If you do not know which phase to select, you can look up some material information on the computer where the Material server is installed and running.

If the failure to the Material server occurs when the **Prompt to Bind a Material Phase Step** dialog box is open, the warning message does not show.

**Look up material information**

If you do not know which phase to select when there is no communication with the Material server, you can look up material information on the computer where the Material server is installed and running.

**To look up material information:**

1. Open the **Material Editor**.
2. Navigate to **Independent Containers or Materials**.
3. Double-click the desired container or material to see its properties.
4. Check the **Materials**, **Containers**, and **Inventory** tabs and compare the values to the **Unit**, **Material**, and **Amount to Charge** to determine which phase can best accommodate the material phase step.
If the failure to the Material server occurs when the **Prompt to Bind a Material Phase Step** dialog box is open, the warning message does not show.

See also

- **Prompt to bind a material phase step** on page 27
- **Batch creation when the Material server is not communicating** on page 25

If the FactoryTalk Batch Server is configured to **Switch to Manual**, you must reestablish communication with the Material server when it becomes available again. When the Material Status indicator changes to Suspect (**MT-S**), the Material server is available and you can attempt reconnection.

**To reestablish communication with the Material Server on manual batches:**

1. Double-click the **Material Server Status** indicator.
2. Click the **Material Server Control** button.
3. Manually reconcile the batch event journal with the material database.
4. Select the **Reestablish Communication with Material Server** check box, and then click **OK**.

See also

- **Acknowledge a prompt** on page 33
Chapter 3

PromptsList Control

The PromptsList control displays a list of prompts waiting to be acknowledged. Prompts alert the operator to the parameter value requests. A control recipe cannot complete execution until the operator acknowledges the pending prompts by providing the requested parameter value(s).

Similar to the Unacknowledged Prompts window in the FactoryTalk Batch View, the operator uses the PromptsList to acknowledge any prompts generated while running a control recipe. Unacknowledged prompts remain in the PromptsList until the operator acknowledges the prompt or until the control recipe has been STOPPED or ABORTED.

To use the PromptsList ActiveX control:

- Place the control within a web browser or other ActiveX container.
- Place the control on a Visual Basic form, and open it as an executable.

Consult your system administrator for information on using these options.

See also

- PromptsList interface on page 31
- PromptsList shortcut menu on page 32
- Acknowledge a prompt on page 33

PromptsList interface

The appearance of the PromptsList varies depending on the container in which it is placed, and how it is configured within that container. Use the control to recognize and respond to any Unacknowledged Prompts.

The following data is shown, if configured:

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td>The time that the prompt occurred.</td>
</tr>
<tr>
<td>Batch ID</td>
<td>The user-entered or system-generated identification of the control recipe that generated the prompt. Define each batch with a unique Batch ID.</td>
</tr>
<tr>
<td>Source Step</td>
<td>The recipe name, procedure, operation and phase where the prompt originated.</td>
</tr>
</tbody>
</table>
Chapter 3  PromptsList Control

The PromptsList ActiveX control provides a shortcut menu that is accessed by right-clicking anywhere within the PromptsList control. Use the shortcut menu to log in to the FactoryTalk Batch View Security and acknowledge prompts. The Acknowledge option on the shortcut menu is specific to the prompt you click on, and is disabled if a prompt is not clicked.

Log in to an ActiveX control

Log in to verify your permissions to use the functions of the ActiveX control, depending on the defined security settings. If the ActiveX control is invoked in a container which already has an established FactoryTalk User authenticated, then the established user is automatically authenticated.

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td>The time that the prompt occurred.</td>
</tr>
<tr>
<td>Description</td>
<td>A description of the prompt.</td>
</tr>
<tr>
<td>Type</td>
<td>The event type (prompt).</td>
</tr>
<tr>
<td>Default</td>
<td>The default value of the prompt.</td>
</tr>
<tr>
<td>EU</td>
<td>The Engineering Units associated with the prompt (i.e., inches, gallons, etc.).</td>
</tr>
<tr>
<td>Area</td>
<td>The area model in which the phase that activated the prompt is running.</td>
</tr>
<tr>
<td>Process Cell</td>
<td>The name of the process cell in which the control recipe that generated the prompt is running.</td>
</tr>
<tr>
<td>Unit</td>
<td>The unit where the prompt originated.</td>
</tr>
<tr>
<td>Phase</td>
<td>The phase where the prompt originated.</td>
</tr>
<tr>
<td>Prompt ID</td>
<td>The unique ID assigned to the prompt when it is generated.</td>
</tr>
</tbody>
</table>

See also

PromptsList Control on page 31

Log in to an ActiveX control on page 16
Important: When the ProcedureView ActiveX control connects to a running FactoryTalk Batch Server which is an incompatible version, the control displays the following message:

"The ProcedureView ActiveX control is incompatible with the current Batch Server. It is strongly recommended that you upgrade your system before continuing."

This is an untested, unsupported state. (See Version compatibility with FactoryTalk Batch Server for more information.)

To log in to an ActiveX control:

1. Right-click the ActiveX control and select Login from the shortcut menu.
2. Enter your user name in the User Name box.
3. Enter your password in the Password box.
4. Click OK to log in.

Tip: To log out of the ActiveX control, log off of the FactoryTalk Directory (if single-signon is enabled) or log off Windows.

See also

- ActiveX Controls library introduction on page 7
- Version compatibility with FactoryTalk Batch Server on page 11

Acknowledge a prompt

Batch execution cannot complete until prompts are acknowledged. The prompt is removed when the acknowledgment is complete.

To acknowledge a prompt:

1. Select Unacknowledged Prompts.
2. Double-click the prompt to be acknowledged.
3. Type the appropriate value for the parameter.
4. Select Acknowledge.

See also

- Acknowledge a material phase step prompt on page 33

Acknowledge a material phase step prompt

Prompts alert operators when requests for parameter values are issued. Control recipe execution cannot complete until the prompt(s) are acknowledged.
To acknowledge a material phase step prompt:

1. Open the **Prompt to Bind a Material Phase Step** dialog box and select a container and phase to use.

2. Click the **Unacknowledged Prompts** button to open the **Unacknowledged Prompts** window.

3. Double-click the prompt. The **Prompt to Bind a Material Phase Step** dialog box opens with the current prompt data shown.

4. Select a lot or storage container to bind the material to and click **OK**. The prompt is acknowledged and removed from the **Prompts List**.

See also

[Acknowledge a prompt on page 33](#)

**Cross invocation**

Cross invocation is the process of passing configured context information to an automation server. This information is relative to the current selection within the ActiveX controls. When cross invocation is enabled and an automation server is identified, the shortcut menu displays the **Cross Invocation** menu items.

