



# FactoryTalk Transaction Manager User Guide

Rockwell Automation Publication FTTM-UM001A-EN-E-July 2021  
Supersedes Publication RSSQL-UM001L-EN-E-September 2020



## Important User Information

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

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

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

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

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

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

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

Throughout this manual, when necessary, we use notes to make you aware of safety considerations.



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

---



**ATTENTION:** Identifies information about practices or circumstances that can lead to personal injury or death, property damage, or economic loss. Attentions help you identify a hazard, avoid a hazard, and recognize the consequence.

---

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

---

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



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

---



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

---



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

---

<b>Welcome To FactoryTalk Transaction Manager</b>	<p><b>Chapter 1</b></p> <p>What Is FactoryTalk Transaction Manager? ..... 9</p> <p>What Can FactoryTalk Transaction Manager Do For Me? ..... 9</p> <p>Understanding FactoryTalk Transaction Manager Concepts .....10</p> <p style="padding-left: 20px;">Transaction Control Manager Service .....10</p> <p style="padding-left: 20px;">FactoryTalk Transaction Manager Service and Control Connectors11</p> <p style="padding-left: 20px;">Control Connectors..... 11</p> <p style="padding-left: 20px;">Enterprise Connectors.....12</p> <p style="padding-left: 20px;">Configuration Server .....12</p> <p style="padding-left: 20px;">Transactions .....12</p> <p>FactoryTalk .....12</p> <p style="padding-left: 20px;">FactoryTalk Services Platform Components..... 13</p> <p>Intended Audience ..... 13</p> <p>Where Can I Go for Help?..... 14</p>
<b>Installing FactoryTalk Transaction Manager</b>	<p><b>Chapter 2</b></p> <p>Hardware and software requirements..... 15</p> <p>Distributed Installations ..... 15</p> <p>Installing FactoryTalk Transaction Manager ..... 15</p> <p style="padding-left: 20px;">Install FactoryTalk Services..... 15</p> <p style="padding-left: 20px;">Install FactoryTalk Transaction Manager .....16</p> <p style="padding-left: 20px;">Activate FactoryTalk Transaction Manager ..... 17</p>
<b>Migrate and Upgrade FactoryTalk Transaction Manager</b>	<p><b>Chapter 3</b></p> <p>Migrate FactoryTalk Transaction Manager from version 10.20 to 13.10 ..... 20</p> <p style="padding-left: 20px;">Step 1: Back up your FactoryTalk Administration Console configuration..... 20</p> <p style="padding-left: 20px;">Step 2: Install FactoryTalk Transaction Manager.....21</p> <p style="padding-left: 20px;">Step 3: Copy the cfgs70.dat filer .....21</p> <p style="padding-left: 20px;">Step 4: Copy the configuration files and folder structure..... 22</p> <p style="padding-left: 20px;">Step 5: Convert the FactoryTalk Transaction Manager configuration ..... 22</p> <p>Migrate FactoryTalk Transaction Manager from version 12.00, 12.10 or 13.00 to 13.10 ..... 24</p> <p style="padding-left: 20px;">Step 1: Back up your FactoryTalk Administration Console configuration..... 25</p> <p style="padding-left: 20px;">Step 2: Back up your FactoryTalk Transaction Manager configuration..... 26</p> <p style="padding-left: 20px;">Step 3: Install FactoryTalk Transaction Manager ..... 26</p>

Step 4: Restore your FactoryTalk Transaction Manager configuration..... 26

Finish the Migration ..... 26

Step 1: Update your FactoryTalk Transaction Manager configuration..... 26

Step 2: Restore your FactoryTalk Administration Console configuration..... 29

Step 3: Configure your database connections..... 29

Upgrade FactoryTalk Transaction Manager on the same computer..... 30

Step 1: Remove FactoryTalk Transaction Manager ..... 30

Step 2: Install the new version of FactoryTalk Transaction Manager ..... 30

Step 3: Convert the FactoryTalk Transaction Manager configuration ..... 30

**Chapter 4**

**Exploring the FactoryTalk Transaction Manager User Interface**

Starting FactoryTalk Transaction Manager.....33

Exploring the FactoryTalk Transaction Manager User Interface .....33

Title Bar.....34

Menu Bar .....34

Toolbar .....35

Configuration Tree .....36

    Configuration and Connector Status.....36

    Configuration Server Status .....36

Workspace .....37

Transaction Definition View.....37

Transaction Monitor View ..... 38

Error Log Files View..... 38

FactoryTalk Transaction Manager system graphic ..... 41

Status Bar.....41

Configuration Checklist .....41

    Step 1: Defining and Naming a New Configuration ..... 42

    Step 2: Defining Connectors .....43

    Step 3: Defining Data Points .....43

    Step 4: Defining Data Objects ..... 44

    Step 5: Defining Transactions ..... 44

    Step 6: Verifying Transactions .....45

Miscellaneous .....45

    Viewing Configuration Properties .....45

    Starting Configurations ..... 46

    Stopping Configurations.....47

    Starting and Stopping Connectors.....47

Monitoring Configurations.....	47
Understanding FactoryTalk Transaction Manager External Files .....	47

## Chapter 5

### Understanding FactoryTalk Transaction Manager Services

Introducing FactoryTalk Transaction Manager Services .....	49
Control Connectors .....	49
FactoryTalk Live Data .....	49
Generic OPC .....	50
Enterprise Database Connectors .....	50
Microsoft OLE DB .....	50
ODBC .....	50
Oracle OCI .....	51
Enterprise Application Connectors .....	51
Time-Series Data Compression .....	51
FactoryTalk Metrics Enterprise Application Connector .....	51
Enterprise Connector Options .....	51
FactoryTalk Transaction Manager Service.....	52
Transaction Control Manager Service.....	52
Configuration Server .....	52

## Chapter 6

### Defining Data Points

Introducing Data Points .....	55
FactoryTalk Live Data Data Points .....	56
Selecting a Collection Mode .....	57
Scheduled: Maintain the Current Subscribed Value.....	57
Device Scheduled: Request the Current Value From the Device	57
Unscheduled: Send Subscribed Value Whenever It Changes ....	57
Consecutive Data Point and Data Block Support.....	58
Selecting Timeout Properties.....	59
Data Valid .....	59
Data Retrieval Timeout .....	59
Selecting a Substitution Option.....	60
Preventing Stale and Mismatched Data.....	60
Specifying Quality.....	61
OPC Data Points .....	61
Generic OPC Data Points .....	62

## Chapter 7

### Defining Data Objects

Introducing Data Objects.....	63
Enterprise Database Objects.....	64
Oracle Call Interface (OCI) Data Objects .....	64
Microsoft SQL Server Data Objects .....	64

ODBC Data Objects ..... 64

Enterprise Application Objects ..... 64

    FactoryTalk Metrics Data Objects ..... 65

Enterprise Connector Error Handling ..... 65

Inserting and Updating Data Table Records ..... 66

Stored Procedures ..... 66

**Chapter 8**

**Creating Transactions**

Introducing Transactions ..... 67

Transaction Types ..... 68

    Unidirectional Transactions ..... 68

    Bidirectional Transactions ..... 69

    Bidirectional or Unidirectional Transactions With Transaction Bindings ..... 70

Transaction Timeout ..... 71

Transaction Completion ..... 72

    Cached Transactions ..... 72

    Real-time Transactions ..... 72

    Transactions With Bound Transaction Results ..... 73

    Database Triggers ..... 73

Using the Expression Editor ..... 73

    Logical and Mathematical Operators ..... 74

    Time Functions ..... 74

    Data Point Range and Advanced Functions ..... 74

    Parse Function ..... 75

Transaction Trigger and Storage Options ..... 75

**Chapter 9**

**Understanding Online Edits**

Introducing Online Edits ..... 79

Understanding Online Edit Concepts ..... 79

    Online Edits Workflow ..... 80

    Configuration That Uses Online Edits ..... 80

    Assembling Pending Edits ..... 81

    Canceling Pending Edits ..... 82

    Pending Edit Alerts ..... 82

        Occurrence Conditions of Pending Edit Alerts ..... 82

Creating a Configuration That Uses Online Edits ..... 83

    Editing Data Points In a Running Configuration That Uses Online Edits ..... 83

        Adding New Data Points ..... 84

        Editing Existing Data Points ..... 84

        Saving Data Point Pending Edits ..... 84

Assembling Data Point Pending Edits .....	84
Canceling Data Point Pending Edits.....	85
Editing Transactions In a Running Configuration That Uses Online Edits .....	85
Saving Transaction Pending Edits .....	86
Assembling Transaction Pending Edits .....	86
Viewing Transaction Differences.....	86
Canceling Transaction Pending Edits.....	87

## Chapter 10

### Advanced Topics

Remote User Interface.....	89
Configuring the Remote User Interface .....	89
Distributed Configurations.....	90
Licensing Required For a Distributed Configuration .....	90
Establishing Microsoft Windows Privileges .....	90
Creating a Distributed Configuration.....	91
Using UNC Paths .....	91
Changing the Transaction Cache File Path.....	91
Changing the Error Log File Path.....	91
Data Point Buffering.....	92
Buffering In the Controller .....	92
Buffering In the FactoryTalk Transaction Manager Control Connector .....	92
Buffering In the Transaction Control Manager Service.....	92
Buffering In the FactoryTalk Transaction Manager Service.....	93
Buffering In Cached Transaction Files .....	93
Increasing Performance .....	93
Control System.....	93
Database .....	94
FactoryTalk Transaction Manager.....	94
Hardware and Operating Environment.....	95
Windows Performance Counters.....	95

## Chapter 11

### FactoryTalk Transaction Manager Sample Applications

External Trigger Sample Application.....	99
Application Contents.....	99
Running the Sample Application.....	99

## Chapter 12

**Securing FactoryTalk  
Transaction Manager Using  
FactoryTalk Security**

Considerations When Using FactoryTalk Transaction Manager With  
FactoryTalk Security .....103  
Using FactoryTalk Transaction Manager With Single Sign-on (SSO) 105  
FactoryTalk Security Permissions To Perform FactoryTalk Transaction  
Manager Tasks.....106  
Writing Product-Specific Security Privileges From a Previous Release  
To a File .....107  
Mapping Old Product-Specific Security Privileges To New FactoryTalk  
Security Permissions.....108

**Glossary**

**Glossary**

A - D ..... 111  
E - O .....113  
P - U ..... 116

**Legal Notices**

**Index**



# Welcome To FactoryTalk Transaction Manager

In this chapter you will learn about the following:

- [What Is FactoryTalk Transaction Manager?](#) on [page 9](#)
- [What Can FactoryTalk Transaction Manager Do for Me?](#) on [page 9](#)
- [Understanding FactoryTalk Transaction Manager Concepts](#) on [page 10](#)
- [FactoryTalk](#) on [page 12](#)
- [Intended Audience](#) on [page 13](#)
- [Where Can I Go for Help?](#) on [page 14](#)

## What Is FactoryTalk Transaction Manager?

FactoryTalk Transaction Manager (previously known as RSSql) is an industrial transaction software engine that shares data between your shop floor systems and your enterprise applications (for example, corporate databases). FactoryTalk Transaction Manager can interact with the following shop floor systems:

- Human Machine Interfaces (HMI).
- Programmable Logic Controllers (PLC).
- ControlLogix Controllers.
- Distributed Control Systems (DCS).

FactoryTalk Transaction Manager can interact with the following database systems:

- Microsoft SQL Server 2012 Standard SP4 64-bit
- Microsoft SQL Server 2014 Standard SP3 64-bit
- Microsoft SQL Server 2016 Standard SP2 64-bit
- Microsoft SQL Server 2016 Enterprise SP2 64-bit
- Microsoft SQL Server 2017 Standard SP2 64-bit
- Microsoft SQL Server 2019 Standard 64-bit
- Microsoft SQL Server 2019 Enterprise 64-bit
- Oracle 11g
- Oracle 12c

## What Can FactoryTalk Transaction Manager Do For Me?

FactoryTalk Transaction Manager helps you manage your manufacturing processes by integrating the data in your control systems with enterprise applications.

The following are examples of FactoryTalk Transaction Manager applications:

- Automate data logging

FactoryTalk Transaction Manager can move large amounts of data in a fast and robust manner. In addition, the software has built-in fault tolerance and the ability to optimize reading and writing of both

control and enterprise data. You can use the software to automate the following types of processes:

- Monitoring performance of control systems such as machine usage.
- Tracking product information such as Work in Progress status and raw material availability.
- Updating real-time process information such as temperature, pressure, and alarm states.
- Control the plant floor using business rules and quality

FactoryTalk Transaction Manager provides the interface for a repository of business rules. A business rule can be any logic required to run your plant such as product specifications or quality parameters. By placing business rules in a database server in a central location, the rules are easier to manage within an enterprise system. Additionally, the software can assure quality data to meet the requirements of today's advanced manufacturing companies.

- Manage recipes

FactoryTalk Transaction Manager moves data (for example, recipe information) from a database to an HMI or control system.

## Understanding FactoryTalk Transaction Manager Concepts

FactoryTalk Transaction Manager consists of several design-time and run-time components. This guide describes the following components in detail:

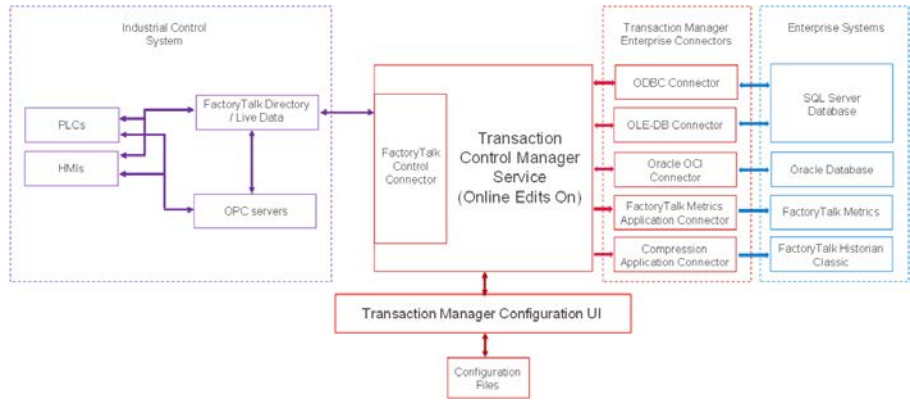
- Transaction Control Manager service.
- FactoryTalk Transaction Manager service.
- Control connectors.
- Enterprise connectors.
- Configuration Server.
- Transactions.

## Transaction Control Manager Service

The Transaction Control Manager is a service that controls and executes FactoryTalk Transaction Manager transactions contained in a configuration, but with the additional functionality of the FactoryTalk Live Data control connector embedded in it. In a configuration enabled for editing, the Transaction Control Manager replaces the separate FactoryTalk Transaction Manager and control connector services.

The Transaction Control Manager service can connect to Rockwell Software products and all OPC servers; therefore, the use of this service is the preferred method for all new FactoryTalk Transaction Manager configurations. For

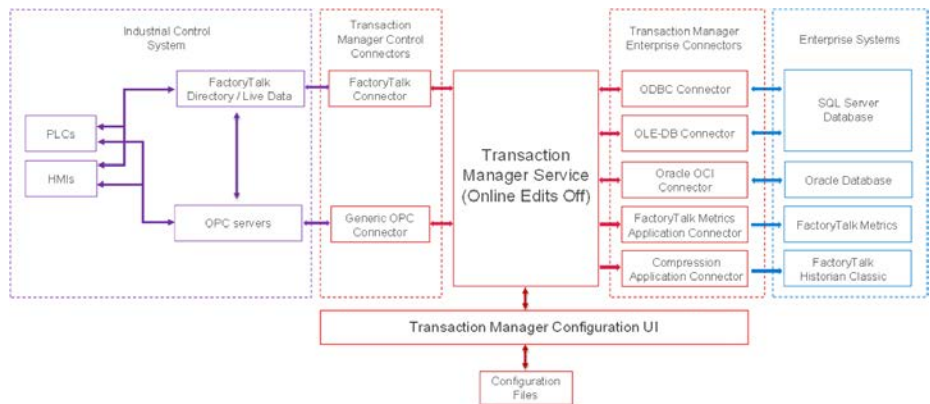
more information on how the Transaction Control Manager service functions, see the following figure.



In a configuration that uses online edits, the Transaction Control Manager service performs the duties of the FactoryTalk Transaction Manager service. For more information on online edits, see [Understanding Online Edits](#) on page 79.

## FactoryTalk Transaction Manager Service and Control Connectors

The FactoryTalk Transaction Manager service is used to control and execute FactoryTalk Transaction Manager transactions contained in configurations that do not require online edits, were created prior to the existence of the newer Transaction Control Manager service, or there is some other reason to not run the Transaction Control Manager service. For more information on how the FactoryTalk Transaction Manager functions, see the following figure.



## Control Connectors

The FactoryTalk Transaction Manager service communicates with the industrial control system device via a control connector. A control connector is a Microsoft Windows service that collects data from a controller and sends it to the FactoryTalk Transaction Manager service in the FactoryTalk Transaction Manager. You can use the following types of control connectors:

- FactoryTalk Live Data
- Generic OPC

Control connectors can be used to reference data points or memory locations within your control or shop floor system.

## Enterprise Connectors

The Transaction Control Manager service and the FactoryTalk Transaction Manager service communicate with enterprise systems, such as databases, via an enterprise connector service. An enterprise connector is a Microsoft Windows service that transfers data between the Transaction Control Manager service or the FactoryTalk Transaction Manager service and a database. You can use the following types of enterprise connectors:

- Open Database Connectivity (ODBC).
- Oracle Call Interface (OCI).
- Microsoft OLE DB.
- Time-series Data Compression.
- FactoryTalk Metrics connector.

Enterprise connectors can be used to create data objects that reference elements in your enterprise system, such as tables or stored procedures.

## Configuration Server

The Configuration Server is a service that runs continuously to provide a single interface to the configuration files (with the **.dat** file extension) which make up the FactoryTalk Transaction Manager configuration. The Configuration Server simplifies access to the configuration files by filtering all changes to the files and by communicating with other FactoryTalk Transaction Manager services. A collection of all changes that affect a configuration is recorded in an audit trail (via either FactoryTalk Diagnostics or the Configuration Server **\*.log** file).

## Transactions

FactoryTalk Transaction Manager creates transactions, or discrete operations that transfer data from the control system to or from the enterprise system. For example, a single transaction can download data from an Oracle database via a stored procedure, to tags in a ControlLogix processor. Alternatively, a transaction can send multiple data points from a distributed control system to a Microsoft SQL Server database to be logged for reporting. Transactions can be modified at runtime. For more information, see [Understanding Online Edits](#) on [page 79](#).

The transaction model organizes the task of data management. FactoryTalk Transaction Manager’s flexibility provides many options for customizing all aspects of a transaction. The software can manage many transactions at once, allowing for sophisticated manufacturing data collection and control applications. By using the software, you can also monitor, modify, and enable or disable individual transactions, making the development and implementation of an application easier.

## FactoryTalk

FactoryTalk is a manufacturing information platform that integrates plant-wide control systems and connects the enterprise with the production facility.

Item	Description
Integrate	FactoryTalk eliminates both functionality gaps and overlaps by providing common services (such as diagnostics and access to real-time data) and by sharing plant resources (such as tags and graphic displays) throughout a production facility.

Item	Description
Communicate	FactoryTalk transforms plant-floor data into useful information, and delivers it to the people who need it, e.g. maintenance engineers or enterprise planners.
Collaborate	FactoryTalk allows defining plant-floor resources once, and then allows simultaneous access to those resources across system boundaries.

## FactoryTalk Services Platform Components

With each coordinated release, additional Rockwell Software products build on the FactoryTalk platform and integrate more of the FactoryTalk components. All of the FactoryTalk components are installed together as a platform, integrated into each FactoryTalk-enabled product installation process:

Item	Description
FactoryTalk Directory	FactoryTalk-enabled products use the FactoryTalk Directory to share a common address book, which automatically finds and provides access to plant-floor resources, such as data tags and graphic displays. Unlike a single database, FactoryTalk Directory provides searchable references to resources stored anywhere across an automation system, offering the benefits of central data storage without the risk of a single point of failure. Changes made to the automation system automatically update across all participants in a FactoryTalk-enabled application.
FactoryTalk Live Data	FactoryTalk Live Data manages connections between FactoryTalk-enabled products and data servers. It notifies clients when a connection is lost, automatically reconnects clients, and combines data from multiple controllers and servers into a single group with a single data server connection. This results in faster real-time data transfer and more reliable, efficient connections to data servers. It also assists in redundancy support for data servers by automatically handling detection and failovers for all FactoryTalk-enabled products.
FactoryTalk Administration Console	The FactoryTalk Administration Console is a stand-alone tool that allows you to configure and manage FactoryTalk-enabled applications.
FactoryTalk Audit and FactoryTalk Diagnostics	FactoryTalk Audit and FactoryTalk Diagnostics provides the ability to log errors, warnings, and other status messages generated throughout a FactoryTalk-enabled system to either local logs or a central location.
FactoryTalk Security	FactoryTalk Security is intended to improve the security of your automation system by limiting access to the users with a legitimate need. FactoryTalk Security authenticates user identities and authorizes user requests to access a FactoryTalk-enabled system. These security services are fully integrated with the FactoryTalk Directory and are included as part of the FactoryTalk Services Platform that is installed with many Rockwell Software products.  FactoryTalk Security replaces the product-specific privilege configuration that was available in previous releases of FactoryTalk Transaction Manager. For more information about using FactoryTalk Transaction Manager with FactoryTalk Security, refer to <a href="#">Securing FactoryTalk Transaction Manager using FactoryTalk Security</a> .

## Intended Audience

For this guide it is assumed that you are a control engineer or a database administrator, and that you are familiar with the following:

- Using personal computers.
- Microsoft Windows operating systems.
- OPC servers.

- Configuration of database connections such as ODBC, OCI, or Microsoft OLE DB.
- FactoryTalk Linx
- FactoryTalk View SE.

## Where Can I Go for Help?

Consult the following resources for additional information about the product:

- Release Notes  
The release notes contain current information about the product, including hardware and software requirements, new features, known and fixed anomalies.
- RSBizWare Administration Guide  
The administration guide helps the RSBizWare administrator install and configure the software as well as understand the architecture of the RSBizWare suite and its components.  
This manual is supplied only if you have purchased another RSBizWare product that uses FactoryTalk Transaction Manager, such as FactoryTalk Metrics or FactoryTalk Historian Classic.
- Online help  
The online help provides general information and step-by-step procedures for working with the product.
- Rockwell Automation Support Center  
The support center provides a variety of services, such as trainings, webinars, and online support that will improve your experience using the FactoryTalk Transaction Manager.

## Installing FactoryTalk Transaction Manager

FactoryTalk Transaction Manager is either distributed as a part of the RSBizWare suite or as a separate product. If you have FactoryTalk Transaction Manager as a part of the RSBizWare suite, then it is installed as part of that suite's installation. If you have FactoryTalk Transaction Manager as a standalone product, then follow the steps in this chapter to install it.

In this chapter you will learn about the following:

- [Hardware and software requirements](#) on [page 15](#)
- [Distributed Installations](#) on [page 15](#)
- [Installing FactoryTalk Transaction Manager](#) on [page 15](#)

### Hardware and software requirements

For details on hardware and software requirements, refer to the *FactoryTalk Transaction Manager Release Notes* and *RSBizWare Release Notes* as applicable.

### Distributed Installations

You may want to run FactoryTalk Transaction Manager or its services on more than one computer. To run the software in a distributed mode, it must be installed on all computers that are referenced in a configuration. For more information, see [Distributed Configurations](#) on [page 90](#).

### Installing FactoryTalk Transaction Manager

When installing FactoryTalk Transaction Manager, you will perform the following steps:

- [Install FactoryTalk Services](#) on [page 15](#)
- [Install FactoryTalk Transaction Manager](#) on [page 16](#)
- [Activate FactoryTalk Transaction Manager](#) on [page 17](#)

To begin the installation, run the FactoryTalk Transaction Manager installation media and select **Required Steps** in the **Welcome** screen.

### Install FactoryTalk Services

FactoryTalk Transaction Manager requires the installation of FactoryTalk Services Platform. It can work with any of the compatible versions of FactoryTalk Services Platform and does not require the installation of the version contained on the FactoryTalk Transaction Manager installation media. Refer to the Release Notes for compatible versions of FactoryTalk Services Platform.

When you install FactoryTalk Services, the following components will be installed:

- FactoryTalk Services Platform with FactoryTalk Directory

FactoryTalk Services Platform is an underlying architecture and a set of common services (such as diagnostic messages, health monitoring services, access to real-time data, and shared plant resources such as tags and graphic displays) that Rockwell Automation products build upon. It is a prerequisite for all FactoryTalk-enabled software products.

- FactoryTalk Activation Manager

FactoryTalk Activation Manager allows you to download activation files using an Internet connection, and transfer the activation files to a computer that does not have an Internet connection. Install this software on the same computer as the FactoryTalk Directory server.

- FactoryTalk Linx

FactoryTalk Linx is a FactoryTalk Live Data server and a device-based alarm and event server. FactoryTalk Linx links Allen-Bradley networks and devices to Microsoft Windows products such as the FactoryTalk View SE (HMI software) and the RSLogix family of device programming software.

- FactoryTalk Alarms and Events (optional)

FactoryTalk Alarms and Events provide a common, consistent view of alarms and events throughout a FactoryTalk system.

For more information on FactoryTalk, refer to *FactoryTalk Help*.

### To Install FactoryTalk Services:

1. Click **Install FactoryTalk Services**.

The installation wizard appears.

2. In the Setup screen, click **Install now**.

3. In the End User License Agreements screen, click **Accept all**.

A screen displays that tracks the progress of the installation.

4. Once the installation is complete, click either **Restart now** or **Restart later** as appropriate for your installation.



Tip: You may omit restarting the computer after the FactoryTalk Services installation is complete. For details on configuring the FactoryTalk Services, see *FactoryTalk Help*.

## Install FactoryTalk Transaction Manager

### To install FactoryTalk Transaction Manager:

1. Click **Install FactoryTalk Transaction Manager**.

The installation wizard appears.

2. Click **Next**.



3. On the **License Agreement** page, accept the terms of the agreement and click **Next**.
4. On the **Customer Information** page, enter your **User Name**, **Organization**, and **Serial Number**, and click **Next**.
5. On the **Setup Type** page, select **Complete**, and click **Next**.
6. On the **Ready to Install the Program** page, click **Install**.

The **Installing** page presenting the installation status appears.

Once the installation is complete, the **InstallShield Wizard Completed** page appears.

7. Click **Finish**.

A message prompting you to restart the computer appears.

8. Click **Yes**.

## Activate FactoryTalk Transaction Manager

You need to activate FactoryTalk Transaction Manager so that you can use its features.

You activate FactoryTalk Transaction Manager by obtaining license activation file(s) from the Rockwell Automation licensing website and assigning them to FactoryTalk Transaction Manager using the FactoryTalk Activation Manager.

### To configure FactoryTalk Activation:

1. Open the FactoryTalk Activation UI and click **How to Activate Rockwell Software Products** in the right panel.

The **Activate Rockwell Software products** topic opens in a browser window.

2. Follow the instructions in the topic to configure your activations.

Please see the FactoryTalk Activation documentation for more details.



