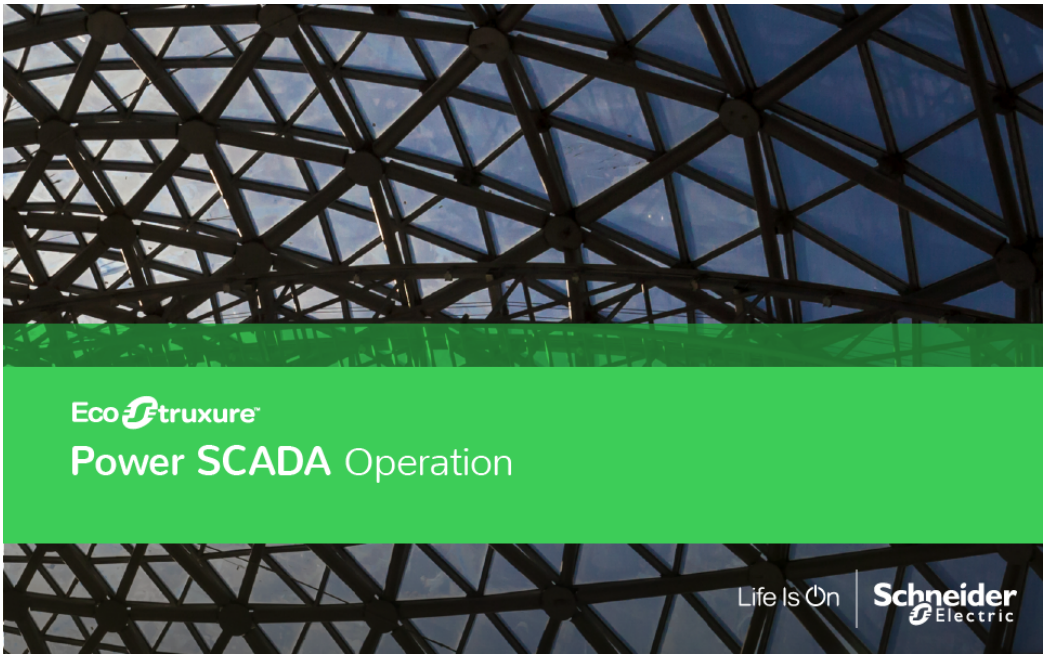
EcoStruxure™

Power SCADA Operation 9.0 with Advanced Reporting and Dashboards

System Guide

7EN02-0413-00

09/2018



Schneider Electric

Legal Information

Safety Information

Important Information

Read these instructions carefully and look at the equipment to become familiar with the device before trying to install, operate, service or maintain it. The following special messages may appear throughout this bulletin or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.

The addition of either symbol to a "Danger" or "Warning" safety label indicates that an electrical hazard exists which will result in personal injury if the instructions are not followed.

This is the safety alert symbol. It is used to alert you to potential personal injury hazards. Obey all safety messages that follow this symbol to avoid possible injury or death.

⚠ DANGER

DANGER indicates a hazardous situation which, if not avoided, **will result in** death or serious injury.

⚠ WARNING

WARNING indicates a hazardous situation which, if not avoided, **could result in** death or serious injury.

⚠ CAUTION

CAUTION indicates a hazardous situation which, if not avoided, **could result in** minor or moderate injury.

NOTICE

NOTICE is used to address practices not related to physical injury.

Please Note

Electrical equipment should be installed, operated, serviced and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

A qualified person is one who has skills and knowledge related to the construction, installation, and operation of electrical equipment and has received safety training to recognize and avoid the hazards involved.

Contents

Safety Precautions

During installation or use of this software, pay attention to all safety messages that occur in the software and that are included in the documentation. The following safety messages apply to this software in its entirety.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Do not use the software or devices for critical control or protection applications where human or equipment safety relies on the operation of the control action.
- Do not use the software to control time-critical functions.
- Do not use the software to control remote equipment without proper access control and status feedback.

Failure to follow these instructions can result in death or serious injury.

⚠ WARNING

INACCURATE DATA RESULTS

- Do not incorrectly configure the software or the devices.
- Do not base your maintenance or service actions solely on messages and information displayed by the software.
- Do not rely solely on software messages and reports to determine if the system is functioning correctly or meeting all applicable standards and requirements.
- Consider the implications of unanticipated transmission delays or failures of communications links.

Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

⚠ WARNING

POTENTIAL COMPROMISE OF SYSTEM AVAILABILITY, INTEGRITY, AND CONFIDENTIALITY

- Use cybersecurity best practices to help prevent unauthorized access to the software.

Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

Work with facility IT System Administrators to ensure that the system adheres to the site-specific cybersecurity policies.

Introduction

The *Power SCADA Operation 9.0 – System Guide* describes the procedures involved in creating a power SCADA monitoring and control system.

Use this guide as a reference to help you prepare, and develop the Power SCADA Operation with Advanced Reporting and Dashboards project that you are deploying. For related information, or where more detail is required, references are made to existing documentation.

NOTE: This guide does not discuss the planning, design, and operation of the electrical power system that is being monitored.

How this guide is organized

The content in this guide is organized into the following functional life-cycle stages:

- "Planning" on page 27
- "Installing and upgrading" on page 91 (includes Licensing)
- "Configuring" on page 147
- "Administering" on page 464
- "Operating" on page 467
- "Troubleshooting" on page 502
- "Reference" on page 525

Reference is a resource chapter that contains detailed information. Use Reference information to deepen your understanding of Power SCADA Operation concepts, and to complete complex procedures that might require additional information. The Reference section content is organized to mirror the structure of the functional life-cycle stages.

Some tools, tasks, or functions are specific to a particular stage, others are part of different stages. For example, setting up Power SCADA projects is done during the Configuration stage. In contrast, basic reports has a Configuration component and a Operating component.

Document updates

This guide is also available online through the Exchange Extranet (access is limited and registration is required.) We may update the online version over time to improve clarity and accuracy. If you see differences between your local copy and the online version, use the online version as your reference. See "Support contacts" on page 26 for contact information.

Assumptions

This guide is intended for application engineers, system integrators, and other qualified personnel that understand and have experience with power SCADA and monitoring systems.

NOTICE
INOPERABLE SYSTEM Ensure that you have received Power SCADA training and understand the importance of the Power SCADA Operation productivity tools and workflows. **Failure to follow these instructions can result in overly complex projects, cost overruns, rework, and countless hours of support troubleshooting.**

NOTE: Power SCADA Operation is build on Citect Studio and includes productivity tools that are designed and optimized to create the tags you need to configure power-based SCADA projects. If you have prior experience using Citect Studio, do not rely exclusively on Citect tools to build a power SCADA project.

Qualified personnel will:

- Have received Power SCADA Operation training
- Understand how to install the various devices used in the project, and how to install the Power SCADA Operation with Advanced Reporting and Dashboards software
- Have completed necessary reading and will have made decisions regarding architecture and hardware specifications.

This content assumes that the system will include:

- Power SCADA Operation with Advanced Reporting and Dashboards
- Extract, Transform, Load module (ETL): the ETL Administration Tool for Power SCADA Operation with Advanced Reporting and Dashboards
- Power SCADA Anywhere (also referred to as CitectSCADA Anywhere)

What's New

Power SCADA Operation 9.0 with Advanced Reporting and Dashboards is a major release that introduces a number of new features and improvements. We highly recommend you upgrade your existing Power SCADA Operation system to version 9.0.

Highlights of the 9.0 release

- New Diagnostics Viewer that lets you view the health of your system and troubleshoot system settings during configuration and at runtime. See "Diagnostics Overview" on page 502 for details.
- New Notifications Settings: See "Notifications" on page 325 for details.

- SQL Server database is no longer required for Power SCADA Operation 9.0 (it is required for Power SCADA Operation 9.0 with Advanced Reporting and Dashboards)
- Support for PSO clustering
- M580 PLC time-stamped event capture: See "OFS system time stamping" on page 435 for details.
- The user interface is translatable (localizable): see "Localizing Power SCADA Operation" on page 374 for details.
- Expanded device support: See "Supported power devices" on page 48 for details.
- Power SCADA Anywhere supports Windows Active Directory Groups
- ETL performance improvements: See "PSO to PME ETL job performance" on page 400 for details.
- Power SCADA Operation 9.0 is built on Citect SCADA 2018. See Citect SCADA 2018 help for more information on the new features.

Resources

NOTE: For a list of commonly used terms, see the "Glossary" on page 656.

Download Center

The following Power SCADA Operation 9.0 with Advanced Reporting and Dashboards documents are available on the Schneider Electric Download Center:

- *Power Monitoring Expert 9.0 System Guide* (English) – Document number 7EN02-0411
- *Power Monitoring Expert 9.0 Web Applications Guide* (Multilingual) – Document number 7EN02-0409

Exchange Extranet

- EcoStruxure Power SCADA Operation portal (product demos, videos, and other product content)
- Power SCADA Operation 9.0 with Advanced Reporting and Dashboards Design and Quote tools:
  - PSO Software Assurance Calculator
  - PSO Database Growth Calculator
  - PSO Commissioning Time Calculator
  - And others.
- Power SCADA Operation 9.0 with Advanced Reporting and Dashboards Install and Maintain documents:

- *Power SCADA Operation 9.0 with Advanced Reporting and Dashboards – IT Guide* (English) – Document number 7EN42-0169
- *Power Monitoring Expert 9.0 – IT Guide* (English) – Document number 7EN42-0168
- *Power Monitoring Expert* 9.0 – *System Guide* (English) – Document number 7EN02-0411

Power SCADA Anywhere

- *Power SCADA Anywhere Server Installation and Configuration Guide*
- *Power SCADA Anywhere Web Client User Guide*
- *Power SCADA Anywhere Quick Start Guide*

McAfee

- (McAfee) Advanced-Parameter-EmbeddedControl-v6
- (McAfee) Installation-Guide--v6.2.0
- (McAfee) Command Line Interface Guide
- McAfee-Embedded-Control-Code-Signing-Guide v1/2
- (McAfee) Product-Guide-v6.2.0

Tofino ConneXium

- Tofino ConneXium TCSEFEA Installation Manual V1
- Tofino ConneXium TCSEFEA User Manual V1

Manuals

In addition to this help manual, the following documents – located on the installation disk – also provide helpful information:

- Citect SCADA Help
- Vijeo Citect 2016 Web Client Guide
- Vijeo Citect Installation Guide
- Release Notes: Includes information specific to this release of the product
- Readme file: Includes late-breaking information about this release.

Help files

In addition to the help file released with this product, there are several related help files. They are located in C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin

Support contacts

Use the My Schneider App or contact your local country office.

.

Planning

Use the information provided in this chapter to prepare for an installation or upgrade of a Power SCADA Operation system.

Use the links in the table to find the content you are looking for.

Topic	Content
"Overview" on page 28	An overview of Power SCADA Operation and how it differs from non-power SCADA applications.
"Components and architectures" on page 30	Design considerations and sample architectures for the Power SCADA Operation components
"Distributed architectures" on page 46	Design considerations for distributed architectures, including time synchronization and clustering.
"Supported devices and protocols" on page 48	Detailed information on supported drivers (native, 3rd party, other devices, and Citect drivers), supported protocols, and waveform file share access.
"Hardware and software requirements" on page 51	Detailed information on Server and Client CPU, RAM, and disk storage requirements, as well as supported operating systems, SQL Server editions, browsers, and virtual environments.
"Device response time" on page 58	recommendations on using Ethernet and Serial communications
"Cybersecurity" on page 59	Recommendations on how to help secure your system from a malicious cyber attack.
"Advanced Reporting and Dashboards" on page 66	Customizing advanced reports and design considerations for device communication in Power SCADA Operation with Advanced Reporting and Dashboards.
"OFS system time stamping" on page 435	Architectural guidelines for implementing system time stamping
"Licensing" on page 79	Licensing information and options
"Localization" on page 83	Design considerations for localizing Power SCADA
"Integrating with other systems" on page 84	Integrating Power SCADA Operation with EcoStruxure Buidling Operation and OPC DA
"Extending Power SCADA" on page 89	Extending Power SCADA Operation through Cicode scripting, CtAPI, and other extensibility points

Overview

Power SCADA Operation is uniquely designed to let you leverage the power of a SCADA for Power Management Applications.

Power SCADA Operation with Advanced Reporting and Dashboards enables the Facilities Team in Power Critical Facilities to monitor, control, and troubleshoot issues in real-time with their electrical distribution systems.

Power SCADA Operation features

Power SCADA Operation with Advanced Reporting and Dashboards includes the following features:

Power Monitoring & Alarming

- High performant real-time communications
- Native system redundancy and scalable architecture
- Extensive protocol support and open data exchange
- Highly customizable with scripting and an open API
- Cyber resilient networks and servers

Source Control

- Monitor complex auto-transfer schemes
- Remotely and safely control breakers

Avoid Disruption via Events Analysis

- Default, rich data integration for connected devices such as Masterpact MTZ, ION9000, PM8000, etc...
- 1 ms Sequence of Events Recording (SER)
- Power Quality Waveform Analysis (COMTRADE)

For designers with a Citect background

Engineers developing Power SCADA with a background in Citect SCADA and process automation may be unaware of the critical importance of the differentiated Power SCADA development tools and the Power Applications that Power SCADA is used for.

NOTICE
INOPERABLE SYSTEM Ensure that you have received Power SCADA training and understand the importance of the Power SCADA Operation productivity tools and workflows. **Failure to follow these instructions can result in overly complex projects, cost overruns, rework, and countless hours of support troubleshooting.**

NOTE: Power SCADA Operation is build on Citect Studio and includes productivity tools that are designed and optimized to create the tags you need to configure power-based SCADA projects. If you have prior experience using Citect Studio, do not rely exclusively on Citect tools to build a power SCADA project.

Ensure that you and your engineers are aware of Power SCADA Operation's unique tooling and workflows. The following features only are supported using Power SCADA tooling and workflows:

- Notifications Settings
- Interoperability with Advanced Reporting
- Interoperability with EcoStruxure™ Building Operation
- LiveView
- Basic Reports
- Advanced one-line configuration
- Power SCADA power graphics libraries
- I/O Device Manager

Components and architectures

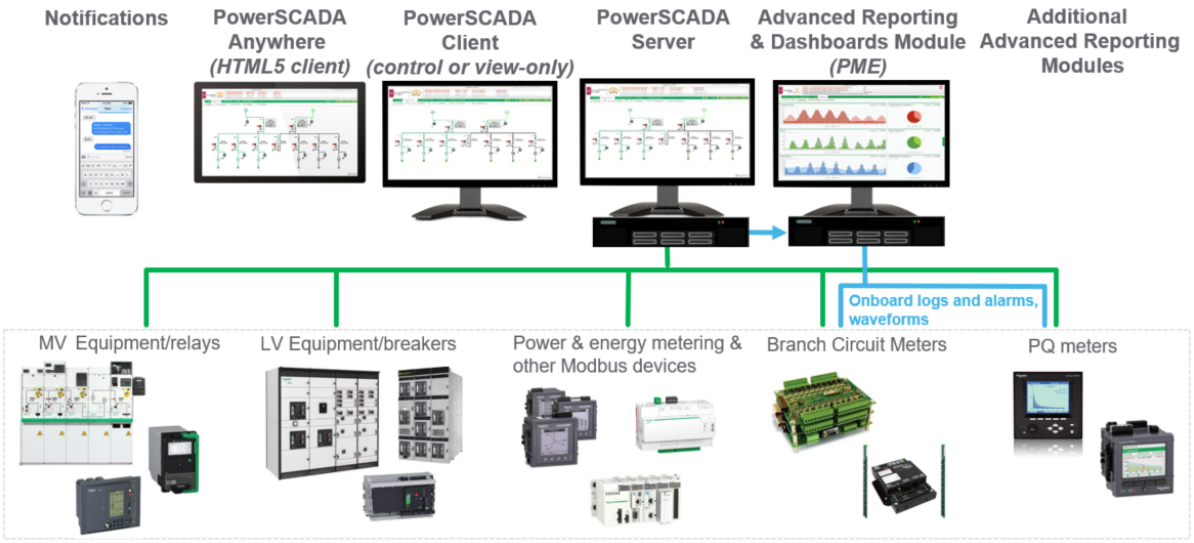
This section provides information on the design considerations for the Power SCADA Operation 9.0 with Advanced Reporting and Dashboards components as well as component architectures.

Use the links in the table below to find the content you are looking for:

"Components overview" on page 31	An overview of the Power SCADA Operation with Advanced Reporting and Dashboards components and data flow
"Power SCADA Server Component" on page 31	Server component purpose, licensing options, design considerations, and how points are calculated
"Server Component Architecture" on page 33	Standalone and redundant server component architectures and data flows
"Power SCADA Client Component" on page 34	Client component purpose, licensing options, design considerations,
"Client Component Architecture" on page 35	Client component architectures and data flows
"Power SCADA Anywhere Component" on page 37	Power SCADA Anywhere component purpose, licensing options, design considerations,
"Power SCADA Anywhere architectures" on page 38	Power SCADA Anywhere component architectures and data flows
"Advanced Reporting and Dashboards Component" on page 41	Advanced Reporting and Dashboards component purpose, licensing options, design considerations
"Advanced Reporting and Dashboards architectures" on page 41	Advanced Reporting and Dashboards component architectures and data flows
"Additional Advanced Reporting Modules Component" on page 44	Component purpose, licensing options, design considerations, description of modules
"Mapping EcoStruxure Power to Advanced Reporting Modules" on page 44	How EcoStruxure Power applications map to the Advanced Reporting and Dashboards Modules.

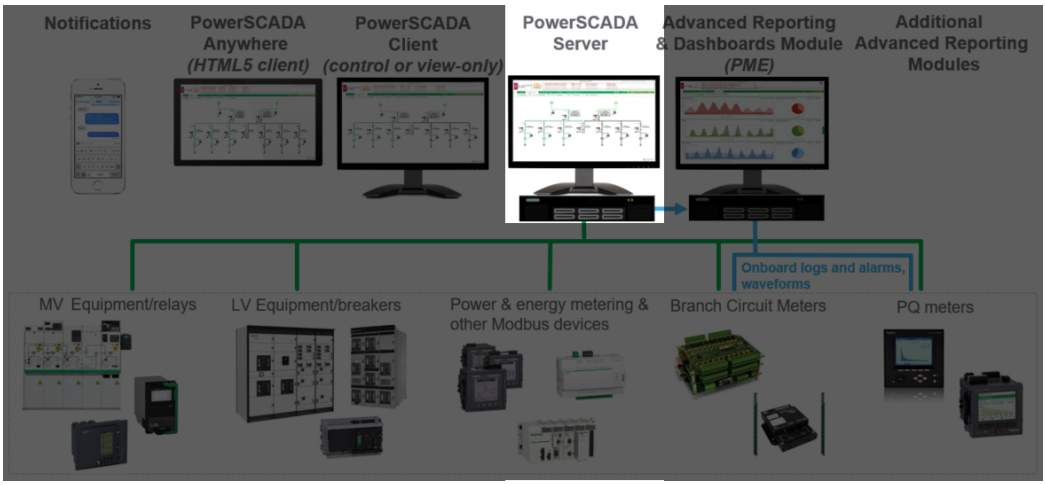
Components overview

Power SCADA Operation with Advanced Reporting and Dashboards is comprised of the following components:



Refer to the topics in this section for detailed information on component purpose, licensing options, design considerations, and architectures,

Power SCADA Server Component



Purpose

The Power SCADA Server is the required base component of any Power SCADA system responsible for data acquisition, alarming and trending of historical data. The Server includes:

- Power SCADA engineering tool suite
- Open data exchange protocols/tools (OPC DA client/server, OPC AE server, EcoStruxure Web Services (EWS) for interoperability w/ SBO, CtAPI)

- Device drivers (Modbus, ION, IEC-61850 master, IEC 60870-5-104 master, BACnet/IP master, SNMP v.2, etc.)
- Basic reporting (Trend, Tabular, Single Device Usage and Multi-Device Usage reports)

Licensing options

Licensed by number of points/tags (options include: 500, 1500, 5000, 15000 and Unlimited tags). For more information on licensing, see "Licensing" on page 79.

Design considerations

- Server redundancy achieved by licensing additional Servers in the design.
- Server license also includes one control client license, which can run on the same machine as the Server.

How points are calculated

The compiler does not generate any static point count any more. CitectSCADA 7.0 counts all I/O device addresses dynamically at runtime. This includes all tags used by alarms, trends, reports, events, pages, in Super Genies, use of the TagRead() and TagWrite() Cicode functions, or read or written to using DDE, ODBC, or the CTAPI. A particular variable tag is only counted towards your point count the first time it is requested. That is, even though you may have configured a certain tag on a particular page in your project, unless you navigate to that page and request the data, the variable tag will not be counted towards your point count.

In addition to this, the following changes were made to the licensing structure in CitectSCADA 7.0:

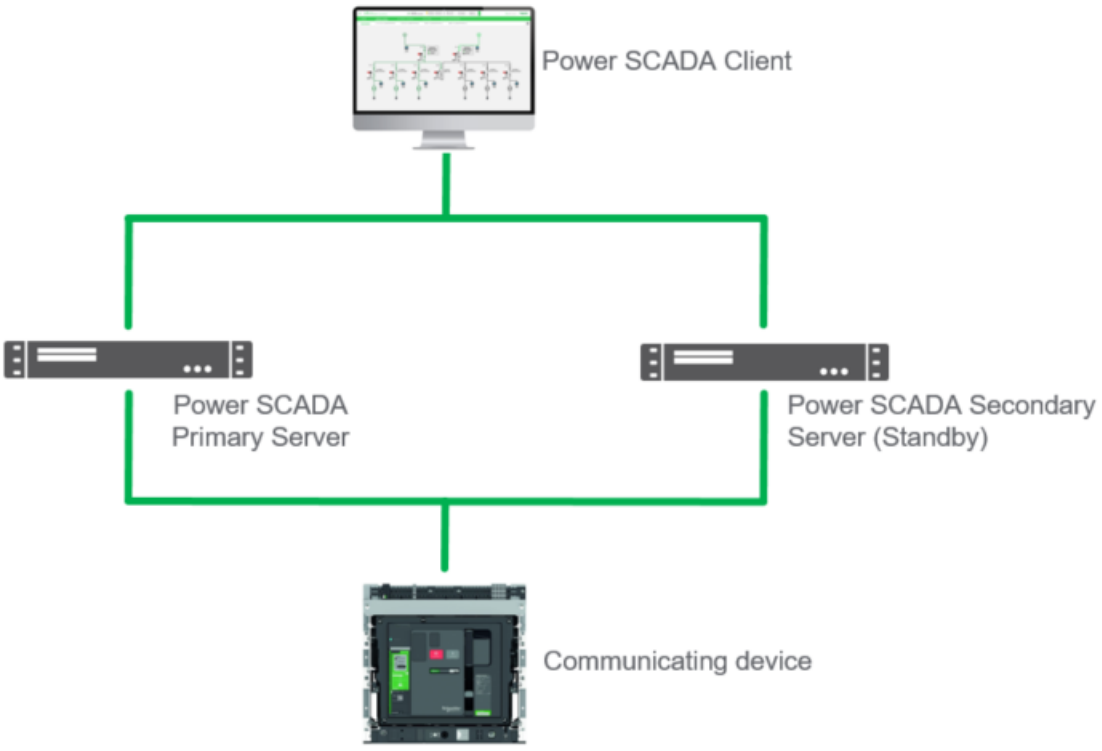
- I/O point count is now tag based not address based. For example, two tags that use the same PLC address will be counted twice. If two trend tags use the same variable tag, it will be counted once. The same applies to alarms.
- For the multi-process mode, each server component will accumulate its own point count. The server component point count is the count added up from all server components. If two server components use the same tags, say alarm and trend, the tags will be counted twice when the point count gets summed.
- For the multi-process mode, the client component will also accumulate its own point count including super genie and CTAPI tags.
- For the multi-process mode, the machine point count will be the point count on the client component or the point count added up from all server components, whichever is bigger. For example, if the total point count for all server components is 100, and the client component point count including CTAPI and super genies is 95, the kernel "General" window will show 100. If the client component point count reaches 120 later and the server component point count still remains 100, the kernel "General" window will show 120.
- Reading properties of a tag with TagGetProperty() will cause that tag to be included in the point count, even if the value is not read.
- Writing to local variables or disk IO variable tags via OPC etc will also increase the point count. For example, if you use an OPC client to write to a local variable, each local variable will be counted once, the first time it is used.