**Tip:** Cross invocation is enabled by setting two properties in the ActiveX Controls Library: **InvocationProgID** and **ShowCustomMenuItems**.

The items in the shortcut menu are formatted using cross invocation strings associated with the equipment resource. These cross invocation strings also indicate the context data to be passed to the automation server.

The FactoryTalk Batch selectable elements for cross invocation from the ActiveX controls are:

<table>
<thead>
<tr>
<th>Control</th>
<th>Selectable Elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>ControlRecipeList</td>
<td>Batch</td>
</tr>
<tr>
<td>ProcedureView</td>
<td>Batch, Recipe Phase, Operation, and Unit Procedure</td>
</tr>
<tr>
<td>PromptsList</td>
<td>Unacknowledged Prompts (not binding requests)</td>
</tr>
</tbody>
</table>

See also

[Perform cross invocation on page 23](#)

**Perform cross invocation**

When you enable cross invocation and identify the automation server, the shortcut menu displays the **Cross Invocation** menu items. The shortcut menu items are formatted using cross invocation strings associated with the equipment resource.
Tip: Cross invocation is enabled by setting two properties in the ActiveX Controls Library: Invocatio
ProgID and ShowCustomMenuItems.

To perform cross invocation:

1. Right-click the appropriate selection in the ActiveX controls. The configured Cross Invocation menu items display.

2. Select a menu or sub-menu item. The context data is automatically passed to the configured automation server.

Tip: Check with your system administrator as to what context data is configured for each of the shortcut menu items displayed on your system.

See also

Cross invocation on page 22
Chapter 4

SignatureList Control

The SignatureList control displays a list of signatures associated with a batch. Signatures are requests for signoffs that require user permissions and optional comments. When a signature is generated against a batch, acknowledge the signature by completing the associated signoffs. There are several types of signatures that can be generated against a batch prior to, during, or after batch execution.

The SignatureList control is part of the Electronic Signatures feature, which enables signatures to be electronically recorded.

To use the SignatureList ActiveX control:

- Place the control within a web browser or other ActiveX container.
- Place the control on a Visual Basic form, and open it as an executable.

See also

- SignatureList Interface on page 37
- SignatureList control shortcut menu on page 38
- Acknowledge a signature on page 39
- Signature cancellation on page 39
- Configure the SignatureList control in an external application on page 40

SignatureList Interface

The appearance of the SignatureList control varies depending on the container, and how it is configured within that container. Use the SignatureList control to recognize and respond to pending signatures in the SignatureList.

The following data is displayed, if configured:

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Time</td>
<td>The time the signature was generated.</td>
</tr>
<tr>
<td>Batch ID</td>
<td>The user-entered or system-generated identification of the control recipe that generated the signature. Define each batch with a unique Batch ID.</td>
</tr>
</tbody>
</table>
### SignatureList Control

**Item** | **Description**
--- | ---
Procedure ID | The ID of the procedure that generated the signature.
Type | The signature type.
State | The signature state.
Signoff Meaning | A short phrase describing the meaning attached to a specific signature signoff.
Signoff Security | The user name and password required for a given signature signoff.
Signature Context | Data specific to the defined signature type.
Action ID | A unique ID assigned to a queued execute request that is awaiting completion of one of more signature requests.

**See also**

SignatureList Control on page 37

**SignatureList control shortcut menu**

The SignatureList ActiveX control provides a shortcut menu, accessed by right-clicking anywhere within the SignatureList control. The shortcut menu allows you to log in to the FactoryTalk Batch View Security.

**See also**

SignatureList Control on page 37

**Log in to an ActiveX control**

Log in to verify your permissions to use the functions of the ActiveX control, depending on the defined security settings. If the ActiveX control is invoked in a container which already has an established FactoryTalk User authenticated, then the established user is automatically authenticated.

**Important:** When the ProcedureView ActiveX control connects to a running FactoryTalk Batch Server which is an incompatible version, the control displays the following message:

"The ProcedureView ActiveX control is incompatible with the current Batch Server. It is strongly recommended that you upgrade your system before continuing."

THIS IS AN UNTESTED, UNSUPPORTED STATE. (See Version compatibility with FactoryTalk Batch Server for more information.)

**To log in to an ActiveX control:**

1. Right-click the ActiveX control and select **Login** from the shortcut menu.
2. Enter your user name in the **User Name** box.
3. Enter your password in the **Password** box.
4. Click **OK** to log in.
To acknowledge a signature, enter the signoff information that is required for validation. Selecting the Close button without completing a signature closes the dialog box and leaves the signature on the signature list.

Tip: Signatures contain: Signoffs (1 to 3), Comments (Required|Optional|None), Last Signoff (Optional - marked with an asterisk).

To acknowledge a signature:

1. Click the flashing Signature List button to display the Signature List window.

2. Double-click the appropriate signature to open the Signature dialog box.

3. Enter the User ID and Password for the selected signoff.

4. Enter a comment for the selected signoff (if required).

5. Click the Sign button.

Tip: The Signer must be listed under Security Requirements or be a member of one of the groups in the list to complete a signoff.

See also

Signature cancellation

If a signature request is cancelled, the action that generated the signature is also cancelled. Selecting the Cancel Signature button in a Signature dialog box displays a warning. If you select the Yes button, the associated action is cancelled and the signature state is changed to Cancelled.

See also

SignatureList Control
Configure the SignatureList control in an external application

In order for the SignatureList control to work in any client application, including the eProcedure Client, that does not call CoInitializeSecurity (this includes Internet Explorer and Visual Basic and potentially HMI applications), configure the SignatureList control on the client computer.

To configure the SignatureList control in an external application:

1. If the FactoryTalk Batch Server runs under a local machine account (workgroup), create an account with the same user name and password on the client machine.

2. Open the Component Services window.
   - Windows XP Professional SP3: Select Start > All Programs > Control Panel > Administrative Tools > Component Services.


4. Click on the [Default] COM Security tab, and click the Edit Default button in the Access Permissions area.

5. Click Add.

6. Select the correct location in the From this location box by doing one of the following:
   - If the FactoryTalk Batch Server is running on a domain, select the Server domain name.
   - If the Server is running on a local user account (workgroup), select the name of the FactoryTalk Batch Client computer.

7. In the Enter the object names to select box, add the account the Server runs as to the list and click Check Names. This verifies that the account exists.
   - If the Server is running on a domain, choose the Server’s domain account.
   - If the Server is running on a local user account (workgroup), select the name of the local account under which the Server runs.

8. Click OK.
9. Click **OK** to close the **Access Permission** and **My Computer Properties** dialog boxes, then close the **Component Services** window.