## Migrate and Upgrade FactoryTalk Transaction Manager

In this chapter you will learn how to perform the migration and upgrade of your FactoryTalk Transaction Manager configuration to version 13.10 in the following environments:

- **Migrate between different computers:** You have a source computer with a previous version of FactoryTalk Transaction Manager (either 10.20, 12.00, 12.10 or 13.00), and you want to move (migrate) the FactoryTalk Transaction Manager's configuration to a new target computer with the current version of FactoryTalk Transaction Manager.
- **Upgrade On a single computer:** You have a computer with a previous version of FactoryTalk Transaction Manager (either 10.20, 12.00, 12.10 or 13.00), and you want to upgrade it to the current version FactoryTalk Transaction Manager.

The steps will vary slightly depending on the versions you are migrating from (either 10.20 to 13.10, 12.00 to 13.10, 12.10 to 13.10 or 13.00 to 13.10). Please see the appropriate section for your environment:

- [Migrate FactoryTalk Transaction Manager from version 10.20 to 13.10 on page 20](#)
- [Migrate FactoryTalk Transaction Manager from version 12.00, 12.10 or 13.00 to 13.10 on page 24](#)
- [Upgrade FactoryTalk Transaction Manager on the same computer on page 30](#)

If you are migrating or upgrading FactoryTalk Transaction Manager as part of an RSBizWare system (FactoryTalk Metrics or FactoryTalk Historian Classic), follow the migration or upgrade instructions in the *RSBizWare Administration Guide*. The instructions in this manual are meant for a standalone FactoryTalk Transaction Manager system.

The following connectors are not supported in this release:

- **Control Connectors:** RSLinxClassic OPC and RSView32
- **Enterprise Database Connector:** Microsoft COM+

If any of these connectors are used in your application, please remove these connectors from the configuration before upgrading or migrating.

## Migrate FactoryTalk Transaction Manager from version 10.20 to 13.10

When migrating from version 10.20 to version 13.10, move all the configuration and data structure directly to the target computer and then update your configuration. To do this, you will need to perform the following steps:

### Start the Migration

[Step 1: Back up your FactoryTalk Administration Console configuration](#) on [page 20](#).

[Step 2: Install FactoryTalk Transaction Manager](#) on [page 21](#).

[Step 3: Copy the cfgs70.dat file](#) on [page 21](#).

[Step 4: Copy the configuration files and folder structure](#) on [page 22](#).

[Step 5: Convert the FactoryTalk Transaction Manager configuration](#) on [page 22](#).

### Finish the Migration

[Step 1: Update your FactoryTalk Transaction Manager configuration](#) on [page 26](#).

[Step 2: Restore your FactoryTalk Administration Console configuration](#) on [page 29](#).

[Step 3: Configure your database connections](#) on [page 29](#).

The following connectors are not supported in this release:

- **Control Connectors:** RSLinx Classic OPC and RSView 32
- **Enterprise Database Connector:** Microsoft COM+

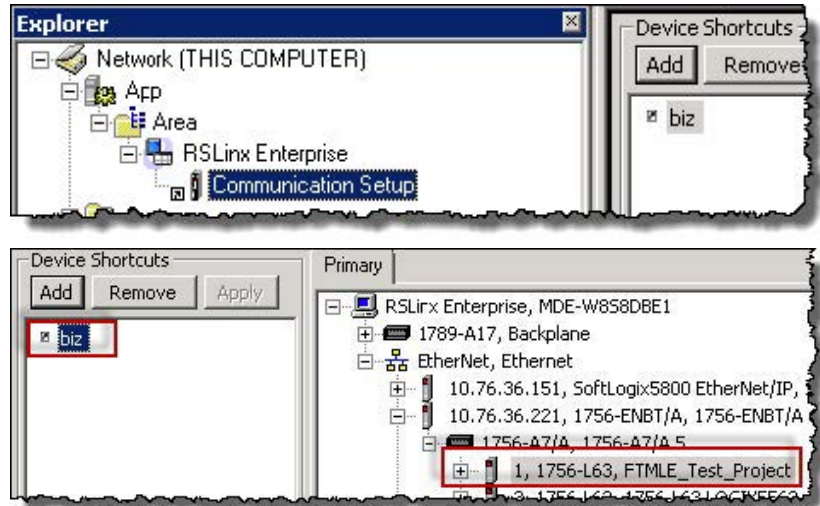
If any of these connectors are used in your application, please remove these connectors from the configuration before upgrading or migrating.

On your source computer, back up the following elements of your FactoryTalk Administration Console configuration:

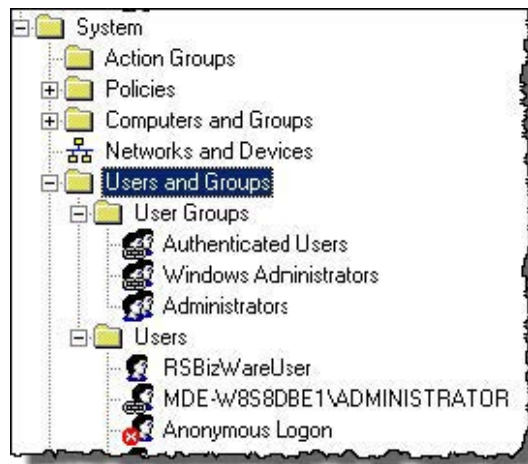
- Applications (Please note that only a single application can be backed at a time, so you may have to perform multiple backups)
- Areas
- Servers

## Step 1: Back up your FactoryTalk Administration Console configuration

- Shortcuts to controllers



- User Groups
- Users



See the FactoryTalk Administration Console documentation for instructions on backing up these items.

## Step 2: Install FactoryTalk Transaction Manager

Install FactoryTalk Transaction Manager on your target machine using the steps provided in [Installing FactoryTalk Transaction Manager](#) on [page 15](#). This machine should be a clean computer where FactoryTalk Transaction Manager has never been installed.

## Step 3: Copy the cfgs70.dat filer

The **cfgs70.dat** file is located in the following path by default on the source computer:

**C:\Program Files (x86)\Rockwell Software\RSSql**

Copy this file from the source computer to the same directory on the target computer.

## Step 4: Copy the configuration files and folder structure

Copy the FactoryTalk Transaction Manager configuration files and folder structure from the source computer to the target computer. When you copy the configuration files and folder structure, make sure that you recreate this structure on the target computer exactly like it was on the source computer. Multiple configurations (folders) may be copied, but make sure the folder names are all exactly the same as they were on the source computer.

### To check the location in which your FactoryTalk Transaction Manager configuration is saved on the source computer:

1. Open FactoryTalk Transaction Manager.
2. On the **Configuration** menu, click **Checklist**.

The **Configuration Checklist** dialog box appears.

3. Under **Define Configuration**, click **Step 1**.

The **FactoryTalk Transaction Manager Configuration** dialog box appears.

In the **Path** text box, the location in which the configuration is saved is provided.

## Step 5: Convert the FactoryTalk Transaction Manager configuration

FactoryTalk Transaction Manager 13.10 supports UNICODE (double byte) strings, so the format of the files that hold information about the FactoryTalk Transaction Manager configurations has changed from version 10.20. Old configuration files must be updated to the new format in order to be recognized and used by FactoryTalk Transaction Manager 13.10.

A conversion utility is provided for updating the existing configuration files.



Note: Perform these steps on the target computer (to which you are migrating FactoryTalk Transaction Manager).

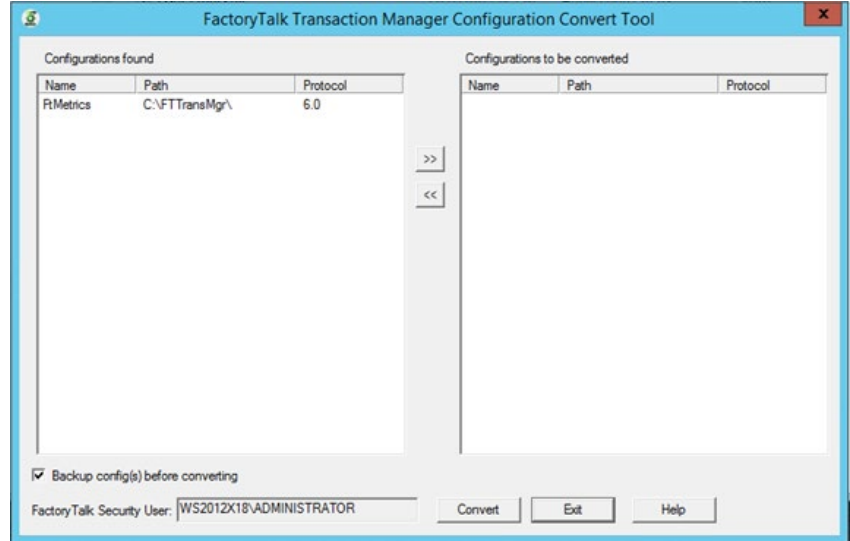
### To convert your FactoryTalk Transaction Manager Configuration:

1. Launch the **FactoryTalk Transaction Manager Configuration Convert Tool** located at the following by default:

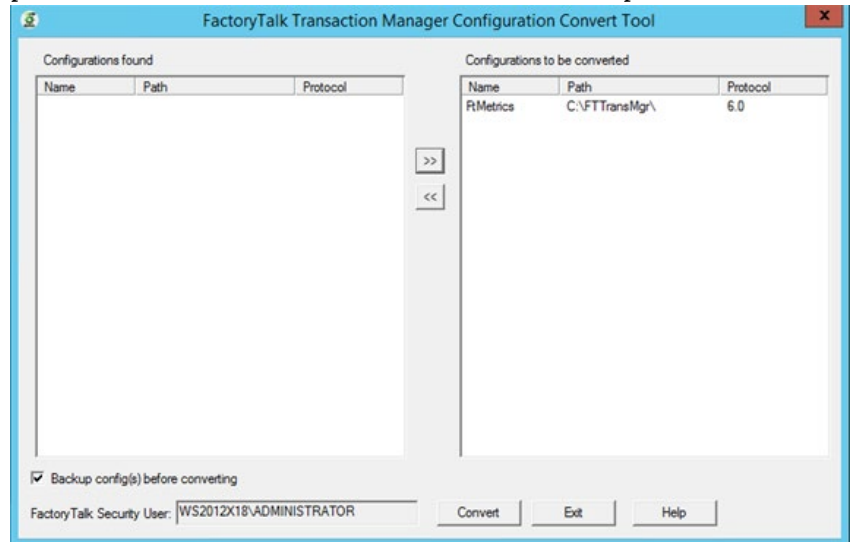
**C:\Program Files (x86)\Rockwell  
Software\RSCCommon\rssql\_convert\_configs.exe**

2. The Tool finds the configurations on the local server that can be upgraded, and lists them in the left panel. Select the configuration(s) to convert by moving them from the left panel to the right panel using

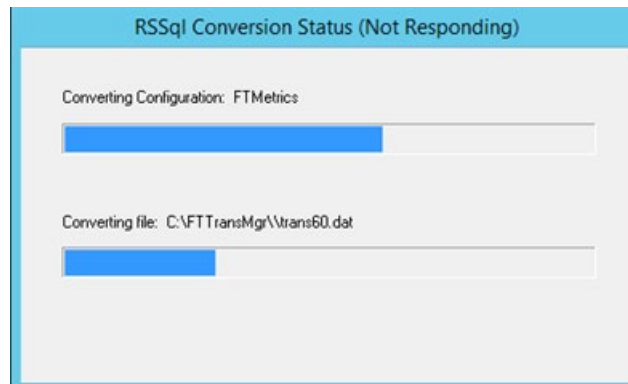
the >> button. Make sure the **Backup config(s) before converting** checkbox is checked.



Note the path of the configuration(s) to be converted listed in the right panel. This information will be needed in future steps.

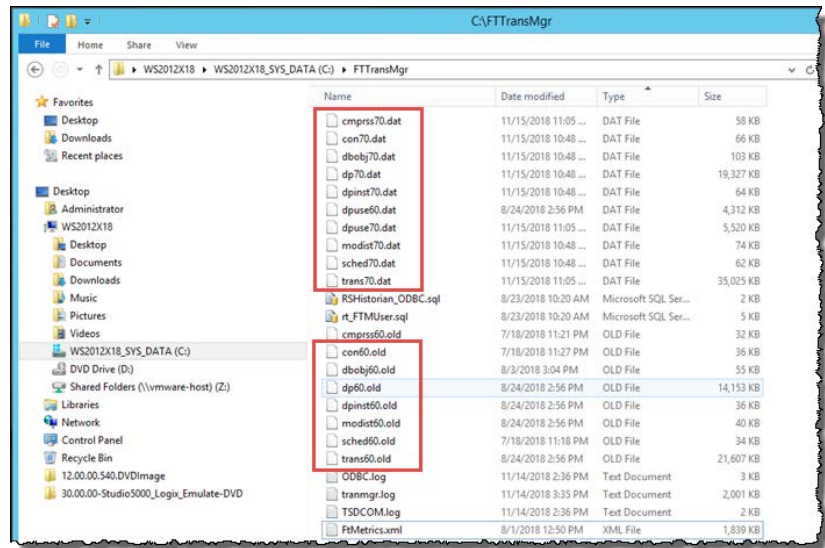


3. Click **Convert** to start the conversion.



4. When the conversion tool is finished, click **Exit** to close the utility.

5. Verify that the conversion files were created by looking in the directory where the FactoryTalk Transaction Manager configuration is stored. Note that after running the conversion:
  - Old configuration files have been given an extension of **.old**. They will no longer be used.
  - New configuration files have been given names containing **70** instead of **60**.
  - If at any point it is necessary to have the old configuration files again, they can be found here. To use them with FactoryTalk Transaction Manager 10.20, change their names from **.old** to **.dat**. Please note that any changes made to the configuration from this point on will not be present in the **.old** files.



Wait until the configuration is converted.

After the conversion is complete, please go to [Finish the Migration](#) on [page 26](#) for the last migration steps.

## Migrate FactoryTalk Transaction Manager from version 12.00, 12.10 or 13.00 to 13.10

When migrating from version 12.00, 12.10 or 13.00 to version 13.10, back up and restore the configuration from your source computer to your target computer and then update your configuration. To do this, you will need to perform the following steps:

### Start the Migration

Step 1: Back up your FactoryTalk Administration Console configuration.

[Step 2: Back up your FactoryTalk Transaction Manager configuration](#) on [page 26](#).

[Step 3: Install FactoryTalk Transaction Manager](#) on [page 26](#).

[Step 4: Restore your FactoryTalk Transaction Manager configuration](#) on [page 26](#).



## Finish the Migration

[Step 1: Update your FactoryTalk Transaction Manager configuration on page 26.](#)

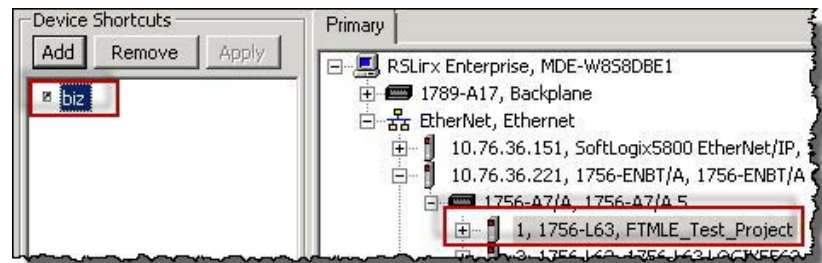
[Step 2: Restore your FactoryTalk Administration Console configuration on page 29.](#)

[Step 3: Configure your database connections on page 29.](#)

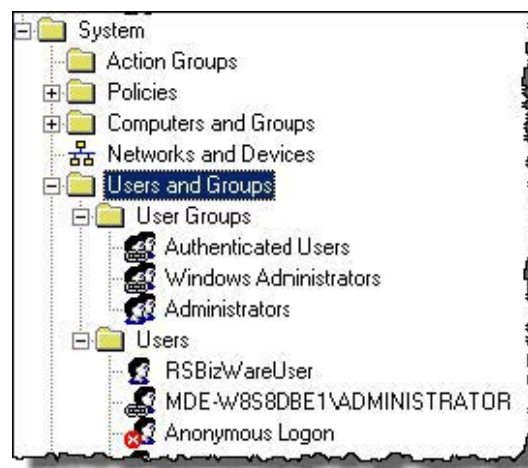
### Step 1: Back up your FactoryTalk Administration Console configuration

On your source computer, back up the following elements of your FactoryTalk Administration Console configuration:

- Applications (Please note that only a single application can be backed at a time, so you may have to perform multiple backups)
- Areas
- Servers
- Shortcuts to controllers



- User Groups
- Users



See the FactoryTalk Administration Console documentation for instructions on backing up these items.

## Step 2: Back up your FactoryTalk Transaction Manager configuration

Back up your FactoryTalk Transaction Manager configuration on your source machine by going to **Configuration** > **Back up** in the FactoryTalk Transaction Manager UI. This will create an rsq file on this server. You will need this backup when you restore it to your target machine.

## Step 3: Install FactoryTalk Transaction Manager

Install FactoryTalk Transaction Manager on your target machine using the steps provided in [Installing FactoryTalk Transaction Manager](#) on [page 15](#). This machine should be a clean computer where FactoryTalk Transaction Manager has never been installed.

## Step 4: Restore your FactoryTalk Transaction Manager configuration

Restore your FactoryTalk Transaction Manager configuration on your target machine by going to **Configuration** > **Restore** in the FactoryTalk Transaction Manager UI. Use the rsq backup file you created in [Step 2: Back up your FactoryTalk Transaction Manager configuration](#) on [page 26](#).

After restoring your configuration, please go to [Finish the Migration](#) on [page 26](#) for the last migration steps.

## Finish the Migration

After you have completed the steps appropriate for your migration, perform the following steps to finish the migration.

## Step 1: Update your FactoryTalk Transaction Manager configuration

### To update your FactoryTalk Transaction Manager configuration:

1. Run Transaction Manager and confirm that your updated Configuration appears in the left-hand window.
2. On the **Configuration** menu, click **Checklist**.

The **Configuration Checklist** dialog box appears.

3. Choose your Configuration from the drop-down menu.
4. Under **Define Configuration**, click **Step 1**.

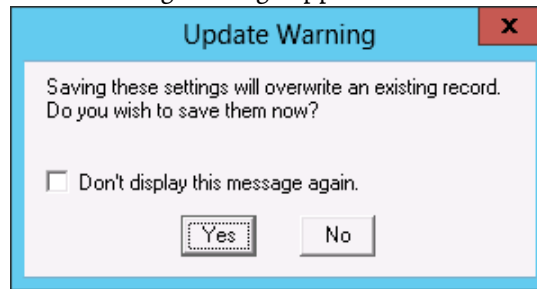
The **FactoryTalk Transaction Manager Configuration** dialog box appears.

5. Make sure that the configuration is the same as it is in your source configuration setting.

Please note that **Time-series Data Compression** is used for FactoryTalk Historian Classic only, and the **FactoryTalk Metrics** connector service is used for FactoryTalk Metrics only.

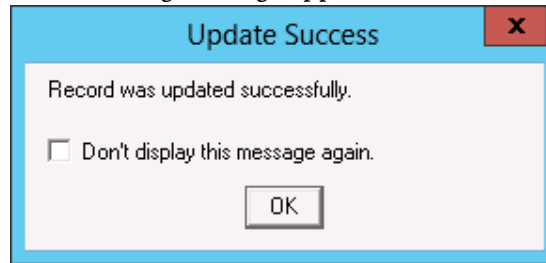
6. Click **Apply**.

The following message appears:



7. Click **Yes**.

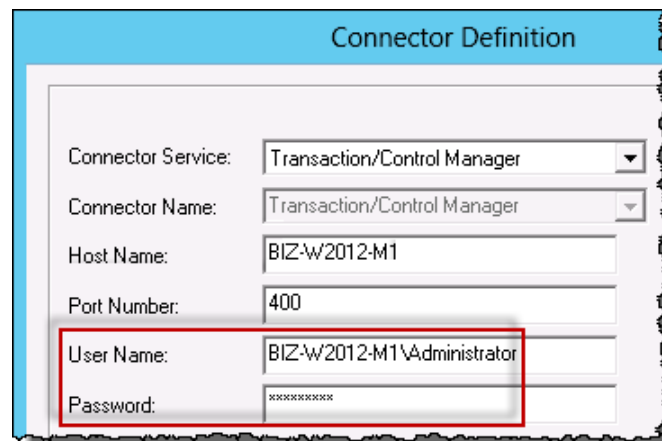
The following message appears:



8. Click **OK**.
9. Click **Close**.
10. In the **Configuration Checklist** dialog box, under **Define Connectors**, click **Step 2**.

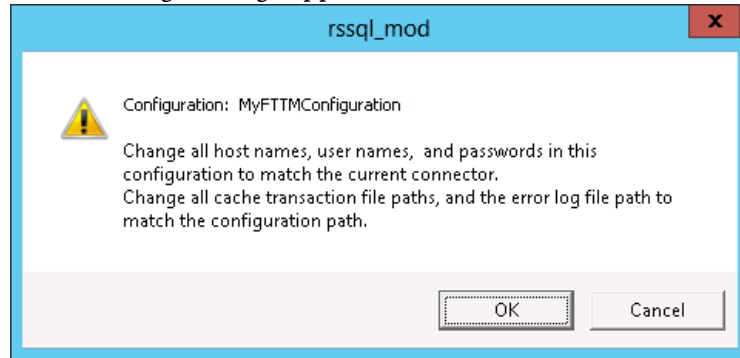
The **Connector Definition** dialog box appears.

11. In the **Host Name** text box, replace the name of the source computer on which you backed up the FactoryTalk Transaction Manager configuration with the name of the target computer on which you restored the configuration.
12. In the **User Name** text box, replace the name of the user of the source computer with the name of the user of the target computer.
13. In the **Password** text box, replace the password of the user of the source computer with the password of the user of the target computer.



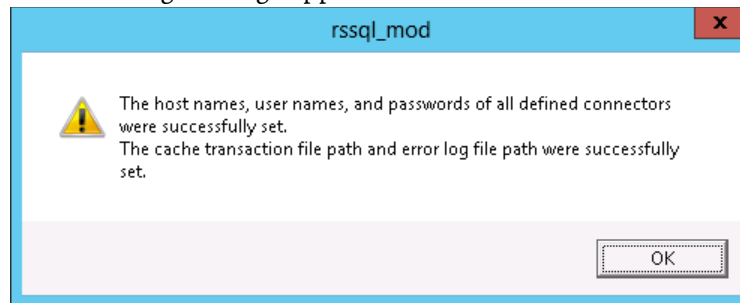
14. Click **Host / User**.

The following message appears:



15. Click **OK**.

The following message appears:

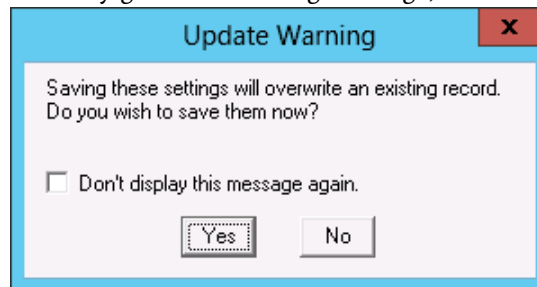


16. Click **OK**.

You have updated the host names, usernames, and passwords for all the connectors defined for your FactoryTalk Transaction Manager configuration.

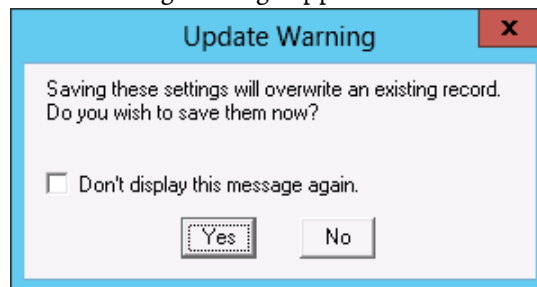
17. Click **Apply**.

You may get the following message, which can be ignored:



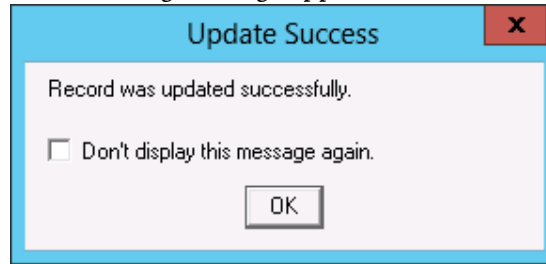
18. Click **OK**.

The following message appears:



19. Click **Yes**.

The following message appears:



20. Click **OK**.

21. Click **Close** to exit the **Configuration Checklist**.

## Step 2: Restore your FactoryTalk Administration Console configuration

To restore your FactoryTalk Administration Console configuration to your target computer, use the backups you created in [Step 1: Back up your FactoryTalk Administration Console configuration](#) on [page 20](#).

### To recreate your configuration:

1. On the Start screen, type any part of the name **FactoryTalk Administration Console**, and then click its shortcut.

The **FactoryTalk Administration Console** window appears.

2. In the **Select FactoryTalk Directory**, click **OK**.
3. Log into FactoryTalk Directory.
4. In the Explorer, right-click the heading of the object you want to restore and select the restore option. For example, to restore an application, right-click **Network**, and then click **Restore Application**.

Please note the following:

- Make sure that all the applications and objects from the source computer are restored.
- Only a single application can be restored at a time, so you may have to perform multiple restore transactions.
- Users that are restored from the source machine may not be valid due to an incorrect machine name. Make sure to check your users and edit their configuration as needed under **Explorer > System > Users and Groups**.

## Step 3: Configure your database connections

Configure your database connections by performing the following as applicable:

- For SQL Server (ODBC connector), configure the ODBC data source (System DSN) to have the same name as the one used on the source computer to make sure the connection to the same database is available.
- For Oracle (OCI connector), use the Net Configuration Assistant to configure the TNS name to have the same name as the one used on the source computer to make sure the connection to the same database is available.

- Check database connections.
  1. In the configuration pane, right-click the configuration, and then click **Start Configuration**.
  2. Confirm that the traffic light is green.
  3. Check the **Transaction Monitor** pane to confirm that the transaction is working properly.

## Upgrade FactoryTalk Transaction Manager on the same computer

In order to successfully upgrade your FactoryTalk Transaction Manager configuration from your current version to the latest version on the same computer, perform the following steps.

1. Back up your configuration.
2. Remove the old version of FactoryTalk Transaction Manager.
3. Install the new version of FactoryTalk Transaction Manager.
4. (Upgrading from 10.20 to 13.10 only) Convert your FactoryTalk Transaction Manager configuration using FactoryTalk Transaction Manager Convert Utility.

The following connectors are not supported in this release:

- **Control Connectors:** RSLinxClassic OPC and RSView32
- **Enterprise Database Connector:** Microsoft COM+

If any of these connectors are used in your application, please remove these connectors from the configuration before upgrading or migrating.

### Step 1: Remove FactoryTalk Transaction Manager

Stop all the configurations and then remove the older version of FactoryTalk Transaction Manager using Control Panel or the Start menu.

### Step 2: Install the new version of FactoryTalk Transaction Manager

Install the new version of FactoryTalk Transaction Manager using the FactoryTalk Transaction Manager installation package. See [Installing FactoryTalk Transaction Manager](#) on [page 15](#) for details.

### Step 3: Convert the FactoryTalk Transaction Manager configuration

---

**IMPORTANT** Only perform this step if you are migrating from FactoryTalk Transaction Manager 10.20 to 13.10. This step is required.