Server Component Architecture

Native architectural redundancy

Power SCADA supports full server redundancy and full communication redundancy. When the Primary Server becomes unavailable, the Standby Server automatically takes over in 2 to 3 seconds.

There is also full data synchronization between servers and historical backfill. If primary goes down and a secondary becomes active, when the primary returns to active state the secondary fills in the primary with any missed information.



NOTE: Multiple NICs are supported on each server and a device may have two communication paths.

Making changes while online

Certain changes and updates to a production Power SCADA system require a restart of the Power SCADA Server processes. For example:

- Adding and removing devices
- Adding and removing tags

For this reason, if the customer requires changes to be made without interruption of service (restarting Power SCADA Server), a redundant architecture is required.

In a redundant architecture, changes can be made without interrupting service by:

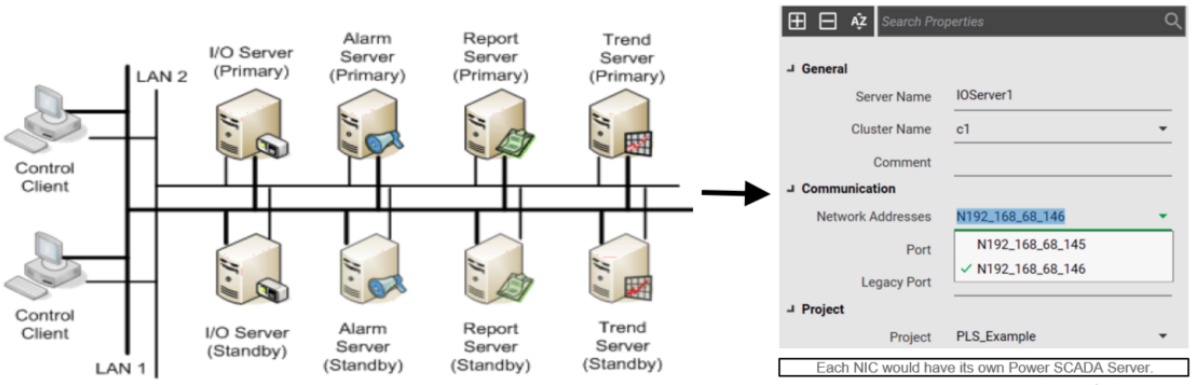
1. Making a change on Secondary Server
2. Restarting Secondary Server
3. Making the updated Secondary Server the Primary Server

Ethernet network redundancy

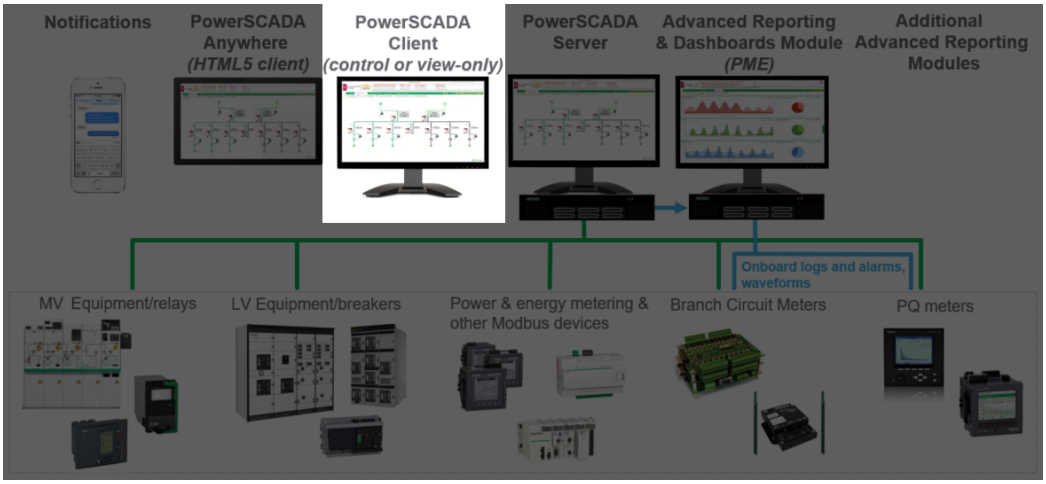
When network redundancy is being considered

Most common approach: Second LAN in parallel to first

If LAN1 becomes inoperative, components will maintain connection using LAN2



Power SCADA Client Component



Purpose

The Client is an optional component that allows operators to access the Power SCADA runtime from a machine other than the Server machine. Clients can be run as either a Windows desktop application or as an Internet Explorer web client. Power SCADA has two types of clients: The Control Client can be used to perform control and/or acknowledge alarms, while the View-only Client can only view the runtime (no control or alarm acknowledge rights).

Licensing options

Control Client is licensed by # of points/tags (options include: 500, 1500, 5000, 15000 and Unlimited tags), while the View-only Client is licensed for Unlimited points/tags only. For systems with Server redundancy, it is recommended to license an equal number of redundant control client licenses for the stand-by Server. For more information on licensing, see "Licensing" on page 79.

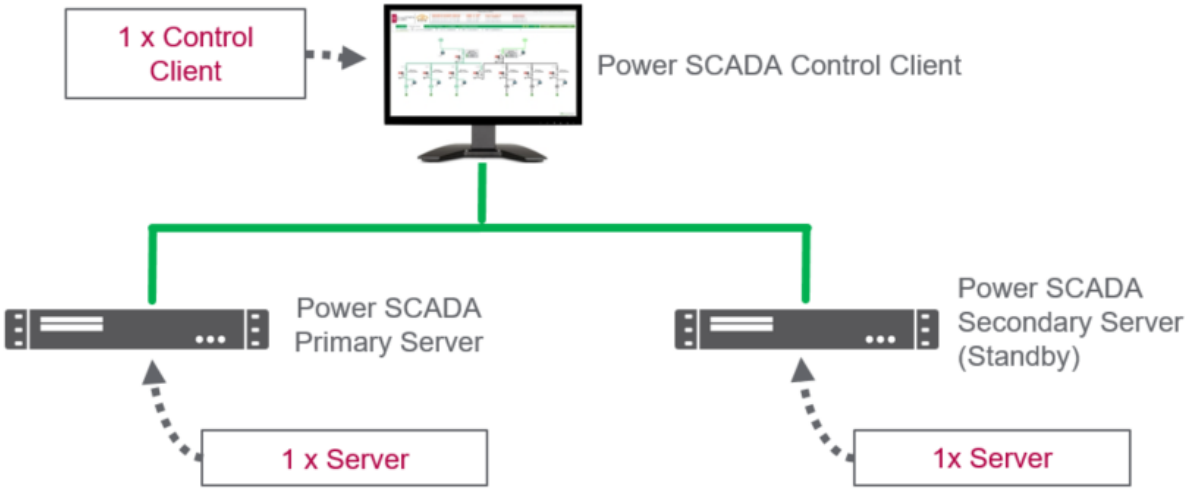
Design considerations

Clients can use either a floating license model (i.e. limited # of licenses can be shared between a number of concurrent users/computers) or a static license model (i.e. license reserved for set number of computers).

Client Component Architecture

Architecture #1: Server redundancy with static Control Client

The following example architecture illustrates server redundancy with static Control Client:



Server redundancy is achieved by installing and licensing a secondary Power SCADA server with the same point/tag count as the primary.

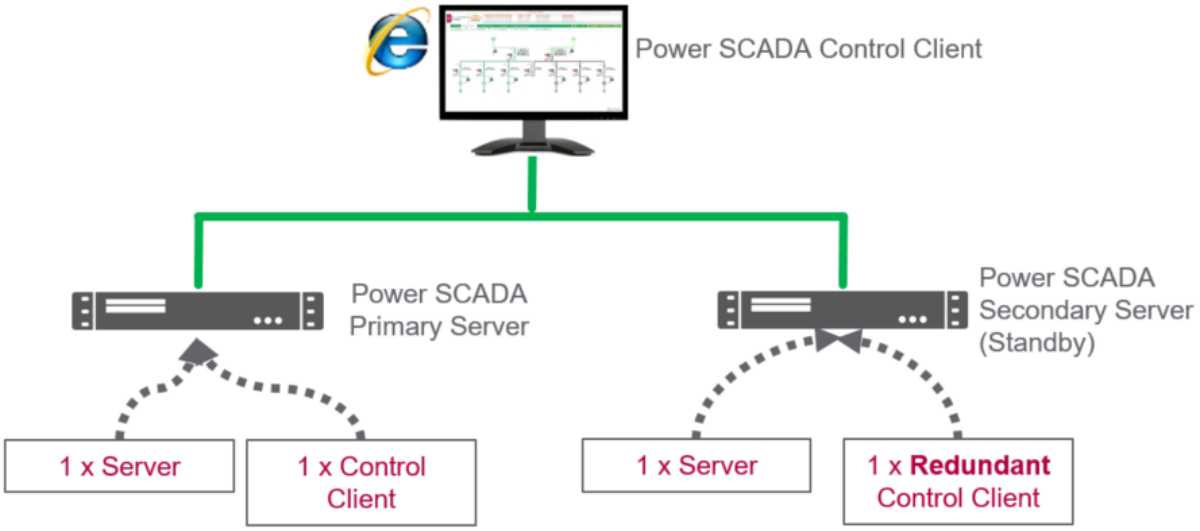
Server software & licenses are installed on the Primary & Secondary Server machines.

Control Client software and license is hosted on Client machine.

By placing Client license on Client machine, the Control Client would be limited to Window desktop application access only instead of a web client access due to static license model being used.

Architecture #2: Server redundancy with floating Control Client

The following example architecture illustrates server redundancy with floating Control Client:



Server software & licenses are installed on the Primary & Secondary Server machines.

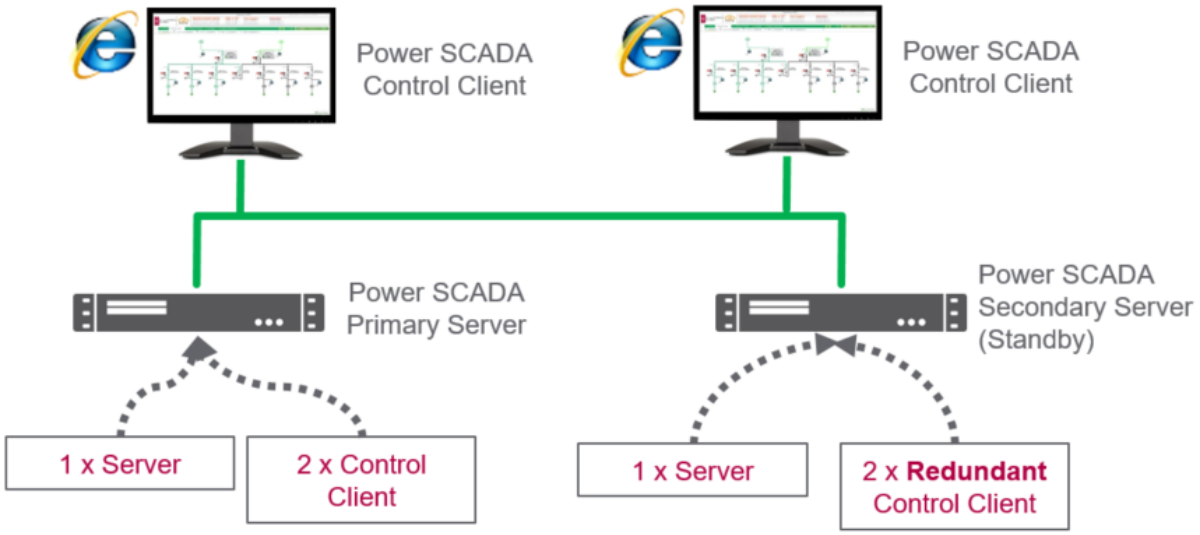
By placing Client license on Server machine (floating license model), Control Client could be accessed via web client or Windows desktop application.

Client connectivity limited to 1 simultaneous connection due to having 1 Client license.

NOTE: The secondary Server hosts a Redundant Control Client license instead of the standard Control Client license.

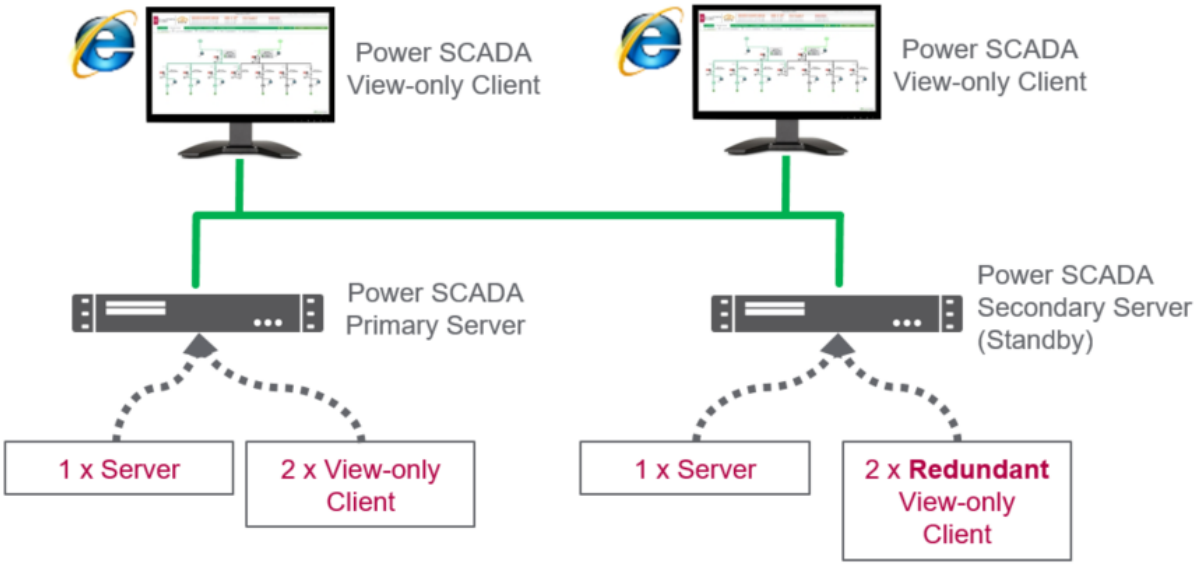
Architecture #3: Server redundancy with 2 floating Control Clients

The following example architecture illustrates server redundancy with 2 floating Control Clients:



Architecture #4: Server redundancy with 2 floating View-only Clients

The following example architecture illustrates server redundancy with 2 floating View-only Clients:



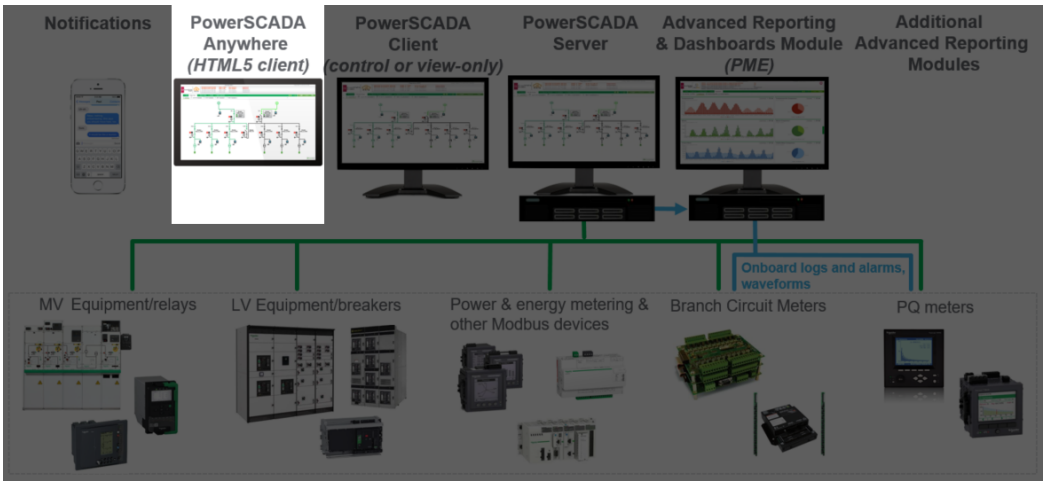
Server software & licenses are installed on the Primary & Secondary Server machines.

By placing Client license on Server machine (floating license model), Control Client could be accessed via web client or Windows desktop application.

Client connectivity limited to 2 simultaneous connections due to having 2 Client license.

NOTE: The secondary Server hosts 2 Redundant Control Client licenses instead of the standard Control Client license.

Power SCADA Anywhere Component



Purpose

Power SCADA Anywhere is an optional component. It is an HTML5 streaming application that allows for the visualization of the Power SCADA runtime from any HTML5 compliant browser (Edge, Chrome, Firefox, etc) by streaming a remote desktop application from a Control Client or View-only Client.

Licensing options

Each Power SCADA Anywhere license allows up to 5 concurrent connections to the runtime via HTML5 web browsers. For more information on licensing, see "Licensing" on page 79.

Design considerations

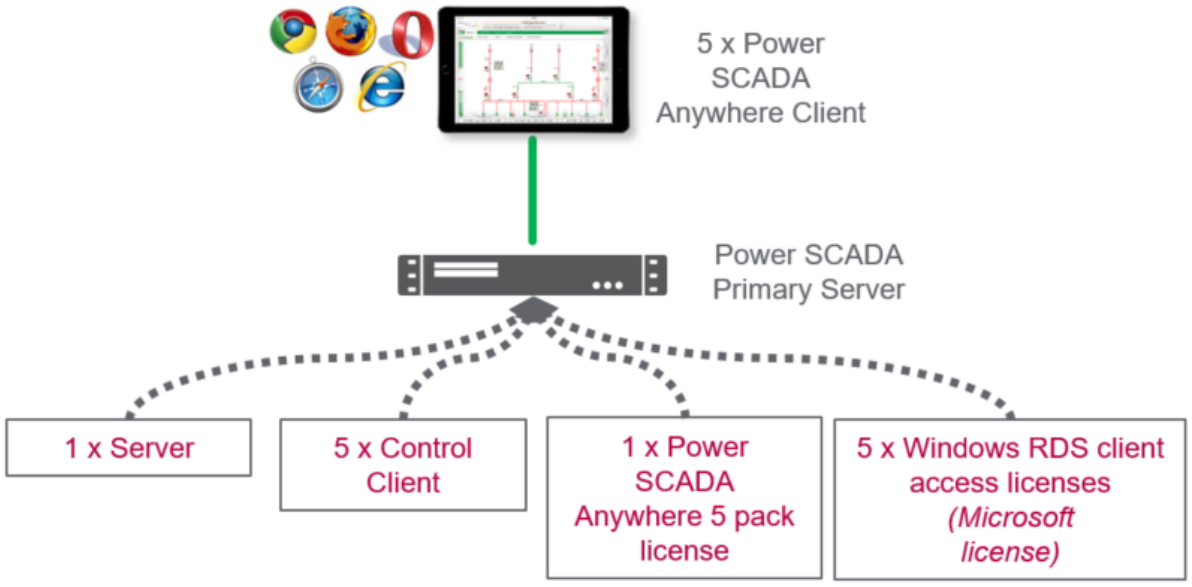
- Power SCADA Anywhere requires an equal number of Power SCADA Control Clients or View-only Clients to be licensed.
- Power SCADA Anywhere requires a domain to exist in order to use Windows Remote Desktop licenses. Note: The Power SCADA Anywhere host may be a domain controller.
- Since Power SCADA Anywhere uses Windows remote desktop connections, it requires an equal number of Windows Remote Desktop Services (RDS) client access licenses (CAL), formerly known as Terminal Services, to be purchased. This can be purchased from similar 3rd party vendors that you purchase Windows OS software from.

Power SCADA Anywhere architectures

NOTE: Power SCADA Anywhere uses Windows Remote Desktop Services licenses. Also, Power SCADA Anywhere requires a domain. Power SCADA Anywhere host may be a domain controller.

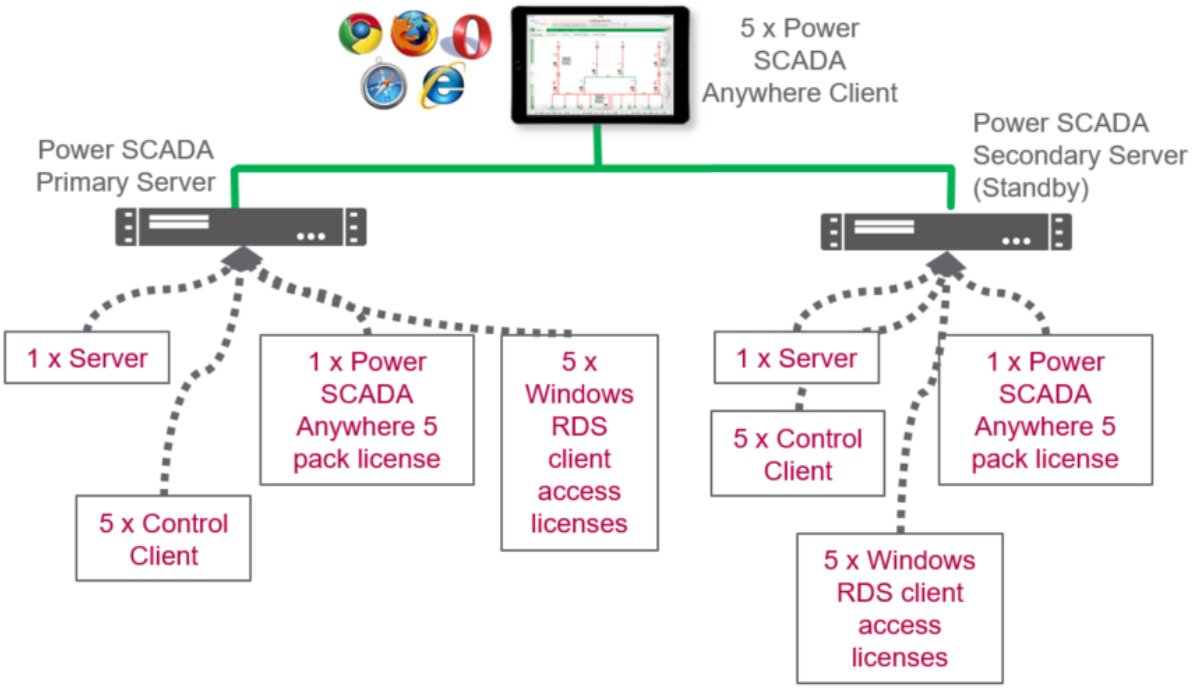
Architecture #1: Power SCADA Anywhere without redundancy

The following example architecture illustrates the simplest Power SCADA Anywhere architecture. All software and licenses are installed on the Server machine including Control Clients, Windows Remote Desktop Services, and Power SCADA Anywhere.