10. Log off and then log back on.

See also

[SignatureList Control](#) on page 37
ProcedureView Control

The ProcedureView control provides the user with a graphical representation of a control recipe that has been placed on the Batch List. Control recipes can be viewed in table format or as a sequential function chart (SFC). You can navigate through all procedural levels of a recipe using the ProcedureView control. Use the control to view a current control recipe, command control recipes, perform manual binding, perform active step changes, and change recipe step parameter values, and view any unit binding requirements and binding preferences.

Like the Procedure as Table and Procedure as SFC windows located within the FactoryTalk Batch View, use the ProcedureView to command and view control recipes that have been added to the Batch List.

To use the ProcedureView ActiveX control:

- Place the control within a web browser or other ActiveX container.
- Place the control on a Visual Basic form, and open it as an executable.

The appearance of the ProcedureView varies depending on the container in which it is placed, and how it is configured within that container. The control functions in the same way regardless of the container in which it is placed.

See also

Table View interface on page 44

SFC View Interface on page 45

ProcedureView navigation on page 47

ProcedureView control recipe properties on page 51
The ProcedureView control’s table view provides a list-based representation of recipe logic flow. Steps are executed in order, starting with the top row, and proceed through the table. Each row consists of an individual phase and its associated information.

The following data is displayed:

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Name</td>
<td>The recipe step name.</td>
</tr>
<tr>
<td>State</td>
<td>The recipe step current state.</td>
</tr>
<tr>
<td>Mode</td>
<td>The recipe step current mode.</td>
</tr>
<tr>
<td>Failure</td>
<td>The failure information associated with the recipe step.</td>
</tr>
<tr>
<td>Unit</td>
<td>The name of the unit in which the recipe step is running.</td>
</tr>
<tr>
<td>Owner</td>
<td>The recipe step’s current owner.</td>
</tr>
<tr>
<td>Step Index</td>
<td>The recipe step’s current step index.</td>
</tr>
<tr>
<td>Message</td>
<td>The recipe step’s message.</td>
</tr>
<tr>
<td>Request</td>
<td>The recipe step’s pending request.</td>
</tr>
<tr>
<td>Parameter</td>
<td>The recipe step’s parameter value. The number of parameters displayed in the table is determined by the DisplayedParameters property.</td>
</tr>
</tbody>
</table>

See also

- ProcedureView Control on page 43
- SFC View on page 47
The **ProcedureView** control’s SFC view provides a graphic representation of recipe logic flow. Steps are executed in order from top to bottom through the SFC. The SFC structure is navigated by double-clicking a recipe step to navigate downward through the recipe hierarchy, or by double-clicking an initial step to navigate upward through the recipe hierarchy.

**See also**

[ProcedureView Control](#) on page 43

[Table View interface](#) on page 44
**ProcedureView shortcut menu**

The **ProcedureView** ActiveX control provides a shortcut menu accessed by right-clicking anywhere within the control. Use the control to log on to the FactoryTalk Batch View Security, and to command and view control recipes that have been added to the Batch List. The menu items on the shortcut menu are specific to the recipe step or step transition that was right-clicked, and they are disabled if the menu item is not valid for the selected element. If the right-click does not occur on a recipe step or step transition, the shortcut menu items will be applicable to the recipe level currently being displayed.

![ProcedureView shortcut menu](image)

**See also**

- [ProcedureView Control](#) on page 43
- [ProcedureView navigation](#) on page 47

**Log in to an ActiveX control**

Log in to verify your permissions to use the functions of the ActiveX control, depending on the defined security settings. If the ActiveX control is invoked in a container which already has an established FactoryTalk User authenticated, then the established user is automatically authenticated.

---

**Important:**  When the ProcedureView ActiveX control connects to a running FactoryTalk Batch Server which is an incompatible version, the control displays the following message:

"The ProcedureView ActiveX control is incompatible with the current Batch Server. It is strongly recommended that you upgrade your system before continuing."

THIS IS AN UNTESTED, UNSUPPORTED STATE. (See Version compatibility with FactoryTalk Batch Server for more information.)
To log in to an ActiveX control:

1. Right-click the ActiveX control and select **Login** from the shortcut menu.
2. Enter your user name in the **User Name** box.
3. Enter your password in the **Password** box.
4. Click **OK** to log in.

**Tip:** To log out of the ActiveX control, log off of the FactoryTalk Directory (if single-signon is enabled) or log off Windows.

**See also**

- [ActiveX Controls library introduction](#)
- [Version compatibility with FactoryTalk Batch Server](#)

**ProcedureView navigation**

The ProcedureView displays the SFC or table for the procedural level defined by the **TopProcedureID** property of the control.

**See also**

- [SFC View](#)
- [Table View](#)

**SFC View**

Similar to the **Procedure as SFC** window within the FactoryTalk Batch View, the SFC view of the ProcedureView displays the sequential function charts (SFCs) of the currently selected control recipe. Use the control to watch a control recipe execute its steps and transitions.

To navigate to a lower recipe level, double-click a step, other than the initial step, in the SFC. The SFC for that step is displayed. Control recipes can be navigated in this way to the operation level. To navigate up through the control recipe, double-click the initial step of the SFC.

The magnification of the displayed SFC can be adjusted by selecting **Zoom** from the shortcut menu and then selecting the zoom percentage. To view the entire SFC within the ProcedureView, select **Zoom** and then select **Whole Procedure**. Text is not included in the depiction of the current procedure when the zoom percentage is at or below 35%.

**See also**

- [ProcedureView Control](#)
- [Table View](#)
Table View

Similar to the Procedure as Table window within the FactoryTalk Batch View, the table view of the ProcedureView displays a control recipe in a table format. It allows you to see components of every procedural element in the selected control recipe. Each row of the table represents a recipe step within the control recipe. Step transitions are not displayed by the table view.

To navigate to a lower recipe level, double-click a step in the table. The table for that step is displayed. Control recipes can be navigated down in this way to the operation level. To navigate up through the control recipe, double-click the table header, located at the top of the table.

See also

ProcedureView navigation on page 47

SFC View on page 47

Control recipe commands

Issue control recipe commands from the ControlRecipeList or ProcedureView.

Use the ControlRecipeList to create and command control recipes, and provide an overview of all control recipes that are running, ready to run, and/or complete. To see the shortcut menu, right-click anywhere in the ControlRecipeList.

Use the ProcedureView to command control recipes currently on the Batch List. Change control recipe states and modes, view and change step formula values, perform manual binding, and perform active step changes. Right-clicking anywhere within the ProcedureView control to open the shortcut menu.

See also

Start a control recipe on page 17

Change the state of a control recipe on page 17

Change the mode of a control recipe on page 21

Start a control recipe

Use the shortcut menu to start a control recipe.