---

FactoryTalk Transaction Manager 13.10 supports UNICODE (double byte) strings, so the format of the files that hold information about the FactoryTalk Transaction Manager configurations has changed. Old configuration files must be updated to the new format in order to be recognized and used by FactoryTalk Transaction Manager 13.10.

A conversion utility is provided for updating the existing configuration files.



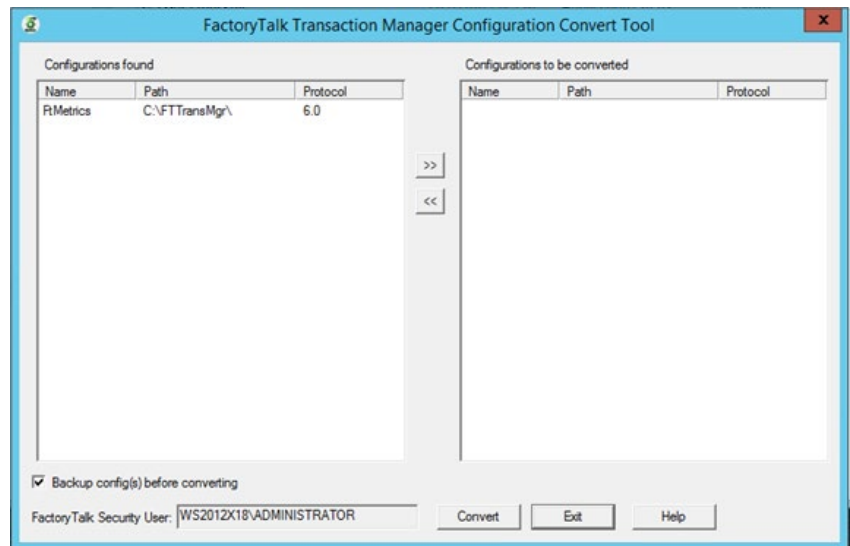
Note: Perform these steps on the target computer (to which you are migrating FactoryTalk Transaction Manager).

## To convert your FactoryTalk Transaction Manager Configuration:

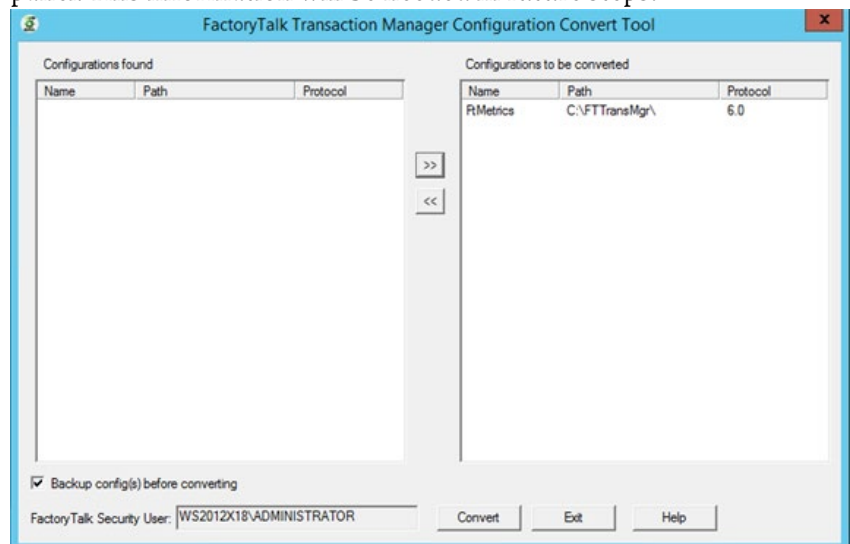
1. Launch the **FactoryTalk Transaction Manager Configuration Convert Tool** located at the following by default:

**C:\Program Files (x86)\Rockwell  
Software\RSCCommon\rssql\_convert\_configs.exe**

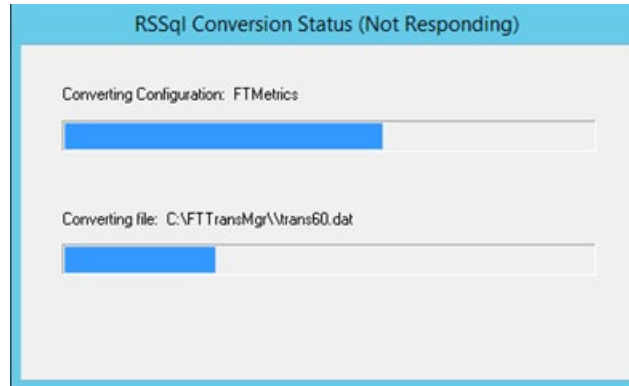
2. The Tool finds the configurations on the local server that can be upgraded, and lists them in the left panel. Select the configuration(s) to convert by moving them from the left panel to the right panel using the >> button. Make sure the **Backup config(s) before converting** checkbox is checked.



Note the path of the configuration(s) to be converted listed in the right panel. This information will be needed in future steps.

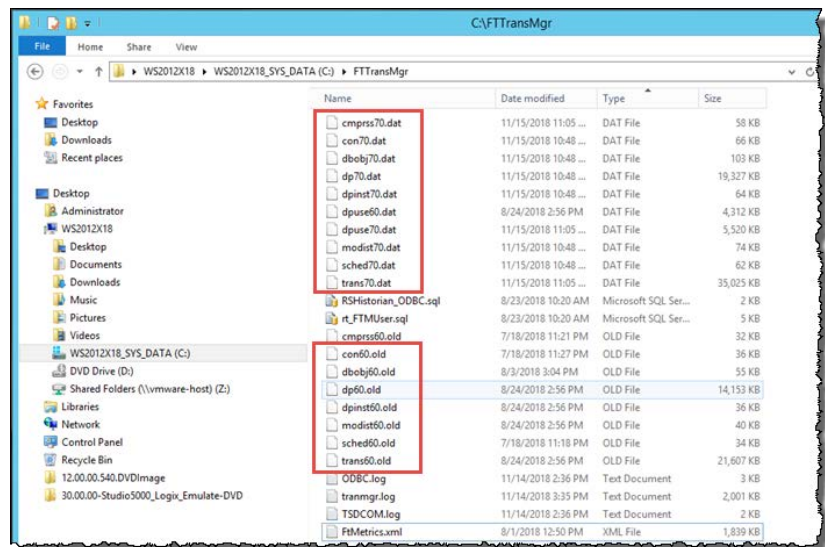


- Click **Convert** to start the conversion.



- When the conversion tool is finished, click **Exit** to close the utility.
- Verify that the conversion files were created by looking in the directory where the FactoryTalk Transaction Manager configuration is stored. Note that after running the conversion:

- Old configuration files have been given an extension of **.old**. They will no longer be used.
- New configuration files have been given names containing **70** instead of **60**.
- If at any point it is necessary to have the old configuration files again, they can be found here. To use them with FactoryTalk Transaction Manager 10.20, change their names from **.old** to **.dat**. Please note that any changes made to the configuration from this point on will not be present in the **.old** files.



- Confirm that the upgrade is successful:
  - In the configuration pane, right-click the configuration, and then click **Start Configuration**.
  - Confirm that the traffic light is green.
  - Check the **Transaction Monitor** pane to confirm that the transaction is working properly.



## Exploring the FactoryTalk Transaction Manager User Interface

In this chapter you will learn about the following:

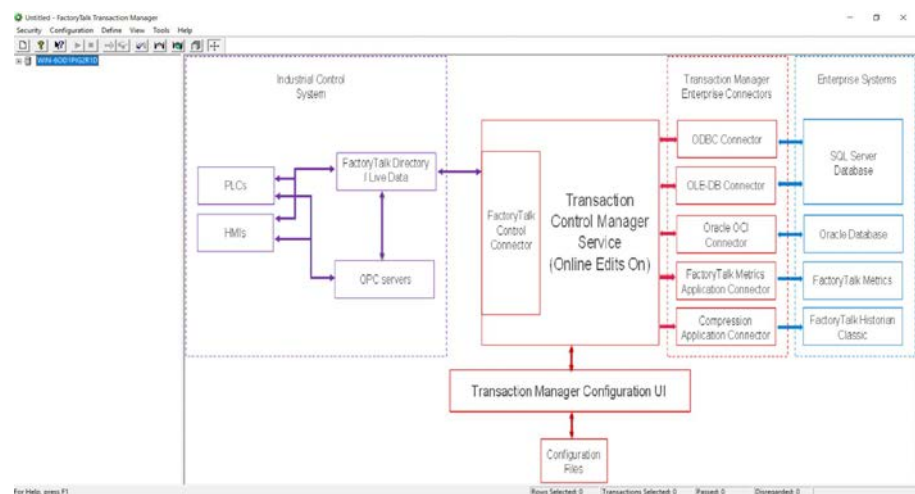
- [Starting FactoryTalk Transaction Manager](#) on [page 33](#)
- [Exploring the FactoryTalk Transaction Manager User Interface](#) on [page 33](#)
- [Configuration Checklist](#) on [page 41](#)
- [Miscellaneous](#) on [page 45](#)
- [Understanding FactoryTalk Transaction Manager External Files](#) on [page 47](#)

### Starting FactoryTalk Transaction Manager

To start FactoryTalk Transaction Manager go to **Start > Programs > Rockwell Software > FactoryTalk Transaction Manager > FactoryTalk Transaction Manager**.

### Exploring the FactoryTalk Transaction Manager User Interface

When you start FactoryTalk Transaction Manager for the first time (and if it is the first Rockwell Software product run on your computer), you are automatically logged on to FactoryTalk Security using your Windows-linked user account information. The system graphic appears in the right pane (or workspace) of the FactoryTalk Transaction Manager user interface.



For more information on FactoryTalk Security, see [Securing FactoryTalk Transaction Manager with FactoryTalk Security](#) on [page 103](#). To view the procedures for configuring and using FactoryTalk Transaction Manager, on the menu, click **Help > Quick Start**.

The FactoryTalk Transaction Manager user interface includes the following elements:

- [Title bar](#) on [page 34](#)
- [Menu bar](#) on [page 34](#)
- [Toolbar](#) on [page 35](#)
- [Configuration tree](#) on [page 36](#)
- [Workspace](#) on [page 37](#)
- Status bar

## Title Bar

The title bar displays the name of the configuration or configuration server, depending on what is selected in the configuration tree.

## Menu Bar

You can access many features from the FactoryTalk Transaction Manager menu bar.



Note: For a description of the available shortcuts, see [Toolbar](#) on [page 35](#).

`S`ecurity `C`onfiguration `D`efine `V`iew `T`ools `H`elp

Review the following for additional information.








Use this menu:	To do the following:
Security	<ul style="list-style-type: none"> <li>• Log on or log off of FactoryTalk Security.</li> <li>• View your FactoryTalk Security permissions on the current configuration server.</li> </ul> <p>For more information on FactoryTalk Security, see <a href="#">Securing FactoryTalk Transaction Manager with FactoryTalk Security</a> on <a href="#">page 103</a>.</p>
Configuration	<ul style="list-style-type: none"> <li>• Create a new configuration.</li> <li>• Set or change properties in a configuration.</li> <li>• Access the Configuration Checklist.</li> <li>• Delete, backup, or restore a configuration.</li> <li>• Stop or start a configuration.</li> </ul>
Define	<p>Define:</p> <ul style="list-style-type: none"> <li>• Connectors.</li> <li>• Data objects.</li> <li>• Data points.</li> <li>• Transactions.</li> <li>• Options for error logging.</li> <li>• Scheduled events.</li> </ul>
View	<ul style="list-style-type: none"> <li>• View: <ul style="list-style-type: none"> <li>• Information about the currently defined transaction.</li> <li>• Diagnostic information about the configuration that is currently running.</li> <li>• Error log files.</li> <li>• The system graphic.</li> </ul> </li> <li>• Switch between large and small icons.</li> <li>• Obtain a status update.</li> <li>• Enable or disable the toolbar or status bar.</li> </ul>






Tools	<ul style="list-style-type: none"> <li>• Create a configuration report.</li> <li>• Verify the selected configuration.</li> <li>• Use wizards to create data logging, duplicate data points, or duplicate transactions.</li> <li>• Set options for messages and the Log Viewer.</li> </ul>
Help	<ul style="list-style-type: none"> <li>• Open the FactoryTalk Transaction Manager Online Help system to get assistance with using FactoryTalk Transaction Manager. You can use the index of the online help system to search for a topic for which you want to view information.</li> <li>• Access:             <ul style="list-style-type: none"> <li>• The FactoryTalk Transaction Manager Release Notes.</li> <li>• The Quick Start Guide for FactoryTalk Transaction Manager.</li> <li>• The product manuals.</li> <li>• Error code resources.</li> </ul> </li> <li>• View information about:             <ul style="list-style-type: none"> <li>• Support and training.</li> <li>• Copy protection.</li> <li>• License type and software version.</li> </ul> </li> </ul>

In addition, you can use this option to obtain FactoryTalk Transaction Manager license and version information.

## Toolbar

The toolbar contains shortcuts to several commonly used FactoryTalk Transaction Manager functions. Each button on the toolbar is a graphical representation of a command that is also available from the FactoryTalk Transaction Manager menu bar.

Icon	Description
	Creates a new FactoryTalk Transaction Manager configuration.
	Displays the About FactoryTalk Transaction Manager dialog box.
	Displays the online help for any design element (click this button and place the cursor on any design element).
	Starts the configuration or connector currently selected in the Configuration tree. If the edit enabled configuration has pending edits, a message appears prompting the user to ignore the pending edits and start the configuration anyway.
	Stops the configuration or connector currently selected in the Configuration tree.
	Assembles all data point and transaction pending edits in the edit enabled configuration. For more information, see <a href="#">Understanding Online Edits</a> on <a href="#">page 79</a> .
	Opens the Pending Edit Alerts dialog box. For more information, see <a href="#">Understanding Online Edits</a> on <a href="#">page 79</a> .

	Verifies the selected configuration. You can verify multiple transactions and choose to save the results to a text file.
	Displays information about the current transactions in the right pane.
	Displays diagnostic information about the configuration that is currently running in the right pane.
	Displays error log information in the right pane.
	Displays the system graphic in the right pane.

## Configuration Tree




The FactoryTalk Transaction Manager user interface is divided into two panes. The left pane is known as the **Configuration tree**. It displays all the items of the current configurations in the form of a tree.

The top level in the item tree is the Configuration Server, with the name of the computer on which it is running. The second level is the configuration name. The third level displays the Transaction Control Manager service (if the configuration uses online edits) or the FactoryTalk Transaction Manager service. The Transaction Control Manager service or the FactoryTalk Transaction Manager service are followed by the control connectors and enterprise connectors defined for the configuration. For more information, see [Understanding Online Edits](#) on [page 79](#).

## Configuration and Connector Status




The traffic lights in the Configuration tree represent the status of FactoryTalk Transaction Manager connectors and configurations, turning from red to green when services are started.

Review the following for additional information.

If you see this icon:	A configuration is:	A connector is:
	Running properly. All connectors in that configuration are running properly. All transactions start based on the triggering rules that you defined.	Running properly.
	Experiencing problems with one or more connectors that are not functioning properly.	This state does not apply to a connector.
	Stopped. All of the connectors are not running. Additionally, the Transaction Control Manager service or the FactoryTalk Transaction Manager service is not running.	Not running properly.

## Configuration Server Status

The icons that are displayed in the Configuration tree represent the status of the configuration servers. The following table shows the status icons, state, and a description of that state:

Icon	Status	Description:
	Running	The configuration server host computer: <ul style="list-style-type: none"> <li>Resides in the FactoryTalk Administration Console.</li> <li>Is working properly (the service is running and FactoryTalk Transaction Manager can connect to it).</li> <li>Has been configured in the FactoryTalk Administration Console so that the current user has <b>permissions</b> to communicate with it.</li> </ul>
	Unknown	The configuration server host computer: <ul style="list-style-type: none"> <li>May or may not reside in the FactoryTalk Administration Console.</li> <li>Cannot be connected to or queried in FactoryTalk Transaction Manager.</li> <li>Cannot provide any information to FactoryTalk Transaction Manager so that the software can determine why it is not working properly.</li> </ul>
	No privilege	The configuration server host computer: <ul style="list-style-type: none"> <li>Resides in the FactoryTalk Administration Console.</li> <li>Is working properly.</li> <li>User does not have <a href="#">permissions</a> on <a href="#">page 106</a> to communicate with the Configuration Server.</li> </ul>

## Workspace

The right pane of the FactoryTalk Transaction Manager user interface is known as the workspace. Depending on the view option that you have selected, it displays:

- The Transaction Definition View.
- The Transaction Monitor View.
- The Error Log Files View.
- The FactoryTalk Transaction Manager system graphic.

## Transaction Definition View

Transactions are displayed in the Transaction Definition view. The states of the transaction definitions are described in the table below:

Item	Description
Current	The current definition of a transaction (with or without pending edits).
Edit Pending	Changes that are made to the current transaction definition, but not assembled.
Add Pending	A new transaction (created online), but not assembled.

The following is a sample Transaction Definition View.

Transaction Name	State	Owner
10SEC_LOG_CLOCKsEC6	Current	FactoryTalk Tr
10SEC_LOG_WALLCLOCK4	Current	FactoryTalk Tr
13SEC_FLOAT0	Current	FactoryTalk Tr
18secBadPoint	Current	FactoryTalk Tr
19secInput1only_logged	Current	FactoryTalk Tr
float0_insert	Edit Pending	FactoryTalk Tr
float1_insert	Add Pending	FactoryTalk Tr
INPUT1_BOOLINDEX	Current	FactoryTalk Tr

### Transaction Monitor View

The following is a sample Transaction Monitor View.

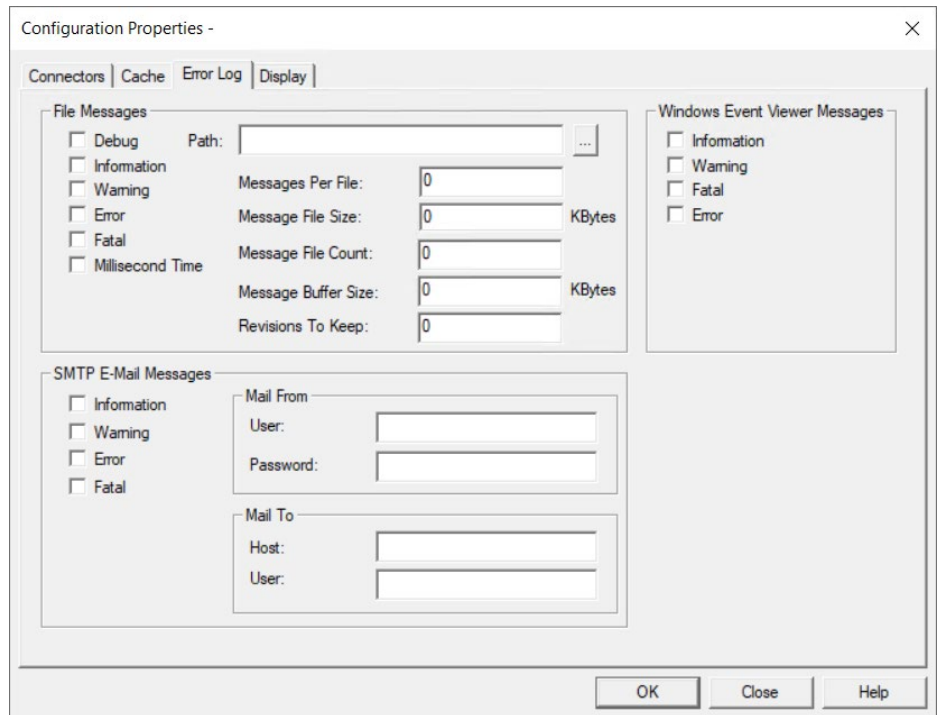
Transaction Name	Total	Passed	Failed	% Passed	Cached	Database Passed	Database Failed	Pending
Totals:	24	24	0	100.00 %	0	26	0	0
Trans1	12	12	0	100.00 %	0	13	0	0
Trans2	12	12	0	100.00 %	0	13	0	0

### Error Log Files View

The following is a sample Error Log Files View.

Date/ Time	Source	Level	Transaction ID	Occurrence ID	Message Text
03/19/2021 09:46:44...	TRX	INF			The software is starting Transaction Manager (version 13.10.00, build 6).
03/19/2021 09:46:44...	TRX	INF			This is the Professional license for unlimited tags.

To specify error log files, click **Define > Error Logging** then the **Error Log** tab appears.



This dialog allows you to specify the logging behavior for the configuration. These settings apply to all the services in the configuration that support

logging. The logs contain information written by the services that is useful for troubleshooting and debugging. The configuration must be stopped to change the Error Logging properties. Three logging destinations can be configured:

1. **File Messages** - writes logs to a local or network drive location
2. **Windows Event Viewer Messages** - writes messages to the Windows Event Logs
3. **SMTP E-Mail Messages** - emails messages to a recipient using SMTP

The following message types can be specified to send to the logging destinations (not all destinations support all types):

1. **Debug** - this is the lowest level of logging supported and is typically used for debugging. Note that it will generate a large number of log messages and should be used only if needed.
2. **Information** - provides informational messages about actions taken within the service.
3. **Warning** - notes possibly abnormal conditions encountered that do not cause an error.
4. **Error** - notes errors encountered.
5. **Fatal** - conditions that cause the service to stop.
6. **Milliseconds Time** - check this to add milliseconds to all timestamps.

## File Messages

This option allows you to create log files on a local or network drive, and to exercise a great deal of control over how much data will be logged and saved.

1. **Message types:** check all the following message types you wish to log to files.
2. **Path:** enter the path or browse to the directory that will contain the log files. For distributed configurations, you should use a UNC path.
3. **Log file size:** two parameters determine the maximum size of a log file. If either of these is reached, the log file will be closed.
  - Messages Per File - the maximum number of messages contained in the file
  - Message File Size - the maximum size of the log file in kilobytes
4. **Message File Count** - the maximum number of log files to keep each time the configuration is run. The Message File Count parameter is a number between 1 and 100 that specifies how many log files will be kept during a single continuous run of a FactoryTalk Transaction Manager configuration (a revision). Once the count is reached, the oldest log file will be overwritten. If the Message File Count is 1 (the default), then a single log file will be continuously overwritten.
5. **Message Buffer Size** - the internal buffer size that determines how often data is written to the log file. A valid range is between 0

(default) and 10 kilobytes. A larger number could increase performance if logging volume is high.

6. **Revisions To Keep** - The number of revisions of log files to keep, where a revision is a single continuous run of a configuration, and each revision could contain 1-100 log files. The Revisions to Keep parameter specifies how many revisions to keep, and when that number is met, when the configuration is run the oldest revision of log files will be deleted. The default is 0, which means no revisions are kept.



Note: It is possible to accumulate a large number of log files requiring considerable space using these parameters. The maximum space for log files is approximately: **Message File Size \* Message File Count \* Revisions to Keep \* number of services that produce log files.**

Log files are not deleted automatically (except as dictated by parameter settings).

The naming convention for log files is as follows:

**SERVICENAMEBB\_#CCYYYYMMDDHHMM.log**

Where:

1. **SERVICENAME** refers to the service that created the log file. Where names were entered for services those names will be used, and default names otherwise.
2. **BB** refers to the version number. For the current version being recorded, **BB** is blank.



Note: The version number of all saved files will be incremented every time the configuration is started.

3. **CC** refers to the file number within the version. For the first file **CC** is blank.
4. **YYYY** - year timestamp.
5. **MM** - month timestamp.
6. **DD** - day timestamp.
7. **HH** - hour timestamp.
8. **MM** - minute timestamp.

## Windows Event Viewer Messages

Choose the message types you would like to send to the Windows Event Log. The messages can be viewed using the Windows Event Log Viewer.

## SMTP Email Messages

Choose the message types you would like to be sent via email. Choose a Sender and Recipient. You should be aware of the volume of email this might create.



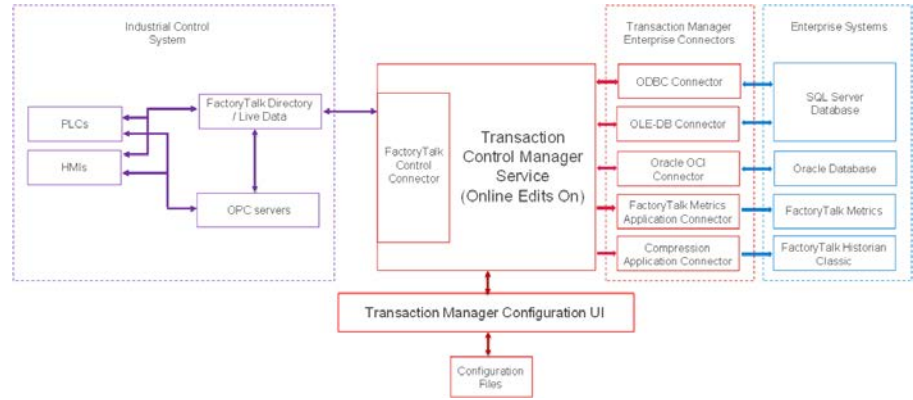


Tip: If your configuration includes the Time-series Data Compression enterprise connector, FactoryTalk Transaction Manager will only support two error log revisions (no matter what value you enter in the **Revisions To Keep** field).

For distributed configurations, you must use a [UNC path](#) on [page 91](#) for the error log file.

## FactoryTalk Transaction Manager system graphic

This is the FactoryTalk Transaction Manager system graphic.



## Status Bar

The status bar at the bottom of the main FactoryTalk Transaction Manager user interface displays the following information:

Item	Description
Rows Selected (includes pending edits)	The number of rows selected before you saved, assembled or canceled pending edits.
Transactions Selected	The number of transactions selected.
Passed	The number of successful operations.
Disregarded	FactoryTalk Transaction Manager does not allow the action to be performed.
Failed	Operations that did not take place because of an internal error.

## Configuration Checklist

A FactoryTalk Transaction Manager configuration consists of a set of transactions that use control and enterprise connector elements required to perform the transactions. You may create many configurations, but the Transaction Control Manager service or the FactoryTalk Transaction Manager service can run only one configuration at a time. Therefore, all the transactions required to implement an application must be contained in a single configuration.

You can create a configuration using the Configuration Checklist, which lists the required steps and displays the progress of your configuration. A green check mark indicates a completed step, while a yellow check mark indicates a partially completed step.

### To display the Configuration Checklist for an already existing configuration:

1. In the Configuration tree, select the configuration name for which you want to display the Configuration Checklist.

2. On the menu, click **Configuration > Checklist**. The **Configuration Checklist** dialog box appears.

### To create a new Configuration Checklist:

On the menu, click **Configuration > New**. The **Configuration Checklist** dialog box appears.

This chapter is not intended to provide step-by-step directions for creating a configuration, but to provide an overview to help you understand configurations. The following sequence mimics the Configuration Checklist. It is recommended, at least initially, that you follow this sequence when creating configurations:

- Define and name a new configuration.
- Define control and enterprise connectors.
- Define the data points that will be used in transactions.
- Define the data objects that will be used in transactions.
- Define transactions, which transfer data between data points, in the control system and data objects.
- Verify transactions.

The following sections provide additional details for each task in the sequence. For more information, see the Quick Start in the FactoryTalk Transaction Manager online help (on the menu, click **Help > Quick Start**).

## Step 1: Defining and Naming a New Configuration

### To define a new configuration:

1. In the **Configuration Name** text field, type the name of the new configuration.
2. Under **Define Configuration**, click **Step 1**. The **FactoryTalk Transaction Manager Configuration** dialog box appears.
3. Enter a name for the configuration and the directory path in which the configuration (\*.dat) files will reside. This directory becomes the default location for the other files used in the configuration.