Architecture #2: Power SCADA Anywhere with Power SCADA Server redundancy

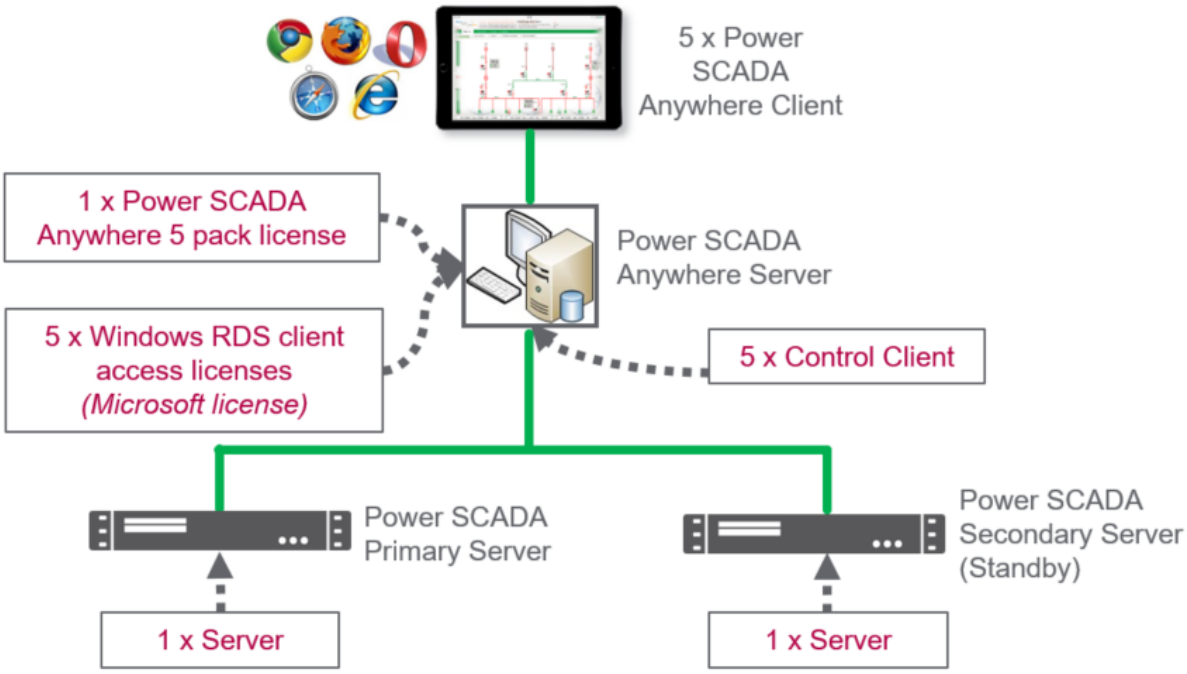
The following example architecture illustrates Power SCADA Anywhere with Power SCADA Server redundancy:



All software and licenses are installed on the Server machine including Control Clients, Windows Remote Desktop Services, and Power SCADA Anywhere.

Architecture #3: Isolated Power SCADA Anywhere with Server redundancy

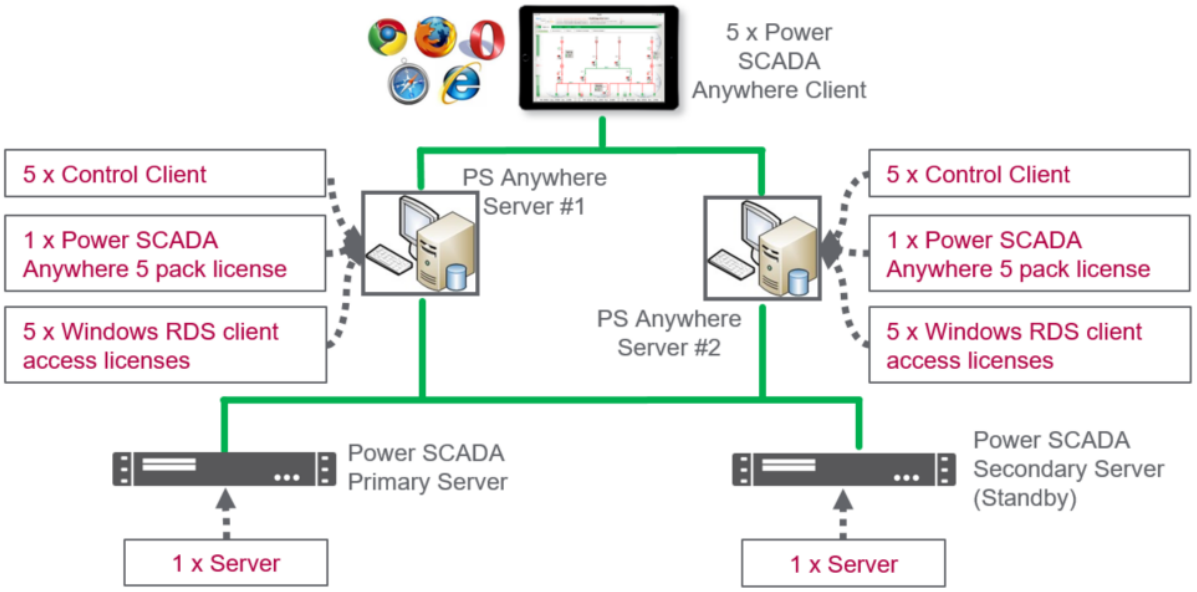
The following example architecture illustrates an isolated Power SCADA Anywhere with Server redundancy:



Power SCADA Anywhere components are isolated using a 3rd machine (Power SCADA Anywhere Server) with software and licenses installed for Control Clients, Windows Remote Desktop Services, and Power SCADA Anywhere.

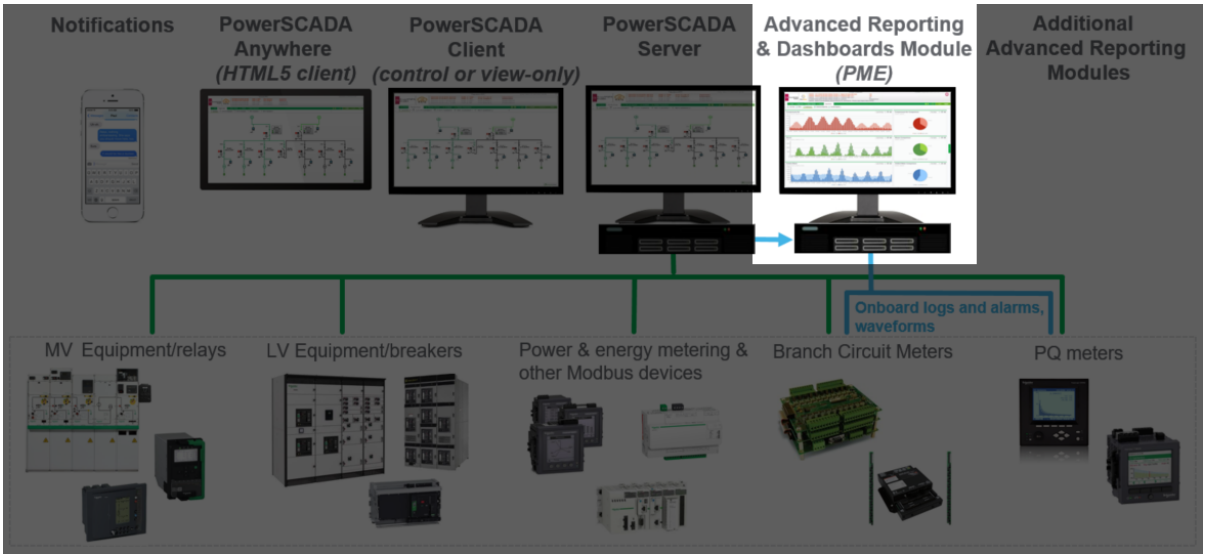
Architecture #4: Power SCADA Anywhere redundancy with Power SCADA Server redundancy

Architecture would be considered if customer wants a stand-by set of Power SCADA Anywhere Servers available in case components on Power SCADA Anywhere Server #1 failed and policies prevented client use on the Power SCADA Server machines.



NOTE: Power SCADA Anywhere clients would use different IP addresses to access Power SCADA Anywhere Server #1 vs. Power SCADA Anywhere Server #2.

Advanced Reporting and Dashboards Component



Purpose

Advanced Reporting and Dashboards Module is a variant of Power Monitoring Expert (PME) that is included on the Power SCADA DVD and can be optionally licensed with Power SCADA. In an architecture with Power SCADA, the Reports and Dashboards components of PME are integrated with the Power SCADA runtime to deliver feature rich “Energy Monitoring Application” experience for the system. Additionally WebReach diagrams are commonly integrated into the Power SCADA runtime as well.

Licensing options

Single license. For more information on licensing, see "Licensing" on page 79.

NOTE: Requires at least 1 Power SCADA Server license for purchase. No additional PME client or device licenses are required for this module as the Power SCADA Server and Client licenses cover the device licenses (i.e. PME DL’s) and client connectivity to the reports and dashboards.

Design considerations

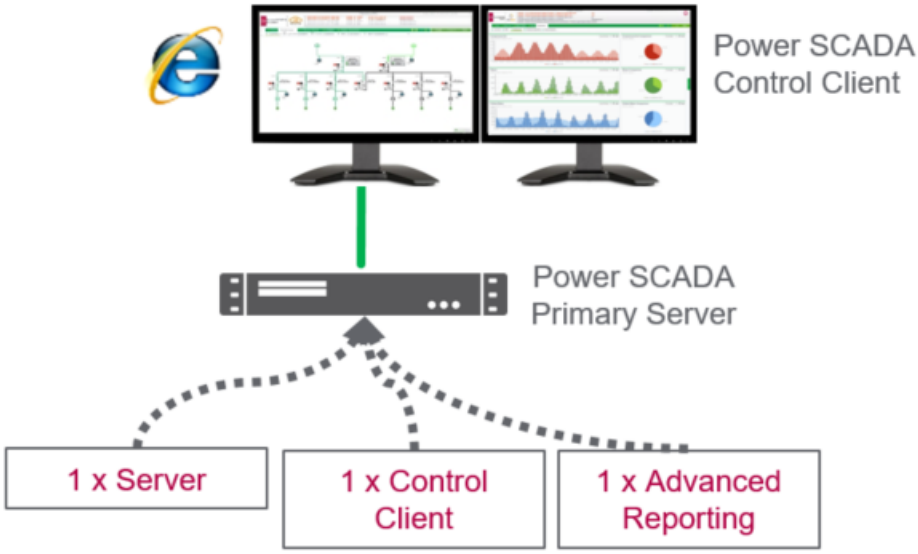
See "Advanced Reporting and Dashboards" on page 66.

Advanced Reporting and Dashboards architectures

Architecture #1: Simple system without redundancy

The following example architecture illustrates the Advanced Reporting & Dashboards Module in a system with a single Power SCADA Server.

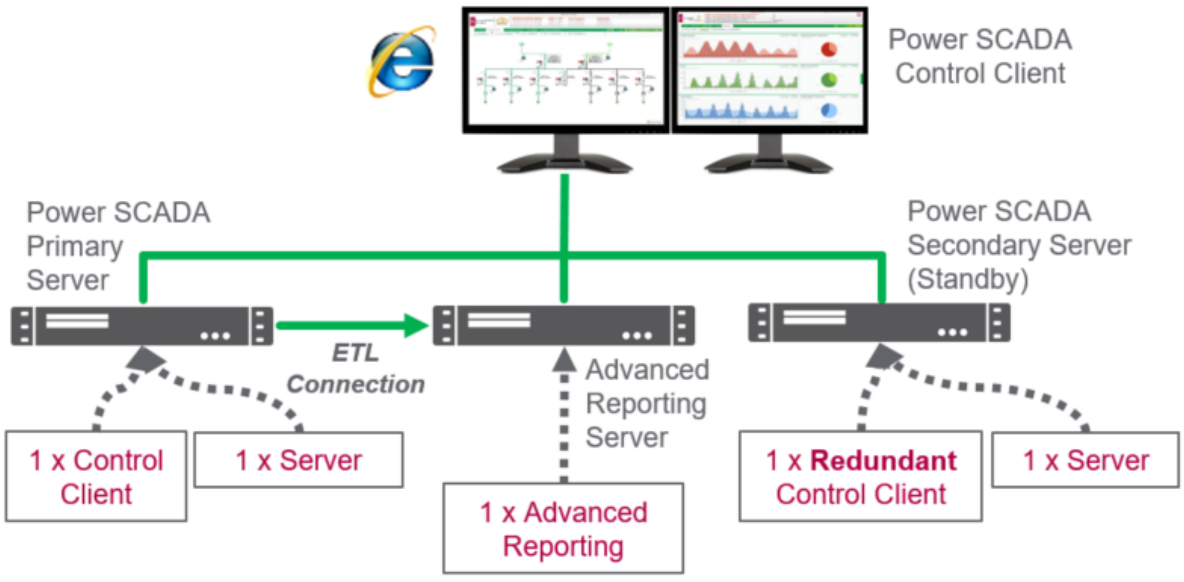
The Power SCADA Server and Advanced Reporting Module are installed on the same machine. Additionally the Control Client license to enable remote web client access is hosted on the Primary Server machine.



Architecture #2: Advanced Reporting with Server redundancy

The following example architecture illustrates the Advanced Reporting and Dashboards Module in a system with Power SCADA Server redundancy:

NOTE: This is the recommended Advanced Reporting architecture.



The Advanced Reporting Server contains both the Advanced Reporting software (PME) and the software key.

NOTE: The ETL used to send information from Power SCADA to PME is installed on the Advanced Reporting machine.

The ETL does not support the concept of communicating with a redundant Power SCADA setup. For this reason, if the Power SCADA Primary Server failed, then the ETL on the Advanced Reporting Server would need to be reconfigured manually to point to the Secondary Server.

Failure scenario: Advanced Reporting with Server redundancy

The following example architecture illustrates the Advanced Reporting ETL where it is not reconfigured to point to the Secondary Server when the Primary Server goes down:

This architecture fails because data cannot be transferred (ETL'd) to the Advanced Reporting Server while the Primary Power SCADA Server is down.

Example failure scenario

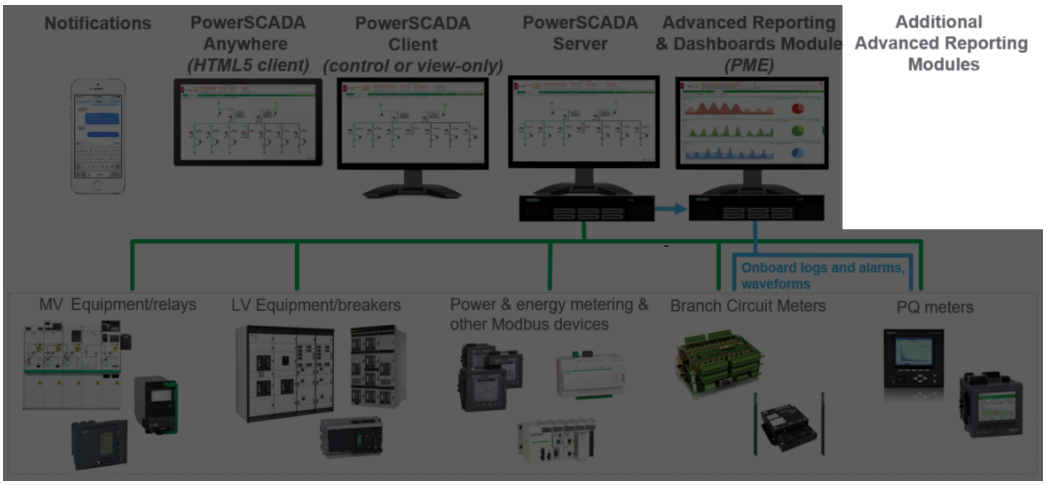
The Primary Power SCADA Server fails on June 1 and is restarted on June 3.

On June 1, the Secondary Power SCADA Server takes over the alarming & one-line diagram visualization. The Advanced Reporting Server is still running with reports, dashboards, and WebReach diagrams. The functionality of Power SCADA and Advanced Reporting would largely remain active from June 1 to 3.

However, when running reports while the Primary Power SCADA Server is down, report and dashboard data would not be present for the June 1 to 3 time period.

Once the Primary Power SCADA Server is recovered on June 3, the Secondary Power SCADA Server fills the Primary Power SCADA Server with the missed trend and historical data. Also, the ETL would start pulling data from the Power SCADA's Primary trend file system. Depending on system size, this June 1 to 3 data would eventually be available in the reports and dashboards.

Additional Advanced Reporting Modules Component



Purpose

Additional software modules compatible with the Advanced Reporting and Dashboards Module are included on the Power SCADA DVD and that can be optionally licensed with Power SCADA. These modules address a variety of electrical network, asset, and energy management needs.

Licensing options

Each module is licensed individually and requires at least 1 Advanced Reporting license. For more information on licensing, see "Licensing" on page 79.

Design considerations

See "Advanced Reporting and Dashboards" on page 66 for details.

Mapping EcoStruxure Power to Advanced Reporting Modules

The following table maps Advanced Reporting Modules to EcoStruxure Power Applications:

EcoStruxure Power Application	Advanced Reporting Module
Insulation Monitoring	Insulation Monitoring Module
Capacity Management	Capacity Management Module
Power Quality Monitoring	PQ Performance Module
Breaker Settings Monitoring	Breaker Performance Module
Energy Usage Analysis	Energy Analysis Reports Module Energy Analysis Dashboards Module
Energy Efficiency Compliance	Energy Analysis Reports Module Energy Analysis Dashboards Module
Cost Allocation	Energy Billing Module
Utility Bill Verification	Energy Billing Module
Backup Power Testing	Backup Power Module

NOTE: The *Power Monitoring Expert 9.0 – System Guide* contains detailed information on how to configure the Advanced Reporting modules.

Distributed architectures

Power SCADA Control Clients and View-only Clients can federate data from multiple Power SCADA Servers. Power SCADA Servers can contain different devices that can be distributed across several sites. Instead of trying to connect devices directly using a remote connection, an I/O sub-component is placed at each site.

In an architecture distributed across time zones, ensure that devices are configured for UTC time.

NOTE: Ensure a stable communication's path that is always-connected and has sufficient bandwidth.

Time synchronization

⚠ WARNING

INACCURATE DATA RESULTS

- Do not incorrectly configure the software or the devices.
- Do not base your maintenance or service actions solely on messages and information displayed by the software.
- Do not rely solely on software messages and reports to determine if the system is functioning correctly or meeting all applicable standards and requirements.
- Consider the implications of unanticipated transmission delays or failures of communications links.

Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

When using multiple machines in Power SCADA systems as outlined in this section, it is important that all machines hosting Power SCADA components be synchronized to the same NTP server (public or private). If time synchronization is not done across Power SCADA components, alarms and notifications may be delayed.

Do not confuse time synchronization as enabling Sequence of Events analysis and recording across devices in a Power Management system that may also be using time synchronization. For example: PTP, IRIG-B, etc...

Clustering

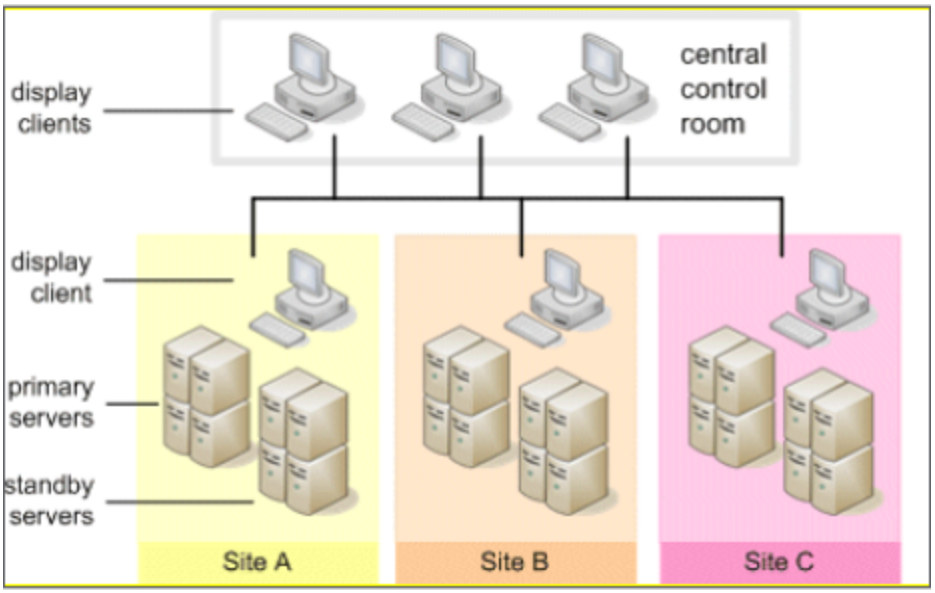
When the components of a Power SCADA project are defined, they are included in a cluster.

- Cluster rules:
  - Each cluster can contain only one pair of redundant Alarm, Report, Trend sub-components. They need to reside on different machines.
  - Each cluster can contain an unlimited number of I/O sub-components

- Within the same Power SCADA project, you can create several clusters. For example: One cluster per site if a customer has several substations or factories
  - Clients can display data from several clusters (simultaneous view of the various sites)

Example: Clustered Control System

The following system is organized into separate sites:

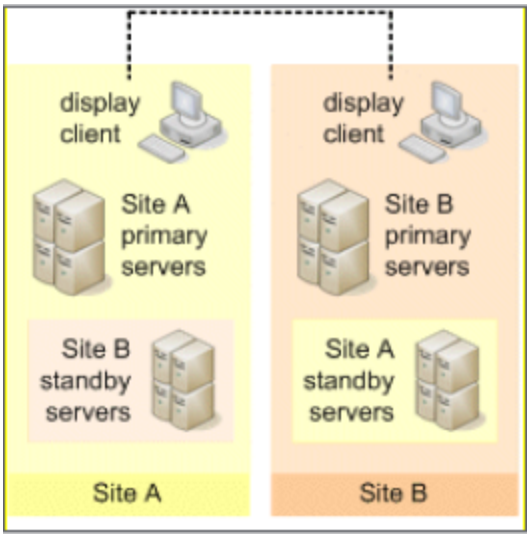


Each site is controlled by local operators, and is supported by local redundant servers. The Display clients from central control room can simultaneously manage all the sites.

Example: Redundant and Distributed Control System

To support several substations, multiple production lines, the following system has separate sites with their own server and clients.

In the following example, the primary and standby servers are distributed across different sites. If the system at one site becomes inoperative, monitoring is done by other site:



Supported devices and protocols

Power SCADA Operation supports concurrent protocol communication; one Power SCADA server can communicate using multiple protocols.

Power SCADA Operation 9.0 supports the following protocols:

- IEC 61850 Master Edition 2
- DNP3 Master
- ION
- Modbus Master
- IEC 60870-5-104 Master
- KNX
- SNMP v2
- BACnet/IP

Power SCADA Operation 9.0 supports the following Open Data Exchanges:

- OPC UA 1.01 (Client)
- OPC DA version 2, version 2.05a (Client and Server)
- OPC AE version 1 (Server)

For a complete list of Citect drivers compatible with Power SCADA see SCADA & MES Global Support DriverWeb.

Supported power devices

Native and other supported devices

For complete details on supported Power SCADA devices, refer to the Power SCADA Device Support Matrix spreadsheet. (XLSX file on Box.)

3rd party devices

3rd party devices can be supported via a variety of protocols using productivity tools not available in the core Citect platform

Protocol	Real time data	Onboard data logs	Onboard alarm time stamps and logs	Alarm time stamp quality	COMTRADE waveforms	Tools used during commissioning
Power Modbus	Yes	No	No	No	via FTP *	Profile Editor

IEC 61850 Ed. 2	Yes	Yes *	Yes *	No	Yes *	Profile Editor
IEC 60870-5-104	Yes	No	Yes *	No	via FTP *	Profile Editor
DNP3	Yes	No	Yes *	No	via FTP *	Profile Editor
SNMP v2	Yes	No	No	n/a	n/a	Power SCADA Studio
BACnet/IP	Yes	No	No	n/a	n/a	Power SCADA Studio

* If supported by the device.

Citect drivers

Citect drivers can be used with Power SCADA Operation. For a complete list of Citect drivers compatible with Power SCADA, refer to SCADA & MES Global Support DriverWeb.

Driver information also contains release notes and currently supported operating systems.

NOTE: Most drivers are licensed via Citect SCADA and are provided at no additional cost. However there are some exceptions where the driver requires an additional purchase cost to license it. Any drivers that require a purchase cost are only commercially available for Citect SCADA and are not commercially allowed for use with Power SCADA Operation.