To start a control recipe:

1. Select a control recipe.

   a. In the BatchList, select the control recipe.

   b. In the ProcedureView, select the procedure level of the control recipe.
Tip: To activate the shortcut menu, if the SFC view is the active view, right-click outside of the SFC (do not click on a recipe step, comment, or transition), or if the Table view is the active view, right-click the table header. The ProcedureView shortcut menu opens.

2. Activate the shortcut menu.
   a. In the BatchList, right-click the control recipe.
   b. In the SFC view, right-click outside of the SFC (do not click on a recipe step, comment, or transition).
   c. In the Table view, right-click the table header.

3. Select Start from the shortcut menu.

See also

Control recipe commands on page 16

Change the state of a control recipe on page 17

Change the mode of a control recipe on page 21

Change the state of a control recipe

Use the shortcut menu to change the state of a control recipe to Started, Restarted, Held, Stopped or Aborted.

To change the state of a control recipe:

1. Select the control recipe.
   a. In the BatchList, select the control recipe.
   b. In the ProcedureView, select the procedure level of the control recipe.

2. Activate the shortcut menu.
   a. In the BatchList, right-click the control recipe.
   b. In the SFC view, right-click outside of the SFC (do not click on a recipe step, comment, or transition).
   c. In the Table view, right-click the table header.

3. Select the appropriate command from the shortcut menu.

See also

Control recipe commands on page 16
Change the mode of a control recipe

Use the shortcut menu to change the mode of a control recipe from AUTO to MANUAL or SEMI-AUTO.

To change the mode of a control recipe:

1. Select the control recipe.
   a. In the BatchList, select the control recipe.
   b. In the ProcedureView, select the procedure level of the control recipe.

2. Activate the shortcut menu.
   a. In the BatchList, right-click the control recipe.
   b. In the SFC view, right-click outside of the SFC (do not click on a recipe step, comment, or transition).
   c. In the Table view, right-click the table header.

3. Select the appropriate state from the shortcut menu.

   Tip: The ProcedureView does not indicate when a control recipe has paused while in SEMI-AUTO or MANUAL modes. To determine if a control recipe is PAUSED, open the ProcedureView shortcut menu. If a control recipe is PAUSED, the Resume command is enabled.

See also

Control recipe commands on page 16

Change the state of a control recipe on page 17

Start a control recipe on page 17
Timer step commands

If the recipe on the batch list contains timer steps and the timer step is in O_AUTO mode, right-click on the timer step to use the Timer-Complete and Timer-Reset commands. The commands are not enabled if the logged in user does not have permission to send the Timer-Complete or Timer-Reset commands. A confirmation dialog is shown if the Timer-Complete or Timer-Reset commands are configured to require confirmation.

Tip: The timer step must be in the RUNNING state for Timer-Complete and the RUNNING or HELD states for Timer-Reset.

In the ProcedureView, the shortcut menu accelerator keys support timer steps. For example, if both TIMER-RESET and RESUME are enabled, toggle between the selected command on the dialog by pressing alt+r and then pressing Enter to select the command. The TIMER-COMPLETE command is accessed the same way: alt+c toggles between TIMER-COMPLETE and CLEAR FAILURES.

See also

Control recipe commands on page 16

ProcedureView control recipe properties

Use the Properties dialog box of the ProcedureView ActiveX control to view properties associated with the different levels of the control recipe. The dialog box contains four tabs: Parameter, Report, Recipe, and Binding.

See also

Parameter tab on page 51
Recipe tab on page 52
Report tab on page 52
Binding tab on page 52

Parameter tab

The Parameter tab contains a list of all phase parameters associated with the selected phase. If the selected recipe step is not a phase, the Parameter tab does not contain any information. If the selected phase is a material-enabled phase, the material parameters of MATERIAL and AMOUNT are displayed.

Use the Parameter tab to view parameters and edit parameter values.

See also

ProcedureView control recipe properties on page 51
Edit parameter values on page 52
Edit parameter values

Edit parameter values using the **Parameter** tab in the **ProcedureView** control.

**To edit parameter values:**

1. Right-click a recipe step at the lowest (operation) level of the control recipe. This step represents a phase.
2. Select **Properties** from the shortcut menu.
3. Double-click a parameter from the **Parameter** tab.
4. Enter a new value in the **Value** box.
5. Click **OK** to save the new value.

**Tip:** You cannot edit the MATERIAL, MATERIAL_CLASS, or CONTAINER values for a material-enabled phase.

6. Click **OK**.

**See also**

- Parameter tab on page 51

Report tab

The **Report** tab contains a list of all report parameters associated with the selected phase. If the selected recipe step is not a phase, the **Report** tab does not contain any information. Displayed report parameters are view-only and are not editable. If the selected phase is a material-enabled phase, the report parameters are displayed.

**See also**

- ProcedureView control recipe properties on page 51

Recipe tab

The **Recipe** tab contains information entered by the recipe author when the master recipe was created (view-only). The information displayed is relevant to the procedural level being displayed within the **ProcedureView** control.

**See also**

- ProcedureView control recipe properties on page 51

Binding tab

The **Binding** tab displays the unit requirements for the step that is selected in the Procedural Hierarchy area or the SFC area. The information on the **Binding** tab includes:

- Unit Requirements for the batch procedure or the unit requirement associated with the step that is selected in the Procedural Hierarchy area or the SFC area.
- Binding Requirements associated with the step selected in the SFC or Table View.
- Binding Preferences associated with the step selected in the SFC or Table View.

At the unit procedure or unit operation level there is only one unit requirement.

See also

Expression values on page 53

ProcedureView control recipe properties on page 51

Expression values

Double-clicking a binding requirement or preference expression opens the Binding Requirement (or Preference) Expression Values view. The view lists the units that are potential legal binding candidates, a value indicating whether those candidates currently meet the selected binding requirement or preference, and shows a list of the inputs used to evaluate the expression.

If viewing an expression for a binding requirement, the content of the Expression is Boolean and evaluates to true or false. If the expression is minimum or maximum, the Result column shows the actual value determined for the minimum or maximum. The values dynamically update as the inputs to the expression change.

If viewing an expression for a binding preference, the content of the expression depends on the type of expression selected:

<table>
<thead>
<tr>
<th>Bind Preference Types</th>
<th>Expression</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boolean Expression</td>
<td>Expression that must evaluate True.</td>
</tr>
<tr>
<td>Minimize Expression</td>
<td>Expression to be minimized.</td>
</tr>
<tr>
<td>Maximize Expression</td>
<td>Expression to be maximized.</td>
</tr>
</tbody>
</table>

See also

Binding tab on page 52

ProcedureView control recipe properties on page 51

Perform an active step change

Change active steps within a control recipe when the recipe level that requires the Active Step Change is in MANUAL mode. Use the ProcedureView control to toggle one or multiple steps between active and inactive.

