RSView32 to RSView Supervisory Edition
Migration Considerations
RSView32 to RSView Supervisory Edition
Migration Considerations

RSView Supervisory Edition™ supports importing and reusing RSView32™ projects. The conversion process successfully retains most of the core RSView32 project, including graphic displays, animated objects, text, tags, alarms, expressions, and macros.

RSView™ SE supports both stand-alone and distributed versions. The stand-alone version, called RSView SE Station™, has characteristics in common with RSView32 runtime software. The RSView SE distributed components include RSView Studio™, RSView SE Server, and RSView SE Client. The distributed version has many of the same capabilities as RSView32 Active Display System™. Keep these distinctions in mind as you convert your projects.

Because RSView32 and RSView Supervisory Edition are designed using different underlying architectures, there are a number of differences in the way in which project elements work within the two products. This document highlights some of the differences to be aware of when converting RSView32 projects to RSView SE applications.

HMI tags and direct-reference tags

In RSView32, tags are user defined and stored in a central tag database. Tags can be imported from controllers, and changes must be updated manually and synchronized. RSView Supervisory Edition still supports user-defined tags. RSView32 tag databases convert successfully to RSView SE.

In addition, RSView SE also makes directly referenced tags available, which are not stored in a central tag database, but are stored in programmable logic controllers where they are created. Part of the underlying FactoryTalk™ platform, FactoryTalk Directory allows access to these tags held in controllers by referencing them through a common address book.
FactoryTalk Directory provides access to both offline and online tags through a tag browser. If a network is not connected to processors, developers can work with offline tags. When working offline, the system reads data points directly from the processor’s project file defined in RSLogix™. Developers can build an entire application offline, without connecting to processors, and then deploy the application to a runtime system later.

If a network is connected to processors, access to online tags is also available. In this case, the system communicates directly with the processor itself and reads data points as they are defined in that processor.

For example, as a control programmer develops a ladder program and creates data points, an HMI engineer might be developing a visualization program in RSView Studio while a database analyst works with RSBizWare™ RSSql™. When the control programmer saves a ladder file, RSLogix automatically updates offline topics in RSLogix™, and the offline data points become available to the RSView and RSSql programmers through the FactoryTalk Directory, even before the ladder project is downloaded to the processor. To use the new data items, RSView and RSSql programmers simply refresh the folders in their tag browsers.
With FactoryTalk Directory, tags need not be recreated or imported in a separate tag database. Changes to tag values and properties update automatically wherever they are used within RSView Supervisory Edition applications and throughout the FactoryTalk-enabled system.

**Memory tags**

Many RSView32 projects use memory tags to store the information to be shown on graphic displays. For example, memory tags might contain variable string or numeric values, such as the graphic display name, current user name, or calculation results. When running RSView32 on a single computer, memory tags act as local variables on that computer. In an RSView32 Active Display System, memory tags act as global variables and are shared across clients.

Memory tags are still available in RSView SE and operate as global variables, just as they do in an RSView32 Active Display System. In a distributed RSView SE system, a single graphic display can be created once, stored on the server, and then accessed on any client. To use variables that are local, rather than global, and specific to each client that accesses a graphic display, use Microsoft® Visual Basic® for Applications (VBA) code, which runs independently on each client.

**VBA and the RSView object model**

Microsoft Visual Basic for Applications (VBA) is an integral part of both RSView32 and RSView Supervisory Edition. However, the way in which VBA is implemented, and the types of object models exposed, are significantly different between the two products. Because of these differences, RSView32 VBA code does not migrate to RSView SE, although VBA code that does not use the RSView32 object model can be copied and reused in RSView SE.
RSView32 has a server-side object model. VBA code triggers actions that run on the server, and not on individual clients. The RSView32 object model supports creating, deleting, and modifying HMI tags, but does not offer any objects, properties, methods, or events for manipulating graphic displays.