Waveform File Share Access and Permissions

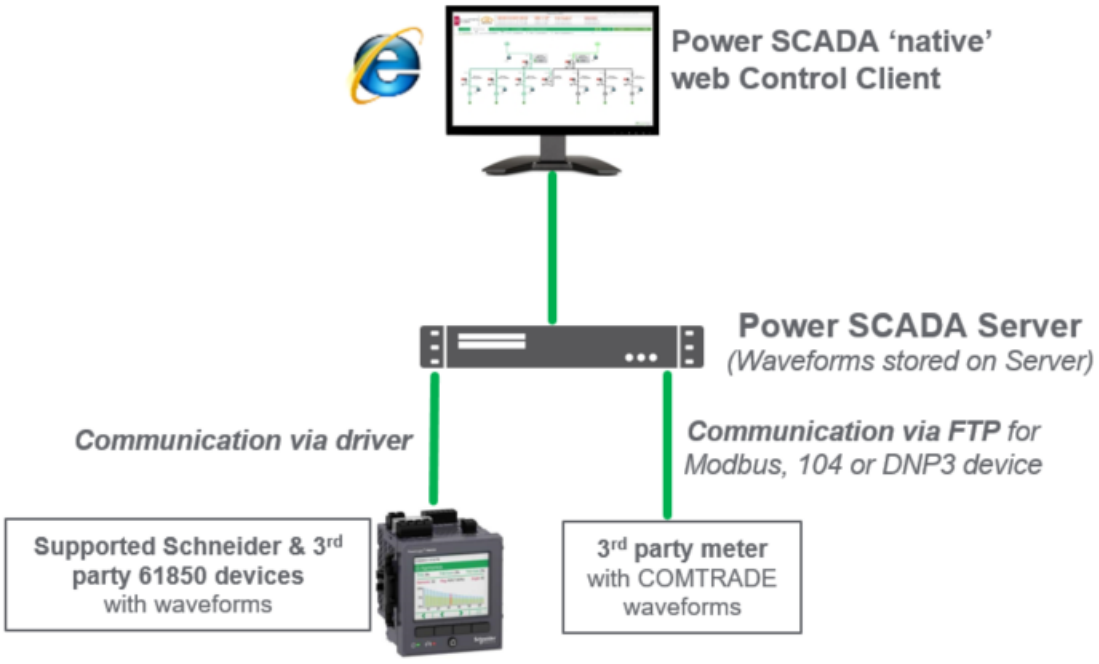
Waveforms are stored in a file share repository on the Power SCADA Server.

The following waveform file share permissions are required:

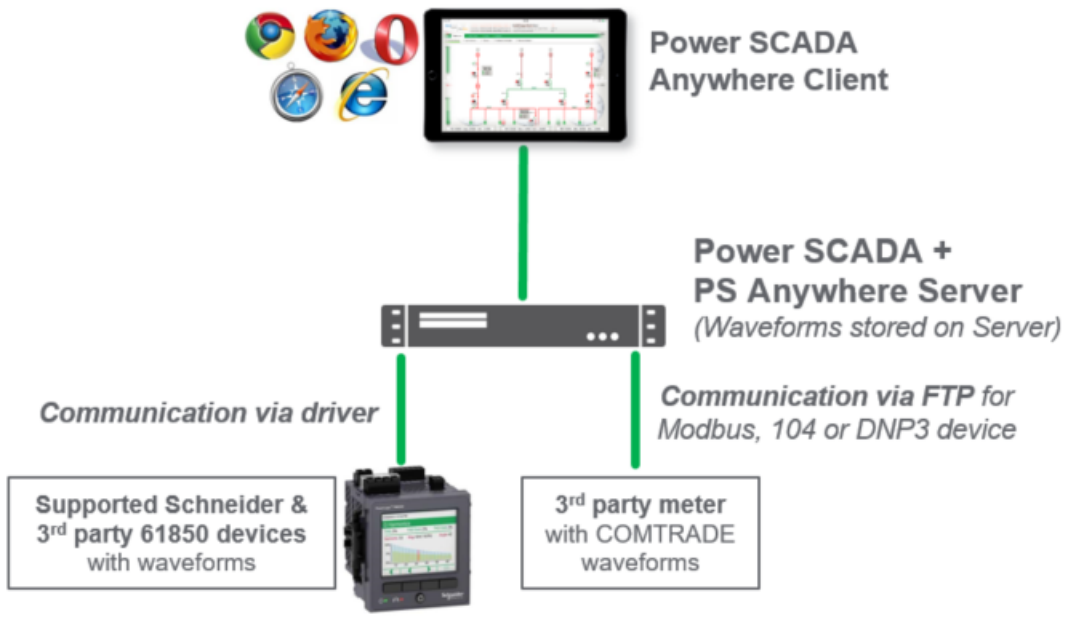
1. The account running Citect on Power SCADA Server requires Full permission (Read/Write/Modify)
2. The VjcaView or VjcaControl Windows user groups containing the Power SCADA user accounts require Read Only permission.
3. Windows user accounts must be linked to Power SCADA user roles that allow Remote Procedure Call (RPC).

   NOTE: This is required to get a list of waveforms from the Server.

Architecture #1: Native Web Client



Architecture #2: Power SCADA Anywhere Client architecture



Hardware and software requirements

This section provides information on the hardware and software requirements for a Power SCADA Operation with Advanced Reporting and Dashboards system.

Use the links in the table below to find the content you are looking for:

"Server CPU and RAM Requirements" on page 51	Server CPU and RAM recommendations for various system architectures
"Client CPU, RAM, and disc requirements" on page 52	Client CPU, RAM, and disk space requirements as well as requirements for monitoring running systems, graphics, and Power SCADA Anywhere
"Server disk storage" on page 53	Required disk space without Advanced Reporting and calculating disk storage size
"Supported Operating Systems" on page 54	Supported Operating Systems
"Supported browsers" on page 55	Supported browsers for web clients, Power SCADA Anywhere
"Supported SQL Server Versions" on page 56	Supported Microsoft SQL Server versions for the Advanced Reporting and Dashboards Module
"Virtualization" on page 57	Supported virtual environments

Server CPU and RAM Requirements

Power Management software needs to be installed on dedicated machines, so that other non-Power Management software applications do not consume machine resources.

When selecting server hardware, carefully review the PassMark score and CPU Clock Speed. The required processor is defined according to an average CPU mark given by PassMark® Software. To check CPU performance, for example a Core i3 CPU, type "PassMark Core i3" in the search engine of a web browser. This will return the CPU's calculated performance as compared to other similar well-known processors.

CPU and RAM recommendations for various system architectures

NOTES:

- The requirements listed in this topic are minimum requirements; we recommend that you consider doubling the RAM requirements listed.
- Power SCADA Anywhere server must have a CPU with SSE2 instruction set support.

Power SCADA Operation

The following table lists the number of CPU cores and RAM required for a Power SCADA Operation system.

NOTE: Use the tag or device number that is higher of the 2 numbers. For example if you have a system using 120,000 tags with 300 devices, use 6 CPU cores and 12 GB of RAM.

Use the larger figure below	CPU PassMark Score	# CPU Cores	RAM (GB)
1,500 tags or 50 devices	1,800	1	4
50,000 tags or 200 devices	4,500	6	8
100,000 tags or 400 devices	8,000	6	12
150,000 tags or 600 devices	8,000	8	12
200,000 tags or 800 devices	8,000	8	12
250,000 tags or 1,000 devices	10,000	10	12
300,000 tags or 1,200 devices	10,000	10	16
350,000 tags or 1,400 devices	10,000	12	16
400,000 tags or 1,600 devices	10,000	12	16
450,000 tags or 1,800 devices	10,000	14	16
500,000 tags or 2,000 devices	10,000	14	20

Power SCADA Operation and Power Monitoring Expert on the Same Machine

The following table lists the number of CPU cores and RAM required for a Power SCADA Operation and Power Monitoring Expert system on the same machine.

NOTE: Use the tag or device number that is higher of the 2 numbers. For example if you have a system using 120,000 tags with 300 devices, use 10 CPU cores and 28 GB of RAM.

Use the larger figure below	CPU PassMark Score	# CPU Cores	RAM (GB)
50,000 tags or 200 devices	8,000	10	16
100,000 tags or 400 devices	8,000	10	28
150,000 tags or 600 devices	8,000	12	36

For systems greater than 150,000 tags or 600 devices, we recommend a distributed architecture with separate physical machines for Power SCADA and Power Monitoring Expert.

Power SCADA Operation and Power Monitoring Expert on separate machines

Refer to the *Power Monitoring Expert 9.0 – System Guide* for specific CPU and RAM requirements when installing Power SCADA and Power Monitoring Expert on separate machines.

Client CPU, RAM, and disc requirements

Power SCADA Control or View Clients have the following minimum requirements:

- CPU: 2 Cores
- RAM: 4 GB
- Disk storage: 10 GB
- Screen resolution: 1920 X 1080

Monitoring CPU for running systems

Optimal performance is achieved when all computers in your Power SCADA Operation network use approximately 40% or lower CPU in normal state. If you have any concerns about system responsiveness or its ability to handle abnormal situations, consider adding resources to lower overall CPU utilization.

Power SCADA Graphics Adapter

128 MB of dedicated VRAM (for systems of any size)

Power SCADA Anywhere

The Power SCADA Anywhere host requirements for disk, CPU, and RAM are negligible.

The Power SCADA Anywhere host must have CPU with SSE2 instruction set support.

Server disk storage

Required disk space without Advanced Reporting

When planning a Power SCADA system without Advanced Reporting (Power Monitoring Expert), you can fine tune your disk storage requirements based on how Power SCADA Operation stores data.

Power SCADA Operation has 2 major consumers of disk storage space:

1. Alarm information which is stored in a propriety database that may grow over time to a size of 1-2 GB.
2. Historical data stored in trend files; flat files on the disk. The size and number of these trend files depend on number of tags in system, logging interval, and number of years to store data.

Trend files are pre-allocated (reserved) on the hard disk the first time that Power SCADA is started. Hard disk space does not "grow" over time by acquiring trend data. In other words, if the hard drive is not big enough for the number of years of trending that you plan for, the system will tell you.

Calculating disk storage

To calculate disk storage size for you system, use the Power SCADA Disk Sizing Calculator.

NOTE: These values include a 2 GB alarm database size and assume that you configure trends to be stored in separate files each week.

Scenario #1: Trend data logged every 15 minutes / Stored for 2 years

	1,500 Tag System	5,000 Tag System	15,000 Tag System	50,000 Tag System	200,000 Tag System
TOTAL disk space required	2.95 GB	5.16 GB	11.49 GB	33.62 GB	128.50 GB

Scenario #2: Trend data logged every 5 minutes / Stored for 2 years

	1,500 Tag System	5,000 Tag System	15,000 Tag System	50,000 Tag System	200,000 Tag System
TOTAL disk space required	4.63 GB	10.75 GB	28.26 GB	89.53 GB	352.14 GB

Supported Operating Systems

The following table lists the compatible operating systems for all versions of Power SCADA Operation and includes all Power SCADA Operation components (Servers, Clients, Advanced Reporting and Dashboards, etc):

NOTE: 64-bit operating systems are recommended for best performance.

Operating System	Power SCADA Operation Version				
	9.0	**8.2**	**8.1**	**8.0**	**7.40**
Windows Server 2016 Standard	✓	✓	–	–	–
Windows 10 Professional/Enterprise	✓ [4]	✓	✓ [1]	✓ [2]	–
Windows Server 2012 R2 Standard/Enterprise	✓	✓	✓	✓ [3]	–
Windows 8.1	–	✓	✓	✓ [3]	–
Windows Server 2012 Standard	✓	✓	✓	✓	✓
Windows Server 2008 R2 SP1	–	✓	✓	✓	✓
Windows 7 Professional/Enterprise	✓ [4]	✓	✓	✓	✓

Operating System	Power SCADA Operation Version				
	9.0	8.2	8.1	8.0	7.40
Windows Server 2008	–	–	–	–	✓

1: Available with PowerSCADA Expert 8.1 update 6 or later

2: Available with PowerSCADA Expert 8.0 Service Release 1 update 3 or later

3: Available with PowerSCADA Expert 8.0 Service Release 1

4: Support for 64-bit Windows only

NOTE: Power SCADA Anywhere is supported on Windows Server 2012 R2, and Windows Server 2008 R2 SP1 only.

Supported browsers

Web client access

Power SCADA has 2 approaches to web client access:

1. Native web clients – If you do not use Power SCADA Anywhere, you will be using the native web clients for the Power SCADA runtime along with the various applications that have different native browser support capabilities.
2. Power SCADA Anywhere client– Enables an HTML5 web client experience by using 3rd party Windows Desktop Services (Terminal Services) to stream a Windows desktop application of the Power SCADA runtime to remote web browsers.

Native web client browser support

Power SCADA Operation native web client supports the following browsers:

	IE 11	IE 10	Edge	Chrome	Firefox	Safari
Power SCADA graphics pages including one-line diagram/engine	Yes	Yes	No	No	No	No
PME reports, PME dashboards, and PME WebReach Diagrams	Yes	Yes	Yes	Yes	Yes	Yes
Power SCADA Basic Reports	Yes	Yes	No	No	No	No
LiveView (Power SCADA Real Time Tables implementation)	Yes	Yes	No	No	No	No

NOTES:

- Limitation: Only one instance of the Power SCADA native web client can be opened at a time in a web browser. In other words, you cannot open multiple tabs with the Power SCADA native web client running.

- PME components—such as PME Real Time Tables and PME Alarms—are only supported in IE as of v8.2. In a combined Power SCADA and PME architecture, these components are not recommended to be used.

Power SCADA Anywhere browser support

Power SCADA Anywhere supports the following browsers:

	IE 11	IE 10	Edge	Chrome	Firefox	Safari
Power SCADA Anywhere	Yes	Yes	Yes	Yes	Yes	Yes

When using Power SCADA Anywhere, we are assuming that the various components listed below are integrated into a runtime experience that is being used in a Control or View-only Client:

- Power SCADA graphics pages including one-line diagram/engine
- PME reports, PME dashboards, and PME WebReach diagrams
- Power SCADA Basic Reports
- LiveView (Power SCADA Real Time Tables implementation)
- Notification Module (configuration tools)

When these components are integrated into a runtime that is being streamed using Power SCADA Anywhere, all HTML5 client browsers listed above are supported.

Unlike the Power SCADA native web client, multiple instances of Power SCADA Anywhere can be opened at the same time in a web browser.

Supported SQL Server Versions

Power SCADA Operation with Advanced Reporting and Dashboards requires a Microsoft SQL Server database. Power SCADA Operation with Advanced Reporting and Dashboards supports the following SQL Server versions:

- SQL Server 2017 Express/Standard/Enterprise/Business Intelligence
- SQL Server 2016 Express/Standard/Enterprise/Business Intelligence
- SQL Server 2014 Express/Standard/Enterprise/Business Intelligence
- SQL Server 2012 Express/Standard/Enterprise/Business Intelligence, SP2

NOTE: Power SCADA Operation 9.0 **without** Advanced Reporting and Dashboards does NOT require a SQL Server database.

Power SCADA Operation with Advanced Reporting and Dashboards installation media (DVD & ISO) includes SQL Server 2016 Express that can be used with Advanced Reporting.

Virtualization

The following table lists the virtualization support for Power SCADA Operation with Advanced Reporting and Dashboards:

	Microsoft Hyper-V	VMWare vSphere
Power SCADA Server (including web server host)	Yes	Yes
Power SCADA Control Client & View-only Client (this refers to Windows Desktop clients)	Yes	Yes
Mobile Notifications	Yes	Yes
Power SCADA Anywhere (this refers to machine hosting Power SCADA Anywhere)	Yes	Yes
Advanced Reporting and Dashboards (ie: PME)	Yes	Yes

When using virtual environments, we recommend that you license all components with software keys, not USB dongles.

NOTE: Power Monitoring Expert is validated with additional virtualization systems, see the *Power Monitoring Expert 9.0 – System Guide* for additional details.

Virtualization configuration notes:

- Set all resource allocation (CPU, memory, and disk) to fixed; dynamic is not supported.
  - Do not share resources between virtual machines via over-allocation.
- You must have a dedicated hard drive used by Power SCADA only.
- You must have a fixed-size disk virtual machine.
- Set host (eg: ESX host) power management to “High Performance”.

Additional virtual machine configuration guidelines vary by hypervisor.

Device response time

We recommend Ethernet communication because it provides the best system performance with devices.

NOTE: Avoid connecting meters serially via gateway to Power SCADA Operation with Advanced Reporting and Dashboards when a direct Ethernet connection is available.

If you have to use Serial communication, you should have a minimum baud rate of 19.2K.

NOTE: Connecting serially significantly reduces the available bandwidth and can prevent multi-mastering.

Cybersecurity

This section provides information on how to help secure your system from a malicious cyber attack.

Use the links in the table below to find the content you are looking for:

"Securing the network and servers" on page 59	Using ConneXium Tofino firewalls to restrict and control traffic between IT, OT, and Internet network zones.
"Securing servers" on page 61	Patching recommendations and whitelisting design considerations
"Securing user access" on page 62	Securing user access through Windows Active Directory, role-based access control, and two-factor authentication
"Awareness and education" on page 64	Additional resources to increase your cybersecurity awareness

Securing the network and servers

⚠ WARNING

POTENTIAL COMPROMISE OF SYSTEM AVAILABILITY, INTEGRITY, AND CONFIDENTIALITY

- Use cybersecurity best practices to help prevent unauthorized access to the software.

Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

Improve security of networked devices by using multiple layers of cyber defense (such as firewalls, network segmentation, and network intrusion detection and protection). Disable unused ports/services and default accounts to help minimize pathways for malicious attackers.

Power SCADA Operation can use ConneXium Tofino firewalls to restrict and control traffic between IT, OT, and Internet network zones.

ConneXium Tofino is an industrial firewall designed for use in industrial control system networks. The firewall offers deep packet inspection of Modbus TCP, allowing restriction at the Modbus command level as defined by the network designer. It is highly configurable using software called ConneXium Tofino Configurator which is (included with Tofino purchase. The software lets you define entire networks, referred to as projects, which can have multiple Tofino firewalls protecting a myriad of devices at different points in the network.

Power SCADA now supports electronic software keys to allow IT departments to lock-down USB ports on server computers.

The configuration setup steps are:

1. Install the Tofino Configurator and create a project.
2. Add all the Tofino devices to your network.

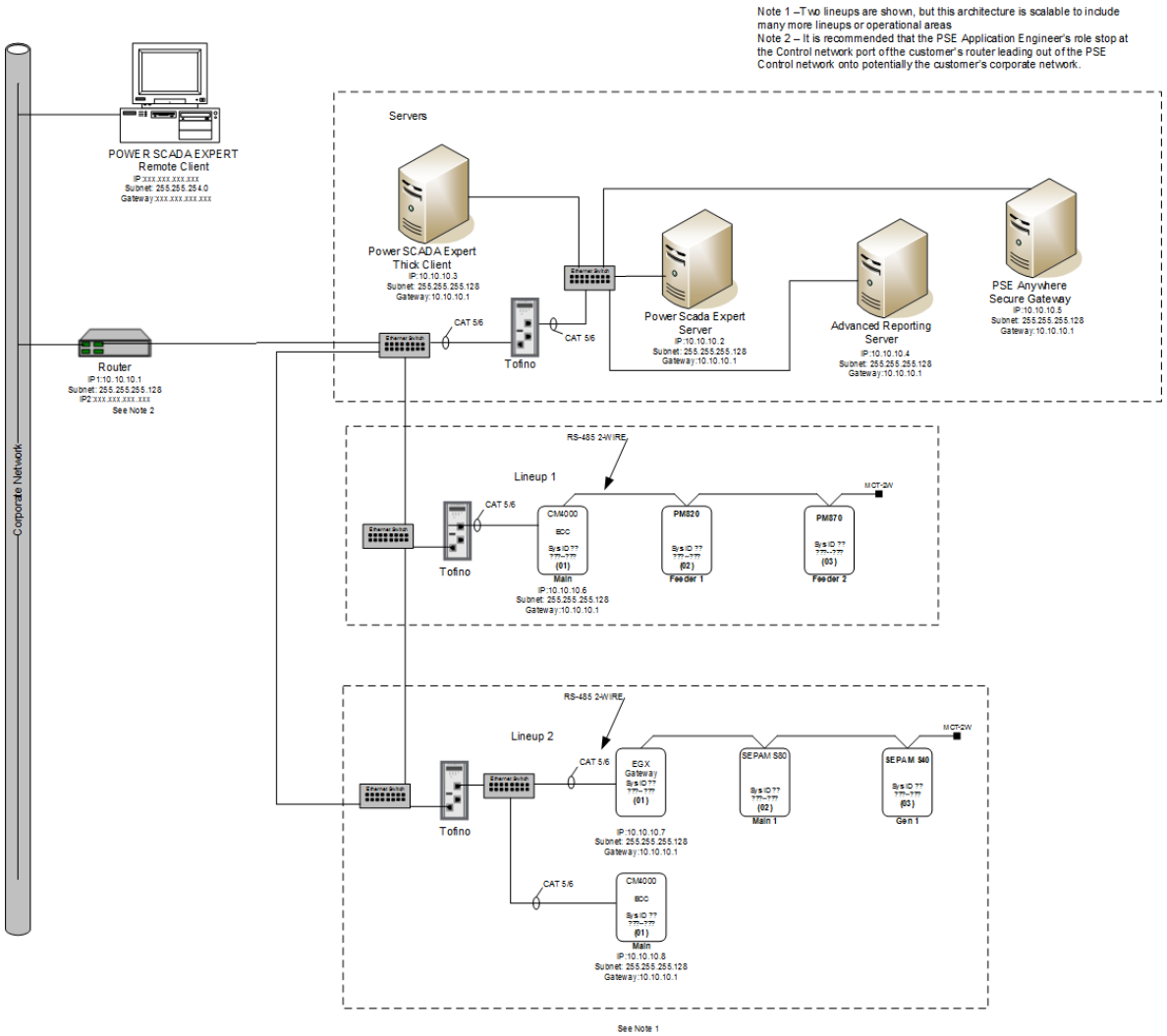
3. Add all the other devices on the network.
4. Configure the rules for the network that define the traffic that can pass through which firewall.

We recommend that you begin with the firewalls in test mode so you can see what would be blocked and then adjust accordingly. The firewall configurations should be then loaded onto a USB flash drive that is used to upload the configuration to each firewall.

Detailed information about the setup and configuration of the Tofino architecture is provided in the ConneXium TCSEFEA User Manual V1.

NOTE: You should not use this firewall as an “edge” device, bridging the Control Network with public networks.

The following is an example architecture that can serve as reference for how one of the networks might be constructed. It is a small network that can be scaled out to fit a much larger system.



Securing servers

Patching

⚠ WARNING

POTENTIAL COMPROMISE OF SYSTEM AVAILABILITY, INTEGRITY, AND CONFIDENTIALITY

Apply the latest updates and hotfixes to your Operating System and software.

Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

Be sure that all Windows updates and hotfixes—especially Windows security updates—are regularly applied to machines running Power SCADA Operation and Power Monitoring Expert.

If compatibility issues are encountered with these Windows updates and Power SCADA Operation and Power Monitoring Expert software, these compatibility issues will be considered high priority for our R&D factories to evaluate and resolve in order to deliver patches to enable the continued use of Windows security updates.

Whitelisting

Zero-day cybersecurity attacks take place before a software vendor is aware of a cybersecurity exploit. Meaning that neither software nor anti-virus programs have been created or updated to protect against the zero-day threat or attack.

Application whitelisting is recommended to protect against Zero Day attacks. Whitelisting specifies an index of approved software applications and processes (in our case Power SCADA) that are permitted to be present and active on a computer system.

Power SCADA has been validated with the McAfee Application Control whitelisting application. Power SCADA and McAfee whitelisting can make your system more resilient to zero-day threats.

Whitelisting Design Considerations

- Power SCADA Servers, Control Clients, View-only Clients, and Advanced Reporting have been validated using McAfee Application Control.
- McAfee Whitelisting can be ordered with Power SCADA using:
  - PSA200100 - McAfee Whitelisting (Embedded Control)

  NOTE: The license is good for one year (12 months). The same part number can be used to purchase a renewal.
- McAfee Whitelisting product documentation can be found on the McAfee Web site.

Securing user access

⚠ WARNING

POTENTIAL COMPROMISE OF SYSTEM AVAILABILITY, INTEGRITY, AND CONFIDENTIALITY

- Use cybersecurity best practices when configuring user access.

Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

Cybersecurity policies that govern user accounts and access – such as least privilege and separation of duties – vary from site to site. Work with the facility IT System Administrator to ensure that user access adheres to the site-specific cyber security policies.