Before you begin:

- One step must be active to execute the Active Step Change.
• Steps that are in the HELD state cannot be removed as the active step.

To perform an active step change:

1. Within the ProcedureView control, navigate to the recipe level that contains the active step to be changed.

2. Open the ProcedureView shortcut menu.
   a. If viewing the control recipe in the SFC view, right-click outside of the SFC structure to open the shortcut menu.
   b. If viewing the control recipe in the Table view, right-click the table header to open the shortcut menu.

3. From the shortcut menu, select Manual to place the current recipe level in MANUAL mode.

4. Open the ProcedureView shortcut menu.

5. From the shortcut menu, select Active Step Change. The cursor changes to a pointing hand.

6. Select all steps that are to be active and deselect all steps that are to be inactive when the active step change is completed.
   a. In the SFC view, steps that were active when the active step change was initiated will be highlighted in yellow hatch with no outline. Selecting an active step causes the step to display a blue hatch background, indicating that the step will be deactivated. Selecting an inactive step (white) causes the step to display a green hatch background, indicating that the step will become active when the active step change is executed. If a step to activate is a material step, you must first rebind the step to a material before doing the Active Step Change. Select each material step to activate and then perform a manual bind. (See Perform material binding for more information.)
b. In the Table view, the ProcedureView control does not use hatching to indicate selected and deselected steps. Instead, the initial active step row is highlighted in yellow, selected (active) step rows are highlighted in bright green, and deselected (inactive) step rows are highlighted in blue.

7. Right-click anywhere within the ProcedureView control. The **Active Step Change** menu opens.

8. Select **Commit Changes** from the menu to execute the active step change. The new active step(s) display a READY state and are in the O_AUTO mode. Any subsequent steps in the recipe that have already completed will be reset. This allows the operator the option to place the batch in the AUTOMATIC mode and process the balance of the batch normally.

   **Tip:** If you did not rebind a material step, you will be prompted to rebind the step to a material, and begin the **Active Step Change** procedure again.

9. Right-click each of the active steps to open the **ProcedureView** shortcut menu, and select **Start** to start the step. The selected step will run and return to a COMPLETED state when it is finished.

   **Tip:** If the control recipe is placed in AUTOMATIC mode, the entire procedural level of the control recipe can be started, and the remainder of the control recipe will continue executing normally.

**See also**

**ProcedureView Control** on page 43

### Manual binding

Manual binding is the process of binding a unit, or rebinding a previously bound unit, to a step within a control recipe without being prompted. Manual binding is performed from within the **Procedure as SFC** view or **Procedure as Table** view or from the **ProcedureView** ActiveX control.

For units, the operator can select **First Available, Prompt** or one of a list of units for binding. For material phases, the operator can select **Automatic, Prompt** or any one of a list of container/phase/lot entries.

**Tip:** All other steps within the recipe that are associated with the same unit requirement are bound to the same unit.

The following criteria are required to perform manual binding:

- The step is associated with a unit class in the recipe equipment requirements.
- The step represents the highest recipe level wholly contained within the unit. (For example, one step in the recipe represents a unit procedure that is associated with the Premixer unit class, and another step that is associated...
with the same Premixer unit class is an operation. As long as the balance of
the manual binding requirements are met, then you can manually bind the
unit procedure, but not the operation, because the unit procedure is the
highest recipe level associated with the unit.)

- The step is inactive or has a binding status of BINDING.
- More than one legal binding choice exists for the unit that is associated with
  the step.

See also

Perform manual binding on page 56

**Perform manual binding**

Manual binding is the process of binding a unit, or rebinding a previously bound
unit, to a step within a control recipe without being prompted to do so.

**Tip:** This procedure applies to steps (not batches) and can only be performed at the step level.

**To perform manual binding:**

1. In the **Batch List** view, select the batch that requires binding.
2. Open the **Procedure as SFC** or **Procedure as Table** view.
3. To bind a unit to a step, you must select the unit procedure in either the
   SFC area, Recipe Table area, or the Procedural Hierarchy area.
4. Click **Bind**.
5. Select the appropriate binding option and click **OK**.

See also

Manual binding on page 55

**Bind a unit procedure to a unit**

Bind a unit, or rebind a previously bound unit, to a step within a control recipe
without being prompted.

**To bind a unit procedure to a unit:**

1. Right-click the unit procedure in either the SFC view or the Table view (if
   the control recipe was created from an operation or unit procedure, select
   the parent of the SFC). The **ProcedureView** shortcut menu opens.
2. If the manual binding criteria have been met, the **Bind** menu option is
   enabled. Select **Bind**.
3. Select the appropriate binding option.
   
   a. Selecting **Operator Prompt** results in an operator binding prompt when unit binding is required during recipe execution.
   
   b. Selecting **First Available** causes the unit procedure to automatically bind to the first available unit when unit binding is required during recipe execution.
   
   c. To specify a particular unit for binding, select the **Specific Unit** option and then select the unit from the list.

4. Click **OK** to save changes.

See also

**Perform manual binding** on page 56

---

If the Material Server is unavailable, a warning message is displayed in the **Bind** dialog box. The unit list displays data supplied by the area model in place of the data normally supplied by the Material Server. When the Material Server is not available, first available binding is serviced as though it was prompt binding. Use extreme caution when operating without the Material Server. Failure to select the correct unit or binding method could result in a hazardous situation, depending on the material.

---

**Material binding**

Material binding is the process of binding activated material-enabled phase steps to material phases. Or, the process of rebinding a previously bound material to a step within a control recipe without being prompted. Material phase binding must be performed from within the **Procedure as SFC** or **Procedure as Table** view or the **ProcedureView** ActiveX control.

**Tip:** This procedure applies to steps (not batches) and can only be performed at the Step level.

The following criteria are required to perform manual binding:

- The step is associated with a material in the recipe.
- The step is inactive or has a binding status of BOUND.

See also

**Perform material binding** on page 57
Perform material binding

Bind active material-enabled phase class steps to material phases. Material binding is performed from the Procedure as SFC or Procedure as Table window in the ProcedureView control or from the Batch List view.

Tip: This procedure applies to steps (not batches) and can only be performed at the step level.

To perform material binding:

1. In the control or view, navigate to the recipe level that contains the material step to be changed.

2. To bind a material to a step, you must select the material step. Right-click the material step, and the shortcut menu opens.

3. If the manual binding criteria have been met, the Bind menu option is enabled. Select the Bind menu option.

4. In the Prompt to Bind a Material Phase Step dialog box, select the material specification that determines the lot or storage container to bind the material step.