RSView Supervisory Edition uses a very different client-side object model. The purpose of the RSView SE object model is to provide flexible access to graphic display objects, and to make it easy to customize and manipulate graphic displays on individual client computers. VBA code, attached to an associated graphic display, triggers actions that run on the client computer and not on the HMI server. The RSView SE object model supports reading and writing values from both directly referenced tags and HMI tags, but does not support creating, deleting, or modifying tags.

When a graphic display opens on an RSView SE client computer, the VBA code executes independently on that client. For example, suppose the same graphic display is opened on two different client computers at the same time. When an operator on the first computer clicks a button on the display, the action causes the VBA code to run. The same display, open on another client computer, does not execute the VBA code until an event on that client triggers it.

Use VBA code to customize the behavior of graphic displays on individual client computers. To allow for independent, local user interaction on each client, replace memory tags and event expressions used in RSView32 graphic displays with VBA code in RSView SE displays.

**Derived tags**

Just as is true in RSView32, in RSView Supervisory Edition a derived tag is an analog, digital, or string tag whose value is determined through the evaluation of an expression. Derived tags set up in RSView32 projects convert successfully in RSView SE applications. However, because RSView SE is a distributed system, be sure to configure derived tags to start when the HMI server starts or when a process starts—and not when a client starts. This guarantees that the derived tags are available for any client that needs them.

**Alarm logging and data logging**

Both RSView32 and RSView Supervisory Edition support logging to native log files and logging directly to an ODBC data source, such as Microsoft SQL Server™, Oracle®, or SyBase®. The ODBC format is same, whether alarms and data are logged from RSView32 or from RSView Supervisory Edition.

While both products support logging to an ODBC database, RSView32 and RSView Supervisory Edition use different native log file formats. As a result, historical data from native RSView32 alarm log and data log files do not convert to RSView SE. You cannot open, view, or log to RSView32 historical log files in RSView SE.

In RSView SE, the native alarm and data logs use an enhanced, proprietary .dat format rather than the standard .dbf format used in RSView32. The .dat format is faster and requires less disk space, but it is not compatible with RSView32’s .dbf format. To view log files in the .dat format,
you must use RSView SE viewers; the .dat format is not compatible with RSView32 or with third-party database tools.

Activity logging

In RSView Supervisory Edition, activity logging is enabled by FactoryTalk Diagnostics. Part of the underlying FactoryTalk platform, FactoryTalk Diagnostics collects activity and diagnostic messages from all participating products throughout a distributed automation system and routes them to a variety of logging destinations, including an ODBC database and a local log on each computer.

In RSView SE, activity logging is no longer concerned with only the current HMI project on the local computer. Instead, FactoryTalk Diagnostics makes it possible to monitor system-wide operator activity, diagnose system-wide error, warning, and information messages, and track system load. FactoryTalk Diagnostics routes data for logging on every computer on which it is running, for all FactoryTalk-enabled products installed on that computer.

To configure and view activity messages in RSView SE, use the new FactoryTalk Diagnostics tools.

Trend charts

RSView32 offers two types of trend charts: a native trend object and a newer TrendX object. RSView Supervisory Edition does not support the RSView32 native trend object, but does support the TrendX object. Trend charts based on the TrendX object convert successfully from RSView32 to RSView SE.

In addition, because RSView SE attaches VBA code to graphic displays, it is now possible to use the TrendX object model to customize and manipulate trend charts displayed on individual client computers during runtime.

Events

An Events component is available in both RSView32 and in RSView Supervisory Edition. In both products, the Events component monitors runtime events and triggers specified actions defined by an Events editor. RSView32 Event files convert successfully to RSView SE Event files. In RSView SE, however, the Events component runs only on an HMI Server and executes only server-side commands, such as tag writes, DerivedOn, DataLogOn, and alarming commands. Events do not trigger and run on client computers.
To cause events to trigger from graphic displays, use VBA code instead. For example, suppose a project includes a graphic display that pops up with operator instructions when a tank reaches a certain fill level. In RSView32, the popup display might be driven by an expression defined in the Event file. In RSView SE, create a numeric display object and write VBA code that executes when the numeric display changes. A numeric display shows tag values, but it can also include expressions, like those used in the RSView32 Event file.