Power SCADA Operation secures user access through:

- Windows Active Directory integration
- Role-based access control
- Two-factor authentication

Power SCADA Operation also includes Power SCADA Runtime user partitioning (8 levels of user privilege) and user event monitoring (log in. log out , shutdown, control, on so on...)

Windows Active Directory

Power SCADA components support user management by using Windows Active Directory groups and by local users.

NOTE: For cybersecurity purposes, it is recommended that you use Windows Active Directory for user access and management.

Power SCADA Anywhere users can only be managed using Windows Active Directory.

NOTE: The machine hosting Power SCADA Anywhere must be installed on a machine that is part of a Windows domain

Role Based Access Control

Power SCADA's implementation of Role Based Access Control (RBAC) leverages the same Power SCADA components that are used as part of a Windows Active Directory support. RBAC is compatible with Schneider Electric devices that support MiCOM Px40, MiCOM Px30 protection relays, MiCOM C264 RTU's, etc.

You connect the Power SCADA Server to the same RBAC appliance that is used with MiCOM devices. Additionally you use the same Power SCADA tools as you would to join to a Windows domain group.

NOTE: The RBAC appliance used with MiCOM devices emulates a Windows Domain Controller.

Two-factor authentication

Two-factor authentication requires users to provide two pieces of proof of identity, such as a password and one other component. This feature allows you to add an additional layer of protection when user credentials are required; such as at log in, shutdown and control functions.

NOTE: For cybersecurity purposes, it is strongly recommended that you configure two-factor authentication in your projects; especially in deployments with control functionality.

Power SCADA Operation uses a one-time password (OTP) to accomplish two-factor authentication. OTP is implemented in Power SCADA Operation using a USB key device called a YubiKey. The YubiKey is designed to fit on a key ring or attached to a badge. It must be plugged into the client machine when the user authenticates.

Power SCADA supports two-factor authentication on isolated networks; the Internet is not required. Additionally, it will work with physical machines, virtual machines, Power SCADA Anywhere, and Web Clients.

How Does It Work?

When a YubiKey is assigned to a Power SCADA Operation user, the YubiKey and the assigned user account share a secret code. The YubiKey uses this secret code to generate encrypted strings of text (the OTPs) when the user presses the button on the YubiKey.

Using the secret code, Power SCADA Operation decrypts the OTP to determine if the OTP is valid (ensuring that it has not been replayed, it is assigned to the current user, etc.). After successful authentication, Power SCADA Operation marks the OTP as expired and will no longer accept it as valid.

YubiKey selection

YubiKeys are not shipped with Power SCADA Operation. Instead, you must buy them from a third-party vendor, such as Amazon. The following table lists the YubiKey models that are compatible with Power SCADA Operation:

Model	Description
YubiKey 4	All form factors including USB-A and USB-C ports supported. YubiKey 4 works with control and web clients with and without FIPs enabled on the servers. i.e. on either local server or server accessing the web client.
YubiKey NEO	YuibKey Neo works with control and web clients with and without FIPs enabled on the servers. i.e. on either local server or server accessing the web client.

Ordering YubiKeys

Keep in mind these points when you are ordering or using a YubiKey:

- You must set "Allow RPC" to TRUE for all roles that are using YubiKey.
- YubiKey is compatible with all thick clients and web clients.
- YubiKey requires access to a USB port at each client.
- Each Power SCADA Operation I/O Server must have Application Services (Core Service Host) running.
- Multiple I/O servers may reside on a physical machine. In this case, only one instance of Application Services resides on the machine.
- YubiKey must be configured and synchronized across all I/O servers (this includes redundant pairs and distributed systems).
- YubiKey is enabled on each client independently. If YubiKey is enabled on a client, all users on that client must authenticate via YubiKey.
- It is possible to configure YubiKey on one machine, export the configuration for all users, and import the configuration to all remaining machines.
- It is not necessary to re-program YubiKey when changing passwords. The YubiKey changes the OTP every time so it is not susceptible to replay attacks.
- YubiKey is authenticated against all servers that contain at least one I/O Server. All servers must successfully authenticate the OTP for success. If a single server does not authenticate (due to misconfiguration, etc.), the user will not be able to log in.
- If a machine (with an I/O Server) is not available, it is not included in the authentication scheme. This means that if a primary server is down, the secondary can still successfully authenticate the OTP.
- If no servers (with I/O servers) are available, the user will not be able to log in on clients that have YubiKey enabled.

To set up one-time passwords, see "Two-Factor Authentication (One-Time Password)" on page 361

Awareness and education

Power SCADA Operation includes cybersecurity features at the network, server, client, and access levels that can be configured to help prevent system compromise. However, knowledge is first step to prevent cyber intrusions. Review the following resources to increase your cybersecurity awareness:

- Schneider Electric Cybersecurity Support Portal
- Securing Power Monitoring and Control Systems (Schneider Electric White Paper)
- Social engineering (security)

The Schneider Electric Cybersecurity Portal contains cybersecurity news, security notifications, and additional resources.

Advanced Reporting and Dashboards

Use the links in the table below to find the content you are looking for:

About Advanced Reporting and Dashboards

The Advanced Reporting and Dashboards Module offers a broad array of reports, dashboard visualizations, and customizable report subscriptions.

Power Monitoring Expert Reporting

- Best in class reporting with more than 30 default reports, including Power Quality reports
- Reports that can be triggered manually, scheduled, or event-triggered
- Save reports as PDF, HTML, or CSV

- Power Monitoring Expert Dashboards
  - End user configurable dashboard view of historical data
  - Ability to embed external web content in a dashboard
  - Kiosk views to let teams see KPI Energy values that are relevant to them
- Power Monitoring Expert WebReach diagrams
  - Diagram-based view of real time device data

When Power SCADA Operation and Power Monitoring Expert are integrated, historical applications from PME (Reports and Dashboards) are integrated into the Power SCADA Operation runtime. WebReach diagrams are also frequently integrated with Power SCADA Operation resulting in a seamless end user experience.

The following table lists how components are used in combined solution:

	Real time information (graphics, tables, trends)	Alarms	Notifications Settings	Waveforms	Historical reports and dashboards	OPC DA and SMNP
Power SCADA Operation	Enabled	Enabled	Enabled	Enabled	Disabled	Enabled
Power Monitoring Expert	Disabled	Disabled	Disabled	Enabled	Enabled	Disabled

NOTE: Power SCADA Operation with Power Monitoring Expert must be the same product version to be integrated. For example, Power SCADA Operation 9.0 and Power Monitoring Expert 9.0 are supported. Power SCADA Operation 9.0 and Power Monitoring Expert 8.2 would not be supported.

Advanced reporting customizations

Power Monitoring Expert reports help customers better understand their electrical network. Sometimes these reports require further customization. Report customization can be divided into the following tiers:

- **Basic** – Colors, logo, toggle on/off report components, target lines, etc...
- **Advanced** – Modify the format/layout of existing report templates, create new basic ones. Excel and Power BI integration.
- **Expert** – Custom report creation. Create completely new reports with existing and new view providers (data sets).

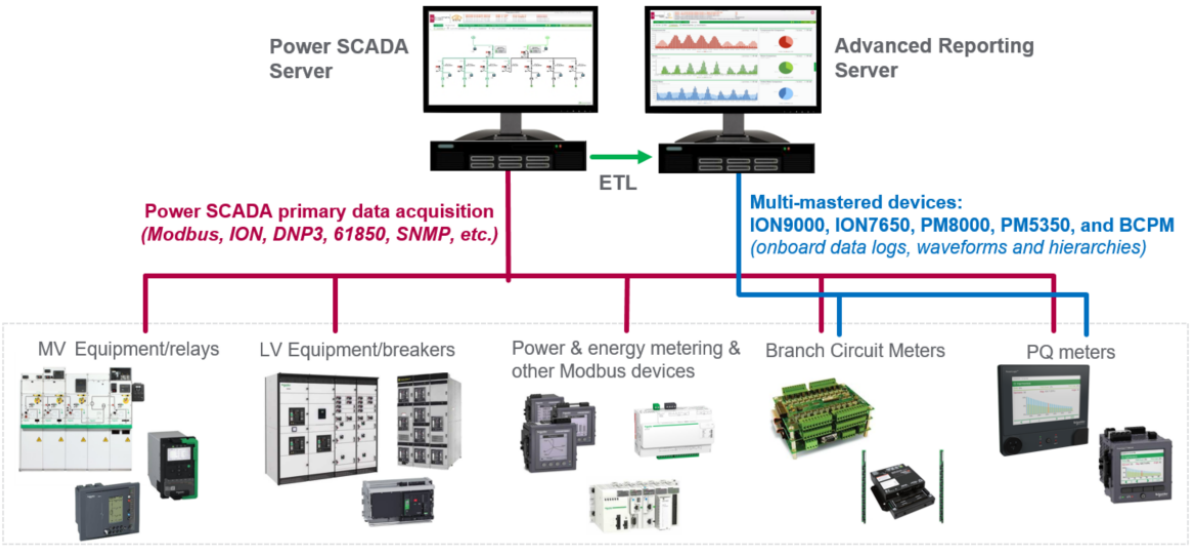
Detailed information on how to customize reports can be found on the Exchange Community (requires log in.)

Device communication

The following device communication architectures can be used when integrating Power SCADA Operation and Power Monitoring Expert:

- **Multi-mastering all devices** – Setting up device communications in both Power SCADA Operation and Power Monitoring Expert
- **Single-mastering all devices** – Setting up device communications with Power SCADA Operation and then transferring device data to Power Monitoring Expert using an Extract, Transform, and Load (ETL) tool

The following image illustrates the recommended device communication architecture for Power SCADA Operation with Advanced Reporting and Dashboards:

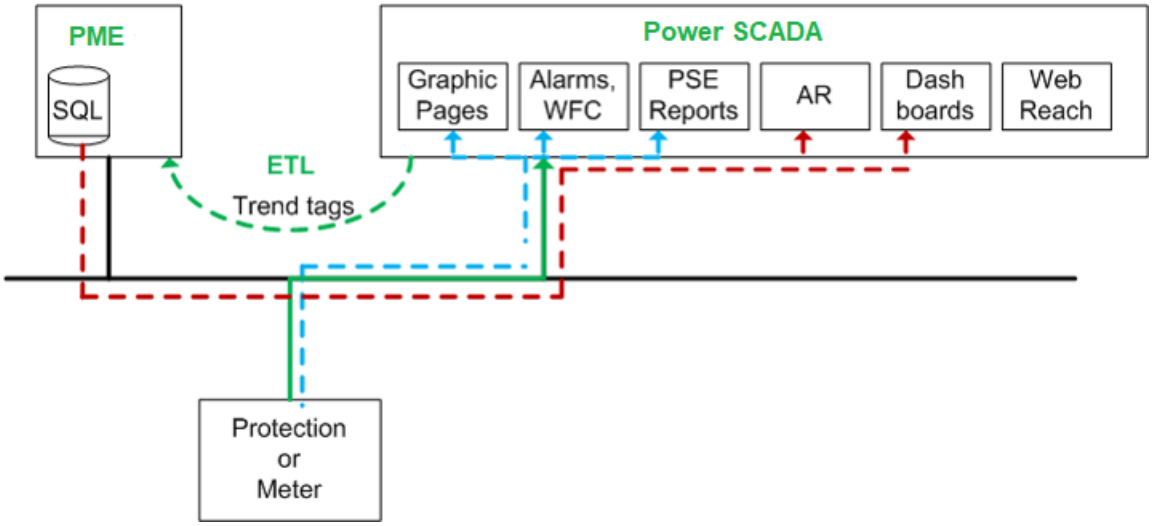


Single-mastering devices

When using single-mastering with Power SCADA:

1. Power SCADA acquires historical (trend) data from all devices.
2. The Extract-Transform-Load (ETL) tool transfers historical data from Power SCADA Operation to Power Monitoring Expert for use in Power Monitoring Expert reports and dashboards.

The following image illustrates single-mastering device communication flow:



Single-mastering is the preferred device communication architecture for the following reasons:

- Improved performance – Power SCADA trend acquisition can be assigned a lower priority than real-time and alarm data thereby reducing CPU and RAM loads
- Increased functionality – Allows PME reporting to be run on devices with protocols not natively supported by PME (for example: IEC-61850, DNP3, SNMP, BACnet, etc.)

- Simplified deployments and maintenance – Devices have to be setup and maintained in Power SCADA only. There is no risk that device names between Power SCADA and PME are inconsistent.
- Recovery from failure scenarios – If the Power Monitoring Expert Server or Power SCADA Primary Server become unavailable, the ETL can still transfer the data.

  In test scenarios where PME communication was unavailable for 1.5 days and then became available again, the ETL when triggered manually took the following times to catch up and re-establish steady state for the following system sizes:
  - 35,000 tags logged every 15 minutes: On average, the system took 30 minutes to recover the lost 1.5 day's worth of data
  - 105,000 tags logged every 15 minutes: On average, the system took 95 minutes to recover the lost 1.5 day's worth of data

NOTE: When using single mastering, it is recommended that you increase the RAM beyond the minimal RAM requirements for the system size.

However, there are exceptions where single-mastering cannot be used. See "Multi-mastering devices" on page 69 for details.

Multi-mastering devices

The devices and Advanced Reporting and Dashboards Modules that require multi-mastering are listed here.

Devices

The following device types cannot be single-mastered by Power SCADA; they must communicate with Power SCADA and Power Monitoring Expert:

- ION9000, ION7650, and PM8000 (Power Quality meters)

  Power Monitoring Expert requires a direct connection to these devices in order to provide data depth in Power Quality Reports.
- BCPMs and PM5350 (multi-channel meters)

  Power Monitoring Expert provides Branch Circuit Reports that leverage hierarchy information.

  NOTE: Trending BCPMs and PM5350 can be reconfigured in the field. For example, instead of using channels 1 to 10, BCPMs can be reconfigured to use channels 1 to 20. This reconfiguration requires restarting the Power SCADA Server.

  NOTE: BCPM historical trends should only be gathered by Power Monitoring Expert, and should be disabled in Power SCADA. If you try to use the ETL to transfer branch circuit power monitor (BCPM) trend data to the Advanced Reports Server, the amount of branch circuit device data can overwhelm the ETL process.
- Any meter that you want to view using WebReach diagrams.

  WebReach diagrams require data acquisition from Power Monitoring Expert to provide real time information.

Advanced Reporting and Dashboards Modules

Certain Advanced Reporting and Dashboards Module require devices to be setup in both Power SCADA and Power Monitoring Expert.

The following table list the modules and devices that require multi-mastering and the reason why:

Module	Devices required on both servers	Reason
Breaker Performance	All Micrologic trip units	Real-time vista diagrams leveraged by module
Billing *	Any device required for billing	Requires data from Hierarchy
Power Efficiency	Any device used in PUE calculation	VIP is used to calculate the total kW and interval energy for the PUE calculation
Power Quality Advisor	All devices	Due to the way PQ Advisor's algorithm works

NOTE: This is addition to ION9000, ION7650, PM8000, PM5350. and BCPMs.

* The Billing Module relies on an energy billing ETL to export Power Monitoring Expert data to be used in 3rd party billing software packages. Since the energy billing ETL requires data from the customer hierarchy, any devices required for the ETL should be added in Power Monitoring Expert and Power SCADA Operation.

OFS system time stamping

Power SCADA Operation provides the System Time Stamping method for the electrical distribution monitoring and control system.

System Time Stamping helps the user analyze the source of abnormal behaviors in an automation system.

The benefits of the system time stamping mode are:

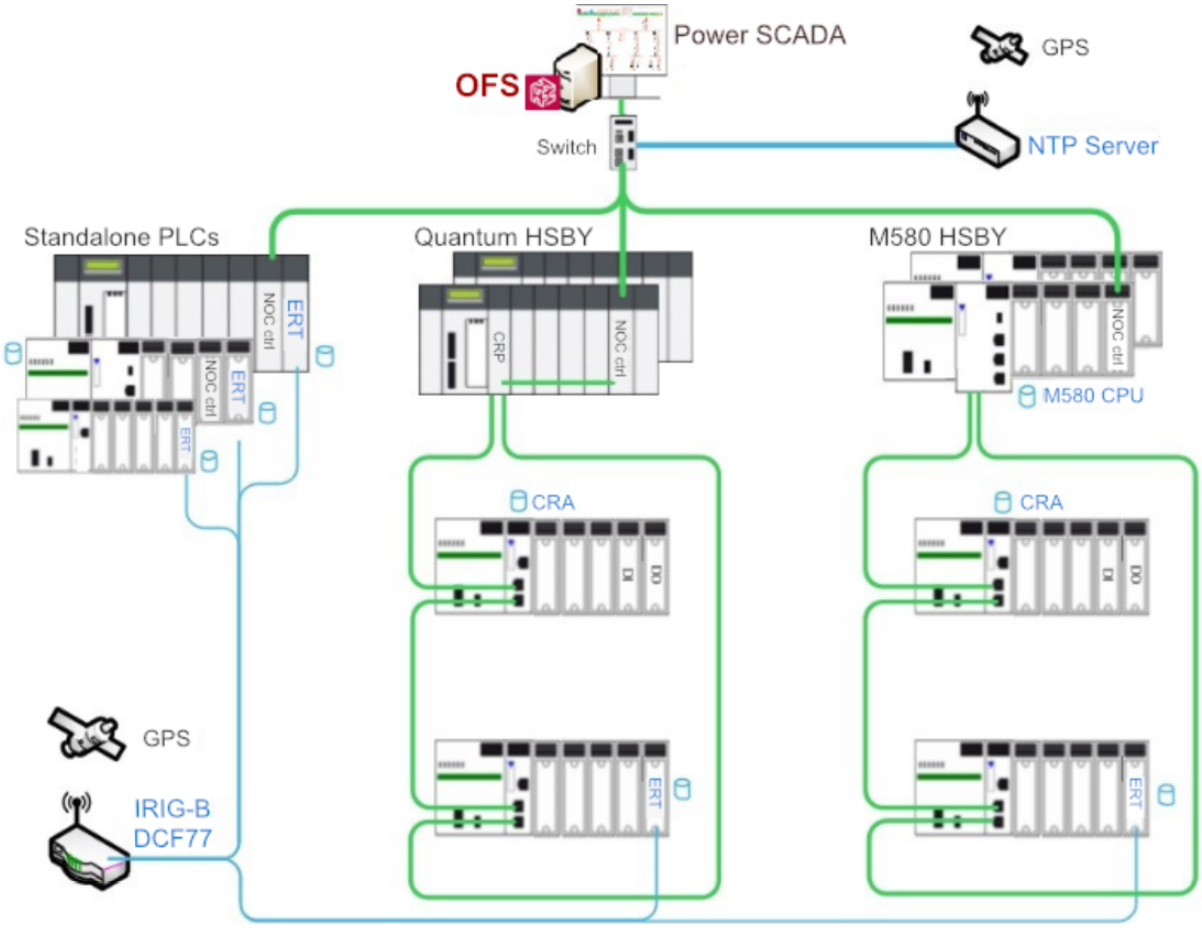
- No PAC programming required: All the time stamped events are managed and transferred automatically by OFS
- Direct communication between the time stamping modules and the client: The available communication bandwidth in the PAC is preserved
- Advanced diagnostic functions:
  - Signaling of uncertain SOE (sequence during which some events may be lost) to the client
  - Time quality information is associated with each time stamped event
- No loss of events in normal operating conditions:
  - An event buffer stores the events in each event source module. The event buffer behavior is configurable

- Both rising and falling edge transitions can be stored for both discrete I/O and PAC internal variables

- Works with both a redundant hot-standby PAC and redundant SCADA

The current limitations of the system time stamping are:

- A communication path between OFS and the time stamping sources is required, so, routing is necessary in multi-layer architectures.
- 2 OPC servers (running for HMI and SCADA) cannot simultaneously access the same time stamping source. A reservation mechanism is implemented.
- No detection of transition edges; the event detection is processed only on both edges.

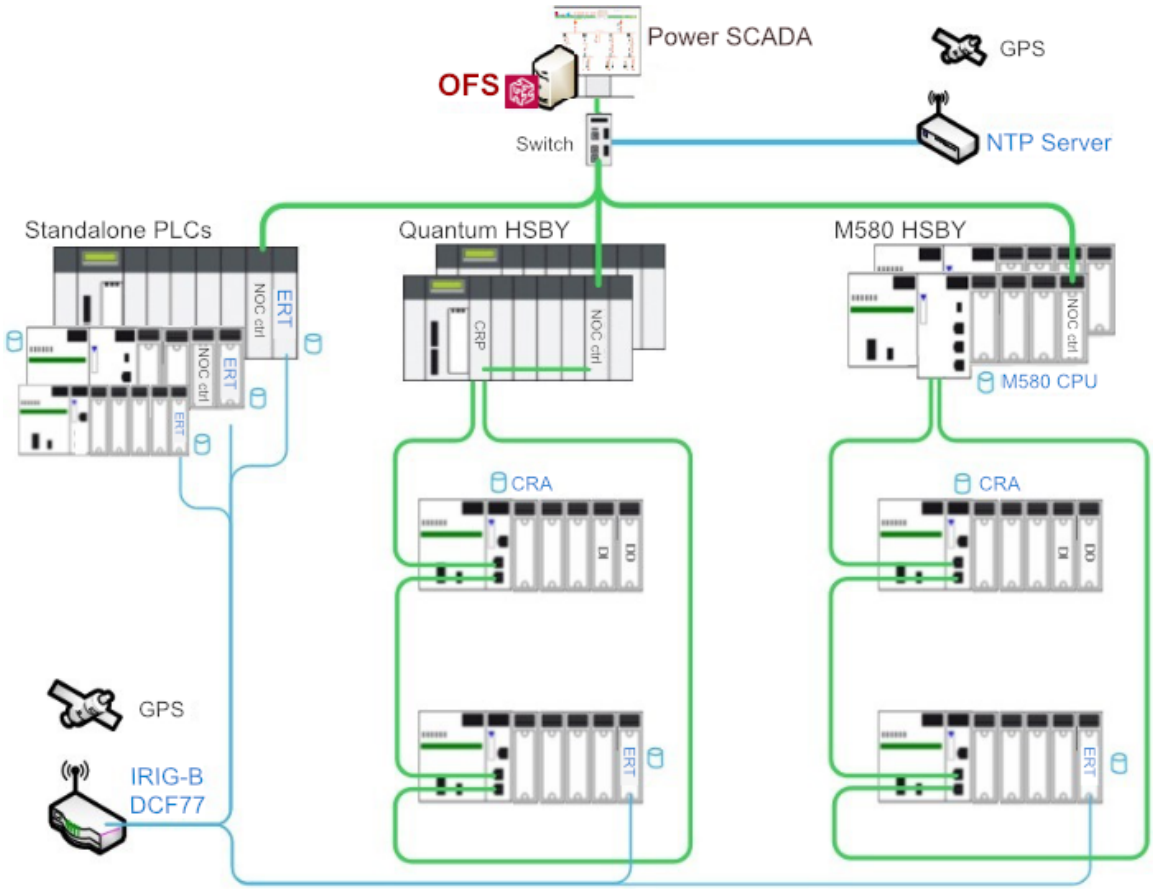


The following table describes the main features and differences between these two methods.