5. Click OK.

See also

Material binding on page 57
Cross invocation

Cross invocation is the process of passing configured context information to an automation server. This information is relative to the current selection within the ActiveX controls. When cross invocation is enabled and an automation server is identified, the shortcut menu displays the Cross Invocation menu items.

Tip: Cross invocation is enabled by setting two properties in the ActiveX Controls Library: InvocationProgID and ShowCustomMenuItems.

The items in the shortcut menu are formatted using cross invocation strings associated with the equipment resource. These cross invocation strings also indicate the context data to be passed to the automation server.

The FactoryTalk Batch selectable elements for cross invocation from the ActiveX controls are:

<table>
<thead>
<tr>
<th>Control</th>
<th>Selectable Elements</th>
</tr>
</thead>
<tbody>
<tr>
<td>ControlRecipeList</td>
<td>Batch</td>
</tr>
<tr>
<td>ProcedureView</td>
<td>Batch, Recipe Phase, Operation, and Unit Procedure</td>
</tr>
<tr>
<td>PromptsList</td>
<td>Unacknowledged Prompts (not binding requests)</td>
</tr>
</tbody>
</table>

See also

Perform cross invocation on page 23

Perform cross invocation

When you enable cross invocation and identify the automation server, the shortcut menu displays the Cross Invocation menu items. The shortcut menu items are formatted using cross invocation strings associated with the equipment resource.

Tip: Cross invocation is enabled by setting two properties in the ActiveX Controls Library: InvocationProgID and ShowCustomMenuItems.

To perform cross invocation:

1. Right-click the appropriate selection in the ActiveX controls. The configured Cross Invocation menu items display.

2. Select a menu or sub-menu item. The context data is automatically passed to the configured automation server.
Tip: Check with your system administrator as to what context data is configured for each of the shortcut menu items displayed on your system.

See also

Cross invocation on page 22
Chapter 6

TimerView Control

The TimerView ActiveX control displays all the timer steps in batches currently on the batch list that satisfy the filter criteria for the ActiveX control. The behavior of the control is configured through the control’s Property Pages during design time. Use the control to change the setpoint value for COUNT_DOWN timers during run time. If filtering is enabled, change the filtering of the ActiveX control through the Filter menu item.

To use the TimerView ActiveX control:

- Place the control within a web browser or other ActiveX container.
- Place the control on a Visual Basic form, and open it as an executable.

The appearance of the TimerView varies depending on the container in which it is placed, and how it is configured within that container. However, the control functions in the same way regardless of the container in which it is placed.

See also

- TimerView shortcut menu on page 63
- Timer Step dialog box on page 64
- Timer step filters on page 65
- Timer step filter patterns on page 65

TimerView interface

The TimerView ActiveX control is available to any ActiveX container including FactoryTalk View SE. The TimerView ActiveX control lists all the timer steps on the batch list that match the filter criteria. The TimerView control is user sizable, in a table-type format.

- Double-clicking anywhere on a non-header row, except the Setpoint box, opens the Timer Step dialog box for the selected timer step.
- Double-click on the Setpoint box for a COUNT_DOWN timer to open Change Setpoint Value dialog box.
- Double-clicking on the Filter Pattern status bar, located below the Timer Step grid, opens the Timer Step Filter dialog box.
• Enable or disable the status bar at design time.
• Hovering the cursor over any row in the timer list displays that timer step’s recipe path in a tooltip.

For each timer step that satisfies the filter criteria, the ActiveX control displays the following information.

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Step Name</td>
<td>The recipe step name.</td>
</tr>
<tr>
<td>Type</td>
<td>The type of counter – COUNT_UP or COUNT_DOWN.</td>
</tr>
<tr>
<td>Elapsed</td>
<td>Time How much time has elapsed since the Timer started running.</td>
</tr>
<tr>
<td>Remaining Time</td>
<td>How much time is left on the Timer; the setpoint value minus the elapsed time. (COUNT_DOWN timers only)</td>
</tr>
<tr>
<td>Setpoint</td>
<td>The configured period of time that the Timer phase is set to run. (COUNT_DOWN timers only)</td>
</tr>
<tr>
<td>Units of Time</td>
<td>The units of time assigned to a Timer step; can be Seconds, Minutes, Hours, or Days.</td>
</tr>
<tr>
<td>State</td>
<td>The recipe step’s current state.</td>
</tr>
<tr>
<td>Unit</td>
<td>The name of the unit in which the Timer step is running (this value may be blank)</td>
</tr>
</tbody>
</table>

The TimerView ActiveX control updates at least once every five seconds. The update value is configured at design time.

If no timer steps match the given filter criteria, the **No Timer steps match the configured filter criteria** message displays. After a period of time, if one or more timer steps match the configured criteria, then those steps are displayed. If the timer steps go out of scope of the filter and there are no timer steps to display, the message displays.

If the TimerView ActiveX control cannot communicate with the configured Batch Server node, a **Communications Lost** message is shown. If communications are reestablished, the ActiveX control displays the timer steps that satisfy the filter criteria.

See also

- TimerView Control on page 61
- Timer Step dialog box on page 64
- Setpoint values on page 64
- Timer step filters on page 65
The TimerView ActiveX control provides a shortcut menu accessed by right-clicking anywhere within the control. Use the shortcut menu to log on to the FactoryTalk Batch View Security, and to filter timers contained in control recipes that have been added to the Batch List. The menu items are disabled if the menu item is not valid for the selected element.

The TimerView shortcut menu shows the following items:

<table>
<thead>
<tr>
<th>Item</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Filter</td>
<td>Opens the Filter dialog box.</td>
</tr>
<tr>
<td>Login</td>
<td>Allows you to log on to the FactoryTalk Batch View security.</td>
</tr>
</tbody>
</table>

See also

**TimerView Control** on page 61

Log in to an ActiveX control

Log in to verify your permissions to use the functions of the ActiveX control, depending on the defined security settings. If the ActiveX control is invoked in a container which already has an established FactoryTalk User authenticated, then the established user is automatically authenticated.

**Important:** When the ProcedureView ActiveX control connects to a running FactoryTalk Batch Server which is an incompatible version, the control displays the following message:

"The ProcedureView ActiveX control is incompatible with the current Batch Server. It is strongly recommended that you upgrade your system before continuing."

THIS IS AN UNTESTED, UNSUPPORTED STATE. (See Version compatibility with FactoryTalk Batch Server for more information.)

**To log in to an ActiveX control:**

1. Right-click the ActiveX control and select **Login** from the shortcut menu.
2. Enter your user name in the **User Name** box.
3. Enter your password in the **Password** box.
4. Click **OK** to log in.