Note: Each configuration must have a unique name and directory path.

4. (Optional). If you want to create a configuration that uses online edits, select the **Enable Online Edits** check box.
5. Select the check boxes next to the control and enterprise connector services which will be used in the configuration.



Note: For a configuration that uses online edits, only the FactoryTalk Live Data control connector can be used. It is selected by default. You may additionally select any enterprise connector. For more information, see [Understanding Online Edits](#) on [page 79](#).

To create a distributed configuration, see [Advanced Topics](#) on [page 89](#).

6. Click **Apply**. A message appears informing you the information was successfully saved.

7. Click **OK**, and then **Close**. A check mark appears next to Step 1 to indicate it is complete.

## Step 2: Defining Connectors

### To define a connector:

1. Under **Define Connectors**, from the drop-down list select the connector service to define.
2. Click **Step 2**. The **Connector Definition** dialog box appears. This dialog box allows you to configure communication and security settings for the Microsoft Windows services.
3. Click **Apply**. A message appears informing you the information was successfully saved.
4. Click **OK**, and then click **Close**. A check mark appears next to Step 2 to indicate it is complete.

**You can create multiple control or enterprise connectors in a configuration, but you can create only one connector of each type on each host computer.** A FactoryTalk Transaction Manager Professional activation is required to distribute connectors.



Note: Only one instance of the Transaction Control Manager service or the FactoryTalk Transaction Manager service can be used by a configuration.

The connector service uses the Microsoft Windows username and password assigned to the connector when it runs on the host computer.

In general, control connectors must run on the computer that contains the data server to which they will communicate. This is not the case with some OPC servers that support DCOM or OPC used in a FactoryTalk architecture.

After this step is complete and you exit the Configuration Checklist, the new configuration appears in the Configuration tree. Each connector service that is used by the configuration appears in the item tree. Individual control and enterprise connectors appear under the connector services. The Configuration tree displays all configurations recognized by each registered configuration server.

## Step 3: Defining Data Points

### To define data points:

1. Under Define Data Points, from the drop-down list select the control connector for which you want to add or edit points.



Note: If you have multiple control connectors, you must configure them individually.

2. Click Step 3. The Select An Application dialog box appears.
3. From the list of applications, select the application for which you want to define the data points, and then click OK. The (FactoryTalk Transaction Manager) FactoryTalk Data Point dialog box appears.

4. Under **Select Tag(s)**, in the **Folders** list, browse to the directory containing the tags you want to use.
5. From the tags list, select the tags you want to use in a configuration, and then click **Add Selected tag(s)**. The selected tags appear in a numbered list at the bottom of the dialog box.



Tip: You can select multiple tags by holding **Ctrl** and clicking on the tags.

6. Click **Apply**, and then click **Close**. A check mark appears next to **Step 3** to indicate it is complete.

For more information about data points, see [Defining Data Points](#) on [page 55](#). In a configuration that uses online edits, you can perform **Step 3** on the **Configuration Checklist** while the configuration is running. For information about distributing control connectors and remote browsing capabilities, see [Advanced Topics](#) on [page 89](#).

## Step 4: Defining Data Objects

### To define a data object:

1. Under **Define Data Objects**, from the drop-down list select the enterprise connector for which you want to define the data object.
2. Click **Step 4**. The **Data Object Definition** dialog box appears. Use this dialog box to configure connections to the database and create a data object.



Note: Depending on the type of the enterprise connector, the items available for configuration in the dialog box may differ.

For more information about enterprise connectors, see [Defining Data Objects](#) on [page 63](#).

3. Select the table, view, or connection (or appropriate enterprise object for your type of connector) to configure the data object to use. Depending on the connector type, different properties are displayed, such as whether to insert or update the rows in the table you select. For information about distributing enterprise connectors, see [Advanced Topics](#) on [page 89](#).
4. Click **Apply**, and then click **Close**. A check mark appears next to **Step 4** to indicate it is complete.

## Step 5: Defining Transactions

You can create transactions that move data between a control system and an enterprise application or database.

### To define a transaction:

1. Click **Step 5**. The **Transaction Definition** dialog box appears.
2. Provide a name for the transaction, and then select the data object to which you want to bind (the process of mapping a column in a database table to a data point) the data points. The data object's columns or parameters appear in the list of available bindings near the bottom of the dialog box.



Note: Each transaction name must be unique.

You can bind individual data points or an expression to a column/field/parameter in a data object. Double-click a non-bound entry in the Data Object Column to open the Filter and Select Data Points dialog box or right-click and select Filter and Select Data Points from the menu. You can limit the list of data points by a connector and/or device. Bind the data point to the data object by dragging it from the Filter and Select Data Points dialog box and dropping it onto the appropriate Data Object Column in the Transaction Definition dialog box.

From the Transaction Definition dialog box, open the Trigger and Storage Parameters dialog box to specify the events that will initiate your transactions and timeout values. For more detailed information about transactions, see [Creating Transactions](#) on [page 67](#).

In a configuration that uses online edits, you can edit existing or create new transactions while the configuration is running. When you have pending edits, you can view the differences between the current and pending definitions on the Transaction Differences dialog box. If the pending edits that you created have caused pending edit alerts, you can view them on the **Pending Edit Alerts** dialog box. For a detailed description of the information displayed on these dialog boxes, see [Understanding Online Edits](#) on [page 79](#).

To verify multiple completed transactions, click **Verify** on the Configuration Checklist. You can also verify transactions individually from the Transaction Definition dialog box, which provides informational messages or warnings about the configuration.

In this section you will learn about the following:

- Viewing Configuration Properties
- Starting Configurations
- Stopping Configurations
- Starting and Stopping Connectors
- Monitoring Configurations

## Step 6: Verifying Transactions

### Miscellaneous

## Viewing Configuration Properties

### To view the properties of a configuration:

1. In the configuration item tree, select the configuration for which you want to view the properties.
2. Do either of the following:
  - On the menu, click **Configuration > Properties**.
  - Right-click the configuration name, and on the shortcut menu click **Configuration Properties**.

## Starting Configurations




The **Configuration Properties** dialog box appears. You can access all configuration level settings, such as enterprise connector options and error logging levels, from individual tabs in this dialog box.

Note: If you are starting a FactoryTalk Transaction Manager configuration, you must be logged into a Microsoft Windows account that has administrative privileges for all computers that are part of the FactoryTalk Transaction Manager system. This is required by the Microsoft Windows Service Control Manager to be able to start and stop FactoryTalk Transaction Manager services.

Only the Transaction Control Manager service or the FactoryTalk Transaction Manager service can run at one time. Further, the Transaction Control Manager service or the FactoryTalk Transaction Manager service can run only one FactoryTalk Transaction Manager configuration at a time. The configuration runs until it is stopped.

### To start a FactoryTalk Transaction Manager configuration do either of the following:

- In the Configuration tree, right-click the configuration you want to start, and on the shortcut menu click Start Configuration.
- In the Configuration tree, select the configuration you want to start,

and then on the toolbar click 



Note: If you are using a distributed configuration, make sure all remote computers that are used in the configuration are running and available before you start the configuration. If the software encounters a remote computer that is not running or available, it will proceed to the next running and available computer.

If your configuration includes FactoryTalk View SE, ensure that these products are running on the appropriate host computer and the project that is used by the configuration is loaded and running before you start the configuration.

Once you have started a configuration manually, you can set the configuration to start automatically.

### To set a configuration to start automatically:


1. Go to **Start > All Programs > Control Panel > Administrative Tools > Services**. The **Services** window appears.
2. From the list of services, select either the **FactoryTalk Transaction Manager Transaction and Control Manager** service or the **FactoryTalk Transaction Manager** service (depending on your configuration), and double-click the service. the **Properties** dialog box appears.
3. From the **Start-up** box, select **Automatic**.
4. Click **OK**.
5. Repeat steps 2 through 4 for each connector service in your configuration.



Note: The Transaction Control Manager service and the FactoryTalk Transaction Manager service cannot be set to Auto-start at the same time.

## Stopping Configurations

To stop a configuration do either of the following:

- In the Configuration tree, right-click the configuration you want to stop, and on the shortcut menu, click **Stop Configuration**.
- In the Configuration tree, select the configuration you want to stop, and then on the toolbar click .

You cannot stop the configuration by exiting FactoryTalk Transaction Manager or logging off from the computer. For more information, see [Advanced Topics](#) on [page 89](#).

## Starting and Stopping Connectors

To start an individual connector when a configuration is running, in the Configuration tree, right-click the connector, and on the shortcut menu click **Start Connector Service**.

To stop an individual connector when a configuration is running, in the Configuration tree, right-click the connector, and on the shortcut menu click **Stop Connector Service**.

## Monitoring Configurations

You can monitor transactions as they are executed while a configuration is running. Click Transaction Monitor on the toolbar to view a summary of cumulative activity organized by transaction.

The following columns are available in the summary:

Item	Description
Total	The total number of transactions that have been triggered.
Total	The number of transactions that have completed without errors.
Failed	The number of transactions that have failed.
% Passed	Passed/Total * 100.
Cached	The number of transactions currently in Transaction Cache files.
Database Passed	The number of transactions that have been successfully executed by the database.
Database Failed	The number of transactions that have experienced a database error.
Pending	The number of transactions that have been started, but are not completed or currently running.



Note: Click any column header, except Total, to sort the list. The list refreshes every 30 seconds, or when you click the column headers.

## Understanding FactoryTalk Transaction Manager External Files

FactoryTalk Transaction Manager generates several types of external files while creating and running a configuration. You can set the location of these files from the FactoryTalk Transaction Manager user interface. You can determine where the configuration files are stored by viewing the Configuration Properties dialog box. For more information on viewing configuration properties, see [Viewing Configuration Properties](#) on [page 45](#).

FactoryTalk Transaction Manager generates the following types of external files:

- **Configuration files (\*.dat)** - FactoryTalk Transaction Manager stores all information associated with a single configuration in a set of \*.dat files. The configuration files have fixed file names, so each configuration generates a set of identically named files. For this reason, configuration files must be stored in a unique directory.
- **Cached Transaction files (\*.rsl)** - FactoryTalk Transaction Manager transactions can update their target database directly from cached transaction files. These files contain completed transactions that are applied to the database as a group. The use of cached transaction files is set individually for each transaction, but the files are associated with an enterprise connector. To edit cached transaction file properties, open the Configuration Properties dialog box, select the Cache tab and double-click a connector to open the Enterprise Connector Options dialog box.
- **Log files (\*.log)** - Each of the FactoryTalk Transaction Manager services generates log files when the configuration is running. The level of error messages contained in these files is set at the configuration level. You can specify error log file parameters from the FactoryTalk Transaction Manager user interface by selecting **Define > Error Logging**. All log files can be viewed by selecting **View > Error Log Files**.
- **SQL files (\*.sql)** - FactoryTalk Transaction Manager generates these SQL files as a result of a failed connection or database error. To include this data in the database, use a database maintenance utility.
- **RSQ files (\*.rsq)** - FactoryTalk Transaction Manager uses these compressed files to backup configurations. An .rsq file is a result of a backup. You can restore an .rsq file by using the restore command.
- **RPB files (\*.rpb)** - These files are generated when you use the Time-series Data Compression enterprise connector. FactoryTalk Historian Classic uses these files to save uncompressed, partial block information.
- **RFB files (\*.rfb)** - These files are generated when you use the Time-series Data Compression enterprise connector. **FactoryTalk Historian Classic** uses these files to save compressed, full block information prior to storage in the database.



## Understanding FactoryTalk Transaction Manager Services

In this chapter you will learn about the following:

- [Introducing FactoryTalk Transaction Manager Services](#) on [page 49](#)
- [Control Connectors](#) on [page 49](#)
- [Enterprise Database Connectors](#) on [page 50](#)
- [Enterprise Application Connectors](#) on [page 51](#)
- [Enterprise Connector Options](#) on [page 51](#)
- [FactoryTalk Transaction Manager Service](#) on [page 52](#)
- [Transaction Control Manager Service](#) on [page 52](#)
- [Configuration Server](#) on [page 52](#)

### Introducing FactoryTalk Transaction Manager Services

FactoryTalk Transaction Manager is designed to run as several services. During design time, the FactoryTalk Transaction Manager user interface sends information to the Configuration Server which writes to the configuration files. At run time, the other FactoryTalk Security services run in the background of the computer(s) involved in the configuration, similar to other Microsoft Windows services. For more information about configuring services, refer to [Exploring the FactoryTalk Transaction Manager User Interface](#) on [page 33](#).

This chapter describes the types of control and enterprise connectors you must define during design time.

### Control Connectors

The control connector services manage the interaction between the industrial control system and the FactoryTalk Transaction Manager service in the FactoryTalk Transaction Manager. The control connector services communicate with the data server using the appropriate protocol. You can use the following control connectors:

- FactoryTalk Live Data.
- Generic OPC.

Rockwell Software recommends using the FactoryTalk Live Data control connector for most user applications. If your application requires the use of any other control connector, please review and consider the following sections before selecting an alternate control connector.

### FactoryTalk Live Data

The FactoryTalk Live Data control connector service is used to interface with data items provided by the FactoryTalk Live Data servers. The FactoryTalk Directory provides a common name space for factory automation products from Rockwell Software, allowing all applications to use the same naming

convention and giving you the capability to browse available data points. FactoryTalk Live Data provides services that allow the efficient transfer of high-speed manufacturing data between processes in the system.

In a configuration that uses online edits, the Transaction Control Manager service performs the duties of the FactoryTalk Transaction Manager service and inherits the functionality of the FactoryTalk Live Data control connector. For more information about performing online edits, see [Understanding Online Edits](#) on [page 79](#).

## Generic OPC

The Generic OPC control connector service can be used to interface with items provided by any OPC server that conforms to the OPC custom interface specifications. The Generic OPC connector is an OPC client that supports OnDataChange subscription callback using either:

- **IAdviseSink**– for OPC 1.0A-compliant servers.
- **IConnectionPoints**– for OPC 2.0-compliant servers.

The Generic OPC connector service tries to establish the IAdviseSink method, and then tries the IConnectionPoints method. Writing data to OPC items is performed using Asynchronous Writes. FactoryTalk Transaction Manager also supports Asynchronous Reads from Device as an option for OPC servers which support this method.

## Enterprise Database Connectors

The enterprise database connector services manage FactoryTalk Transaction Manager interaction between a database and the Transaction Control Manager service or FactoryTalk Transaction Manager service. Each enterprise database connector service can manage connections with multiple databases on multiple computers. You can use the following enterprise connectors:

- [Microsoft OLE DB](#) on [page 50](#).
- [ODBC](#) on [page 50](#).
- [Oracle OCI](#) on [page 51](#).

For transactions that update existing records, the data values that are used to look up a record in the database may update zero or more records in the database. If one or more records is selected, the transaction updates all of the selected records. If no records are selected, the transaction converts from Update to Insert mode, and the transaction is inserted into the database. This is not an error condition.

## Microsoft OLE DB

The Microsoft OLE DB connector allows you to browse a Microsoft SQL Server database without a DSN. Please note that the datetime2 data type is not supported on this connector.

## ODBC

An acronym for Open Database Connectivity. ODBC is a widely accepted API for database access that is based on the Call-Level Interface (CLI) specifications from X/Open and ISO/IEC APIs, and uses Structured Query Language (SQL) as its database access language.

## Oracle OCI

The Oracle Call Interface (OCI) enterprise database connector allows you to connect to database objects from an Oracle server. If you are using Oracle OCI in FactoryTalk Transaction Manager, you need to locally install either of the following Oracle services:

- SQL\*NET 2.3x or later
- Net8
- Oracle Net

You can then connect to an Oracle database.

## Enterprise Application Connectors

The enterprise application connector services manage FactoryTalk Transaction Manager interaction with the enterprise application connectors (Time-series Data Compression or FactoryTalk Metrics), and the Transaction Control Manager service or FactoryTalk Transaction Manager service.

You can use the following enterprise application connectors:

- Time-Series Data Compression.
- FactoryTalk Metrics.

## Time-Series Data Compression

The Time-Series Data Compression enterprise application connector is a service that compresses data from the control system using a lossless algorithm to conserve space. This connector can be configured and used only by FactoryTalk Historian Classic.

## FactoryTalk Metrics Enterprise Application Connector

The FactoryTalk Metrics enterprise application connector is only used with FactoryTalk Metrics. The FactoryTalk Metrics enterprise application connector can only be configured through the Service Console.

## Enterprise Connector Options

You can set additional options for enterprise connectors in the Enterprise Connector Options dialog box.

**To access the Enterprise Connector Options dialog box do either of the following:**

- In the Configuration Checklist dialog box:
  1. Click **Step2**. The **Connector Definition** dialog box appears.
  2. Click **Options**. The **Enterprise Connector Options** dialog box appears.
- In the Configuration tree:
  1. Right-click the selected configuration, and on the shortcut menu click **Configuration Properties**. The **Configuration Properties** dialog box appears.
  2. On the **Connectors** tab, double-click the appropriate enterprise connector. The **Connector Definition** dialog box appears.
  3. Click **Options**. The **Enterprise Connector Options** dialog box appears.

You can set the following options:

Item	Description
Number of real-time threads	Specifies the number of real-time threads used by this connector. Increasing this value permits multiple real-time transactions to execute simultaneously in the enterprise connection with each thread having its own database connection. Do not increase this value unless the real-time transactions are not executing to the database fast enough. To make a transaction use real-time threads, select the Use Real Time Thread option in the Transaction Definition dialog box. For more information see Step 5: <a href="#">Defining Transactions</a> on <a href="#">page 44</a> .
SQL buffer size	Specifies the size of the SQL buffer in kilobytes. The SQL buffer specifies the number of bytes necessary to build the command that will be executed. The default value only needs to be modified if database errors occur and the SQL file shows only a partial command.
Maximum transactions per file	Indicates that a file is sent to the enterprise connector for processing when Maximum Transactions Per File or Maximum Time Between Files value is reached, whichever comes first. Setting this value to a field greater than 1 allows the enterprise connector to use array inserts on databases that support them. Array inserts increase database performance by allowing multiple inserts/updates in a single database command.
Maximum time between files	Indicates that a file is sent to the enterprise connector for processing when the Maximum Time Between Files or Maximum Transactions Per File is reached, whichever comes first.

## FactoryTalk Transaction Manager Service

The FactoryTalk Transaction Manager service performs the following tasks:

- Controls the execution of all FactoryTalk Transaction Manager transactions.
- Collects and sends data to and from all connector services.
- Controls the scheduling and execution of the transactions.
- Controls data manipulation, if required.

## Transaction Control Manager Service

The Transaction Control Manager is a service that controls and executes FactoryTalk Transaction Manager transactions contained in a configuration, but with the additional functionality of the FactoryTalk Live Data control connector embedded in it. In a configuration enabled for editing, the Transaction Control Manager replaces the separate FactoryTalk Transaction Manager and control connector services.

## Configuration Server

The Configuration Server is a service that runs continuously to provide a single interface to the configuration files (with the **.dat** file extension) which make up the FactoryTalk Transaction Manager configuration. The Configuration Server simplifies access to the configuration files by filtering all changes to the files and by communicating with other FactoryTalk Transaction Manager services. A collection of all changes that affect a configuration is recorded in an audit trail (via either FactoryTalk Diagnostics or the Configuration Server **\*.log** file).

Since the service is always running, functions such as configuration diagnostics and remote file browsing are easier. One of the benefits of the

Configuration Server is consolidated file access. The Configuration Server is the focal point for all interactions with, and manipulation of the configuration files.



## Defining Data Points

In this chapter you will learn about the following:

- [Introducing Data Points](#) on [page 63](#)
- [FactoryTalk Live Data Data Points](#) on [page 56](#)
- [OPC Data Points](#) on [page 61](#)

### Introducing Data Points

Data points are specific data locations or registers in the control system that are made available to FactoryTalk Transaction Manager transactions. The software can read information from, and write to data points through the embedded FactoryTalk Live Data control connector (Transaction Control Manager service) or a control connector (FactoryTalk Transaction Manager service). The control connector then communicates with a data server (based on OPC or FactoryTalk) that communicates with the control system devices. Once defined, a data point can be used by multiple transactions.

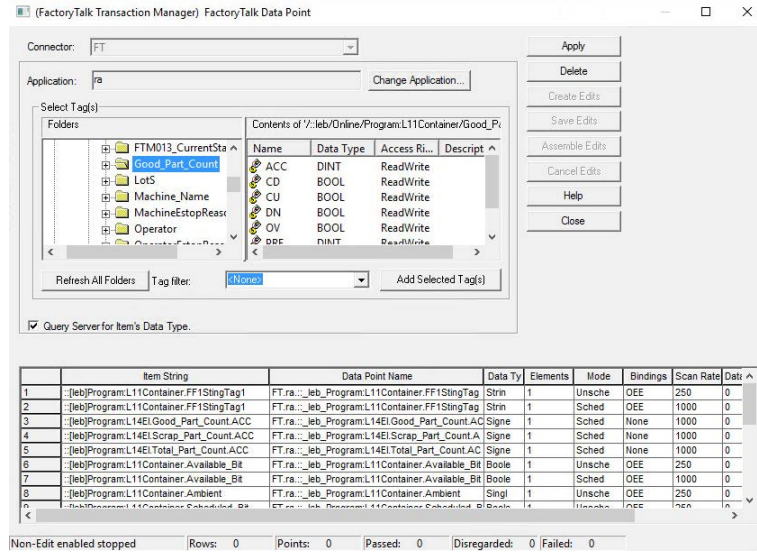
Using FactoryTalk Transaction Manager, you can assign the following attributes to a data point:

- Item string or address.
- Data point name.
- Data type.
- Number of elements.
- Mode and Scan rate.
- Retrieval timeout.
- Substitution options.

These attributes are assigned in **Step 3** of the Configuration Checklist by defining data points in the **Data Point Definition** dialog boxes. In these dialog boxes you can:

- Browse for data locations serviced by a data server.
- Add data points to the configuration.
- Edit existing data points.
- Delete unwanted data points.

The following figure shows a sample FactoryTalk Data Point dialog box.



Notice the following fields on the status bar at the bottom of the FactoryTalk Data Point dialog box:

Item	Description
Configuration status	The status of the current configuration.
Rows	The number of rows selected before you saved, assembled, or canceled pending edits.
Points	The number of data points that specific row represents. For example, two rows may be two different data points or they may be the current definition and the pending definition of the same data point.
Passed	The number of successful operations.
Disregarded	The number of rows on which FactoryTalk Transaction Manager does not allow the action to be performed. For example, selecting a row that does not have a pending edit and clicking Cancel Edits.
Failed	Operations that did not take place because of an internal error.

For more information on online edits, see [Understanding Online Edits](#), on [page 79](#)

## FactoryTalk Live Data Data Points

FactoryTalk Transaction Manager uses FactoryTalk Live Data to communicate with the FactoryTalk Live Data servers. With access to the FactoryTalk Directory, the FactoryTalk Transaction Manager service can browse available data items directly. This is the only method that can be used to transfer data to and from FactoryTalk View SE or FactoryTalk Linx, and is the recommended method to transfer data to and from RSLinx Classic.

In a configuration that uses online edits, you can use only FactoryTalk data points. For further details about using FactoryTalk data points in a configuration that uses online edits, see [Understanding Online Edits](#) on [page 79](#).

Rockwell Software recommends using the FactoryTalk Live Data control connector for most user applications. The majority of this chapter will focus



on information related to FactoryTalk Live Data data points. For information on all of the other data point types, see [OPC Data Points](#) on [page 61](#).

## Selecting a Collection Mode

You can select a collection mode for FactoryTalk data points. This section provides more detail about the following collection modes:

- [Scheduled](#) on [page 57](#)
- [Device Scheduled](#) on [page 57](#)
- [Unscheduled](#) on [page 57](#)

### Scheduled: Maintain the Current Subscribed Value

In the scheduled collection mode, each data point is continuously scanned at a configurable rate. The data server sends any change in value or quality to the control connector. The control connector retains the current value in a buffer, and provides it to the FactoryTalk Transaction Manager service when requested. The Transaction Control Manager service also buffers the data internally and uses it as needed.

The data server reads the value of the data point at the Subscription Scan Rate that is set in the **FactoryTalk Data Point** dialog box; if a new value is found, it is sent to the data client. If the value has not changed in the data server, no data is transferred. Thus network bandwidth is not used to transmit the same value between the data server and the control connector.

Typically, this collection mode is used to support a transaction that logs data constantly and rapidly (for example, logging a data point every second while an assembly line is running).

### Device Scheduled: Request the Current Value From the Device

In the device scheduled collection mode, the data points are not scanned, and the data is not buffered in the control connector or data server. The data server reads directly from the device. Each value is read from the controller only when the Transaction Control Manager service or FactoryTalk Transaction Manager service requests it from the control connector. The value is then retrieved from the controller and passed to the control connector.

Typically, this collection mode is used to support a transaction that logs data infrequently.

### Unscheduled: Send Subscribed Value Whenever It Changes

This collection mode is driven by the data server, and each data point is continuously scanned at the specified rate. When the server detects a change to a data point value or quality, it sends the value to the control connector, which passes it to the Transaction Control Manager service or the FactoryTalk Transaction Manager service. This collection mode is event-based, not time-based.



Note: Keep in mind that any unsolicited message sent by the controller that does not include a change in value or quality will never be sent to FactoryTalk Transaction Manager.

The Subscription Scan Rate specifies how fast new data can arrive. For example, if the Subscription Scan Rate is 150 milliseconds, the data server cannot send changes faster. Even if the underlying data value is changing

faster, the control connector can only see the current value every 150 milliseconds.

Typically, this collection mode is used for a transaction trigger that executes at a high or low transition, or exceeds a valid range.

## Consecutive Data Point and Data Block Support

In your application, you can quickly add multiple consecutive data points (for example, N7:0 through N7:6).

### To add multiple consecutive data points:

In the **FactoryTalk Data Point** dialog box, select the data points from the Contents of window.

- Click **Add Selected Tags**.

The data points are added to the FactoryTalk data point grid.

You can create a single data point with blocked data, such as a data point that has multiple consecutive elements. For example, if you want to create a data point with 10 elements, type **tag 1 .Data[0], L10**. This feature applies to pure data table files and not to structures (for example, **T3:0.ACC,L8** is not supported). The maximum size of a data block passed to the connector from a data point is 512 bytes for transactions without online edits, and 4096 bytes for transactions with online edits. If you defined a contiguous set of ASCII registers from a PLC processor, this data point contains a complete string (when used in a transaction). On the other hand, if you defined multiple consecutive elements with numeric data types in the contiguous registers of the PLC processor, you have an array. You can use the Parse() expression to extract each element from the block data.

This is an example of creating a data point with 10 elements.