Process	System Time Stamping
1. Synchronize the time clock	ERT module is synchronized by IRIG-B/DCF77 link and x80CRA & M580 CPU are synchronized by the NTP server
2. Time stamping of events generation	I/O events are stamped by x80 ERT modules & CRA Internal variable values are stamped by the M580 CPU

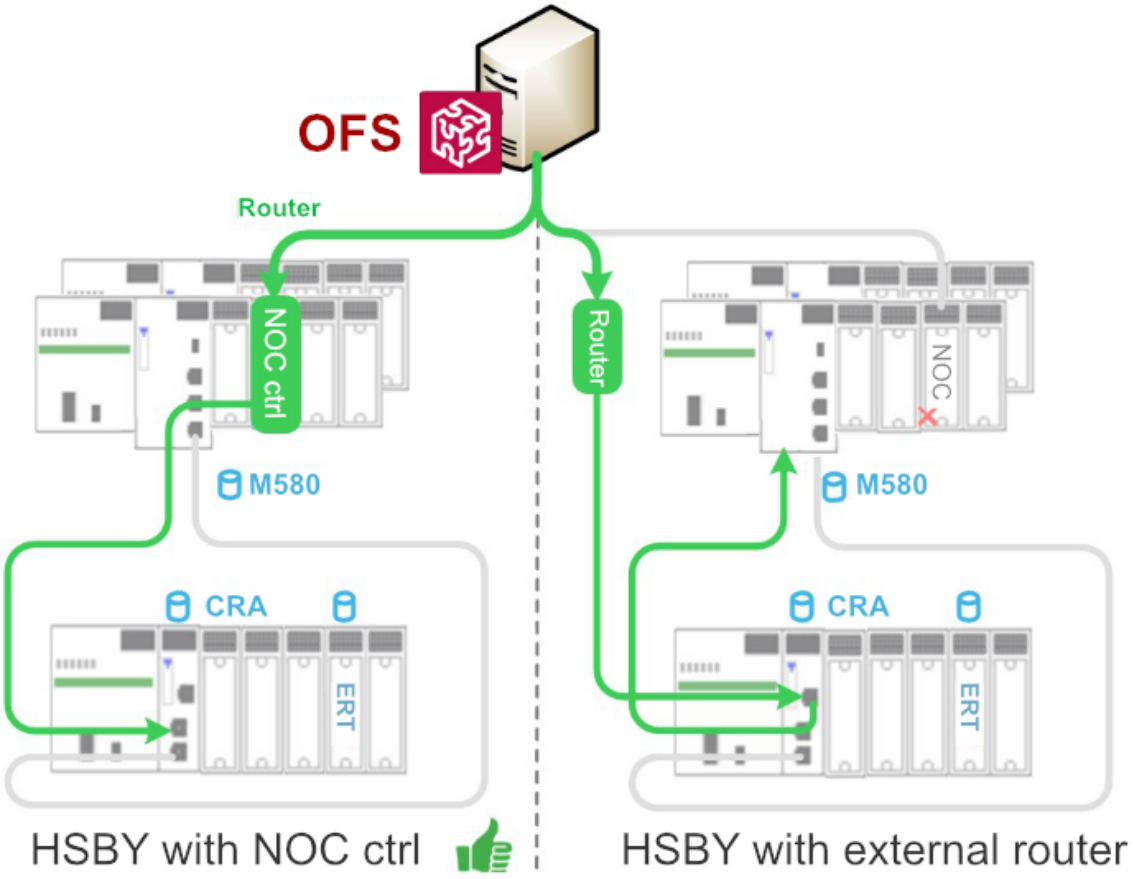
Process	System Time Stamping
3. Manage the time stamped events in PAC buffer	Events are managed and transferred to Power SCADA automatically by OFS
4. Transfer time stamped events from PAC to SCADA	Events are managed and transferred to Power SCADA automatically by OFS

Architecture selection

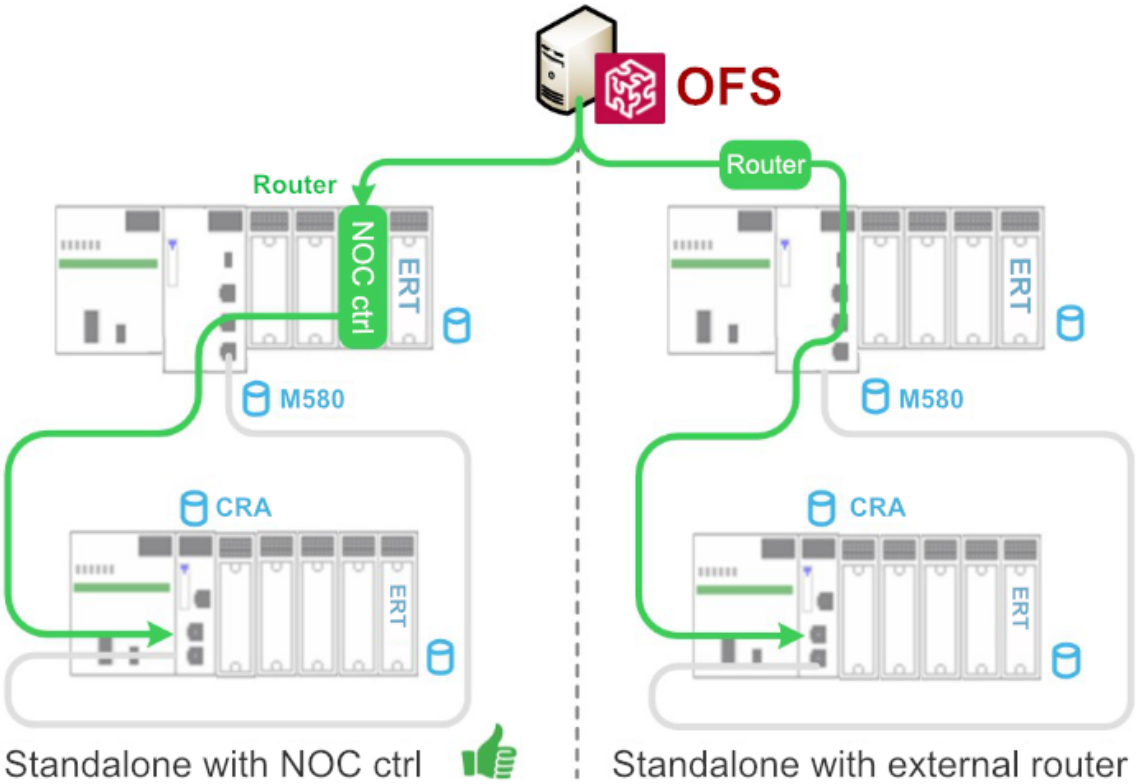


There are three types of modules which are supported by the system time stamping solution, including the M340/eX80ERT, eX80CRA, and M580 CPU. In the system time stamping architecture, OFS is used to automatically transfer the events from the time stamping module to the SCADA. As the time stamping module and OFS are on separate subnets, it is necessary to select a router to link these two subnets.

- In the standalone architecture, we can either select the NOC control module or a third-party router connected to the CPU service port/NOC module which is linked to RIO network in order to set up the connection between OFS and the time stamping module.

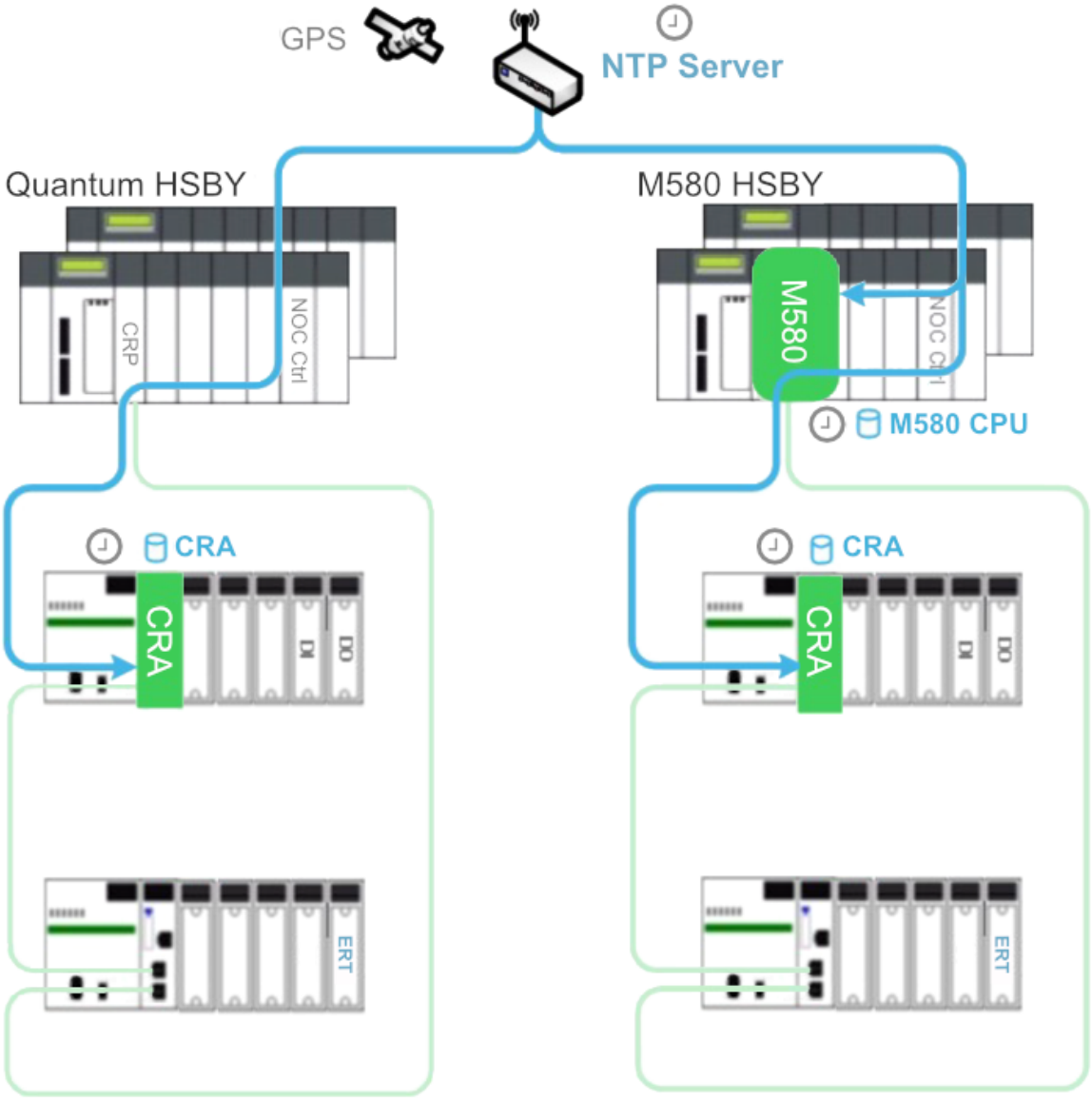


- In the HSBY architecture, we can either select the NOC control module as a router, or select a third-party router directly connected to the RIO network to set up the connection between OFS and the time stamping module.

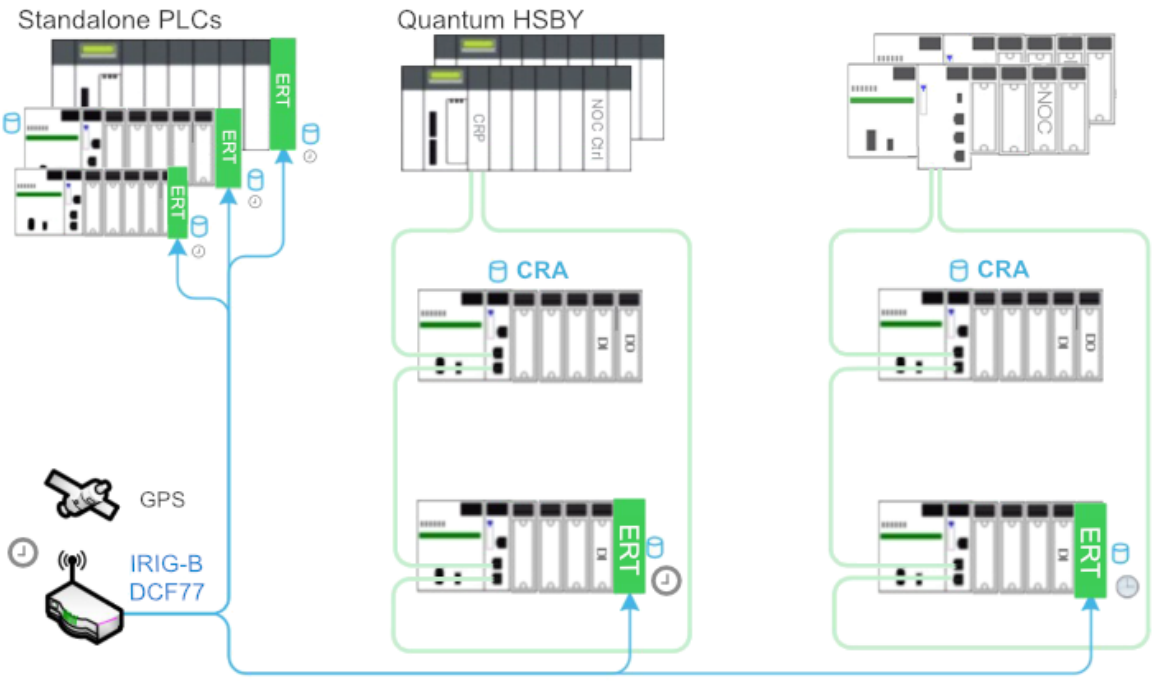


Time synchronization

- The external NTP server provides the time clock for the CPUs and CRAs. We have to configure the NTP server's IP address and polling period for each NTP client. In the M580 architecture, the M580 CPU can act as an NTP server to synchronize its CRA module's time clock.

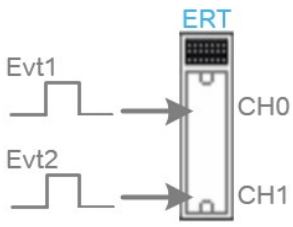
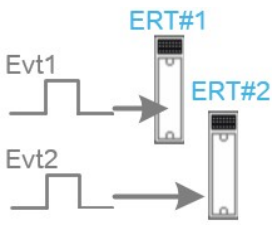
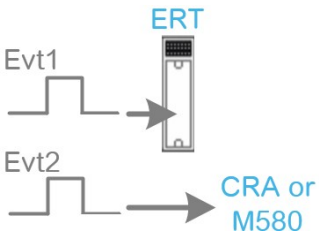
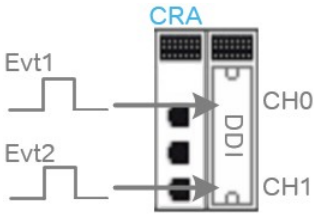
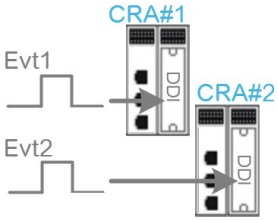
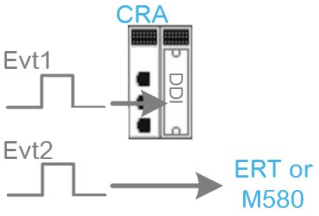
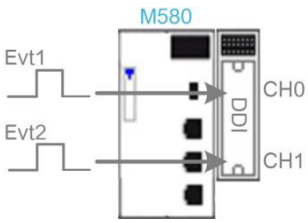
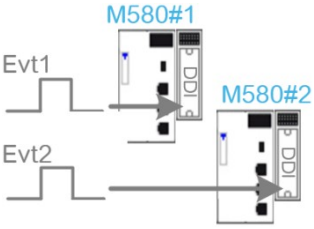
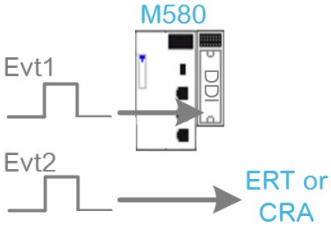


- The IRIG-B 004/5/6/7 or DCF77 signals generated by the GPS receiver are used to synchronize the ERT module's time clock.


Standalone PLCs
Quantum HSBY
ERT
ERT
ERT
CRP
NOC Ctrl
NOC
CRA
CRA
ERT
ERT
GPS
IRIG-B
DCF77

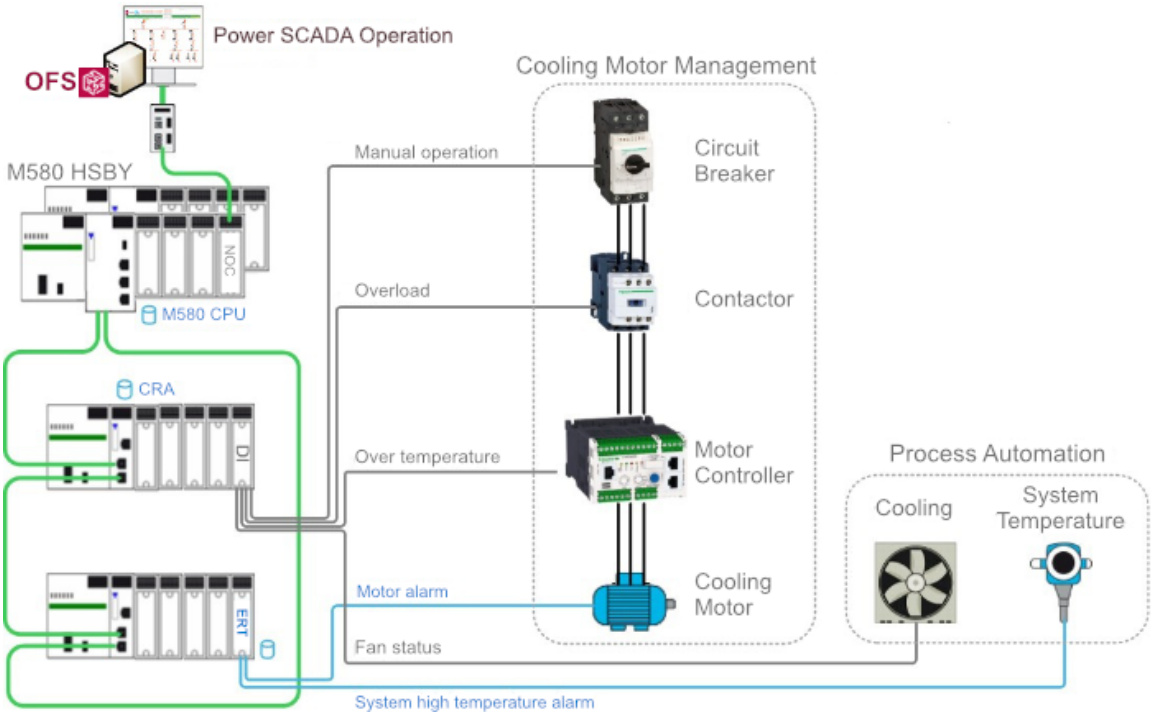
Event resolution

The resolution time is an important parameter for the time stamping application as it impacts the precision of the sequence of events. Below is the list of the resolution times depending on where the events are detected.

TS source module	Events recorded by one module	Events recorded by two modules of the same type	Events recorded by two modules of different types
M340/x80 ERT			
	Min 1ms resolution	Min 2ms with IRIG-B 004/5/6/7 Min 4ms with DCF77	Depends on CRA or M580 scan time
(e)X80 CRA			
	CRA scan time, average 3ms	Average 10ms resolution	Depends on CRA or M580 scan time
M580 CPU			
	CPU MAST task scan time	Depends on large M580 scan time	Depends on CRA or M580 scan time

SOE architecture design

This guide uses the M580 HSBY architecture as an example to design an SOE function.

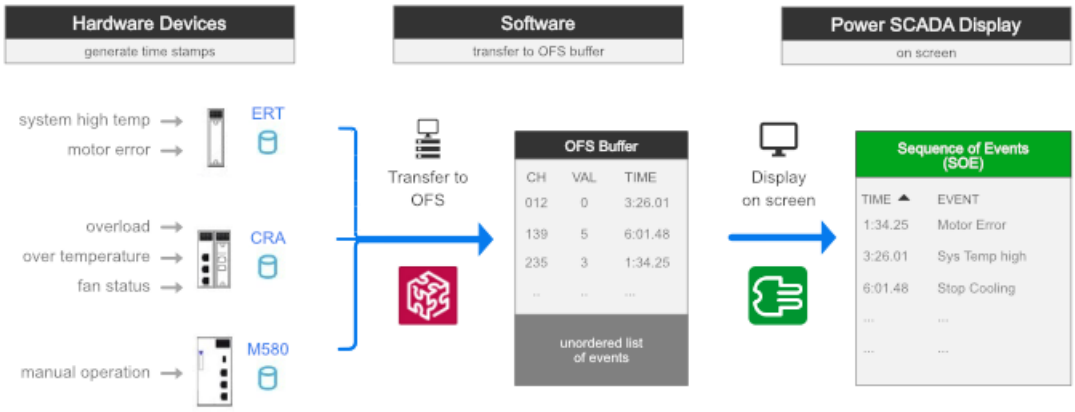


In the above diagram, a cooling control system includes a circuit breaker, a contactor, a motor controller, a motor, and a fan. The fan is used to cool down the system temperature when the temperature is higher than the pre-set value. For the process automation monitoring, some device statuses and process values need to be acquired by the PAC. Meanwhile, these statuses need to be time stamped by the PAC for building an SOE service. The first step to designing the SOE function is to define which time stamping module will be used to monitor the status of the devices, and the process for generating the time stamping events. The table below shows which time stamping module is associated with which event.

Event level	Event name	Source devices	TS module
Process events	High temperature alarm	System temperature instrument	M340/eX80 ERT module
Device events	Motor alarm	Motor	
	Overload	Contactor	eX80 CRA with RIO module
	Fan status	System cooling fan	
	Over temperature	Motor controller	
	Manual operation	Circuit breaker	M580 CPU with RIO module

Data flow design

The following image shows the flow of the time stamped data from the devices to the SCADA using the system time stamping solution:



1. Events are detected and time stamped by the time stamping module
2. Manage the time stamping events using OFS
3. Transfer these events to SCADA using OFS, and display them on the SCADA pages

Licensing

Every Power SCADA Operation component purchased must have an associated license.

The following license option is available:

1. USB license dongle (PSA109921: Power SCADA USB dongle)

   A USB dongle can host licenses for several components of a Power SCADA system. However, only one physical USB dongle is allowed on a machine.

NOTE: For a list of Power SCADA Operation with Advanced Reporting and Dashboards commercial references, see "Commercial References" on page 81.

Licensing Support for Power SCADA Components

The following table lists the supported license matrix for software keys and USB dongles:

Power SCADA Component	Software Key License Option Availability	USB Dongle License Option Availability	Location where license can be hosted
Server	Yes	Yes	On machine where Server is installed
Client	Yes	Yes	Floating license model: On machine where Primary Server is installed Static license model: On machine where Client is installed
Redundancy Client	Yes	Yes	On machine where Stand-by Server is installed
Power SCADA Anywhere	Yes	No	On machine hosting Power SCADA Anywhere
Advanced Reporting	Yes	No	On machine hosting Advanced Reporting
Additional Advanced Reporting Modules	Yes	No	On machine hosting Advanced Reporting

Licensing Advanced Reporting and Dashboards Module

Advanced Reporting and Dashboards Module is included on the Power SCADA Operation 9.0 with Advanced Reporting and Dashboards DVD and can be purchased as a single commercial reference with Power SCADA Operation.

Advanced Reporting and Dashboards Module requires no additional Power Monitoring Expert clients or Power Monitoring Expert device licenses to be purchased.

Internal License Keys for system development

Consider obtaining the Power SCADA development license key. The Power SCADA development license (PSA109502) is intended for internal use by application engineers within country teams and by partners as they develop Power SCADA systems for end users. These development keys are NOT available to end users. The Power SCADA development license includes:

- Allows 8 days of continuous use.
- Expiry date set to 12 months from date of purchase.
  - To renew an expired key, email scada.orders@schneider-electric.com with key information (a screen shot of the licensing screen) and a request that the key needs to be renewed.
- As with any Power SCADA component, the development key requires an associated USB or software key license. In addition to the development key, you will also need to also order a license key either:
  - PSA109921: Power SCADA USB dongle

    NOTE: If ordering USB dongle, some components will still be delivered as software entitlement certificates as they do not support USB dongles.

Power SCADA development key (PSA109502) contains:

- 2 x PSA101199: Power SCADA Server, Unlimited Points
- 20 x PSA102099: Power SCADA Control Client, Unlimited Points
- 20 x PSA103099: Power SCADA View-only Client, Unlimited Points
- 1 x PSA104112: Advanced Reporting and Dashboards Module
- 1 x PSA104114: Billing Module
- 1 x PSA104115: Breaker Performance Module
- 1 x PSA104116: Energy Analysis Module
- 1 x PSA104118: EPSS Module
- 1 x PSA104119: UPS Performance Module
- 1 x PSA104120: Generator Performance Module
- 1 x PSA104121: Power Capacity Module
- 1 x PSA104122: Power Efficiency Module
- 1 x PSA104123: IT Billing Module
- 1 x PSA104124: Power Quality Advisor Module

NOTE: Energy Analysis Dashboards Module MUST be purchased from Victoria plant using PME license portal. This is the only Advanced Reporting Module not available for purchase via SOC order point.