**Tip:** To log out of the ActiveX control, log off of the FactoryTalk Directory (if single-signon is enabled) or log off Windows.

See also

**ActiveX Controls library introduction** on page 7

**Version compatibility with FactoryTalk Batch Server** on page 11
The **Timer Step** dialog box is shown at run time when you double-click anywhere in a timer row, except the **Setpoint**. The **Timer Step** dialog box shows information for the selected timer.

The name of the timer displays in the header. The **State**, **Type**, **Hold Behavior**, **Unit**, **BatchID**, **Units of Time**, **Elapsed Time**, **Remaining Time** (COUNT_DOWN Timers only), and **Recipe Path** are displayed (view-only).

If the timer is a COUNT_DOWN Timer, the **Setpoint** box is enabled; double-click in the box to open the **Change Setpoint Value** dialog box.

**See also**

[TimerView Control on page 61](#)

[Setpoint values on page 64](#)

### Setpoint values

The **Change Setpoint Value** dialog box can be displayed two ways from the TimerView ActiveX control. For COUNT_DOWN Timers, double-click the **Setpoint** box from the table view, or double-click the Setpoint text box on the **Timer Step** dialog box.

Use the **Change Setpoint Value** dialog box to see the Minimum and Maximum values (view-only) for the Setpoint parameter and enter a new value in the **Value** box.

Once a new value is entered, click **Apply** to immediately update the timer’s setpoint value or **Recalculate** to recalculate the estimated remaining time, then click OK.

**Tip:** The value entered must be within the displayed minimum and maximum values for the parameter. If the value is not between the minimum and maximum values when the user clicks **OK**, an error message requests **Please enter a value between the configured Minimum [Minimum Value] and Maximum [Maximum Value] values**. Upon clicking **OK** for the error message, enter a new value.

**See also**

[TimerView interface on page 61](#)
Timer step filters

Timer step filters are preconfigured on the Appearance tab of the Property Pages dialog box at design time, or in the Timer Step Filter dialog box, which is accessed from the Filter menu at run time. The Filter menu option is enabled at design time.

Tip: You should select only one filter, but if more than one filter is selected, the control filters in the following order: FilterOnAreaModel (Unit Name), FilterOnBatchID (Batch ID), and then FilterOnBatchList (Batch Recipe Path).

The Timer Step Filter dialog box consists of a tree control to apply a filter to any one of three distinct areas. Filters cannot be combined with any other “filter by” value.

- **Unit Name:** Filters timer steps based on the unit in which they are running. This filter is applied to the value appearing in the Unit column.
- **Batch ID:** Filters timer steps based on the ID of the batch. This filter is applied to only the Batch ID of a timer step.
- **Batch Recipe Path:** Filters timer steps based on the batch’s Recipe Path. This filter is applied to the recipe path of a timer step.

**Important:** It is important that the filter pattern is correct for the “Filter By” option. A filter mismatch occurs if the incorrect “Filter By” option is set using the wrong filter pattern.

As items are selected from the tree, the filter pattern is built and displayed in the Filter Pattern box at the bottom of the dialog box. Edit the text in this box to further refine the filter criteria.

Tip: The run-time setting of the filter is only valid for the current running session of the control. If a filter pattern is set at run-time, the filter value does not persist and the default filter value set in configuration mode is used the next time the ActiveX control is initiated.

See also

- Timer step filter patterns on page 65
- TimerView Control on page 61

Timer step filter patterns

Timer step filter patterns are strings. A group of one or more characters (charlist) enclosed in brackets ([ ]) is used to match any single character in the string and includes almost any character code, including digits.

<table>
<thead>
<tr>
<th>Characters in pattern</th>
<th>Matches in string</th>
</tr>
</thead>
<tbody>
<tr>
<td>?</td>
<td>Any single character.</td>
</tr>
<tr>
<td>*</td>
<td>Zero or more characters.</td>
</tr>
</tbody>
</table>
Characters in pattern | Matches in string
---|---
# | Any single digit (0-9).
[charlist] | Any single character in the charlist
[!charlist] | Any single character not in the charlist

To match the special characters left bracket ([), question mark (?), number sign (#), and asterisk (*), enclose them in brackets. The right bracket (]) cannot be used within a group to match itself, but it can be used outside a group as an individual character. By using a hyphen (-) to separate the upper and lower bounds of the range, charlist can specify a range of characters.

For example, [A-Z] results in a match if the corresponding character position in string contains any uppercase letters in the range A-Z. Multiple ranges are included within the brackets without delimiters. The meaning of a specified range depends on the character ordering valid at run time.

An exclamation point (!) at the beginning of charlist means that a match is made if any character except the characters in charlist is found in the string. When used outside brackets, the exclamation point matches itself. A hyphen (-) can appear either at the beginning (after an exclamation point if one is used), or at the end of charlist to match itself. In any other location, the hyphen is used to identify a range of characters.

When a range of characters is specified, they must appear in ascending sort order (from lowest to highest). [A-Z] is a valid pattern, but [Z-A] is not. The character sequence [] is considered a zero-length string ("").

See also

TimerView Control on page 61

Timer step filters on page 65
Legal Notices

Copyright notice

© 2017 Rockwell Automation Technologies, Inc. All rights reserved. Printed in USA.

This document and any accompanying Rockwell Software products are copyrighted by Rockwell Automation Technologies, Inc. Any reproduction and/or distribution without prior written consent from Rockwell Automation Technologies, Inc. is strictly prohibited. Please refer to the license agreement for details.

End User License Agreement (EULA)

You can view the Rockwell Automation End-User License Agreement ("EULA") by opening the License.rtf file located in your product's install folder on your hard drive.

Trademark Notices

Allen-Bradley, CompactLogix, ControlLogix, Connected Components Workbench, eProcedure, FactoryTalk, GuardLogix, GuardPLC, Micro800, MicroLogix, MobileView, MobileView Guard, PanelBuilder, PanelView, PhaseManager, PlantLink, PlantPax, PLC-2, PLC-3, PLC-5, PowerMonitor, RSBizWare, RSBizWare BatchHistorian, RSBizWare Historian, RSBizWare MaterialTrack, RSLinx, Rockwell Automation, Rockwell Software, SLC, SoftLogix, Studio 5000, and XWare trademarks of Rockwell Automation, Inc.

Any Rockwell Automation software or hardware not mentioned here is also a trademark, registered or otherwise, of Rockwell Automation, Inc.