	Item String	Data Point Name	Data Ty	Elements	Mode	Bindin
1	[[1]Program.L13Capper.FTM007_Shift_GoodParts	FT.app..._1_Program.L13Capper	Singl	1	Sched	OEE
2	[[1]Program.L13Capper.FTM006_Shift_ScrapParts	FT.app..._1_Program.L13Capper	Singl	1	Sched	OEE
3	[[1]Program.L13Capper.Running_State	FT.app..._1_Program.L13Capper	Boole	1	Unsche	OEE
4	[[1]Program.L13Capper.Available_Bit	FT.app..._1_Program.L13Capper	Boole	1	Sched	OEE
5	[[1]Program.L13Capper.Available_Bit	FT.app..._1_Program.L13Capper	Boole	1	Sched	OEE
6	[[1]Program.L13Capper.Running_State	FT.app..._1_Program.L13Capper	Boole	1	Sched	OEE
7	[[1]Program.L13Capper.BEventTrigger	FT.app..._1_Program.L13Capper	Boole	1	Unsche	OEE
8	[[1]Program.L13Capper.BEventTrigger	FT.app..._1_Program.L13Capper	Boole	1	Sched	OEE
9	[[1]Program.L15Label.Counter	FT.app..._1_Program.L15Label.C	Signe	1	Sched	OEE
10	[[2]fact1.DATA[0],L10	FT.app..._2_tag1.DATA_0	Signe	1	Sched	None



Note: This feature also applies to the Generic OPC control connector.

## Selecting Timeout Properties

Keep the following sections in mind when selecting timeout properties.

### Data Valid

Regardless of the trigger mechanism, once a transaction is started, the Transaction Control Manager service or FactoryTalk Transaction Manager service checks each required data point to determine if the value in its local cache is still valid.

A data point value is not valid if the data valid time has passed or is set to zero. The following calculation demonstrates the validity requirements.

$$\text{Data is valid} = \text{Time Received} + \text{Data Valid value} < \text{Current Time}$$

Once the data point value is no longer valid, the Transaction Control Manager or FactoryTalk Transaction Manager service requests the data point from a data server, and starts a timer for the retrieval of the data point. If the data point is not returned by the time specified in the data retrieval parameter, the rules of substitution are applied.

When a data point is unscheduled, it is never requested, and the transaction fails if the data valid time has expired. Unscheduled data points do not have substitution values. If a data point is to be used as a transaction trigger as well as in other transactions, set the data valid parameter to a value that is large enough to keep the value fresh. Alternatively, you can create another data point using the same address, and make it a scheduled data point so that the Transaction Control Manager FactoryTalk Transaction Manager service can request it as needed.



Note: An unscheduled trigger is considered to be fresh when it starts a transaction.

If the Transaction Control Manager or FactoryTalk Transaction Manager service starts a transaction, and determines that it has already requested a data point but not received a reply yet, it will not request the data point again; it will use the same value for both transactions when the value is received.

### Data Retrieval Timeout

The data retrieval timeout is measured from the transaction start to the data point arrival. The data point timeout may be affected if there is communication latency between any of the following:

- The Transaction Control Manager service and the data server
- The FactoryTalk Transaction Manager service and the control connector, as well as the control connector and the data server

An example of this latency type is using a device-scheduled data point in FactoryTalk, where the following actions are performed:

1. The Transaction Control Manager service or FactoryTalk Transaction Manager service requests the data point from the control connector.
2. The control connector sends a notification to the FactoryTalk Live Data server.
3. The FactoryTalk Live Data server sends a read request to the controller, which in turn returns the data value.
4. The FactoryTalk Live Data server sends the value to the control connector, which sends it to the Transaction Control Manager service or FactoryTalk Transaction Manager service.

If the data point is not returned by the time specified in the data retrieval parameter, the rules of substitution are applied.

## Selecting a Substitution Option

All data points need to have valid values to allow a transaction to complete successfully. If a data point is not retrieved and times out, the Transaction Control Manager service or FactoryTalk Transaction Manager service uses a substitution value. Unscheduled data points do not have substitution policies. If unscheduled data points are invalid, the transaction will fail.

### To select a substitution option:

1. In the **Data Point Definition** dialog box, right-click the data point, and select **Edit Selected Collection Parameters**.
2. Choose one of the following substitution options for scheduled data points:

Item	Description
No Substitution	Specifies that the transaction fails if valid data is not available for this data point. This occurs when the Transaction Control Manager service or FactoryTalk Transaction Manager service times out waiting for data, or when you have bad quality data.
Substitute Previous Value	Instructs the FactoryTalk Transaction Manager service to use the last good value for this data point.
Substitute Value	Allows you to specify the substitution value to use.

A substitution value of Null (default) causes the enterprise database connector to leave the value for the column empty. A data point that has a Null substitution policy causes a transaction to fail if the data point is used in an expression. A Null value is not a null string or a zero. It is a value that does not exist; therefore, the expression evaluator cannot use it to calculate a result.

## Preventing Stale and Mismatched Data

Stale data is data that no longer matches the value in the control system. Mismatched data refers to a set of data in which individual data values from different times were collected, and they are not synchronized. Several strategies exist to eliminate stale and mismatched data, depending on the transaction type. For scheduled transactions, data may be read while values are changing. For most applications, this should not be a concern (except for the case of high-speed data changes). If the data is changing at a high rate, you need to switch the transactions to unscheduled.

Unscheduled transactions offer better protection against stale and mismatched data via the use of ladder logic. The controller dictates when the data is read, and it can lock values into its registers prior to triggering the transaction. This helps prevent stale data from being read as long as the data valid time is set to zero for a non-trigger data point.

The best way to prevent stale or mismatched data is to use unscheduled data point blocks. All data is handled as a single unit, managed by the controller and parsed into separate units using the Parse function. This method may also reduce traffic on the controller network because the data is not scanned at a constant rate.

## Specifying Quality

For the FactoryTalk Live Data connectors, choose one of the bad quality options if you want the Transaction Control Manager service or FactoryTalk Transaction Manager service to use bad quality values. Otherwise, select **Use Substitution Option for Bad Quality** in the **Edit Collection Parameters** dialog box. If you do not select a bad quality value, the Transaction Control Manager or FactoryTalk Transaction Manager service will use one of the substitution policies listed earlier in this chapter when it receives a bad quality value for this data point.

If you choose to allow bad quality values, use the QualityOf() expression in the **Expression Editor** dialog box to bind the quality value to a column in your database. In addition, the bad quality status is saved to the Transaction Control Manager service log file or the control connector's log file (when using the FactoryTalk Transaction Manager service) at an error level. The QualityOf() expression provides the following quality values:

Server returns:	QualityOf() returns:
Bad (0)	1
Uncertain (1)	2
N/A (2)	3
Good (3)	0



Note: This function only applies to FactoryTalk Live Data and OPC data servers. If you are using the bad quality substitution option and the controller is switched off, set the **Data Retrieval** property to a value large enough so that the server has enough time to respond.

## OPC Data Points

OPC data points are typically used by the Generic OPC connectors, but can also be used by FactoryTalk Live Data connectors.

The OPC Specification is a non-proprietary technical specification that defines a set of standard interfaces based upon Microsoft's OLE/COM technology. The application of the OPC standard interface facilitates the interoperability between automation/control applications, field systems/devices and business/office applications. For further information, visit the OPC Foundation web site (<https://opcfoundation.org/>).

## Generic OPC Data Points

The Generic OPC control connector service is an OPC client. It communicates with an OPC server according to OPC standards. In the **OPC Data Points** dialog box, you can browse a list of servers supported by the Generic OPC control connector. If the OPC server supports tag browsing, you can browse for the desired OPC items to create data points. The Generic OPC control connector has full DCOM support, which allows the OPC server to reside on a remote PC provided that the DCOM security is set correctly.

## Defining Data Objects

In this chapter you will learn about the following:

- [Introducing Data Objects](#) on [page 63](#)
- [Enterprise Database Objects](#) on [page 64](#)
- [Enterprise Application Objects](#) on [page 64](#)
- [Enterprise Connector Error Handling](#) on [page 65](#)
- [Inserting and Updating Data Table Records](#) on [page 66](#)
- [Stored Procedures](#) on [page 66](#)

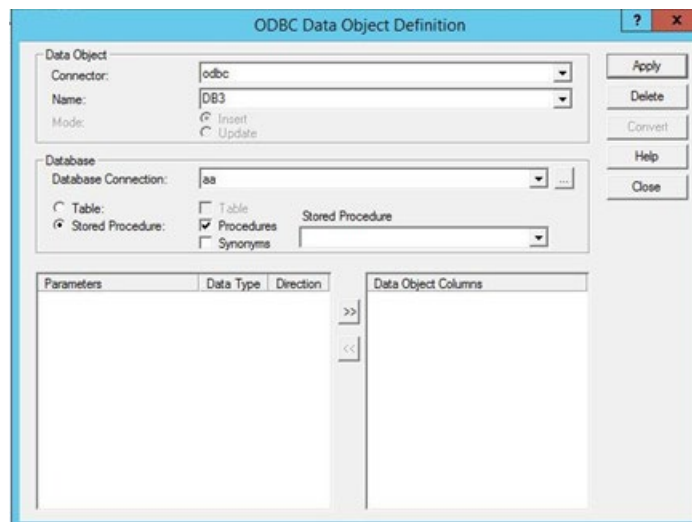
### Introducing Data Objects

A FactoryTalk Transaction Manager data object references a specific object in an enterprise system. For enterprise database connectors, this object can be a database table, view, or stored procedure. For enterprise application connectors, this object can be a FactoryTalk Metrics function. The objects can be used in transactions. A data object may contain many columns or parameters that are used as a unit. Enterprise database objects can also include an action (for example, inserting or updating a record).

In FactoryTalk Transaction Manager, you can define data objects in the **Data Object Definition** dialog box. To access the dialog box, click **Checklist** on the **Configuration** menu, and then click **Step 4**.

You can assign the following attributes to a data object:

- Data object name
- Enterprise system connection
- Mode (Insert or Update)
- Database tables or views
- Database columns or parameters



The target database tables, views, and stored procedures must already exist in the database to be used in data objects.



Note: If you modify target database tables, views, or stored procedures in the database after you create a data object in FactoryTalk Transaction Manager, you need to click **Apply** in the **Data Object Definition** dialog box so that the data object recognizes the database modifications.

Enterprise database connectors (ODBC, Oracle OCI, or Microsoft OLE DB) can communicate with databases located on other computers. Multiple data objects can reference the same database using a connection to the database system.

The following sections describe how to connect to database objects for the enterprise database connectors.

## Enterprise Database Objects

### Oracle Call Interface (OCI) Data Objects

The Oracle Call Interface (OCI) allows you to connect to database objects from an Oracle server. In the OCI Data Object Definition dialog box, you can create, modify, or delete FactoryTalk Transaction Manager data objects. When you create a data object, you need to enter a valid connection for the Oracle OCI server.

FactoryTalk Transaction Manager can communicate with a local or remote Oracle database via the Oracle network of client tools, such as SQL\*NET, Net8, or Oracle Net Services.



Note: You can install SQL\*NET/Net8 on the computer that contains the enterprise connector, and communicate with remote Oracle databases running on any operating system supported by Oracle.

### Microsoft SQL Server Data Objects

The Microsoft OLE DB connector service allows you to connect to database objects from Microsoft SQL Server and browse the database without entering a DSN. Please note that the datetime2 data type is not supported on this connector.

### ODBC Data Objects

The ODBC connector service allows you to connect to database objects using an ODBC 2.0 or a newer version of a compliant server. You need to use a valid system DSN, which is available to all Microsoft Windows services.

If the FactoryTalk Transaction Manager user interface resides on a computer that is remote from the computer(s) containing the enterprise connector, the system DSN list comes from the computer where the enterprise connector resides.



Note: Rockwell Software does not recommend using ODBC data objects if you have an Oracle database. Instead, use Oracle OCI data objects.

## Enterprise Application Objects

The following sections describe how to connect to database objects for the enterprise application connectors.



## FactoryTalk Metrics Data Objects

You can only connect to FactoryTalk Metrics data objects from the FactoryTalk Metrics Server. This connector is different from the other connectors because it preprocesses the data before sending it to the database. For more information, see the *FactoryTalk Metrics User Guide*.

## Enterprise Connector Error Handling

The following are the most likely FactoryTalk Transaction Manager error conditions that can affect the enterprise connector:

- **Lost connection with the enterprise connector** - The Transaction Control Manager service or FactoryTalk Transaction Manager service cannot communicate with the enterprise connector. This may occur because the enterprise connector service has stopped running, or the enterprise connector is located on another computer and the communication between the computers has been interrupted. The Transaction Control Manager or FactoryTalk Transaction Manager service will create cache transaction files for transactions that use historical logging (if the **Use Cached Transaction Files** check box is selected in the **Transaction Definition** dialog box). Therefore, you should store configuration cache files on the same computer as the Transaction Control Manager or FactoryTalk Transaction Manager service.
- **Lost connection with the enterprise database** - The enterprise connector cannot communicate with the database. This may occur because the database service has stopped running, or the database is located on another computer and the communication between the computers has been interrupted. The enterprise database connector stores the data to be reapplied when the database connection returns.



Note: If you select the **Use Cached Transaction Files** check box in the **Transaction Definition** dialog box, a \*.sql file is generated, which you need to manually apply to the database. The software keeps generating \*.rsl files until the connection returns. FactoryTalk Transaction Manager does not process any further \*.rsl files if there is no connection.

To access diagnostic information about the currently running configuration, click **Transaction Monitor** on the **View** menu while the configuration is running. The **Transaction Monitor** dialog box appears.

Transaction Name	Total	Passed	Failed	% Passed	Cached	Database Passed	Database Failed	Pending
Totals:	28	28	0	100.00 %	0	18	11	0
FT_Trans	28	28	0	100.00 %	0	18	11	0

The following columns display failed transactions:

- The **Failed** column - Lists transactions that were not successfully processed.
- The **Database Failed** column - Lists transactions that were processed correctly, but could not be applied to the database due to a database error.

For more information on error conditions that can affect enterprise connectors, as well as how FactoryTalk Transaction Manager handles transaction types and storage methods, see [Creating Transactions](#) on [page 67](#).

## Inserting and Updating Data Table Records

When logging data to a table, you can use FactoryTalk Transaction Manager to directly insert data (creating new records in the table), or update existing data in the table.

In the **Data Object Definition** dialog box, the **Insert** mode is selected by default. When you select the **Update** mode, some of the data points may be used as criteria for selecting rows to be updated, while other data points will be used to update the values in the selected columns. If no rows match the given criteria, the data will be inserted into a new row.

## Stored Procedures

A stored procedure is a user-defined function or program that is executed in the database. It can consist of any of the components of a structured language that enable you to define data behavior. A stored procedure can be simple like a single select command, or complicated like validating all data before it is inserted into a table.

A stored procedure works like a function that is stored in a database. Most databases provide a comprehensive stored procedure language that combines the data query capabilities of SQL and some kind of procedural control (e.g., the "If...Then" statement). As most kinds of function calls, a stored procedure can have both inputs and outputs. In a transaction that is connected to a stored procedure, the values bound to inputs are collected from the control system, and the outputs are returned to the control system.

When you select the **Stored Procedure** option in the **Data Object Definition** dialog box under **Database**, FactoryTalk Transaction Manager queries the database for all stored procedures accessible by the chosen database user account, and they appear in the **Stored Procedure** list. When you select a stored procedure, the input and output parameters are displayed under **Parameters**. Since all parameters for a stored procedure are required, the software automatically adds them under **Data Object Columns** when they are selected.

## Creating Transactions

In this chapter you will learn about the following:

- [Introducing Transactions](#) on [page 67](#)
- [Transaction Types](#) on [page 68](#)
- [Transaction Timeout](#) on [page 71](#)
- [Transaction Completion](#) on [page 72](#)
- [Using the Expression Editor](#) on [page 73](#)
- [Transaction Trigger and Storage Options](#) on [page 75](#)

### Introducing Transactions

FactoryTalk Transaction Manager transactions move data between the control system and the enterprise system. Binding is the process of mapping:

- A column in a database table to a data point in a control system, expression, or a literal string.
- A parameter in a stored procedure to a data point, expression, a literal string, or a null value.

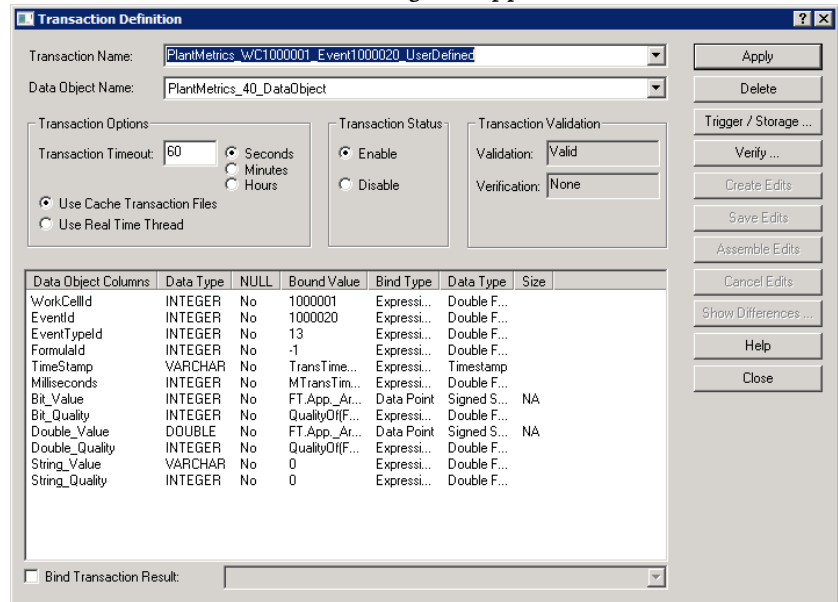
A configuration can contain any number of transactions, each running independently based on a trigger event. Although the number of transactions in a configuration is unlimited, you cannot exceed the tag count for which you are licensed (see Activation Options). In a configuration that uses online edits, you can edit existing or create new transactions while the configuration is running. For more information, see [Understanding Online Edits](#) on [page 79](#).

You can assign attributes to a transaction in the **Transaction Definition** dialog box.

#### To access the dialog box:

1. On the **Configuration** menu, click **Checklist**.  
The **Configuration Checklist** dialog box appears.
2. Enter the configuration name and then click **Step 5**.

The **Transaction Definition** dialog box appears.



In the dialog box, you can assign the following transaction attributes:

- Transaction name
- Transaction timeout
- Data object name
- Bindings of data object elements to data points/expressions
- Trigger/Storage
- Transaction (cached or real-time) and storage options

## Transaction Types

FactoryTalk Transaction Manager supports two types of transactions, regardless of the trigger mechanism:

- Unidirectional transactions
- Bidirectional transactions

For either transaction type, you can optionally specify the transaction result to be bound back to a control system.

## Unidirectional Transactions

Unidirectional transactions are transactions that use information from the control system to add records to a database table, or to update the contents of existing records. They do not return data to the control system.

Unidirectional transactions are commonly used to log production data to a database, including:

- The monitoring of performance
- The sampling of quality analysis
- The collecting of real-time production information
- The tracking of material consumption
- The product tracking
- The reporting of the end of job/batch/shift

Unidirectional transactions are the simplest transaction type. They can be used to perform the database Insert or Update commands. They can also be

used to perform simple stored procedures, provided no output parameters or return codes exist. In most applications, unidirectional transactions account for the majority of transaction volume. Data may be collected frequently from a large number of data points.

## Bidirectional Transactions

Bidirectional transactions in FactoryTalk Transaction Manager take data from the control system, and then call a stored procedure, which exercises some logic and provides the software with output values that can be written back to the control system. It is possible to set up bidirectional transactions so that the data is not sent to the database, but downloaded from the database to a control system.

Bidirectional transactions are the most powerful transaction type, because they allow transactions to interact with a database stored procedure.

Bidirectional transactions let you perform:

- The downloading of product parameters.
- The dynamic routing.
- The dynamic production scheduling.
- The controller centralizing.
- The production floor interacting.
- The warehousing of automated storage and retrieval.

A bidirectional transaction with input/output bindings implements a data transfer from the enterprise system and sends it to the control system, binding the input and output parameters of the database stored procedure. The control system data serves as an input to a stored procedure. The results of a stored procedure can be written back to data points in the control system (enabling the creation of sophisticated transactions that allow a high degree of interaction between a database system and the factory floor).

For transactions with output bindings, the **Real Time Thread Storage** option needs to be selected in the **Transaction Definition** dialog box, because the procedure or method must complete before the data can be returned to the Transaction Control Manager service or FactoryTalk Transaction Manager service. FactoryTalk Transaction Manager allows you to specify the number of database threads used by a connector to improve performance.

When creating a transaction using input/output bindings, parameters are bound in the same manner as columns in a table. The input data points are collected, expressions are evaluated, and the procedure is called. If the procedure succeeds, output parameters are written to the control system.

Transactions with input/output bindings use the following parameter types:

- **Input parameters.** In a stored procedure, these parameters must have a binding (a data point, expression, or a null). Right-click a data object parameter and select **Bind Data Point** or **Bind Null Value**. A transaction must have all input parameters bound to it before it can be

enabled. Binding a null value to an input eliminates the need to gather a value from the database.

- **Output parameters.** These parameters do not require a binding. The results of the bound value are displayed in the **Bound Value** column. If an output parameter is not bound, or is bound to a null, the value is ignored and the column is empty. A procedure must succeed before output parameters can be returned to the control system. Normally, you bind outputs to a data point in the control system; upon a successful completion of a transaction, a value is written from the stored procedure to the data point.
- **Input/output parameters.** These parameters are handled as a single binding unless they are separated using the **Separate Input/output** option. This option allows separate data points to be bound to the same procedure parameter. This implies that the address from which the input parameter is derived is different from the address to which the output parameter is written. This also implies that if an input/output parameter is used only for the output, then the input portion can be bound to a null value. Leaving an input/output parameter bound to a single data point causes the value to be read prior to executing the procedure, and the output value to be written upon the transaction completion.

The Microsoft SQL Server RETURN\_CODE contains data that can be bound as an output to a stored procedure. This value is only available if the procedure executes successfully. A successful return code does not guarantee a successful transaction as the transaction is not yet complete.

## Bidirectional or Unidirectional Transactions With Transaction Bindings

A bidirectional or unidirectional transaction with a transaction result binding implements a data value that is written back to the control system; the data value allows you to determine whether the transaction has been completed successfully. The control system can then take appropriate actions based on the success or failure of the transaction. For example, consider a high-liability manufacturing environment where a verifiable quality record is a requirement for each item produced. At various steps in the manufacturing process, a transaction may send a test result to be stored in the central database. If the test result is not recorded successfully, the part on the production line may become worthless.

The transaction result, which notifies the control system that the transaction was successful, is sent variably based on the transaction type. In a real-time transaction (where data flows straight to the database), the transaction result is sent upon the data being successfully logged into the database. However, in the case of a cached transaction, where a cache is used to ensure data integrity, the transaction result is sent once the data has been logged into the cache, and not necessarily into the database. The transaction result is used to confirm that the test result was recorded. If it indicates that the test result was not recorded, the control system can respond appropriately by alerting the

operator, changing the part's routing, or retrying the transaction. The following are two examples:

- Validated data logging
- Closed-loop quality tracking

The **Transaction Result Binding** option allows a transaction to return a transaction result code to the control system. To enable the option, select the **Bind Transaction Result** check box in the **Transaction Definition** dialog box, and then in the list select a data point to accept the transaction result code. The control system should take the appropriate action, depending on the transaction result.

The transaction result code is a 16-bit integer. Bit 0 is the least significant, Bit 15 is the most significant. The transaction result code consists of the following:

Item	Description
Bit 0: the Done Bit	A value of 1 indicates the transaction has completed, 0 indicates that it has not. No information is given on the transaction's success or failure.
Bit 1- The Error Bit	A value of 1 indicates that there was an error, 0 indicates that no errors occurred.
Bits 2 through 15- The Error Code	If Bit 1 contains 1, the bits contain the error code. Otherwise, the bits are zeros. If you have chosen to use the transaction result code, you can run the FactoryTalk Transaction Manager error utility from the Startup menu. This eliminates the need to decode the integer to determine the FactoryTalk Transaction Manager error code.

### To convert the error code to a proper error number:

1. Click **Start > Programs > Rockwell Software > FactoryTalk Transaction Manager > Error Messages**.

The **FactoryTalk Transaction Manager Error Messages** dialog box appears.

2. Enter the transaction result code number (binary or decimal).
3. Select **Bind Transaction Result Error**.
4. Click **Apply**.

For example, a transaction result code of 0000000000000001 (binary) or 1 (decimal) indicates that the transaction has been completed without errors.

If the transaction result code is 000001011101111 (1519 decimal), the FactoryTalk Transaction Manager error code is 33147.

## Transaction Timeout

The transaction timeout parameter specifies how long the Transaction Control Manager service or FactoryTalk Transaction Manager service should wait for a transaction to complete. The timeout for an unscheduled transaction does not affect other copies of the same transaction, because multiple copies of an unscheduled transaction can be executed at the same time.



Note: Setting the transaction scan rate for a scheduled transaction lower than the transaction timeout may cause the second transaction to fail, because only one scheduled transaction can run at a time.

## Transaction Completion

It is important to determine when the Transaction Control Manager service or FactoryTalk Transaction Manager service considers a transaction to be complete, because the services run only one occurrence of a scheduled transaction at a time. The next occurrence cannot start until the current transaction is complete. Additionally, if a transaction has a transaction result binding, the transaction result is written when the transaction completes.

## Cached Transactions

With the **Use Cache Transaction Files** option selected in the **Transaction Definition** dialog box under **Transaction Options**, unidirectional transactions complete as soon as the data is written to the cache file. The data is not stored to the database yet, but it is on the disk.

## Real-time Transactions

With the **Use Real Time Thread** option selected in the **Transaction Definition** dialog box under **Transaction Options**, unidirectional transactions complete when the Transaction Control Manager service or FactoryTalk Transaction Manager service receives a reply from the enterprise connector that the data has been stored. This means that the time necessary for the enterprise system to store the values is included in the transaction completion time. This may cause the transaction to time out if the Transaction Control Manager service or FactoryTalk Transaction Manager service has not received the reply from the enterprise connector in the allotted time. The transaction timeout does not determine if the data was logged to the database. The transaction can time out prior to sending the data to the database or after the data was successfully stored.

The table below shows how FactoryTalk Transaction Manager handles each transaction type and storage method.

Transaction type	Transaction storage method	Lost connection with enterprise connector	Lost connection with enterprise system
Unidirectional	Cached transaction files	Cache files are applied when the connection is restored.	One cache file can be converted to an .sql file and returns Database Failed. Remaining cache files are processed when connection is restored.(1)
Unidirectional	Real time thread	The transaction fails and transaction data is lost.	Transaction returns Database Failed, transaction data is written to an .sql file.(2)
Bidirectional	Real time thread	The transaction fails and transaction data is lost.	Transaction returns Database Failed, transaction data is written to an .sql file.(2)
Transaction Result Binding	Cached transaction files	The transaction data is written to a cache file. The transaction returns a successful result to the controller. The cache files are processed when the connection is restored.	Transaction returns a successful result. One cache file can be converted to an .sql file and returns Database Failed. Remaining cache files are processed when the connection is restored.(1)
Transaction Result Binding	Real time thread	The transaction fails and transaction data is lost. The failure is sent to the controller.	Transaction returns Database Failed, transaction data is written to an .sql file.(2)

(1) If multiple database connections are defined in the configuration, the cache files continue processing. All data for the failed connection is written to an .sql file.

(2) Data is stored in separate .sql files, depending on whether the **Use Real Time Thread** or the **Use Cache Transaction Files** option was selected in the **Transaction Definition** dialog box. This allows you to recover the data.



For more information on the enterprise connector error handling, see [Defining Data Objects](#) on [page 44](#).