Transferring Licenses

If you design and order a system with USB dongles and later want to change to software keys or vice versa, you can transfer the Power SCADA component licenses to a different license type by ordering PSA109401: License Transfer

NOTE: PSA109401 would need to be ordered for each component whose license is being transferred.

Commercial References

Power SCADA Server

- PSA101199 - Power SCADA Server, Unlimited Points
- PSA101115 - Power SCADA Server, 15000 Points
- PSA101114 - Power SCADA Server, 5000 Points
- PSA101113 - Power SCADA Server, 1500 Points
- PSA101112 - Power SCADA Server, 500 Points

Power SCADA Clients

- PSA102099 - Power SCADA Control Client, Unlimited Points
- PSA102015 - Power SCADA Control Client, 15000 Points
- PSA102014 - Power SCADA Control Client, 5000 Points
- PSA102013 - Power SCADA Control Client, 1500 Points
- PSA102012 - Power SCADA Control Client, 500 Points
- PSA102088 - Power SCADA Control Client Redundant license
- PSA103099 - Power SCADA View-only Client, Unlimited Points
- PSA103088 - Power SCADA View-only Client Redundant license

Power SCADA Anywhere

- PSA105100 - Power SCADA Anywhere, 5 User Pack

Advanced Reporting & Dashboards Module and Additional Advanced Reporting Modules

- PSA104112 - Advanced Reporting and Dashboards Module
- PSA104114 - Energy Billing Module
- PSA104115 - Breaker Performance Module
- PSA104116 - Energy Analysis Reports Module
- PSA104121 - Capacity Management Module
- PSA104124 - Power Quality Advisor Module
- PSA104125 - Insulation Monitoring Module
- PSA104126 - Backup Power Module

Localization

The following table lists the Power SCADA components that are localizable:

Power SCADA Component	Translatable	Comments
Power SCADA runtime	Yes	Can be done and has been done successfully by country organizations (DBF files can be updated by application engineers; this includes the alarm text.)
Power SCADA Power Applications (LiveView, Basic Reports)	Yes	By translating RESX files. (Contact factory for additional detailed steps to support this.)
Power SCADA engineering tools	No	
Power SCADA Anywhere	Not confirmed *	This is the translation of the web client log in screen
Advanced Reporting & Additional Advanced Reporting Modules	n/a	Available in the following languages: • English • Spanish • French • German • Swedish • Italian • Polish • Czech (excluding online help) • Russian • Simplified Chinese • Traditional Chinese

* Indicates that the factory has not tested using this configuration. This may work but we do not officially support the translation of the Power SCADA Anywhere log in screen using multi-byte characters

Integrating with other systems

This section provides information on the different approaches and technologies for integrating Power SCADA Operation with other systems.

Use the links in the table below to find the content you are looking for:

"EcoStruxure Building Operation" on page 84	Integration architecture, component usage, data flows, and communication design.
"Power SCADA OPC DA" on page 87	Standalone and redundant architectures and data flow

EcoStruxure Building Operation

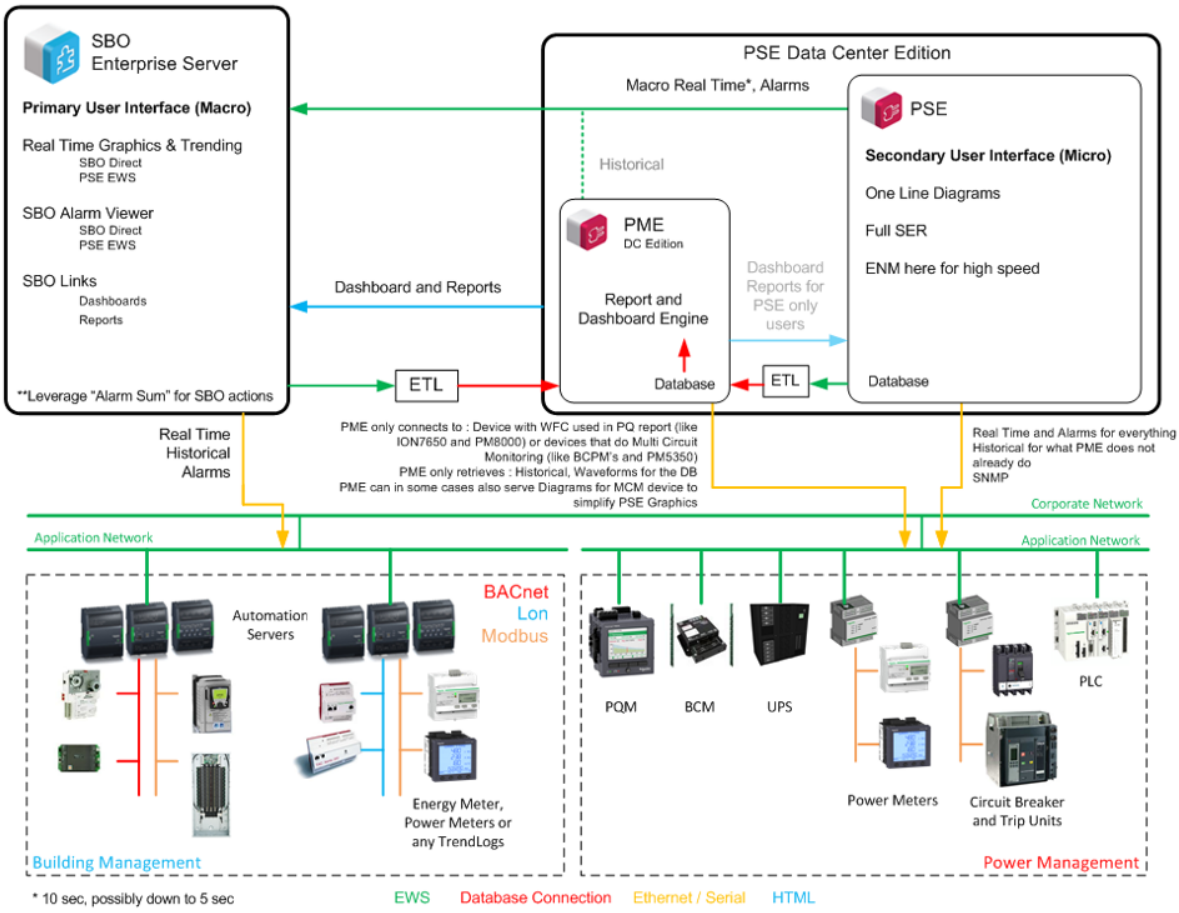
EcoStruxure Buidling Operation integrated with Power SCADA Operation with Advanced Reporting and Dashboards combines electrical and mechanical systems into a single advanced solution.

The main integration points in EcoStruxure Buidling Operation and Power SCADA Operation with Advanced Reporting and Dashboards architecture are:

- EcoStruxure Web Services (EWS) provides alarm data and high level real time data from Power SCADA Operation to EcoStruxure Buidling Operation graphics screens.

  NOTE: On average, expect to a 10 second alarm and real time data update time between EBO and PSO systems.

- The EcoStruxure Buidling Operation to Power Monitoring Expert ETL sends mechanical data to the historical database for display in dashboards and reports within PSO or EBO.
- Integration of Reports and Dashboards from Power Monitoring Expert to EcoStruxure Buidling Operation to view electrical data



The following table lists how components are used in a combined solution:

	Real Time Information	Alarms	Waveforms	Historical Reports and Dashboards	OPC DA an d SNMP
EcoStruxure Building Operation	Enabled (graphics screens for macro level real-time data and EBO trendin g	Enabled EBO aggregate s alarms from PSO and PME using EWS	n/a	n/a	n/a

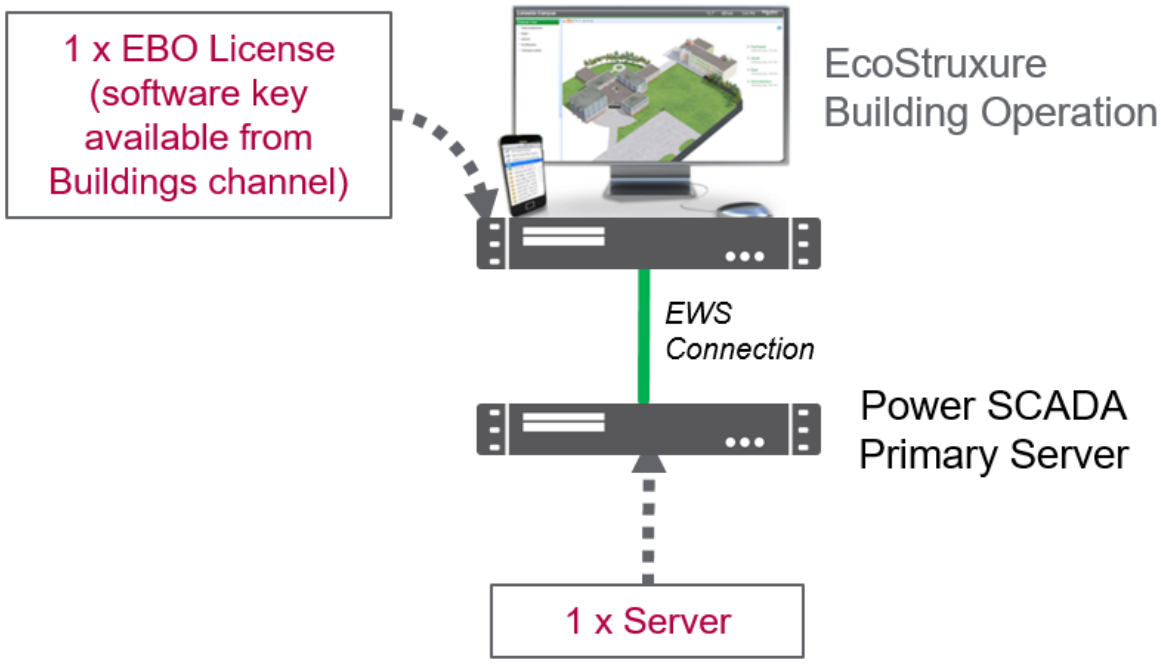
Power SCADA Operation	Enabled (animated one-line, LiveView)	Enabled	Enabled (used for Sequence of Events analysis)	Disabled	Enabled (native drivers without Power SCADA are used)
Power Monitoring Expert	Disabled (Vista, PME real time tables, real time trends)	Disabled	Enabled (used by PME Power Quality reports)	Enabled (PME Web Reports and Dashboards integrated into EBO or PSO runtime)	Disabled (optional drivers for PME not required)

Architecture #1: Simple EcoStruxure Buidling Operation system without redundancy

The following image represents the simplest system that can be configured for Power SCADA and EcoStruxure Buidling Operation. As a best practice EcoStruxure Buidling Operation is installed on a separate machine from Power SCADA.

EcoStruxure Web Services (EWS) sends Power SCADA alarm data to EcoStruxure Buidling Operation. EcoStruxure Buidling Operation operators can acknowledge these alarms. EcoStruxure Buidling Operation acknowledgments are then sent back to Power SCADA.

NOTE: EWS for Power SCADA must always be installed on a Power SCADA Server.



EcoStruxure Web Services (EWS)

EcoStruxure Web Services (EWS) for Power SCADA Operation shares real-time, historical, and alarm data with EcoStruxure™ Building Operations (EBO) and historical data with Power Monitoring Expert (PME). Do not confuse this feature with the EWS Server that was released as a part of PowerSCADA Expert/Vijeo Citect version 7.40 (which is for tag level process data).

EWS uses web-based HTTP protocol to transfer data. It enables two-way data transfers, which allows the acknowledgment of alarms from EBO. To include this new EWS implementation in your installation, select the EWS Server check box during installation.

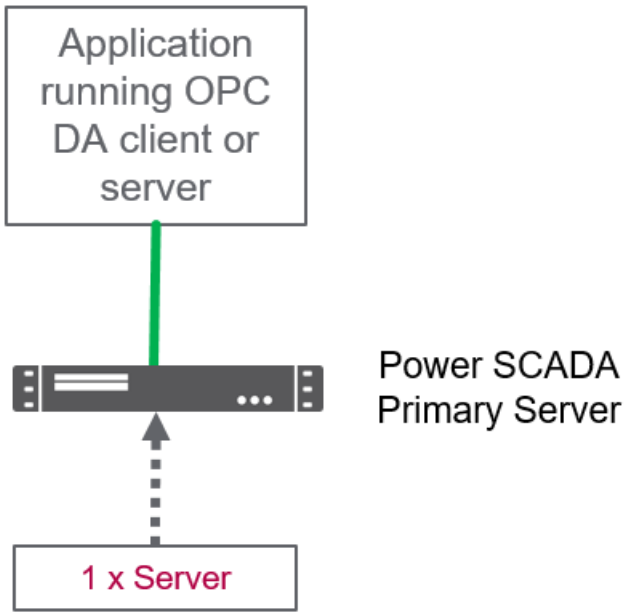
EWS is set up and configured using the Application Configuration Utility.

Power SCADA OPC DA

Architecture #1: Simple system without redundancy

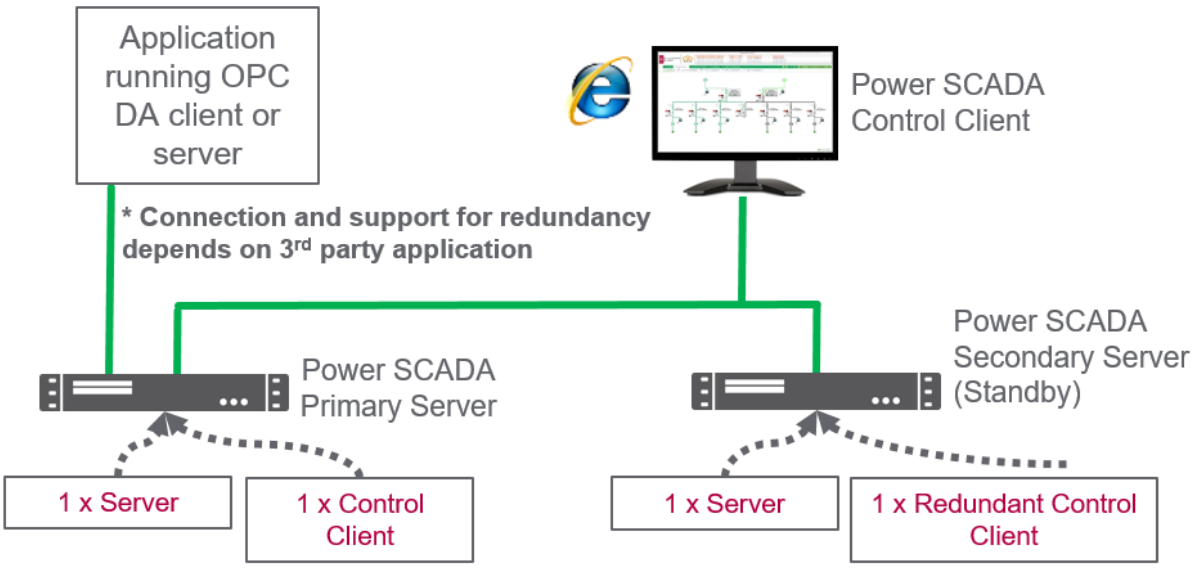
The following image illustrates the simplest system that can be configured for a 3rd party application that is consuming or sending OPC information to the Power SCADA Server.

NOTE: OPC DA client/server is included and hosted on the Power SCADA Server.



Architecture #2: OPC DA client/server with Server redundancy

The following image illustrates a redundant Power SCADA system that is sending and consuming data from OPC DA to a 3rd party application.



The ability to support the redundant Power SCADA architecture depends on the 3rd party application. If the 3rd party application does not have a concept of working with redundant systems, then you should connect to the Primary Server, as pictured. Otherwise you can configure the 3rd party application to connect to both Primary and Secondary Servers.

NOTE: OPC DA client/server is included and hosted on the Power SCADA Server.

Extending Power SCADA

Power SCADA Operation offers several means to extend and customize your system.

CiCode scripting

CiCode allows you to access all real time data within Power SCADA. It is a built-in and well-documented scripting language requiring no previous programming experience to use.

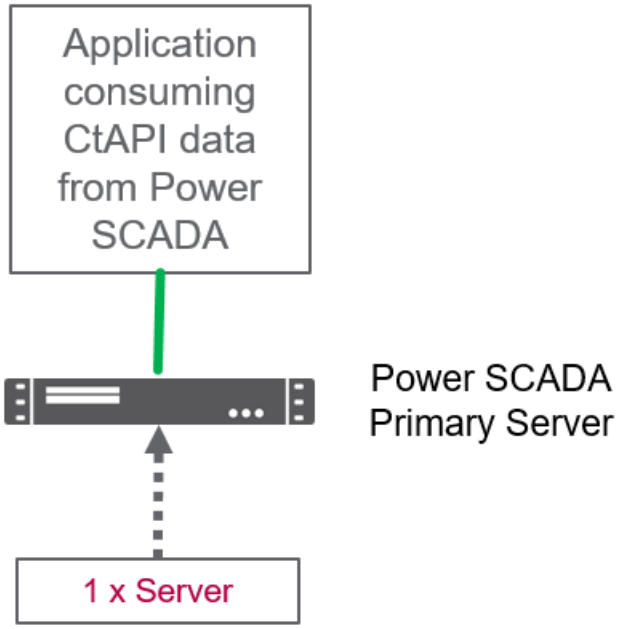
CtAPI

CtAPI is an Application Program Interface (API) for programmers to create applications that extend Power SCADA by using industry standard programming languages such as C, C#, etc... Using CtAPI requires programming experience.

NOTE: CtAPI data can be obtained from the Power SCADA Server or thick Client. A Power SCADA Server or Client can support up to 10 concurrent CtAPI connections.

Architecture #1: Simple system without redundancy

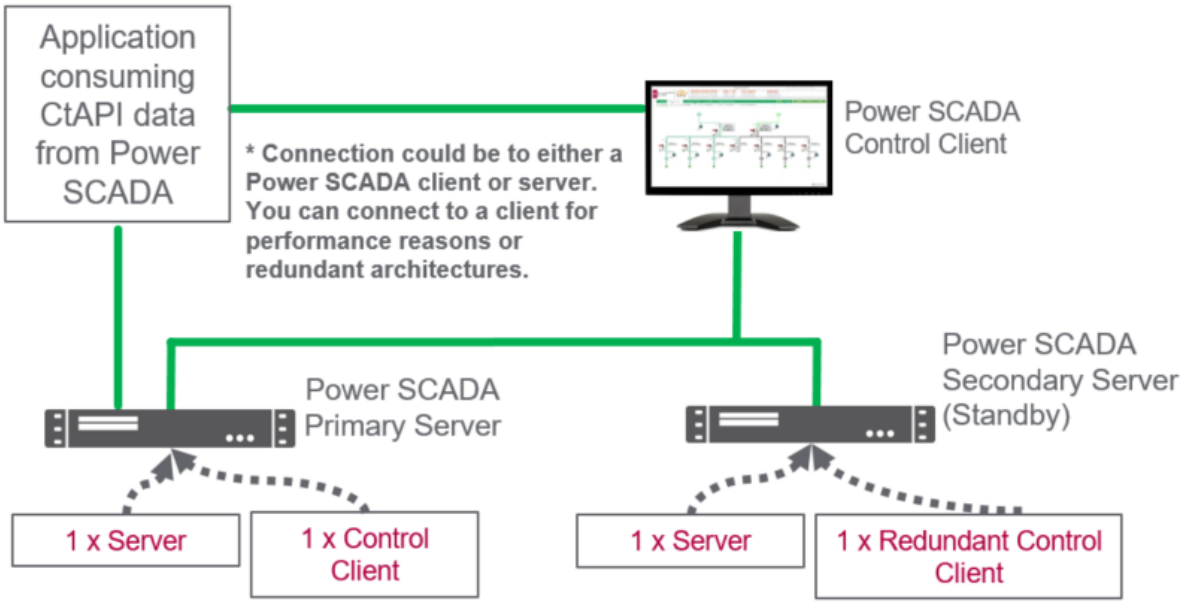
The following image illustrates the simplest system that can be configured for a 3rd party application that is consuming CtAPI data from the Power SCADA Server:



CtAPI documentation can be obtained from this download link.

Architecture #2: CtAPI client with Server redundancy

The following image illustrates a redundant Power SCADA system that can be configured for a 3rd party application that is consuming CtAPI data from the Power SCADA Server or Client:



The ability to support the redundant Power SCADA architecture depends on the 3rd party application. If the 3rd party application does not have a concept of working with redundant systems, then you should connect to the Primary Server (as pictured). Otherwise you can configure the 3rd party application to connect to both Primary and Secondary Servers or a separate Client.

CtAPI data can be obtained from the Power SCADA Server or thick Client. A Power SCADA Server or Client can support up to 10 concurrent CtAPI connections.

Other extensibility resources

A complete list of Power SCADA extensibility points can be obtained from the Power SCADA Integration Map. (download link)

Installing and upgrading

Use the information provided in this chapter for installing, upgrading, and licensing a Power SCADA Operation 9.0 system.

Use the links in the tables below to find the content you are looking for:

Installing

Section	Description
"Installation process" on page 92	An overview of the installation process
"Before installing the software" on page 92	Steps to prepare prior to installing Power SCADA Operation 9.0
"System software order of installation" on page 95	The order of installing Power SCADA Operation and its components
"Installing the software" on page 97	How to install Power SCADA Operation
"Installing the ETL Administration Tool" on page 98	How to install the PSO to PME ETL
"After installing the software" on page 100	Guidelines for how to keep your system up to date and getting started in Power SCADA Operation
"Uninstall and reinstall Power SCADA Operation" on page 101	How to uninstall and reinstall Power SCADA Operation 9.0

Upgrading

Section	Description
"Upgrading" on page 102	A general overview of the steps involved in upgrading your system to 9.0
"Upgrade Method" on page 104	Guidelines for choosing an upgrade methods: Offline or Online.
"Upgrade Path" on page 105	A description of the number of versions to which you need to upgrade to get from your current version to Power SCADA Operation 9.0
"Offline Upgrade in Test Environment" on page 106	General steps for performing an upgrade in a test environment
"Offline Upgrade" on page 107	The upgrade process to perform for an Online Upgrade.
"Online Upgrade" on page 115	The upgrade process to perform for an Online Upgrade.
"Migration Tools" on page 131	A description of the migration tools you must use to upgrade your pre-existing projects for use in Power SCADA Operation 9.0.

Also refer to the "Upgrading Reference" on page 525 section for detailed upgrading information you need to consider, as well as Cicode functions and Citect INI settings that have changed with each successive Power SCADA Operation version.

Licensing

Section	Description
"Licensing" on page 141	An overview of licensing a Power SCADA Operation system
"Update a Sentinel Key with CiUSAFE" on page 142	Updating USB keys
"Activate Licenses Using the Floating License Manager" on page 143	Using the Floating License Manager to activate licenses.
"Dynamic Point Count" on page 144	A description of how Power SCADA Operation uses a dynamic point count
"Specify the Required Point Count for a Computer" on page 145	A description of how points are specified by the computer role
"Run the software in demo mode" on page 145	A list of Power SCADA Operation features you can run without a hardware key

Installation process

You can install Power SCADA Operation with Advanced Reporting and Dashboards as a new product only.

Power SCADA Operation does not support different versions running side-by-side. If you are upgrading from an earlier version of Power SCADA Operation, back up your existing project files. These files include LiveView templates; reporting configurations (such as email addresses); and Profile Editor custom tags, device types, profiles, and units (in the Program Data folder).

Uninstall prior versions before installing V9.0.

Remove existing Floating License Managers installations before installing the new version.

Before proceeding with the installation of Power SCADA Operation with Advanced Reporting and Dashboards and optional components, refer to "Before installing the software" on page 92 for detailed installation prerequisite information.

Before installing the software

This section describes the requirements for hardware, operating system software, and system configuration prior to installing Power SCADA Operation with Advanced Reporting and Dashboards and any of its components.