For a complete list of products and their respective trademarks, go to: http://www.rockwellautomation.com/rockwellautomation/legal-notices/overview.page?tab4

Warranty

This product is warranted in accordance with the product license. The product's performance may be affected by system configuration, the application being performed, operator control, maintenance, and other related factors. Rockwell Automation is not responsible for these intervening factors. The instructions in this document do not cover all the details or variations in the equipment, procedure, or process described, nor do they provide directions for meeting every possible contingency during installation, operation, or maintenance. This product’s implementation may vary among users.
This document is current as of the time of release of the product; however, the accompanying software may have changed since the release. Rockwell Automation, Inc. reserves the right to change any information contained in this document or the software at any time without prior notice. It is your responsibility to obtain the most current information available from Rockwell when installing or using this product.

Environmental compliance


Contact Rockwell Automation

Customer Support Telephone — 1.440.646.3434

Online Support — http://www.rockwellautomation.com/support/
Index

A
abort command 43
active step change 43, 51
command 43
perform 51
ActiveX
components 8
overview 7
ActiveX controls
using 9
add a control recipe 18
area 29
auto command 15, 43

B
batch creation errors 19
Batch ID 13, 29
batch scale 18
Batch Server
incompatible 11
bind command 43
binding
preferences 50
requirements 50

C
change
timer setpoint 62
clear failures command 15, 43
command 43
commands
timer step 48
comment command 43
control recipe
add 18
command 18
filter 18
ControlRecipeList
add a control recipe 18
menu 14
remove a control recipe 21

D
DCOM
distributed communications 8
security 8
default 29
distributed communications 8

E
edit
parameter values 49
EU (Engineering Units) 29
equation values 50

F
filter 62
clock recipe 18
filter patterns
timer step 63

H
hold command 15, 43

I
incompatible
Batch Server 11
install ActiveX components 8

L
Login command 43

M
manual command 43
menu
ControlRecipeList shortcut 14
ProcedureView shortcut 43
PromptsList shortcut 30

Rockwell Automation Publication FactoryTalk Batch ActiveX Controls User Guide - February 2017 69
Index

N
no legal units 19

P
parameter values
  edit 49
ProcedureView
  active step change 43, 51
  menu 43
  SFC view 43
  Table view 42
  view step properties 48
PromptsList
  menu 30
  properties 43
Properties dialog box 48
  Binding tab 50
  Parameter tab 48
  Recipe tab 49
  Report tab 49

R
redistribute files 8
remove a control recipe 21
restart command 15, 43
resume command 43

S
scale 18
security 8
semi-auto command 43
setpoint
  change timer 62
SFC view 43
SignatureList
  interface 35
Source Step 29
start command 15, 43
stop command 15, 43

T
table view 42

display items 42
time 29
Timer steps
  change a setpoint 62
  commands 48
  filter patterns 63
Timer-Complete command 43
Timer-Reset command 43
TimerView
  filter patterns 63
  filters 62
  window 59
type 29

U
using ActiveX controls 9

V
version checking 11
view
  binding preferences 50
  binding requirements 50
  expression values 50
Rockwell Automation support

Rockwell Automation provides technical information on the web to assist you in using its products. At [http://www.rockwellautomation.com/support](http://www.rockwellautomation.com/support) you can find technical and application notes, sample code, and links to software service packs. You can also visit our Support Center at [https://rockwellautomation.custhelp.com](https://rockwellautomation.custhelp.com) for software updates, support chats and forums, technical information, FAQs, and to sign up for product notification updates.

In addition, we offer multiple support programs for installation, configuration, and troubleshooting. For more information, contact your local distributor or Rockwell Automation representative, or visit [http://www.rockwellautomation.com/services/online-phone](http://www.rockwellautomation.com/services/online-phone).

Installation assistance

If you experience a problem within the first 24 hours of installation, review the information that is contained in this manual. You can contact Customer Support for initial help in getting your product up and running.

<table>
<thead>
<tr>
<th>United States or Canada</th>
<th>1.440.646.3434</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outside United States or Canada</td>
<td>Use the Worldwide Locator available at <a href="http://www.rockwellautomation.com/locations">http://www.rockwellautomation.com/locations</a>, or contact your local Rockwell Automation representative.</td>
</tr>
</tbody>
</table>

New product satisfaction return

Rockwell Automation tests all of its products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning and needs to be returned, follow these procedures.

<table>
<thead>
<tr>
<th>United States</th>
<th>Contact your distributor. You must provide a Customer Support case number (call the phone number above to obtain one) to your distributor to complete the return process.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outside United States</td>
<td>Please contact your local Rockwell Automation representative for the return procedure.</td>
</tr>
</tbody>
</table>

Documentation feedback

Your comments will help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete the feedback form, publication **RA-DU002**.

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

[www.rockwellautomation.com](http://www.rockwellautomation.com)
Rockwell Automation support

Rockwell Automation provides technical information on the web to assist you in using its products. At [http://www.rockwellautomation.com/support](http://www.rockwellautomation.com/support) you can find technical and application notes, sample code, and links to software service packs. You can also visit our Support Center at [https://rockwellautomation.custhelp.com](https://rockwellautomation.custhelp.com) for software updates, support chats and forums, technical information, FAQs, and to sign up for product notification updates.

In addition, we offer multiple support programs for installation, configuration, and troubleshooting. For more information, contact your local distributor or Rockwell Automation representative, or visit [http://www.rockwellautomation.com/services/online-phone](http://www.rockwellautomation.com/services/online-phone).

Installation assistance

If you experience a problem within the first 24 hours of installation, review the information that is contained in this manual. You can contact Customer Support for initial help in getting your product up and running.

<table>
<thead>
<tr>
<th>United States or Canada</th>
<th>1.440.646.3434</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outside United States or Canada</td>
<td>Use the Worldwide Locator available at <a href="http://www.rockwellautomation.com/locations">http://www.rockwellautomation.com/locations</a>, or contact your local Rockwell Automation representative.</td>
</tr>
</tbody>
</table>

New product satisfaction return

Rockwell Automation tests all of its products to ensure that they are fully operational when shipped from the manufacturing facility. However, if your product is not functioning and needs to be returned, follow these procedures.

<table>
<thead>
<tr>
<th>United States</th>
<th>Contact your distributor. You must provide a Customer Support case number (call the phone number above to obtain one) to your distributor to complete the return process.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Outside United States</td>
<td>Please contact your local Rockwell Automation representative for the return procedure.</td>
</tr>
</tbody>
</table>

Documentation feedback

Your comments will help us serve your documentation needs better. If you have any suggestions on how to improve this document, complete the feedback form, publication RA-DU002 [http://literature.rockwellautomation.com/idc/groups/literature/documents/du/ra-du002_-en-e.pdf](http://literature.rockwellautomation.com/idc/groups/literature/documents/du/ra-du002_-en-e.pdf).

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

[www.rockwellautomation.com](http://www.rockwellautomation.com)