## Transactions With Bound Transaction Results

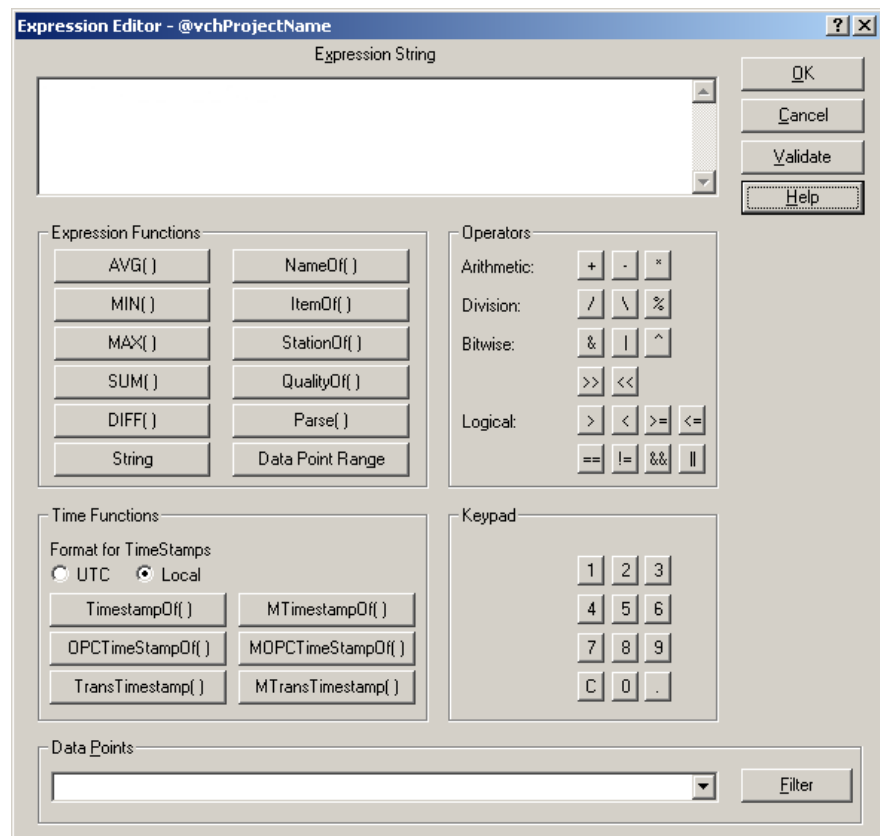
Bidirectional transactions that use a transaction result binding are not complete until the Transaction Control Manager service or FactoryTalk Transaction Manager service receives a reply from the control connector that the bound result was written. If a transaction times out after the bind transaction result is sent to the control connector, all data is moved successfully to the controller, and a message indicates that the transaction failed.

## Database Triggers

Database triggers are functions that are executed by the database whenever the triggering operation occurs. For example, a trigger can be set in such a way that whenever a value is inserted into a table, the data is verified, and then another value is updated with the verified data. The amount of time necessary to process the trigger and the associated function is charged against the transaction timeout. In this case, the database insert does not return control to the enterprise connector until the insert and its triggered function are complete. In other words, a real-time transaction is considered complete when the FactoryTalk Transaction Manager transaction and any database triggers caused by the transaction are complete.

## Using the Expression Editor

Use the Expression Editor to define calculations or formulas with mathematical operators and functions that can be bound to a data object. To display the Expression Editor, right-click a data object in the **Transaction Definition** dialog box, and then select **Bind Expression**.



## Logical and Mathematical Operators

Mathematical operators define simple expressions, which perform calculations that are evaluated after all data has been collected (or substituted). There is also a set of operators for both bitwise and logical operations. Once an expression has been defined, the syntax and semantics are checked to determine if it can be evaluated at runtime. After the data points have been collected, their current values are used to evaluate the expression. The results are then passed to the database for processing.

## Time Functions

The Expression Editor provides several functions for storing the current time. Select time functions can be expressed in either Coordinated Universal Time (UTC) or system local time.

The `TimestampOf()` function logs the time when a data point is read. This may differ from the time the transaction executed because the data point may have a data valid time of greater than zero.

The `TransTimestamp()` function returns the transaction execution time. This may differ from the time the data was inserted into the database, because completed transactions may be buffered in cached transaction (\*.rsl) files.

The `MTimestampOf()` and `MTransTimestamp()` functions log the millisecond portions of the two previous times. These can be stored in separate columns, which permits the accurate trend analysis for databases that do not store time values to the millisecond.

The `OPCTimeStampOf()` and `MOPCTimeStampOf()` functions return an OPC timestamp value that indicates when the OPC data server (or FactoryTalk data server) received (or read) the data from the controller. If the data server is RSLinx Classic, this is the time when RSLinx Classic provided data to FactoryTalk Transaction Manager. If the data server is FactoryTalk Linx, this is the time when FactoryTalk Linx acquired the data from the controller. This value is accurate to the nearest second.

## Data Point Range and Advanced Functions

The Expression Editor supports a data point range syntax that compares a data point value over a series of transactions. Each time a transaction runs, a new value is added to the data range for a given transaction. The expression is then evaluated using the range of values. This allows the expression to calculate an average of the previous 10 transaction values (`avg(datapoint[0,9])`), as well as calculate the minimum (`min`) or maximum (`max`) value of a data point over several transactions.



Note: You must run the transaction to reflect the changes in the historical values. However, you do not need to store the results.

Using the Store on Every N Transactions option, you can collect the data needed for an average, but not store the data to the database. If a transaction executes every second but only stores its data every 60 times, and there is an `avg(datapoint[0,59])`, the value that is stored once a minute is the average of the values taken every second.

In a running configuration, if you make changes to transactions (or their bound data points) using the Data Point Range function in conjunction with the Avg function in an expression, the transaction resets, or behaves as though it is starting for the first time when you assemble the pending edits. For more information, see [Understanding Online Edits](#) on [page 79](#).

## Parse Function

The Parse function in the Expression Editor ensures that all the data for a transaction is synchronized. The input for the Parse function is a block of data, and the output is a parsed subset. This allows the control system to manage all the data into a single data point, which can be sent to the control connector using an unsolicited message. The control connector then sends the data as a single unit to the Transaction Control Manager service or FactoryTalk Transaction Manager service. The service uses the data point arrival as a trigger, and then parses out the individual values as needed. Since the control system gathered all of the data into a single block and the block was sent to the service as a single unit, it is synchronized. The Parse function can also be used on scheduled data points and in scheduled transactions to separate data values from a single data point.

## Transaction Trigger and Storage Options

The screenshot shows the 'Trigger And Storage Parameters' dialog box. It is titled 'Trigger And Storage Parameters' and has a question mark icon and a close button (X) in the top right corner. The dialog is divided into two main sections: 'Transaction Triggers' and 'Transaction Stores Data'.

**Transaction Triggers:**

- On Scheduled Event:**
  - Starting Event: SYS\_STARTUP
  - Stopping Event: SYS\_SHUTDOWN
  - Scan Rate: 1
    - Seconds
    - Minutes
    - Hours
- On Unscheduled Event:**
  - On High Transition
  - On Low Transition
  - On Change
    - Positive Delta Value to Equal or Exceed: 0
    - Negative Delta Value to Equal or Exceed: 0
    - Seconds
    - Minutes
    - Hours
  - Scan On High, Rate: 1
    - Seconds
    - Minutes
    - Hours
  - Use Bad Quality Unscheduled Events
  - Ignore the First Unscheduled Event
- On External Application Trigger**

**Transaction Stores Data:**

- On Completion**
- On True Expression:** [Empty text box]
- On Number of Completed Transactions:** 1
  - Seconds
  - Minutes
  - Hours
- On Data Change and/or Rate:** 1
  - Seconds
  - Minutes
  - Hours

Buttons: OK, Cancel, Help

The Trigger and Storage Parameters dialog allows you to select from the following options the type of event you want to initiate your FactoryTalk Transaction Manager transaction.



Tip: If a configuration not using online edits is running, you cannot make the following changes.

- On Scheduled event

The transaction is initiated by an internal timer that can be set using the following parameters:

Starting event: The transaction starts when the specified scheduled event occurs.

Stopping event: The transaction stops when the specified scheduled event occurs.

Scan rate: The transaction executes at the specified scan rate between the starting event and the stopping event (inclusive).



Note: If the online enabled configuration is running, you can change the type of event, but not its definition. However, you can change the scan rate.

- On Unscheduled event

The transaction is initiated by an external event such as a change in value of a controller data point.

The following options are available:

On High Transition: The transaction is triggered only when the value changes from zero to not-zero. Transaction starts when the specified scheduled event occurs.

On Low Transition: The transaction is triggered only when the value changes from not-zero to zero.

On Change: The transaction is triggered when a positive or negative delta is exceeded.

Scan On High Rate: The transaction is triggered at the specified rate when the data point value is any number other than 0. A scan rate of 0 is invalid.

If you are using Scan On high Rate and you want to log the trigger in the same transaction, create a new data point pointing to the same tag, set the mode to scheduled, and then use the scheduled data point in the transaction binding.

A controller can send an unsolicited message to RSLinx Classic and have it trigger a FactoryTalk Transaction Manager transaction. This allows the controller complete control over when the transaction is triggered.

Ignore the First Unscheduled Event: Using this option in a running, edit enabled configuration may cause unexpected results. See the Known anomalies in the Release Notes for further details.

- On External Application Trigger

The transaction is initiated by an external source, such as another application program. To use an external source as a trigger, you must use a FactoryTalk Transaction Manager API function call from an application written in Visual Basic, Powerbuilder, or Visual C++.

In the Transaction Stores Data group, you may select one of the following options for storing data to the database.

- On Completion

The transaction is sent to the enterprise connector when it has completed. This is the default operating mode.

- On True Expression

The transaction is sent to the enterprise connector when the result of an expression is true. This field allows you to use the expression editor to build an expression that can be validated as true (any number other than 0) or false (0). For example, check a data point that is equal to a constant.

- On Number of Completed Transactions

The transaction is sent to the enterprise connector after a predetermined number of transactions occur. This feature, which is useful in trending calculations, saves the first few transactions in memory but does not write them to the database. This feature is intended to be coupled with any transactions that use the Data Range function to examine values during a specified time period. With this capability, you can write data for an average of a number of transactions.

- On Data Change and/or Rate

The transaction is sent to the enterprise connector when the value of the data point changes or a specified time expires. This feature allows you to design a transaction that monitors a set of conditions every few seconds, but only log data to the database when a change is detected. Only new values are sent to the enterprise connector, which reduces the number of records in the database table. This function also provides a "keep alive" timer-based backup that states that even if the values have not changed, the database is updated periodically at a rate determined by the user. You can write the value after a specified amount of time to indicate that the data is still active.



## Understanding Online Edits

In this chapter you will learn about the following:

- [Introducing Online Edits](#) on [page 79](#)
- [Understanding Online Edit Concept](#) on [page 79](#)
- [Creating a Configuration That Uses Online Edits](#) on [page 83](#)

### Introducing Online Edits

The ability to change data points and transactions in a running FactoryTalk Transaction Manager configuration is known as online edits. The use of online edits allows continuing the data collection in your automation system, while you add new or modify existing data points and transactions.

Online edits include current and pending edits. A current edit reflects how a data point or a transaction is currently defined in the running configuration. A pending edit reflects a new definition of a data point or transaction after you change and save it (before it has been assembled). Pending edits do not affect the currently running configuration until they are assembled. When assembled, the pending edits replace the current definition (i.e., become the new current definition).

When discussing online edits, you need to understand the difference between how data points and transactions are defined in the currently running configuration, and what changes take place when you assemble.

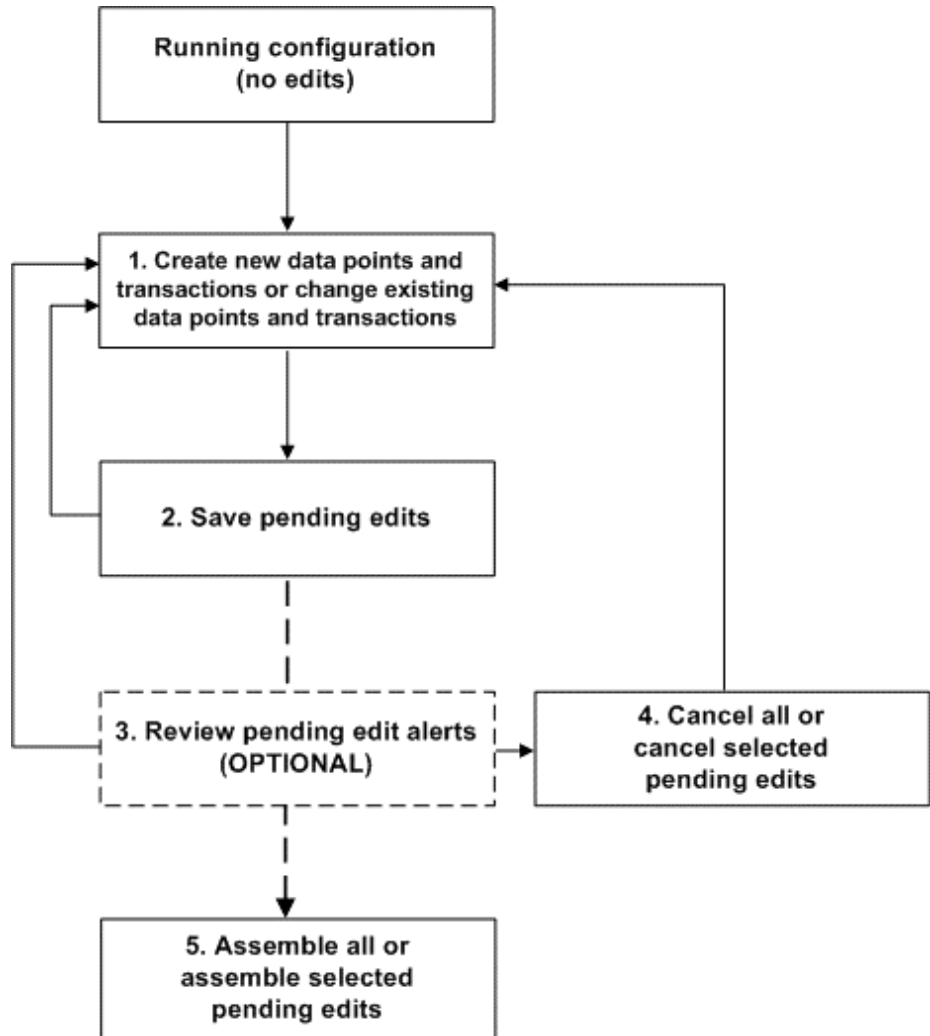
### Understanding Online Edit Concepts

Item	Description
Configuration that uses online edits	A configuration you can change while it is running.
Current definition	The definition of a currently running transaction or data point.
Pending definition	The changes made to data points or transactions in a running configuration that uses online edits. Pending edits must be saved before they can be assembled. Pending edits do not affect the running configuration until they have been assembled.
Assemble pending edits	The process of changing the running, current definition of a transaction or data point in a configuration that uses online edits to the pending definition. You cannot assemble pending edits until you save them first.
Cancel pending edits	The process of removing pending edits made to data points or transactions. Since pending edits do not affect the running configuration until they have been assembled, the definition of a data point or a transaction reverts to the current definition, and there is no effect on the running configuration.
Pending edit alerts	Informational messages that describe the side effects of the pending data points or transaction edits made on running transactions.

Transaction Control Manager	Similar to the FactoryTalk Transaction Manager service, but with the additional functionality of the FactoryTalk Live Data control connector embedded in it. In a configuration that uses online edits, the Transaction Control Manager service replaces the separate FactoryTalk Transaction Manager service and FactoryTalk Live Data control connector services.
-----------------------------	---

## Online Edits Workflow

The following diagram illustrates the multi-step process of changing a configuration that uses online edits.



The running configuration is not affected until the end of the process when the pending edits are assembled. Saving, reviewing, or canceling pending edits prior to assembling only affects the configuration files, not the running configuration itself.

## Configuration That Uses Online Edits

A configuration that uses online edits allows you to add new data points and transactions, as well as change existing data points and transactions while it is running. These changes are referred to as pending edits. In defining configurations that use online edits, you automatically use the Transaction Control Manager, which communicates exclusively with FactoryTalk Live Data data servers.

In a running configuration that uses online edits, you **can**:



- Create new and modify existing data points.
- Create new and modify existing transactions.
- Enable and disable transactions.
- Save and assemble pending edits.
- View differences between current transaction definitions and transactions with pending edits.

In a running configuration that uses online edits, you **cannot**:

- Modify configuration properties.
- Modify error logging properties.
- Add new, modify, or delete existing enterprise connectors.
- Modify enterprise or control connector properties.
- Use any control connector except FactoryTalk Live Data.
- Modify database connection properties.
- Add new, modify, or delete existing data objects.
- Modify a starting or stopping event definition.
- Change data point names or modes (scheduled, unscheduled, or device-scheduled).
- Delete data points.
- Delete transactions (although you can disable them).



Note: If you have a running configuration with pending edits and you stop it, you cannot make any further changes to the items with pending edits until you cancel or assemble the pending edits.

## Assembling Pending Edits

Assembling pending edits is the process of making the running configuration aware of the changes made to data points or transactions. The process may affect different types of pending edits, depending on the location in which they are assembled. To start assembling pending edits, you need to save them first; otherwise, a warning message appears.

You can assemble pending edits in the following locations:

- The main FactoryTalk Transaction Manager window.  
When assembling pending edits from the toolbar in the main FactoryTalk Transaction Manager window, all pending edits for transactions and data points are assembled.
- The **FactoryTalk Data Point** dialog box.  
Assembling in this dialog box affects only the selected data point rows and data point pending definitions. Transaction pending definitions are not assembled.
- The **Pending Transaction Definition** dialog box.  
Assembling in this dialog box affects only the pending definitions for the transaction you are currently viewing. Data point pending definitions are not assembled.

- The **Pending Edit Alerts** dialog box.  
When assembling transaction or data point pending edits that have caused side effects you may not be aware of, FactoryTalk Transaction Manager prompts you to review the pending edits. If you click **Assemble All** in the **Pending Edit Alerts** dialog box, all data point and transaction pending edits are assembled (including the ones that have not caused alerts), and they are no longer displayed in the dialog box.

For more information about assembling pending edits, refer to the FactoryTalk Transaction Manager online help.

## Canceling Pending Edits

Canceling pending edits is the process of removing pending edits made to data points and transactions in a running configuration that uses online edits. Canceling pending edits does not affect the running configuration. Data points or transactions return to their original definitions.

You can cancel pending edits in the **FactoryTalk Data Point**, **Pending Transaction Definition**, and **Pending Edit Alerts** dialog boxes. To cancel pending edits, you need to save them first. You cannot cancel pending edits if they are already assembled.

## Pending Edit Alerts

Informational messages describing that transactions with state information will lose the state information when the associated data point or transaction is assembled.

## Occurrence Conditions of Pending Edit Alerts

Some transactions require the state information for their correct execution. If you make some changes to the transactions or the bindings they use, the transactions reset or behave as though they are starting for the first time when you assemble pending edits. FactoryTalk Transaction Manager warns you that the transaction will be reset by displaying a pending edit alert message.

A transaction requires the state information if it has any of the following characteristics:

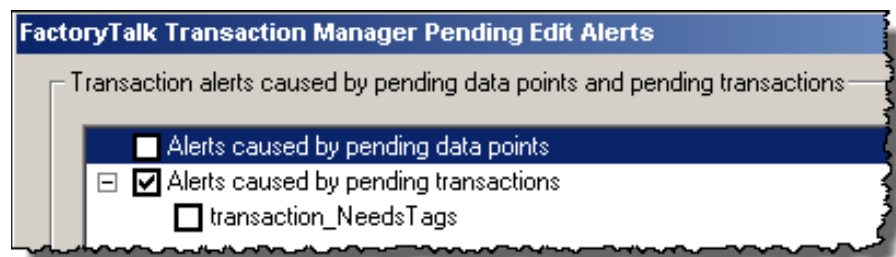
- Ignore First Unscheduled Event
- Transactions Stores Data On Number of Completed Transactions
- Transactions Stores Data On Data Change and/or Rate
- An expression using the DIFF function
- An expression using the Data Point Range function used in conjunction with the Min, Max, or Avg functions

A pending edit alert occurs if your enabled transaction includes any of the characteristics listed above, and you:

- Perform any of the following tasks:
  - Trigger type
  - Unscheduled event data point trigger
  - Add a binding
  - Delete a binding
  - Data point to expression

- Expression to data point
- Data type of binding
- Data point in a binding
- Expression in a binding
- Order of bindings.
- Merge input/output parameters in a binding.
- Separate input/output parameters in a binding.
- Change any of the following data point parameters:
  - Number of elements in an array
  - Size of string
  - Address of a data point
  - Data type of a data point

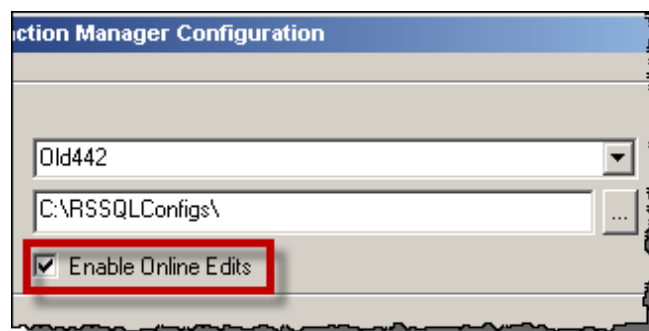
In the **Pending Edit Alerts** dialog box, you can review the transactions that have pending edit alerts. The transactions which do not have alerts are not displayed in the dialog box. You can assemble or cancel the selected transactions or all the transactions in the configuration (even the ones not displayed in the dialog box).



## Creating a Configuration That Uses Online Edits

To create a configuration that uses online edits:

- Use the FactoryTalk Live Data control connector exclusively
- Select the **Enable Online Edits** check box



## Editing Data Points In a Running Configuration That Uses Online Edits

You can edit data points (or create pending edits) in a running configuration by adding new data points or editing existing data points in the **FactoryTalk Data Point** dialog box. The FactoryTalk Connector and Application fields cannot be edited.

## Adding New Data Points

To add new data points, navigate to the correct area in your FactoryTalk Directory in the Select Tags group and double-click to open the folder/area.

Select a tag in the Contents of window, and then click **Add Selected Tag(s)**.

The new data points appear in red in the data point grid. This new data point is considered a pending edit.

In creating new data points and saving them, saving pending edits does not affect the running configuration because they have not been used in a transaction. Assembling new data points that are not used in a configuration that uses online edits does not affect the running configuration because FactoryTalk Transaction Manager does not collect data for data points that are not used. Data points must be assembled before they can be used in a new transaction.

## Editing Existing Data Points

While editing existing data points, you may change any of the data point parameters except the data point name and mode (scheduled, unscheduled or device-scheduled). To open the Edit Collection Parameters of Selected Row(s) dialog box and change the properties of data points, use one of the methods described below:

- Select the data point row in the data point grid, right-click and select Edit Selected Collection Parameters from the menu.
- Double-click in the data point row to which you want to make the change.
- Select the data point row in the data point grid and select Create Edits.
- Select the data point row in the data point grid, and copy or paste data points from Excel.

After you have finished modifying the data point parameters, you must save the changes before you close the FactoryTalk Data Point dialog box.

## Saving Data Point Pending Edits

When you are finished adding new data points or changing existing data points, you must save the pending edits. Select the data point rows in the data point grid and click **Save Edits** or select the data point rows in the grid, right-click and select Save Selected Edit(s) from the menu. You must always save pending edits before you assemble.

## Assembling Data Point Pending Edits

To make the pending edits effective in the running configuration, you must assemble them. On the FactoryTalk Data Point dialog box, select the data point rows in the grid and click **Assemble Edits** or select the data point rows in the grid, right-click and select Assemble Edits from the menu.

It is not critical to assemble pending edits at a specific time in the online edits process. If you have already created data point pending edits, you can close the FactoryTalk Data Point dialog box without assembling and proceed to make changes to transactions on the Pending Transaction Definition dialog box. However, data point pending edits do not display on the Pending Transaction Definition dialog box until they have been assembled. It is a good idea to save and assemble data point pending edits before creating

transaction pending edits. Assembling new data points does not affect a running configuration because the new data points are not used in a running transaction yet.

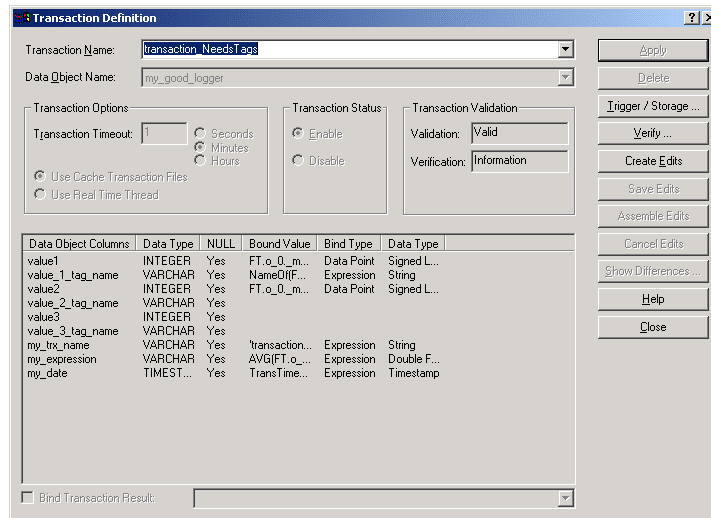
You can also click **Assemble Edits** on the main FactoryTalk Transaction Manager user interface to assemble data point pending edits.

## Canceling Data Point Pending Edits

To cancel pending edits, select the data point rows in the grid and click **Cancel Edits** or select the data point rows in the grid, right-click and select **Cancel Edits** from the menu. You can cancel pending edits after you have saved. You cannot cancel pending edits after you have assembled.

## Editing Transactions In a Running Configuration That Uses Online Edits

When you open the Transaction Definition dialog box to edit an existing transaction, notice that the fields are all disabled. You must first click **Create Edits** to begin making changes to the transaction. Notice that the title changes to Pending Transaction Definition dialog box. Then you can change transaction bindings and parameters such as scan rate or timeout, and even enable or disable the transaction. If you want to change the transaction name or the data object that the transaction uses, you must create a new transaction. When you are finished modifying the transaction, click **Save Edits** to save your changes. You must always save pending edits before you assemble. The following figure is an example of the Transaction Definition dialog box.



In the main FactoryTalk Transaction Manager window you can edit multiple transactions in a running configuration that uses online edits. Select one or more transactions in the Transaction Definition View in the right pane, right-click, and select one of the following menu options.

Item	Description
Enable Transaction or Disable Transaction	A new pending edit row is displayed with the new state. The pending edit is automatically saved, but it must be assembled manually.
Edit Transaction Parameters	The parameters displayed in the <b>Trigger and Storage Parameters</b> dialog box are FactoryTalk Transaction Manager default values, not the values of the selected transactions. Once again, the pending edit is automatically saved, but it must be assembled manually.

## Saving Transaction Pending Edits

When you finish changing the transaction, you need to save the pending edits. You must save pending edits before assembling. Saving pending edits does not affect the running configuration, it only saves the pending definition.

## Assembling Transaction Pending Edits

To make the pending edits effective in the configuration, you must assemble them. In the **Pending Transaction Definition** dialog box, click **Assemble Edits**. You can also click **Assemble Edits** in the main FactoryTalk Transaction Manager window to assemble transaction pending edits.