**Startup configuration**

RSView Supervisory Edition does not convert the RSView32 startup configuration file. You must reconfigure startup preferences in RSView SE.

**Other migration considerations**

When planning to reuse an RSView32 project in an RSView Supervisory Edition application, keep these additional considerations in mind.

**Communications options**

RSView32 requires purchasing RSLinx Professional or RSLinx Gateway separately. RSView Supervisory Edition does not require a separate RSLinx purchase; RSLinx for RSView is included for no additional charge with RSView SE Server and RSView SE Station software. RSLinx for RSView, however, does not allow connections to remote third-party clients. If you need that functionality, use either RSLinx Professional or RSLinx Gateway.


**RSView32 extensions not supported**

RSView32 extensions are not currently supported in RSView Supervisory Edition. This includes RSView32 Messenger™, RSView32 RecipePro™, RSView32 SPC™, and RSView32 WebServer™. Similar functionality is planned for RSView Supervisory Edition, but is not yet available.

**RSView Supervisory Edition advantages**

RSView Supervisory Edition provides a number of advantages that are not available with RSView32 software. A few of these advantages are introduced below.

*Shared editor and compatibility with RSView Machine Edition*

In addition to RSView Supervisory Edition, the RSView Enterprise Series also includes RSView Machine Edition™, a machine-level HMI product for monitoring and controlling individual machines and small processes. RSView Studio provides a common development system used for creating both machine-level projects and supervisory-level applications. With RSView Studio, entire machine-level applications can be imported into supervisory-level...
applications, and individual machine-level components can be dragged and dropped right into supervisory projects.

Edit applications online

RSView Studio allows editing components of a project online, while the application is running. The RSView Supervisory Edition system inherits changes to the control system automatically, without having to shut down running processes or make separate configuration changes. For example, if a processor is moved or a tag structure is changed in RSView Studio, the changes are immediately reflected in the RSView SE runtime application.

Edit applications from remote locations

For distributed applications, RSView Supervisory Edition allows HMI engineers to create and deploy applications remotely, from any computer on the network. Multiple developers can access the application simultaneously, reducing overall development time.

Built-in failure detection and recovery

RSView Supervisory Edition HMI servers, RSLinx and other OPC Data Access 2.0 servers, and FactoryTalk Directory servers can all be configured to run on both primary and secondary computers. Once configured, each redundant pair of servers is available to all FactoryTalk-enabled clients within the system. Adding additional clients does not require any additional configuration.

During runtime, if the connection to a primary computer fails, FactoryTalk automatically switches all clients to a backup computer within 30 seconds, and can automatically switch them back to the primary computer when the connection is restored—all without any custom programming, client configuration, or operator intervention.

Audit trails, electronic signatures, and regulatory compliance

RSView Supervisory Edition offers a number of features that make it easier to comply with government regulations, including the U.S. government’s 21 CFR Part 11 specification. Complying with industry and government regulations often requires capturing and archiving operator actions and changes that occur to a running system. FactoryTalk Diagnostics enables this capability for not only RSView Supervisory Edition applications, but also for all FactoryTalk-enabled products participating in an industrial automation system.

Some critical operations, such as set point changes, RSView commands, and recipe downloads, require that an operator’s identity be verified before proceeding. RSView Supervisory Edition can require an operator, and an optional approver, to enter a username and password before performing an operation. Operator activity and system changes are then logged through FactoryTalk Diagnostics.

For more information


- RSView Supervisory Edition Technical Data
- FactoryTalk press release
- FactoryTalk-Enabled Solutions brochure
- FactoryTalk Spoken Here: Capabilities Overview brochure
- Industry White Paper: Ensuring system availability in RSView Supervisory Edition applications
Rockwell Software

For more information about the latest pricing or a demonstration of any Rockwell Software package, please contact your local Rockwell Automation Sales office or Allen-Bradley distributor. For the very latest about Rockwell Software products, visit us at:

www.software.rockwell.com