Note: There may be a delay between the time you assemble the pending definition and the time the pending definition actually becomes effective, regardless of when the FactoryTalk Transaction Manager user interface indicates the changes have been made.

The rules governing when the actual transaction is changed are complex, because they take into consideration the management of the currently running transactions. Keep the following in mind:

- If a transaction is not currently being executed, the software will apply pending edits immediately.
- If a transaction is currently being executing, the software will wait until the transaction has been completely executed or its timeout has been reached before applying pending edits. No additional transaction instances will be executed until the pending edits have been applied.

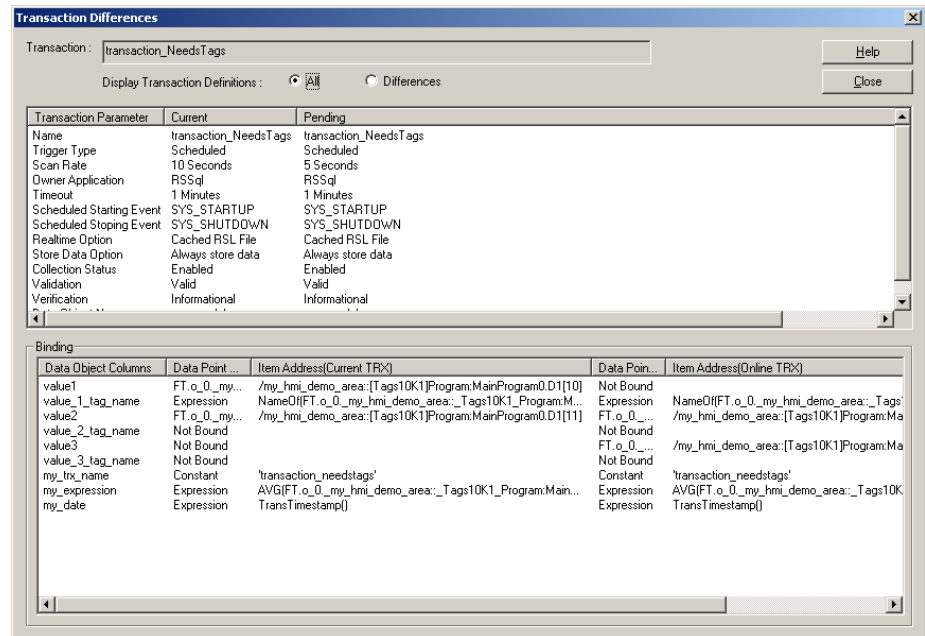
The Transaction Control Manager log file contains the entry displaying the time the transaction in question has been assembled. To view the log file in the main FactoryTalk Transaction Manager window, select the Transaction Control Manager in the Configuration tree, and then click **Log Files** on the toolbar.

## Viewing Transaction Differences

In the **Transaction Differences** dialog box, you can see the differences between the current definition and the pending definition for a specific transaction. You need to save pending edits before you view the transaction differences. To view the differences, click **Show Differences** in the **Pending Transaction Definition** dialog box.

Transaction properties are displayed at the top of the dialog box. Bindings (including the address location of the data points, not just the data point names) are displayed at the bottom of the dialog box. The default option is All, but you may select Differences to display only the properties or bindings that are different between the current and pending definitions. You can view the

differences between the current and pending transaction definitions any time after you save, but before you cancel or assemble pending edits.



You can also view transaction differences in the Transaction Definition View in the main FactoryTalk Transaction Manager window. To view the differences, right-click a transaction and select **Show Transaction Differences**.

## Canceling Transaction Pending Edits

You cannot cancel pending edits after they have been assembled. Canceling pending edits does not affect the running configuration; it only removes the pending definition. If you stop a running configuration with pending edits, you cannot make any further changes to the configuration until you cancel or assemble the pending edits.

To cancel pending edits after saving them in the **Pending Transaction Definition** dialog box, click **Cancel Edits**.





## Advanced Topics

In this chapter you will learn about the following:

- [Remote User Interface](#) on [page 89](#)
- [Distributed Configurations](#) on [page 90](#)
- [Increasing Performance](#) on [page 93](#)

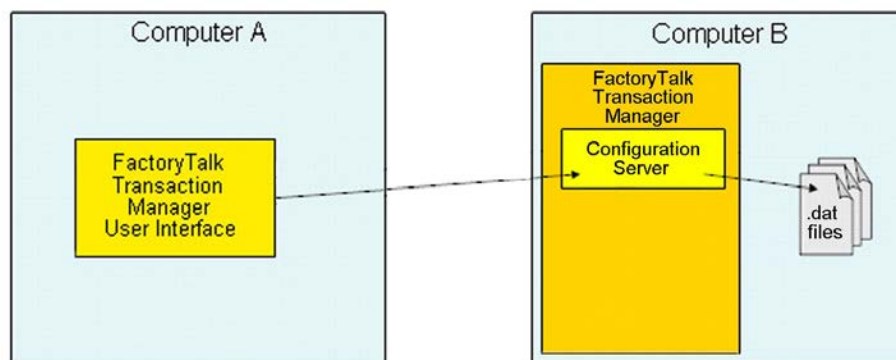
### Remote User Interface

A remote FactoryTalk Transaction Manager user interface is used to configure FactoryTalk Transaction Manager services and configuration (.dat) files to run on another computer. For example, the FactoryTalk Transaction Manager user interface runs on Computer A, but the FactoryTalk Transaction Manager services and FactoryTalk Transaction Manager configuration files are located on Computer B. No additional licensing is required to perform this function remotely.



Note: Although you can use the Demo or Trial versions of FactoryTalk Transaction Manager to configure a remote user interface, you cannot run a configuration using any of those versions.

The remote FactoryTalk Transaction Manager user interface communicates directly with the Configuration Server, which then reads from and writes to the configuration files.



Remote browsing allows you to browse DSNs and Oracle connection strings that cannot be found on the local computer. All the browsing for FactoryTalk data points and databases is performed in the context of Computer B.

### Configuring the Remote User Interface

#### To configure the remote user interface:

1. While running the FactoryTalk Transaction Manager user interface on the remote computer (the computer not running the Configuration Server), right-click the computer name in the **Configuration** tree and select **Register Configuration Server**.

The **Register Configuration Server** dialog box appears.

2. Type the host computer name and then click **Register**.



Note: If you are not using the FactoryTalk Live Data connector, we recommend that you do not attempt to use a remote FactoryTalk Transaction Manager user interface for configuring your data points.

When using the remote FactoryTalk Transaction Manager user interface, you need to ensure that all of the computers involved in your configuration belong to the same FactoryTalk Directory.

To ensure the proper functioning of the FactoryTalk Security, the local FactoryTalk Transaction Manager user interface and remote FactoryTalk Transaction Manager user interface need to share the same Configuration Server in the FactoryTalk Directory.

## Distributed Configurations

A distributed configuration exists when FactoryTalk Transaction Manager services are used on different computers. The advantage of using a distributed configuration is that the processing of large amounts of data can be distributed across multiple computers. This is very useful when a single computer processor is not able to handle the increased amount of data, or you want to use multiple control connectors or enterprise connectors of the same type.

You can create a distributed configuration in step 2 of the [Configuration Checklist](#) on [page 41](#), by choosing different computers to run different FactoryTalk Transaction Manager services.

## Licensing Required For a Distributed Configuration

A FactoryTalk Transaction Manager Professional license is required to distribute control and enterprise connector services among multiple computers. You need to run the FactoryTalk Server Activation software on the network computer(s) that will act as the activation server(s), and run the FactoryTalk Client Activation software on each of the client computers. Then you will direct the client computers to the activation server computer.



Note: The license must not be installed on a mapped drive, otherwise the Transaction Control Manager service or FactoryTalk Transaction Manager service will not be able to use it.

## Establishing Microsoft Windows Privileges

The FactoryTalk Transaction Manager user interface must be logged into a Microsoft Windows account that has administrative privileges for all the computers being part of the FactoryTalk Transaction Manager system. This requirement gives the FactoryTalk Transaction Manager user interface access to the Microsoft Windows Service Control Manager, in order to start and stop the FactoryTalk Transaction Manager services.

## Creating a Distributed Configuration

### To create a distributed configuration:

1. In Step 1 of the [Configuration Checklist](#) on [page 41](#), define the FactoryTalk Transaction Manager configuration files. The path that you use must be on the same machine as the Configuration Server.
2. Select the control and enterprise connector services.
3. In Step 2 of the Configuration Checklist, define the host computer on which each service will run. You should have FactoryTalk Transaction Manager installed on each of the host computers before you begin.

## Using UNC Paths

When defining distributed configurations, you should use the Universal Naming Convention (UNC) for the error log (.log) and transaction cache (.rsl) file paths. Follow the format below:

```
\\servername\sharename\path
```

For example:

```
\\Computer 1\c$\rssql_config
```



Note: It is important that the account you are currently logged into (and the one that FactoryTalk Transaction Manager services will run as) has read and write privileges to the shared folder.

## Changing the Transaction Cache File Path

### To change the transaction cache file:

1. Select the configuration name in the tree, and then on the **Configuration** menu, click **Properties**.  
The **Configuration Properties** dialog box appears.
2. On the **Cache** tab, double-click the connector.  
The **Enterprise Connector Options** dialog box appears.
3. Under **Cached Transaction Files**, change the file path in the **Path** box.



Note: The transaction cache file must be located on the same computer as the FactoryTalk Transaction Manager service.

## Changing the Error Log File Path

### To change the error log file path:

1. Select the configuration name in the tree, and then on the **Configuration** menu, click **Properties**.  
The **Configuration Properties** dialog box appears.
2. Click the **Error Log** tab.
3. Under **File Messages**, change the file path in the **Path** box.



Note: To help reduce the network traffic, you can store the error log files on the remote FactoryTalk Transaction Manager computer. Unfortunately, you may not be able to view those log files from the local FactoryTalk Transaction Manager computer; in such a case, view the files directly on the remote FactoryTalk Transaction Manager computer.

## Data Point Buffering

Simple logging applications include many locations for buffering data. This means that there is a risk of discrepancies between the values in your controller and the values in your database. The sections that follow describe different areas in which data point buffering can be used.

### Buffering In the Controller

Values may change between scans, or, more likely, the data server may read a series of related values while the controller is updating them. In such a case, the values will not be synchronized. This problem can be solved by either blocking the data in a single message sent by the controller or by making sure that the controller does not trigger the transaction until all the values are set. You should also use a transaction result to alert the controller that the values have been read and can now be changed.

### Buffering In the FactoryTalk Transaction Manager Control Connector

The FactoryTalk Transaction Manager control connector maintains a copy of the data. Data servers, which may have separate copies of the data, send changed values to the control connector. Once in the control connector, the data is either sent to the FactoryTalk Transaction Manager service (unscheduled data points) or is buffered until it is requested (scheduled data points).

Unscheduled data points sent to the FactoryTalk Transaction Manager service are used in currently running transactions, as transaction triggers, or used to update the current value table of the FactoryTalk Transaction Manager service. If the data points are not immediately needed, they can be overwritten before the data is used.

Scheduled data points are only used when a transaction is executed. Scheduled data points are stored in the control connector until the FactoryTalk Transaction Manager service requests them. If the transaction is executed slower than the data is changing in the controller, the data may be overwritten in the control connector's buffer. In some circumstances, lost data is acceptable (for example, when the temperature is recorded every five minutes, but fluctuates every few seconds). The control connector is aware of every change, while the FactoryTalk Transaction Manager service is only aware of the current value every five minutes.

### Buffering In the Transaction Control Manager Service

The Transaction Control Manager service has a local data point cache that contains the current values for every data point and the time the value was collected.

If the **Data Valid** option in the **FactoryTalk Data Point Definition** dialog box is set to zero, data is not requested from the FactoryTalk Live Data server, because the server automatically provides updated data values when they change. Hence, the values for the Transaction Control Manager services are always correct.

Transactions buffer their own values, and only request values from the master value table of the Transaction Control Manager service when the Data Valid

timeout occurs. The exceptions to this rule is device scheduled data points, which are read from the controller by the FactoryTalk Live Data server once it receives the read request from the Transaction Control Manager service. Another exception is that the FactoryTalk Transaction Manager service never requests the current value of an unscheduled data point from the FactoryTalk Live Data server.

## Buffering In the FactoryTalk Transaction Manager Service

The FactoryTalk Transaction Manager service has a local data point cache that contains the current values for every data point and the time the value was collected.

If the **Data Valid** option in the **FactoryTalk Data Point Definition** dialog box is set to zero, data is requested from the control connector every time it is needed. Using this data, the FactoryTalk Transaction Manager service knows which data points are still fresh (i.e., still in the valid timeout range), and which data points need to be requested from the control connector. Note that the data is not read from the controller at this time, but from the control connector's buffers. The exception to this rule is device scheduled data points, which are read from the controller by the data server once it receives the read request from the control connector. Another exception is that the FactoryTalk Transaction Manager service never requests the current value of an unscheduled data point from the control connector.

Individual transaction buffers are maintained for each transaction that is running. If two copies of the same transaction are running at the same time, the data values from the second transaction do not overwrite the data values from the first one. Only unscheduled transactions can execute more than one copy of the same transaction at the same time.

## Buffering In Cached Transaction Files

Complete transactions that are not configured for real-time storage are buffered in cached transaction (\*.rsl) files prior to storage in the database. Therefore, data will not be available from a query until it has been removed from the cache file and written to the database.

It is possible to control the rate that the cache transaction files are applied at. In the **Transaction Definition** dialog box, you can set the number of complete transactions to a smaller value, or decrease the time between cache transaction files. This will improve the timeliness of the data in your database, however, the load on your database will increase.

## Increasing Performance

The sections that follow discuss recommendations for increasing performance when running FactoryTalk Transaction Manager configurations.

### Control System

For control system, adhere to the following recommendations:

- Store data in the control system in consecutive locations. This allows the control data server to read and write the entire block of data one time, instead of reading and writing data several times for each transaction.

- Enable the data server optimization of the control system for reading and writing data.
- Use event-driven communication (unsolicited messages) instead of fast polling. For example, if the control data does not change often, set the control system to send data only when it changes, instead of continuously polling.
- If you must poll control data, use an appropriate poll rate (e.g., do not poll every 10 milliseconds for data that is saved every 10 seconds).

## Database

For database, adhere to the following recommendations:

- Use a commercial database (e.g., Microsoft SQL Server) rather than a personal database (such as Microsoft Access). Please see the *FactoryTalk Transaction Manager Release Notes* for a list of supported databases.
- Distribute the database to a different computer than the one running FactoryTalk Transaction Manager.
- Use an appropriate data model for your application.
- Configure the database. Take the time to understand when to use indices and how to archive data. If need be, consult your database administrator for assistance.
- Optimize queries, triggers, and stored procedures executed by FactoryTalk Transaction Manager.

## FactoryTalk Transaction Manager

For FactoryTalk Transaction Manager, adhere to the following recommendations:

- Use unscheduled data change as transaction triggers instead of scheduled transaction execution.
- Use the FactoryTalk Device Scheduled collection mode, which improves data accuracy and reduces network traffic. However, note that this collection mode will cause transactions to run slower.
- Distribute both control connectors and enterprise connectors on multiple computers.
- When using real-time transactions, use multiple real-time threads. The disadvantage is that this consumes extra connections to the database. Some databases are licensed based on the number of simultaneous connections.
- Use the Oracle OCI connection instead of the ODBC connection to the Oracle database.
- Use cached transaction files instead of real-time threads. This enhances the performance of commercial databases that allow inserting array values.
- Modify the parameters of cached transaction files (the number of transactions per log file and the time between caching the files) to optimize the scheduling and volume of transactions issued to your database.
- Disable debug error logging in FactoryTalk Transaction Manager.

- When collecting data from FactoryTalk Linx and FactoryTalk View, some controller values may exist in FactoryTalk Linx and in the FactoryTalk View tag database. Collect these points from FactoryTalk Linx, not from FactoryTalk View. This allows FactoryTalk Linx to optimize the data collection from the controller by reading the data once and passing it to FactoryTalk View and FactoryTalk Transaction Manager.
- Delete unused data objects and database connections. The enterprise connector automatically tries to connect to these databases even if they are not used in the configuration.
- If a controller register is assigned to more than one data point in a transaction, use the same data point name so that the Transaction Control Manager service or FactoryTalk Transaction Manager service does not have to request the data more than once.

## Hardware and Operating Environment

For hardware and operating environment, adhere to the following recommendations:

- Use the fastest CPU, the most RAM, and the fastest disk controller as possible, as well as multiple fast disks.
- Run the FactoryTalk Transaction Manager services, especially the Transaction Control Manager service or FactoryTalk Transaction Manager service, on a multi-processor computer. The FactoryTalk Transaction Manager services are multi-threaded and can take advantage of multiple-processors.
- Optimize the Ethernet traffic. Use a local subnetwork based on switched Ethernet.
- Use the fastest Ethernet connection possible.
- Turn off the ODBC trace and SQL trace facilities.
- Place the Microsoft Windows operating system and paging files, the cached transaction files, and the database along with its associated files on separate physical disks.

## Windows Performance Counters

Use Windows performance counters to track transaction run-time status in FactoryTalk Transaction Manager. The following status information are exposed via the Windows performance counters:

- **Transactions Total:** the total number of transactions that have been triggered.
- **Transactions Passed:** the number of transactions that have been completed without error.
- **Transactions Failed:** the number of transactions that have failed.
- **Transactions Passed %:** the percentage of transactions that have completed without errors.
- **Transactions Cached:** the number of transactions currently in the transaction cache files.
- **Database Passed:** the number of transactions that have been successfully executed by the database.

- **Database Failed:** the number of transactions that have experienced a database error.
- **Database Pending:** the number of transactions that have been started but are not completed or currently running.

Install and remove these performance counters by manually running the appropriate .bat file as an Administrator. The following files are located in **C:\Program Files (x86)\Rockwell Software\RSSql:**

- InstallPerformanceCounters\_OnlineEditsOff.bat
- Run this .bat file if you have not enabled online edits in your configuration.
- InstallPerformanceCounters\_OnlineEditsOn.bat
- Run this .bat file if you have enabled online edits in your configuration.
- RemovePerformanceCounters\_OnlineEditsOff.bat
- Run this .bat file if you have not enabled online edits in your configuration.
- RemovePerformanceCounters\_OnlineEditsOn.bat
- Run this .bat file if you have enabled online edits in your configuration.

#### To create Windows performance counters in FactoryTalk Transaction Manager

1. Navigate to the folder **C:\Program Files (x86)\Rockwell Software\RSSql**, run one of the following BAT files before creating Performance Counters:
  - If you have not enabled online edits in your configuration, choose **InstallPerformanceCounters\_OnlineEditsOff.bat**
  - If you have enabled online edits in your configuration, choose **InstallPerformanceCounters\_OnlineEditsOn.bat**
1. Open **Performance Monitor**.
2. On the left pane of **Performance Monitor**, expand **Data Collector Sets**, right-click **User Defined**, and then select **New > Data Collector Set**.
3. Enter the name for the Data Collector Set, select **Create manually**, and then click **Next**.
4. Select **Create data logs > Performance counter**. Click **Next**.
5. Click **Add**.
6. In **Available counters**, expand **FactoryTalk Transaction Manager** or **FactoryTalk Transaction and Control Manager** to select the following counters.



Tip: If you run InstallPerformanceCounters\_OnlineEditsOff.bat, the performance counters are listed under **FactoryTalk Transaction Manager**. If you run InstallPerformanceCounters\_OnlineEditsOn.bat, the performance counters are listed under **FactoryTalk Transaction and Control Manager**.

- **Transactions Total**
- **Transactions Passed**
- **Transactions Failed**



- **Transactions Passed %**
  - **Transactions Cached**
  - **Database Passed**
  - **Database Failed**
  - **Database Pending**
1. Click **Add**.
  2. Click **OK**.
  3. Type **Sample interval**, and then click **Next**.
  4. Configure the root directory, and then click **Next**.
  5. Select **Save and close**, and then click **Finish**.
  6. On the left pane of **Performance Monitor**, right-click the data collector set that is created in step 3, and then click **Start**.

#### **To visualize your performance counters in Performance Monitor**

1. In **FactoryTalk Transaction Manager**, start your configuration.
2. In **Performance Monitor**, expand **Monitoring Tools**, select **Performance Monitor**. Click **Add**.
3. On the page of **Add Counters**, in **Available counters**, expand **FactoryTalk Transaction Manager** or **FactoryTalk Transaction and Control Manager** to select one or more counters.
4. Click **Add > OK**.



# FactoryTalk Transaction Manager Sample Applications

In this chapter you will learn about the following:

- [External Trigger Sample Application](#) on [page 99](#)
- [Application Contents](#) on [page 99](#)
- [Running the Sample Application](#) on [page 99](#)

## External Trigger Sample Application

The External Trigger sample application illustrates how an external application can trigger a FactoryTalk Transaction Manager transaction. You can use external triggering to create a custom user interface to the FactoryTalk Transaction Manager application, or to integrate FactoryTalk Transaction Manager functionality into an existing software system.

The **Extras** directory on the FactoryTalk Transaction Manager product CD contains the External Trigger sample application discussed in this chapter.

## Application Contents

The sample application contains the following elements:

- A SQL Server single table (ExternalTriggerDatabase) in Master database that contains two columns:
  - DataPointValue
  - TransTimestamp
- A FactoryTalk Transaction Manager configuration that contains a single unidirectional transaction named ExternalTriggerTransaction. The transaction obtains a data value from Control system, and appends a record in the Microsoft SQL database. The transaction is defined so that it can be triggered by an external application. You will have to edit the transaction to use a DataPoint from your control system.
- Three sample programs (in Microsoft Visual Basic, C Language, and Microsoft Visual C++) that allow you to trigger the transaction by entering a transaction name.

## Running the Sample Application

To run the application:

1. Open the SQL Server Management Studio. Logon with Windows authentication or SQL Server authentication account that has permission to operate master database, select **File > Open > File**, and then open the script file from **\Documentation\Extras\Application Trigger Sample\SQLServer\ExternalApplication.sql**. This will create ExternalTriggerDatabase table in master database.



Note: This sql file does not create any user or assign any user permissions to the tables, which must be done manually.

2. Create a System DSN (use ODBC Data Sources 32-bit on the NT Control Panel) named `ExtTrigger` that points to the Microsoft SQL Server database called `Master`.
3. Open the table `ExternalTriggerDatabase` and return all rows.
4. Run FactoryTalk Transaction Manager and restore the FactoryTalk Transaction Manager configuration contained in the file **Documentation\Extras\ApplicationTriggerSample\RSSql\ExternalTriggerSQL.rsq**. Change all of the Connectors to run on your local computer with a system administrator username and password. (This is best done from the Connector Definition screen. Change the username and password for one of the connectors, then click on the Host/User button to set them all.) From the ODBC Data Object Definition screen, check the Database Connection field to make sure it is connecting to the new System DSN `ExtTrigger`. Check the username and password.
5. Open **FactoryTalk Administration Console**. Create a new application. Create a new FactoryTalk Linx server, and then create a shortcut and bind it to a controller which has program running. Click step 3 in the configuration checklist. Change application to the one that you have created. Select one integer datapoint and add it to the list.

Double click "ExternalTriggerTransaction" transaction to open the transaction definition dialog, bind the integer data point you have added to the "DataPointValue" column. Save the transaction.

From the FactoryTalk Transaction Manager main menu, start the Configuration, and wait for all the traffic light indicators to turn to green.

6. Select which programming environment you will use, and open the corresponding sample program from its subdirectory (`VB_Example.vbp`, `C_Example.dsw`, `CPP_Example.dsw`).
7. Run the program, and enter the transaction name "ExternalTriggerTransaction". Hit the "Trigger" button to trigger the transaction. The transaction will take the value from the DataPoint and append a record in the database containing this value.
8. From the SQL Server Management Studio, refresh the display of the `ExternalTriggerDatabase` table to see the records that have been created in the database.

The sample illustrates the following two methods in the FactoryTalk Transaction Manager Application Program Interface (API) for triggering transactions:

- **RSSqlUnconnectedTrigger() function:** The only parameter is the name of the transaction to be triggered (case-sensitive). The function establishes a connection to the FactoryTalk Transaction Manager service, sends the trigger request, and stops the connection. The

function works well when the number of trigger requests is small. The Visual Basic version of this call is `RSSqlUnconnectedTriggerVB()`.

- **`RSSqlConnectedTrigger()` function:** The only parameter is the name of the transaction to be triggered (case-sensitive). In addition, the function requires that the calling function use the `RSSqlConnect()` and `RSSqlDisconnect()` functions to manage the connection. The function is suitable for applications in which a large number of transactions must be triggered. The Visual Basic version of this call is `RSSqlConnectedTriggerVB()`, and the related calls are `RSSqlConnectVB()` and `RSSqlDisconnectVB()`.

For more information, refer to section "API Calls" in the FactoryTalk Transaction Manager online help.



## Securing FactoryTalk Transaction Manager Using FactoryTalk Security

FactoryTalk Security is intended to improve the security of your automation system by limiting access to the users with a legitimate need. FactoryTalk Security authenticates user identities and authorizes user requests to access a FactoryTalk-enabled system. These security services are fully integrated with the FactoryTalk Directory and are included as part of the FactoryTalk Services Platform that is installed with many Rockwell Software products.



Tip: For more information on configuring or overriding security services using FactoryTalk Security, see the FactoryTalk Security online help.

By default, the help file is located in the following directory:

C:\Program Files (x86)\Common Files\Rockwell\HelpFTSecurityEN.chm

### Considerations When Using FactoryTalk Transaction Manager With FactoryTalk Security

Please keep the following in mind when configuring FactoryTalk Transaction Manager for use with FactoryTalk Security:

- FactoryTalk Transaction Manager inherits its security settings from Network (also called Distributed) applications and/or the FactoryTalk Network Directory. Any changes that you make via FactoryTalk Security affect FactoryTalk Transaction Manager and all other products that are connected to the same FactoryTalk Directory computer.



Note: If you must change the FactoryTalk Directory computer location, reboot your computer to synchronize the Configuration Server with the FactoryTalk Directory computer.

- The Configuration Server is the only FactoryTalk Transaction Manager-specific component to which you can apply security permissions via the FactoryTalk Administration Console. One Configuration Server (in FactoryTalk Transaction Manager) is equivalent to one computer (in FactoryTalk). All configurations that are displayed under a single Configuration Server have the same security settings.



Note: For information on product-specific, FactoryTalk Security-related permissions that are necessary for external components used by FactoryTalk Transaction Manager (e.g., FactoryTalk Live Data), refer to the online help for respective components.

- FactoryTalk Transaction Manager inherits its FactoryTalk Security settings from the computer that hosts the Configuration Server. The settings must be configured using the FactoryTalk Administration Console.



Note: The computer that hosts the Configuration Server may be configured to inherit from a higher level in the FactoryTalk Directory (e.g., Application or Area). For more information, refer to the FactoryTalk online help.

- In addition to the standard Read and Write permissions, FactoryTalk Transaction Manager also supports one custom action: Start, Stop, and Assemble Configurations. For more information, see [FactoryTalk Security Permissions To Perform FactoryTalk Transaction Manager Tasks](#) on [page 106](#), [Writing Product-Specific Security Privileges From a Previous Release To a File](#) on [page 107](#).
- All FactoryTalk Transaction Manager-specific privilege information from previous releases will be ignored in existing configurations, and it will not be converted directly to FactoryTalk Security attributes. You can view all FactoryTalk Transaction Manager-specific privilege information using the tool described in [Writing Product-Specific Security Privileges From a Previous Release To a File](#) on [page 107](#), and then you can use that data to establish similar permissions in the FactoryTalk Administration Console.
- FactoryTalk Security is set in the FactoryTalk Directory. Therefore, if you move a configuration from one FactoryTalk Directory to another FactoryTalk Directory, your FactoryTalk Security permissions will not be maintained.
- If the administrator changes your individual security permissions in the FactoryTalk Administration Console, or if your user permissions are altered, the changes will be reflected in FactoryTalk Transaction Manager without requiring you to log off and log on to FactoryTalk Security. However, if the administrator makes changes in the FactoryTalk Administration Console to a group membership (of which you are a member), you need to log off and then log on again to FactoryTalk Security to enforce the permission changes.
- FactoryTalk Transaction Manager includes the **Security** menu that allows you to log on and log off from FactoryTalk Security from within FactoryTalk Transaction Manager, as well as check the permissions of the current user (by clicking **Permissions** on the **Security** menu).
- If you want to share configuration tasks across multiple computers, the FactoryTalk Directory servers on all of the computers must match. For example, if the FactoryTalk Directory server and the FactoryTalk Transaction Manager Configuration Server are on your computer (computer A), and you want to communicate with another computer (computer B) that is configured to use another FactoryTalk Directory server, data can be shared between computer A and computer B only if the FactoryTalk Directory servers are the same.
- All security permissions must be assigned in the FactoryTalk Administration Console.

The username and password associated with the FactoryTalk Security permissions to the FactoryTalk Transaction Manager user interface may be



completely independent from the username and password associated with each connector.



Tip: For more information on configuring FactoryTalk Security, see the FactoryTalk Security online help, section "About security".

By default, the help file is located in the following directory:

C:\Program Files (x86)\Common Files\Rockwell\HelpFTSecurityEN.chm

## Using FactoryTalk Transaction Manager With Single Sign-on (SSO)

The single sign-on (SSO) is a FactoryTalk Security policy setting that allows you to log on to the first product that you run in the FactoryTalk system, and then to be automatically logged on (without being prompted) to each subsequent Rockwell Software product that you run, using the same user account and password.



Note: If you want to use FactoryTalk Transaction Manager and FactoryTalk Security effectively, you must have a FactoryTalk user account in the FactoryTalk Directory.

When you log on to FactoryTalk Transaction Manager, all ties between FactoryTalk Transaction Manager and the machine-wide SSO are cut. It means that any changes made to the machine-wide SSO user do not affect the user that is already logged on to FactoryTalk Transaction Manager.

When you log off from FactoryTalk Transaction Manager, the machine-wide SSO user is not affected either.

When you log on to FactoryTalk Transaction Manager, the SSO behavior varies depending on the user account that you are using:

- **If you have a Windows-linked user account in the FactoryTalk Directory.**

If you already have a Windows-linked user account in the FactoryTalk Directory, you will be logged on as the SSO user, even if FactoryTalk Transaction Manager does not recognize you as the SSO user at startup.

If you log off from FactoryTalk Transaction Manager by clicking **Logoff** on the **Security** menu, and then you are logging on again by clicking **Logon**, you will be logged on to FactoryTalk Transaction Manager automatically.

- **If you have a FactoryTalk user account and are logged on to the FactoryTalk Directory.**

If you are already logged on to the FactoryTalk Directory with your FactoryTalk user account, and you are logging on to FactoryTalk Transaction Manager, you will be logged on as the SSO user, because FactoryTalk Transaction Manager will recognize you as the SSO user at startup.

If you log off from FactoryTalk Transaction Manager by clicking **Logoff** on the **Security** menu, and you are logging on again by clicking **Logon**,

you will be logged on to FactoryTalk Transaction Manager automatically.

- **If you do not have a Windows-linked user account.**

If you are not recognized by FactoryTalk Security as the SSO user, and you do not have a Windows-linked user account in the FactoryTalk Directory, you will be prompted to type your FactoryTalk username and password at the FactoryTalk Transaction Manager startup in the **Log On to FactoryTalk** dialog box.

If you log off from FactoryTalk Transaction Manager by clicking **Logoff** on the **Security** menu, and you are logging on again using the **Logon** option, you will be prompted to type your FactoryTalk username and password in the **Log On to FactoryTalk** dialog box.



Tip: For more information on user account types and the way they interact with FactoryTalk Security, see the FactoryTalk Security online help, section "About user, computer, and group accounts".

By default, the online help is located in the following directory:

C:\Program Files (x86)\Common Files\Rockwell\HelpFTSecurityEN.chm

## FactoryTalk Security Permissions To Perform FactoryTalk Transaction Manager Tasks

To perform specific tasks in FactoryTalk Transaction Manager, you need to have specific FactoryTalk Security permissions. Use the following table to determine the initial action (indicated by "x") that you or your administrator must take in the FactoryTalk Administration Console to ensure that you will be able to perform the specified FactoryTalk Transaction Manager tasks.

To perform this FactoryTalk Transaction Manager task:	Select the following item on the Explorer pane in the FactoryTalk Administration Console:	Right-click the selected item on the Explorer pane in the FactoryTalk Administration Console, select Security, and then set the following actions in the Security Settings dialog box that is displayed:			
		Factory Talk Transaction Manager > Start, Stop, and Assemble Transactions	Common > List Children	Common > Read	Common > Write
View a configuration.	Computer in the Computers group that hosts the FactoryTalk Transaction Manager Configuration Server.			X	
	FactoryTalk Area (located under <b>Network &gt; App</b> ) that you want to browse for tags.(1)		X	X	
Change a non-running configuration.	Computer in the Computers group that hosts the FactoryTalk Transaction Manager Configuration Server.			X	X
	FactoryTalk Area (located under <b>Network &gt; App</b> ) that you want to browse for tags.(1)		X	X	

To perform this FactoryTalk Transaction Manager task:	Select the following item on the Explorer pane in the FactoryTalk Administration Console:	Right-click the selected item on the Explorer pane in the FactoryTalk Administration Console, select Security, and then set the following actions in the Security Settings dialog box that is displayed:			
		Factory Talk Transaction Manager > Start, Stop, and Assemble Transactions	Common > List Children	Common > Read	Common > Write
Change a running configuration (i.e., perform online edits).	Computer in the Computers group that hosts the FactoryTalk Transaction Manager Configuration Server.	X		X	X
Start or stop a configuration.	Computer in the Computers group that hosts the FactoryTalk Transaction Manager Configuration Server.	X		X	X
Run a service that is not FactoryTalk Live Data enabled.					
Run a service that is FactoryTalk Live Data enabled (Transaction Control Manager or FactoryTalk Live Data connector) for read only access to the controller.(2)	FactoryTalk Area (located under <b>Network &gt; App</b> ) that you want to browse for tags.		X	X	

(1) If you specify permissions at the application level in the FactoryTalk Administration Console, they will be inherited by all of the areas included in the application. You can also set the permissions for each area separately. For more information, see the FactoryTalk Security online help.

(2) The user specified in the control connector must be a Microsoft Windows-linked user. This Microsoft Windows-linked user does not have to be defined via FactoryTalk Security.

## Writing Product-Specific Security Privileges From a Previous Release To a File

Prior to FactoryTalk Transaction Manager 9.00.00, the software used its own security mechanism to secure configurations. If you used this product-specific security implementation to secure your configurations in a previous release, you can now use a utility to write your previous FactoryTalk Transaction Manager security settings to a file.

The utility name is **RSSql\_WritePriv**, which is located in the following directory:

**C:\Program Files (x86)\Rockwell Software\RSSql.**

When using the tool, you can specify the file name and location. The file will be formatted as follows:

- Line 1 - the file title.
- Line 2 - the Configuration Server name.
- Line 3 - the configuration name.
- Line 4 and subsequent lines - the security level, the user or group name, and any file notification options.

For example:

-----  
-----

```

FactoryTalk Transaction Manager Security Privilege
Information
Configuration Server = ussewbob
Configuration = TrackerSecretPrivs
-----
ADMIN Users/Groups
Count = 1
BOB
-----
MODIFY Users/Groups
Count = 0
-----
VIEW Users/Groups
Count = 1
Mail - FactoryTalk Transaction Manager Beta

```

After generating the file, you can review the old product-specific security privileges and use the content to establish new FactoryTalk Security permissions in the FactoryTalk Administration Console.

For more information on writing product-specific security privileges to a file, see [FactoryTalk Security Permissions To Perform FactoryTalk Transaction Manager Tasks](#) on [page 106](#).

## Mapping Old Product-Specific Security Privileges To New FactoryTalk Security Permissions

The following table provides information on the way the old product-specific security privileges map to the new FactoryTalk Security permissions.

If you used this old product-specific security privilege:	It has been replaced by this FactoryTalk Security permission or FactoryTalk Transaction Manager custom action:	This security permission allows you to:
RSSQL_ADMIN	FactoryTalk Transaction Manager > Start, Stop, and <a href="#">Assemble Configurations</a> on <a href="#">page 106</a> <b>Note:</b> This attribute also requires you to have the <b>Common &gt; Read</b> and <b>Common &gt; Write</b> permissions.	Start, stop, or assemble (on-line editing) a running configuration.
RSSQL_MODIFY	Common > <a href="#">Write</a> on <a href="#">page 106</a> <b>Note:</b> This attribute also requires you to have the <b>Common &gt; Read</b> permission.	Make offline changes to a configuration.
RSSQL_VIEW	Common > <a href="#">Read</a> on <a href="#">page 106</a>	View a configuration.

If you used this old product-specific security privilege:	It has been replaced by this FactoryTalk Security permission or FactoryTalk Transaction Manager custom action:	This security permission allows you to:
No privilege	No permissions needed.	View the following FactoryTalk Transaction Manager options (since security permissions have not been configured): <ul style="list-style-type: none"> <li>• Security &gt; Logon</li> <li>• Configuration &gt; Exit</li> <li>• Help Menu (all enabled)</li> <li>• Configuration Tree (empty)</li> <li>• System View Graphic</li> </ul>
If you used this old product-specific security privilege:	It has been replaced by this FactoryTalk Security permission or FactoryTalk Transaction Manager custom action:	This security permission allows you to:
		<ul style="list-style-type: none"> <li>• System View Graphic</li> </ul>



## **Glossary**

### **A - D**

#### **- A -**

#### **Assemble Pending Edits**

The process of changing the running, current definition of a transaction or data point in a configuration that uses online edits to the pending definition. In order to assemble pending edits, you need to save them first.

#### **Audit Trail**

A record of changes made to a FactoryTalk Transaction Manager service in the FactoryTalk Transaction Manager configuration, including the information by whom and when the changes were made, as compiled by the Configuration Server log file. The changes are displayed in FactoryTalk Diagnostics.

#### **- B -**

#### **Binding**

The relationship between a single data object element (a table column or a stored procedure parameter) and its corresponding data point or expression in a transaction.

#### **Bound Value**

The data to be written to or read from a table column or a stored procedure parameter.

**- C -****Configuration**

A FactoryTalk Transaction Manager configuration consists of a set of transactions, as well as the connectors, data points, and data objects required to implement the transactions. All configuration information is stored in configuration files. You can create an unlimited number of configurations, but the FactoryTalk Transaction Manager can run only one configuration at a time.

**Configuration Server**

The Configuration Server is a service that runs continuously to provide a single interface to the configuration files (with the .dat file extension) which make up the FactoryTalk Transaction Manager configuration. The Configuration Server simplifies access to the configuration files by filtering all changes to the files and by communicating with other FactoryTalk Transaction Manager services. A collection of all changes that affect a configuration is recorded in an audit trail (via either FactoryTalk Diagnostics or the Configuration Server \*.log file).

**Configuration That Uses Online Edits**

A configuration using the Transaction Control Manager service to communicate exclusively with FactoryTalk Live Data servers. The configuration can be changed while it is running.

**Control Connector**

A service that moves data between a data server in the control system and the FactoryTalk Transaction Manager service.

**Control System**

Typically includes a network of controllers and/or HMI servers that collect data from machines in a manufacturing plant and control their operation.



---

**- D -****Data Object**

A subset of columns in a database table, a database view, or a set of stored procedure parameters, along with the database connection information to access the database. A FactoryTalk Transaction Manager transaction acts on a single data object, so all of the necessary database information for a transaction must be contained in a single data object. Data objects are defined in enterprise connectors.

**Data Point**

Data locations in the control system. Data points are associated with control connectors and also contain collection parameters and other attributes. Data points can serve as transaction triggers, supply input data for transactions, and receive data as an output from a transaction.

**Database View**

A filter on selected fields in database table(s) outside the FactoryTalk Transaction Manager software.

**DSN**

An acronym for Data Source Name (i.e., the name of the database being used). A system DSN is available to all users and Microsoft Windows services, while a user DSN is available only to the user who configured it. The ODBC enterprise database connector requires a system DSN to connect to an ODBC data source.

**E - O****- E -****Enterprise Connector**

A service that moves data between the FactoryTalk Transaction Manager service and database(s) or an enterprise system.

## **Expression Editor**

A FactoryTalk Transaction Manager utility used to create expressions prior to binding them in a transaction.

**- F -**

## **FactoryTalk Metrics Enterprise Application Connector**

The FactoryTalk Metrics enterprise application connector is only used with FactoryTalk Metrics and stores data into the Service Console. The FactoryTalk Metrics enterprise application connector can only be configured through the Service Console.

## **FactoryTalk Security**

FactoryTalk Security is intended to improve the security of your automation system by limiting access to the users with a legitimate need. FactoryTalk Security authenticates user identities and authorizes user requests to access a FactoryTalk-enabled system. These security services are fully integrated with the FactoryTalk Directory and are included as part of the FactoryTalk Services Platform that is installed with many Rockwell Software products.

## **FactoryTalk Transaction Manager Service**

A service that controls and executes FactoryTalk Transaction Manager transactions contained in a configuration.

## **FactoryTalk Transaction Manager User Interface**

The user interface that you use to create, run, control, and monitor FactoryTalk Transaction Manager configurations.

**- G -**

## **Generic OPC Control Connector**

A service that moves data between the FactoryTalk Transaction Manager service and an OPC-compliant server.

---

**- L -****Local FactoryTalk Transaction Manager Computer**

A computer that is running the FactoryTalk Transaction Manager user interface.

**- M -****Microsoft OLE DB Enterprise Database Connector**

A service that moves data between the FactoryTalk Transaction Manager service and Microsoft SQL Server.

**- O -****OCI**

An acronym for Oracle Call Interface. OCI is an Application Programming Interface (API) used for developing software that can interface natively to Oracle databases.

**ODBC**

An acronym for Open Database Connectivity. ODBC is a widely accepted API for database access that is based on the Call-Level Interface (CLI) specifications from X/Open and ISO/IEC APIs, and uses Structured Query Language (SQL) as its database access language.

**ODBC Enterprise Database Connector**

A service that moves data between the FactoryTalk Transaction Manager service and an ODBC-compliant database. The ODBC enterprise database connector is currently written to the ODBC version 2.0 specification, and should support any ODBC driver that is version 2.0 compliant or greater.

## **OLE DB**

A Component Object Model (COM)–based database architecture that provides universal data integration over an enterprise network (from mainframe to desktop), regardless of the data type.

## **Oracle OCI Enterprise Database Connector**

A service that uses OCI to move data between the FactoryTalk Transaction Manager service and an Oracle SQL\*NET–compliant database.

## **P - U**

### **- P -**

## **Pending Edits**

Changes made to data points or transactions in a configuration that uses online edits. Pending edits must be saved before they can be assembled. Pending edits do not affect the running configuration until they have been assembled.

## **Pending Edit Alerts**

Informational messages describing that transactions with state information will lose the state information when the associated data point or transaction is assembled.

### **- R -**

## **Remote FactoryTalk Transaction Manager Computer**

A computer that is used to configure FactoryTalk Transaction Manager services and configuration (.dat) files to run on another computer (local FactoryTalk Transaction Manager computer). The computer communicates directly with the Configuration Server, which then writes to the configuration files. The FactoryTalk Transaction Manager user interface does not run on the computer.

---

**- S -****SQL**

An acronym for Structured Query Language. SQL is an ANSI/ISO standard language for querying, updating, inserting, deleting, controlling access to, and defining storage containers for data.

**- T -****Table**

In relational database terms, a unit of storage containing columns and rows with specific names and data types.

**Tag**

A collection of information for a single data point.

**Time-series Data Compression Enterprise Database Connector**

A service that compresses data from the control system using a lossless algorithm to conserve space. This connector can be configured and used only by FactoryTalk Historian Classic.

**Transaction**

An exchange of data between data points and a data object. Transactions also include triggering information and other attributes that govern its behavior. Transactions can be unidirectional or bidirectional.

**Transaction Control Manager Service**

The Transaction Control Manager is a service that controls and executes FactoryTalk Transaction Manager transactions contained in a configuration, but with the additional functionality of the FactoryTalk Live Data control connector embedded in it. In a configuration enabled for editing, the Transaction Control Manager replaces the separate FactoryTalk Transaction Manager and control connector services.

### **Transaction Result Binding**

Implements data transfer from the control system to the enterprise system (and possibly the other way round, logging a return code to the control system that is bound to a data point, and reports the success or failure of the transaction). The control system can then take the appropriate action based on the success or failure of the transaction.

### **Transaction Result Code**

A data value assigned to a transaction by FactoryTalk Transaction Manager in order to communicate successful transaction completion.

**- U -**

### **UNC**

An acronym for Universal Naming Convention.

### **Unidirectional Transactions**

Unidirectional transactions are transactions that use information from the control system to add records to a database table, or to update the contents of existing records. They do not return data to the control system.

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

### 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.

The default location of this file is:

C:\Program Files (x86)\Common Files\Rockwell\License.rtf.

### Open Source Software Licenses

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

You can view a full list of all open source software used in this product and their corresponding licenses by opening the chm file located your product's Help folder on your hard drive.

The default location of this file is:

C:\Program Files (x86)\Rockwell Software\RSSql\Help\rssqlrnotes.chm

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

Alternatively, you may obtain complete Corresponding Source code by contacting Rockwell Automation via the **Contact** form on the Rockwell Automation website: <http://www.rockwellautomation.com/global/about-us/contact/contact.page>. Please include "Open Source" as part of the request text.





# Index

## A

Adding New Data Points 84  
 Advanced Topics 89  
 Application Contents 99  
 Assembling Data Point Pending Edits 84  
 Assembling Pending Edits 81  
 Assembling Transaction Pending Edits 86

## B

Bidirectional or Unidirectional  
     Transactions With Transaction  
     Bindings 70  
 Bidirectional Transactions 69  
 Buffering In Cached Transaction Files 93  
 Buffering In the Controller 92  
 Buffering In the FactoryTalk Transaction  
     Manager Control Connector 92  
 Buffering In the FactoryTalk Transaction  
     Manager Service 93  
 Buffering In the Transaction Control  
     Manager Service 92

## C

Cached Transactions 72  
 Canceling Data Point Pending Edits 85  
 Canceling Pending Edits 82  
 Canceling Transaction Pending Edits 87  
 Changing the Error Log File Path 91  
 Changing the Transaction Cache File Path  
     91  
 Configuration and Connector Status 36  
 Configuration Checklist 41  
 Configuration Server 52  
 Configuration Server Status 36  
 Configuration That Uses Online Edits 80  
 Configuration Tree 36  
 Configuring the Remote User Interface 89  
 Consecutive Data Point and Data Block  
     Support 58

Considerations When Using FactoryTalk  
     Transaction Manager With  
     FactoryTalk Security 103  
 Control Connectors 49  
 Control System 93  
 Creating a Configuration That Uses Online  
     Edits 83  
 Creating a Distributed Configuration 91  
 Creating Transactions 67

## D

Data Point Buffering 92  
 Data Point Range and Advanced Functions  
     74  
 Data Retrieval Timeout 59  
 Data Valid 59  
 Database 94  
 Database Triggers 73  
 Defining Data Objects 63  
 Defining Data Points 55  
 Distributed Configurations 90  
 Distributed Installations 15

## E

Editing Data Points In a Running  
     Configuration That Uses Online Edits  
     83  
 Editing Existing Data Points 84  
 Editing Transactions In a Running  
     Configuration That Uses Online Edits  
     85  
 Enterprise Connector Error Handling 65  
 Enterprise Connector Options 51  
 Enterprise Database Connectors 50  
 Enterprise Database Objects 64  
 Establishing Microsoft Windows Privileges  
     90  
 Exploring the FactoryTalk Transaction  
     Manager User Interface 33  
 External Trigger Sample Application 99

## F

FactoryTalk 12  
 FactoryTalk Live Data 49  
 FactoryTalk Live Data Data Points 56  
 FactoryTalk Metrics 51  
 FactoryTalk Metrics Data Objects 65

FactoryTalk Security Permissions To Perform FactoryTalk Transaction Manager Tasks 106  
 FactoryTalk Services Platform Components 13  
 FactoryTalk Transaction Manager 94  
 FactoryTalk Transaction Manager Sample Applications 99  
 FactoryTalk Transaction Manager Service 52

## G

Generic OPC 50  
 Generic OPC Data Points 62  
 Getting Started 15  
 Glossary 111

## H

Hardware and Operating Environment 95

## I

Increasing Performance 93  
 Inserting and Updating Data Table Records 66  
 Installing and Activating FactoryTalk Transaction Manager 15  
 Introducing Data Objects 63  
 Introducing Data Points 55  
 Introducing FactoryTalk Transaction Manager Services 49  
 Introducing Online Edits 79  
 Introducing Transactions 67

## L

Licensing Required For a Distributed Configuration 90  
 Logical and Mathematical Operators 74

## M

Maintain the Current Subscribed Value 57  
 Mapping Old Product-Specific Security Privileges To New FactoryTalk Security Permissions 108  
 Menu Bar 34  
 Microsoft OLE DB 50  
 Microsoft SQL Server Data Objects 64  
 Migrate and Upgrade 19  
 Migrate FactoryTalk Transaction Manager from version 10.20 to 12.10 24

Migrate FactoryTalk Transaction Manager from version 12.00 to 12.10 20  
 Miscellaneous 45  
 Monitoring Configurations 47

## O

Occurrence Conditions of Pending Edit Alerts 82  
 ODBC 50  
 ODBC Data Objects 64  
 Online Edits Workflow 80  
 OPC Data Points 61  
 Oracle Call Interface (OCI) Data Objects 64  
 Oracle OCI 51

## P

Parse Function 75  
 Pending Edit Alerts 82  
 Preventing Stale and Mismatched Data 60

## R

Real-time Transactions 72  
 Remote User Interface 89  
 Request the Current Value From the Device 57  
 Rockwell Automation Support 99

## S

Saving Data Point Pending Edits 84  
 Saving Transaction Pending Edits 86  
 Securing FactoryTalk Transaction Manager Using FactoryTalk Security 103  
 Selecting a Collection Mode 57  
 Selecting a Substitution Option 60  
 Selecting Timeout Properties 59  
 Send Subscribed Value Whenever It Changes 57  
 Specifying Quality 61  
 Starting and Stopping Connectors 47  
 Starting Configurations 46  
 Starting FactoryTalk Transaction Manager 33  
 Step 1  
   Back up your FactoryTalk Administration Console configuration 20  
   Defining and Naming a New Configuration 42

- Remove FactoryTalk Transaction Manager 30
  - Update your FactoryTalk Transaction Manager configuration 26
  - Step 2**
    - Defining Connectors 43
    - Install FactoryTalk Transaction Manager 21
    - Install the new version of FactoryTalk Transaction Manager 30
    - Restore your FactoryTalk Administration Console configuration 29
  - Step 3**
    - Configure FactoryTalk Activation 17
    - Configure your database connections 29
    - Convert the FactoryTalk Transaction Manager configuration 30
    - Copy the cfgs60.dat file 21
    - Copy the configuration files and folder structure 22
    - Defining Data Objects 44
    - Defining Data Points 43
    - Install FactoryTalk Services 15
    - Install FactoryTalk Transaction Manager 26
    - Upgrade your FactoryTalk Transaction Manager configuration 30
  - Step 5**
    - Convert the FactoryTalk Transaction Manager configuration 22
    - Defining Transactions 44
  - Step 6**
    - Verifying Transactions 45
  - Stopping Configurations 47**
  - Stored Procedures 66**
- T**
- Time Functions 74**
  - Title Bar 34**
  - Toolbar 35**
  - Transaction Completion 72**
  - Transaction Control Manager Service 10**
  - Transaction States 37**
  - Transaction Timeout 71**
  - Transaction Trigger and Storage Options 75**
  - Transaction Types 68**
  - Transactions 12**
- Transactions With Bound Transaction Results 73**
- U**
- Understanding FactoryTalk Transaction Manager External Files 47**
  - Understanding FactoryTalk Transaction Manager Services 49**
  - Understanding Online Edit Concepts 79**
  - Understanding Online Edits 79**
  - Unidirectional Transactions 68**
  - Upgrade FactoryTalk Transaction Manager on the same computer 30**
  - Using FactoryTalk Transaction Manager With Single Sign-on (SSO) 105**
  - Using the Expression Editor 73**
  - Using UNC Paths 91**
- V**
- Viewing Configuration Properties 45**
  - Viewing Transaction Differences 86**
- W**
- Welcome To FactoryTalk Transaction Manager 9**
  - What Can FactoryTalk Transaction Manager Do For Me? 9**
  - Where Can I Go for Help? 14**
  - Workspace 37**
  - Writing Product-Specific Security Privileges From a Previous Release To a File 107**

# Rockwell Automation support

Use these resources to access support information.

<b>Technical Support Center</b>	Find help with how-to videos, FAQs, chat, user forums, and product notification updates.	<a href="http://rok.auto/support">rok.auto/support</a>
<b>Knowledgebase</b>	Access Knowledgebase articles.	<a href="http://rok.auto/knowledgebase">rok.auto/knowledgebase</a>
<b>Local Technical Support Phone Numbers</b>	Locate the telephone number for your country.	<a href="http://rok.auto/phonesupport">rok.auto/phonesupport</a>
<b>Literature Library</b>	Find installation instructions, manuals, brochures, and technical data publications.	<a href="http://rok.auto/literature">rok.auto/literature</a>
<b>Product Compatibility and Download Center (PCDC)</b>	Get help determining how products interact, check features and capabilities, and find associated firmware.	<a href="http://rok.auto/pcdc">rok.auto/pcdc</a>

## Documentation feedback

Your comments help us serve your documentation needs better. If you have any suggestions on how to improve our content, complete the form at [rok.auto/docfeedback](http://rok.auto/docfeedback).

## Waste Electrical and Electronic Equipment (WEEE)



At the end of life, this equipment should be collected separately from any unsorted municipal waste.





Rockwell Automation maintains current product environmental information on its website at [rok.auto/pec](http://rok.auto/pec).

Allen-Bradley, expanding human possibility, Logix, Rockwell Automation, and Rockwell Software are trademarks of Rockwell Automation, Inc.

EtherNet/IP is a trademark of ODVA, Inc.

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

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

Connect with us.    

**rockwellautomation.com** ————— expanding **human possibility**™

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

EUROPE/MIDDLE EAST/AFRICA: Rockwell Automation NV, Pegasus Park, De Kleetlaan 12a, 1831 Diegem, Belgium, Tel: (32) 2 663 0600, Fax: (32) 2 663 0640

ASIA PACIFIC: Rockwell Automation, Level 14, Core F, Cyberport 3, 100 Cyberport Road, Hong Kong, Tel: (852) 2887 4788, Fax: (852) 2508 1846