These requirements will vary subject to the components of Power SCADA Operation with Advanced Reporting and Dashboards that you intend to install on any computer. This section identifies the basic system software requirements, as well as requirements specific to each particular component. Refer to "Core components selection" on page 94 to determine the components that you want to install.

⚠ WARNING

POTENTIAL COMPROMISE OF SYSTEM AVAILABILITY, INTEGRITY, AND CONFIDENTIALITY

- Apply all Windows security updates on machines running Power SCADA Operation and Power Monitoring Expert.

Failure to follow these instructions can result in death, serious injury, equipment damage, and permanent loss of data.

Before you begin to install Power SCADA Operation with Advanced Reporting and Dashboards, it is important that you install the latest updates from Microsoft for your operating system and system software. See "Preparing servers" on page 93 for more information. Also see the Operating System Matrix that shows the operating systems that are compatible with various versions of Power SCADA Operation.

Supported environments

Review the "Hardware and software requirements" on page 51 section to ensure that your hardware and system software meet the requirements for your selected installation.

Preparing servers

The software Installer performs many of the setup and configuration tasks during installation to ensure that the prerequisites for your Power SCADA Operation with Advanced Reporting and Dashboards system are met. Complete the following before proceeding with the installation.

Updating the operating system

⚠ WARNING

POTENTIAL COMPROMISE OF SYSTEM AVAILABILITY, INTEGRITY, AND CONFIDENTIALITY

Apply the latest updates and hotfixes to your Operating System and software.

Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

Run the Windows Update service to install the latest security patches and hotfixes from Microsoft.

Advanced Reporting and Dashboards Module Server

You need to install Advanced Reporting and Dashboards Module on a separate server.

For more information on server requirements and preparation, see the "Advanced Reporting and Dashboards Component" on page 41.

Component selection

Decide which Power SCADA Operation with Advanced Reporting and Dashboards components and add-ons you want to install.

Core components selection

The installer provides a list of options to help you select the appropriate components during installation. The options are described here.

Runtime Environment

Selects Runtime, Sentinel Driver, and Communications Drivers for installation. It is an installation which will install the runtime components for both a Server and Client. Such an installation will include runtime infrastructure files, Client and I/O Server, Alarm Server, Trend Server and Reports Server.

Select this option if this is an installation of Power SCADA Operation that will act as a server to service a number of client installations.

Configuration and Development Environment

Installs the design-time configuration environment. Users who have sufficient security privileges can set up graphics pages, create reports, etc. The configuration tools include: Power SCADA Studio, Application Configuration Utility, IO Device Manager, Project Setup, Project Backup/Restore and the Power SCADA Runtime.

Deployment Client

Installs the Deployment Client component, which allows projects to be deployed to this machine remotely.

Deployment Server

Installs the Deployment Server component, which allows projects to be administered, versioned, and deployed to other remote Deployment Client machines from this machine. The server has the ability to roll out project changes to the various computers in your project.

Sentinel Driver

Install the USB sentinel driver, which allows you to use a physical USB dongle to license Power SCADA Operation. The Schneider Electric License Manager option will install the license manager, which enables software-based Power SCADA Operation licensing.

Add-ons selection

Once you have selected the core components that you want to install, select any add-ons that you want to include in your installed system. The options are described here:

Project DBF Add-in for Excel

Installs an Add-In for Microsoft Excel. When this Add-In is loaded into Excel, it allows you to browse, open, edit and save Power SCADA Operation .dbf files in the correct format. This is only available for selection if Microsoft Excel 2007 or above is installed on the computer. Otherwise, it is visible but is deselected and disabled.

Power SCADA Operation Web Server for IIS

Installs a Web Server running on Microsoft Internet Information Service (IIS). The Web Server performs the server-side functionality of a Web Service to the Web Client. As well as facilitating communication, it directs a client to the graphical and functional content of a Power SCADA Operation project, and the location of the runtime servers. This information is stored on the Web Server when a Power SCADA Operation project is deployed. A Web Server can contain multiple deployments.

NOTE: If the Web Server and Power SCADA Operation Server are set up on different machines, and it is not possible to establish a trust relationship between them, the two machines need to be on the same domain so that the Web Server can access the directory on the Power SCADA Operation Server that is hosting the web deployment files.

If a trust relationship can be established between the Web Server and the Power SCADA Operation server, they can be on different domains as long as the Web Server has read access to the project folder on the Power SCADA Operation Server.

Power SCADA Operation Reporting

Installs the Power SCADA Operation basic reports.

The Power SCADA Operation Profile Editor

Installs the Profile Editor. Profile Editor lets you create tags, device types, devices, and projects outside of the Power SCADA Studio environment.

The Power SCADA Operation LiveView

Installs LiveView. LiveView lets you create table templates for real-time system readings.

System software order of installation

This section provides an overview of the general steps required to install:

- Power SCADA Operation
- Advanced Reporting and Dashboards Module files: Advanced Reporting and Dashboards
- Extract, Transform, and Load (ETL): Use this module to extract historical data from Power SCADA Operation and transform it into a format that can be used in the Advanced Reporting and Dashboards Module.
- Power SCADA Anywhere
- McAfee Application Control

Before you begin, you need the following items:

- Installation medium for Power SCADA Operation with Advanced Reporting and Dashboards and Power SCADA Operation 9.0 Installation Guide.
- Installation medium for ETL and Power SCADA Anywhere (included on the Power SCADA Operation with Advanced Reporting and Dashboards ISO).
- Installation medium for .NET Framework 4.6.1, downloaded from Microsoft.
- Installation for Microsoft SQL Server (SQL Express is included on the Power SCADA Operation with Advanced Reporting and Dashboards ISO, however, SQL Server must be obtained from Microsoft.)

On the Power SCADA Operation Server Computers

The following table lists software that you will install on each of the servers and clients in your project.

Power SCADA Primary Server	Power SCADA Secondary Server	Power SCADA Anywhere	Advanced Reporting and Dashboards Server
Power SCADA Operation 9.0	Power SCADA Operation 9.0	Power SCADA Operation 9.0 control client only	SQL Server
		Power SCADA Anywhere *	Advanced Reporting and Dashboards (from the Power SCADA Operation ISO)
		Windows Terminal Services must be enabled.	ETL

Power SCADA Operation Server Computers

Install all operating system updates before you install Power SCADA Operation.

On the server that you will use for Power SCADA Operation, install software in the following order:

- Verify that you have the correct Internet Explorer version for your operating system. See "Supported browsers" on page 55 for more information.
- Install .NET 4.6.1
- If you want to have Matrikon Explorer on the computer, install Matrikon before you install Power SCADA Operation.
- Install Power SCADA Operation

On the Advanced Reporting and Dashboards Computer

On the server that you will use for the Advanced Reporting and Dashboards Module, install the software in the following order:

- Microsoft SQL Server: You must install SQL Server on the Advanced Reporting and Dashboards server. Refer to the *Power Monitoring Expert 9.0 – System Guide* for information.
- Advanced Reporting and Dashboards Module: Use the Power SCADA Operation with Advanced Reporting and Dashboards installation medium and installation guide.
- On the Advanced Reporting and Dashboards Module server only, install ETL. See "Installing the ETL Administration Tool" on page 98 for details.

NOTE: The installation medium is located on the same DVD or .ISO as the Power SCADA Operation installation, in the Advanced Reporting and Dashboards Module folder.

On the Power SCADA Anywhere Server Computers

You need to install Power SCADA Anywhere on a remote client computer. See "Configure the Power SCADA Secondary Server" on page 462 for directions.

Installing the software

NOTE: Do not have Windows Update running when you install Power SCADA Operation.

When you begin the installation, if any required system software is not detected, you must install it before you can begin the Power SCADA Operation with Advanced Reporting and Dashboards (PSO) installation. For example, if you have not yet installed .NET Framework 4.6.1, you will be prompted to install it first.

To install Power SCADA Operation with Advanced Reporting and Dashboards:

1. Insert the Power SCADA Operation with Advanced Reporting and Dashboards DVD into the DVD drive (or launch the installation media). If you have autorun enabled, the initial Setup dialog will display. If this does not occur, use Windows Explorer to navigate to the root directory of the DVD. Then click `MainSetup.exe` to display the Setup dialog.
2. If your system does not have Microsoft .NET Framework 4.6.1:
   a. Install .NET Framework 4.6.1.
   b. After installing the .NET Framework, restart your system.
   c. After the restart, double-click MainSetup.exe to launch the PSO installer again.
3. When all required software is installed and you launch the `Mainsetup.exe`, the Welcome screen appears. Click **Next**.
4. Select the core components that you want to install.

   For a description of each component, see "Core components selection" on page 94.
5. Click **Next** and then select the add-on components that you want to install.

NOTE: **Project DBF Add-in for Excel** can only be selected if Microsoft Excel 2003, 2006, 2010, or 2013 is installed on the computer.

For a description of each add-on component, see "Add-ons selection" on page 94.

6. Click **Next** and then review the default installation destination folders. Optionally change the installation location if desired.

   If you change the default paths, you can return it to the default setting by clicking Reset.

7. Click **Next**. The **Check System** window appears. This window indicates whether or not the installation will be successful.

   If the installation is unsuccessful:

   a. Click **Open Log** to review where the installation stopped.

   b. Note the files that need to be corrected, and correct them in the order they are presented.

   c. After you make the corrections, click **try again** to re-install PSO.

   d. Repeat this step, as necessary, until all problems are solved.

8. When **System Verified** appears, click **Next**.

9. In **Ready to Configure**, review the component list and make sure you are satisfied with the installation configuration.

10. Click **Install** to continue with the installation or click **Back** to change any of the items.

11. In **Configure System**, click **Next** when the configuration is complete.

    The **Complete** screen indicates that the installation completed successfully.

12. Click **Close**.

Depending on your system architecture, complete the installation of the Power SCADA Operation with Advanced Reporting and Dashboards system components.

Installing the ETL Administration Tool

NOTE: Do not install ETL on the Power SCADA server.

After installing the ETL (PSO to PME) you will need to allow the ETL to remotely access the Power SCADA server. See "Allowing ETL remote access to the PSO Server" on page 392 for details.

Install the ETL Administration Tool on the Power Monitoring Expert server using a Windows Administrator account.

To install ETL for PSO:

1. In Windows Explorer, navigate to \Power SCADA with Advanced Reports ETL.

2. Copy the PSO to PME ETL EXE to the PME server.

3. Double-click `SegApps_ETL_PowerSCADA-xxx.exe`.

   (Where *xxx* is the build number.)

4. **Application Language:** Select your preferred application language from the drop-down list and click **Next**.

   NOTE: The ETL Administration Tool supports English only.

5. **Welcome:** Review the steps and click **Next**.
6. **License Agreement:** Read the End User License Agreement and if you accept the terms of the agreement, click **I Agree** to proceed.
7. **Setup Type:** ETL: Power SCADA 9.0 can only be installed with the **Standalone Server** option. Click **Next**.
8. **File Destination:** Click **Next** to install the ETL tool to the default location. To select a different location, click the ellipsis button and then select a new location. Click **OK**.
9. **Check System:** The installer checks the operating system. If a condition affecting installation is detected, the installer notifies you to correct it. When verification is successful, click **Next**.
10. **Ready to Configure:** A summary of your configuration choices for the installation. Ensure that all items are correct before proceeding.
11. Click **Install** to continue or click **Back** to move back through the installer and change any items.

    The **Copy Files** screen appears and the ETL files are copied to the system.

12. **Configure System:** The selected configuration settings are applied.
13. Click **Next**.
14. **Complete:** The Complete page appears after the install is successful. Click **Installation Log** to view details recorded for the installation process.
15. Click **Close** to finish.

Install Citect Anywhere Server

Power SCADA Anywhere allows a remote desktop session using a Web browser to the Power SCADA server. It is accessible only in the Power SCADA Runtime.

Power SCADA Anywhere is a rebranded name for Citect Anywhere. The term PowerSCADA Anywhere will appear only in the end user-facing Web browser, at the login screen and the launch screen. Everything that is not end user-facing will be referred to as Citect Anywhere, including the installer, the configuration tool, and various file paths.

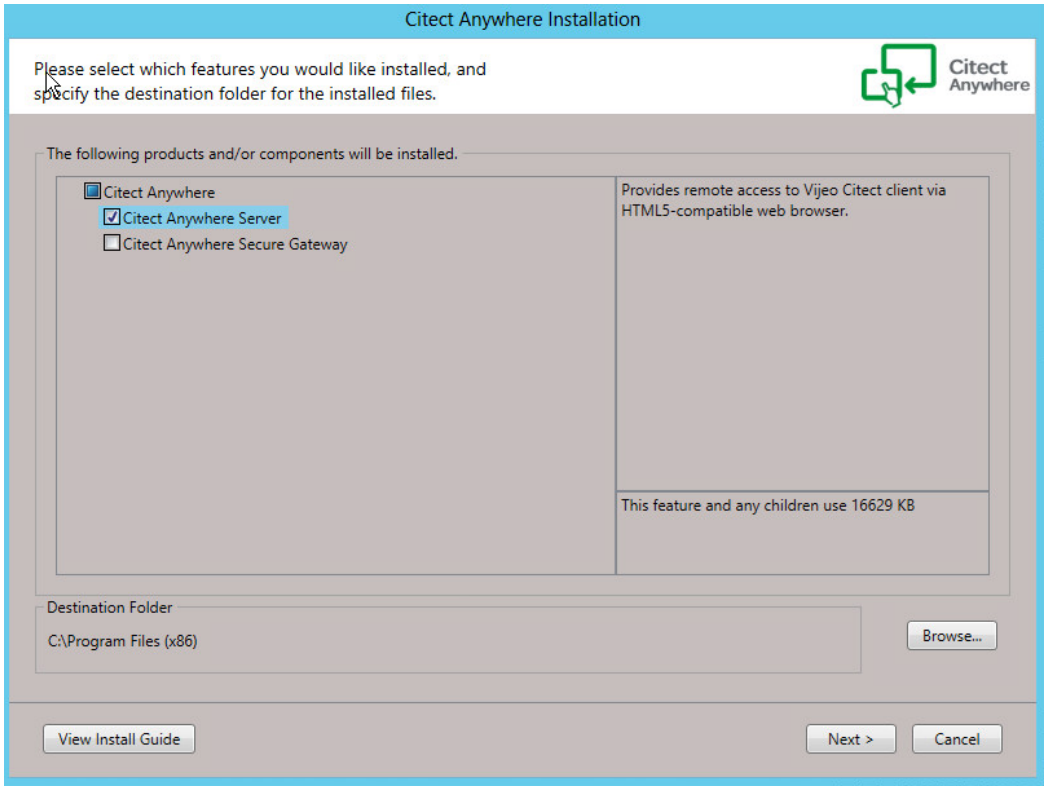
Prerequisites

- Before installing Power SCADA Anywhere, you must first install the Power SCADA Anywhere Server.
- Install a Power SCADA Operation Control Client. For the Power SCADA Operation Control Client, run the Power SCADA Operation install and select the control client-only installation. This installation requires a floating license. It must be on one of these operating systems:

- Windows Server 2008 R2 SP1 Standard, Enterprise (64-bit)
- Windows Server 2012 Standard

To install Power SCADA Anywhere:

1. On the machine where the Power SCADA Anywhere server is installed, launch the installer from the Power SCADA Anywhere installation folder: double-click setup.exe.
2. Click **Citect Anywhere Server**:



3. Accept the license agreement and click **Next** on each screen of the installation. If a pre-requisite is missing, it will be installed for you.
4. When installation is complete, you see a confirmation screen. Click **Finish** to close the install.

For detailed instructions on installing and using the Power SCADA Anywhere Server, see the following documents:

- Power SCADA Anywhere Quick Start Guide.pdf
- Power SCADA Anywhere Installation and Configuration Guide.pdf

These documents are located in the Power SCADA Anywhere Installer folder.

After installing the software

Maintaining system currency

After you install and configure Power SCADA Operation 9.0 with Advanced Reporting and Dashboards and deploy it as your production system, it is very important that you keep your software up to date. Schneider Electric will periodically publish updates in the form of service

releases, hot fixes, or advisories relating to safety, security, and functionality of Power SCADA Operation.

Getting started with Power SCADA Operation

Power SCADA Operation is a suite of tools that lets you develop, design, and deploy power SCADA systems. Built on the Citect SCADA platform, Power SCADA Studio is the main power SCADA development portal. Use Power SCADA Studio to:

- Create, manage, and customize Power SCADA projects
- Create and manage I/O devices
- Design Power SCADA Runtime elements
- Manage user access
- Open other Power SCADA productivity tools.

Power SCADA Operation is shipped with a project that has example page configuration.

To launch Power SCADA Studio:

- Click Start > Schneider Electric > Power SCADA Studio

  OR
- From the desktop, open the Power SCADA Operation folder and then open Power SCADA Studio.

(Optional) Install and Deploy the Power SCADA Operation Web Client

To install and deploy a remote Power SCADA Operation Web Client, follow the steps outlined in the Vijeo Citect 2015 Web Client Guide.

Uninstall and reinstall Power SCADA Operation

Use Add/Remove Programs in the Windows Control Panel to uninstall these programs:

- Power SCADA Operation 9.0 (if you uninstall this, you also uninstall the Profile Editor)
- Power SCADA Operation Profile Editor
- Any additional Power SCADA Operation programs, such as the WebServer, that you installed

If you uninstall programs after you have already created projects, the project data will not be deleted. It is in [Project Drive]\ProgramData\Schneider Electric\Power SCADA Operation\v9.0\User. The first time you launch the application after you re-install it, it will locate the project data and re-link it.

Upgrading

NOTICE
LOSS OF DATA Backup your project and other relevant historical data files from all servers in the system. **Failure to follow these instructions can result in a loss of data.**

Carefully follow this guide. See also Back up your current project and relevant files for details on which files to back up.

The upgrade steps are:

1. "Before Upgrading" on page 103: Ensure you have the required installation files and other collateral before proceeding.
2. "Upgrade Method" on page 104: Depending on whether your system can afford downtime and loss of data, choose an upgrade method: "Offline Upgrade" on page 107 or "Online Upgrade" on page 115.
3. "Upgrade Path" on page 105: Determine the upgrade path. Upgrade path refers to the number of versions to which you need to upgrade to get from your current version to Power SCADA Operation 9.0. For upgrading to intermediate versions specified in the upgrade path (for example, v7.20 or v2015), refer to the documentation for those versions.
4. "Offline Upgrade in Test Environment" on page 106 to upgrade and migrate the existing project to Power SCADA Operation 9.0.

   NOTE: It is recommended that this action be taken before going to the Production site, or in a test environment before performing a Production upgrade.
5. Complete the Offline Upgrade or the Online Upgrade in the Production environment.

NOTES:

- For version v7.20 onwards, cross version compatibility is not available for alarms.
- When updating the computer with a new product version, backup the existing projects and uninstall the existing installation. Install the new version and restore projects into the new version.
- The new version you are installing may have a service pack released. The service pack may have a fix for the automatic upgrade and may be required to be installed before restoring the project. Please refer to the service pack documentation.
- With branding changes being introduced in Power SCADA Operation 9.0, path names may be different from those used in previous versions. It is recommended that you verify the source/destination paths carefully while performing operations such as backup and restore during the upgrade
- For instructions related to previous versions of Power SCADA Operation, such as backing up a or restoring a project, consult the documentation for that version.

Before Upgrading

Required Installation Files

You must have the following files to complete the recommended preparation work before going to the production site to perform the "Offline Upgrade" on page 107 or "Online Upgrade" on page 115:

- Power SCADA Operation 9.0 ISO – Available for download from the Exchange Community
- Power SCADA Operation to Power Monitoring Expert ETL – Found on the Power SCADA Operation ISO

Available Tools

- Easy RoboCopy: a graphical user interface for the RoboCopy utility available from the Citect Knowledgebase

Preparation Work

- Ensure that the production servers' and clients' hardware/OS/software on each machine meets the requirements for Power SCADA Operation 9.0 outlined in the Installation Guide. Be sure to also reference the "Planning" on page 27 section to incorporate additional machine resources when adding Advanced Reports and Dashboards modules to the Power SCADA servers.
- Upgrade the license keys for the project. Find the server and client license key serial numbers. Generate upgrade authorization codes using the online license generator and save the codes and the serial numbers to a text file. This ensures production site is in support and will be allowed to upgrade and operate the Power SCADA Operation software. It also ensures all the keys are registered to the correct site.
- Determine the version of Power SCADA Operation (formerly PowerSCADA Expert) currently in use at the production site, in order to determine the correct "Upgrade Path" on page 105. To do this use the Help > About menu in the Project Explorer and using the Technical Info tab identify the Power SCADA Operation version. Check that the version of "Citect32.exe" matches the product version in the table below.

Product Version	File Check	File Version	Notes
7.20	Citect32.exe	7.20.1.33	
7.20 SR1	Citect32.exe	7.20.4.38	
7.30 SR1	Citect32.exe	7.30.0.601 or 7.30.1.94	Either version satisfies as indicating v7.30 SR1 is installed.
7.40	Citect32.exe	7.40.1.239	
7.40 SR1	Citect32.exe	7.40.1.239	

Product Version	File Check	File Version	Notes
8.0	Citect32.exe	7.40.1.239	This the same file version as in v7.40. As an additional check, in Programs & Features "PowerSCADA Expert 8.0" should be listed.
8.0 SR1	Citect32.exe	7.50.0.4107	
8.1	Citect32.exe	7.50.0.4150	
8.2	Citect32.exe	8.0.0.2065	
9.0	Citect32.exe	8.10.0.2086	

Upgrade Method

Before you plan to upgrade to Power SCADA Operation 9.0, consider whether your SCADA system can afford downtime and whether all of your historical information needs to be available at all times. The upgrade method you choose will depend upon this.

There are 2 upgrade types:

- "Offline Upgrade" on page 107: This method requires your system to be shut down for the duration of the upgrade. If your system can afford downtime, and depending on whether all of your historical information needs to be available at all times, use this method. This is the basic upgrade process that will be required even if you use the online upgrade method.

  NOTE: It is strongly recommended that the Offline Upgrade process is exercised in a test environment, and even before traveling to the Production site, in order to discover potential problems in the upgrade process that can be fixed before attempting an Online Upgrade. This will minimize server downtime in the Online Upgrade process or save time and effort if completing an Offline Upgrade in the Production environment.

NOTICE
LOSS OF DATA Backup your project and other relevant historical data files from all servers in the system. **Failure to follow these instructions can result in a loss of data.**

  Carefully follow this guide. See also Backup your current project and relevant files for details on which files to back up.

- "Online Upgrade" on page 115: If you need your system to be available at all times, use this method. To be able to conduct an online upgrade, you need to have at least one pair of redundant servers. For details and other pre-requisites, see "Prerequisites for Online Upgrade" on page

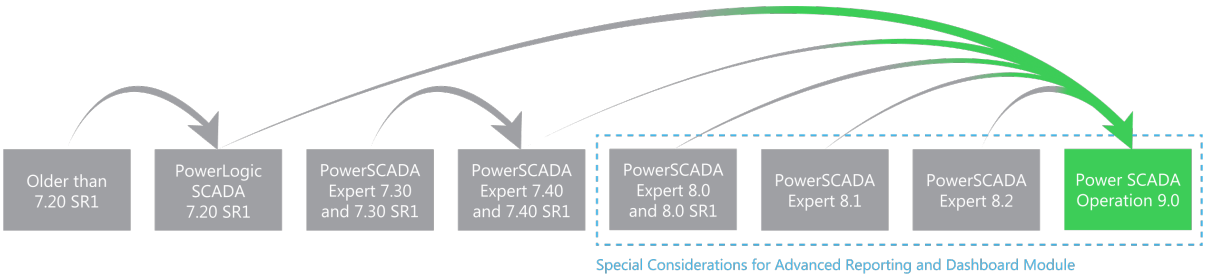
116. If the Offline Upgrade was earlier performed in a test environment, as noted above, the upgraded version 9.0 project will be migrated to production during the Online Upgrade process as the final step in the Upgrade Path (when Power SCADA Operation 9.0 is finally installed on the Production servers).

Upgrade Path

Upgrade path refers to the number of versions to which you need to upgrade to get from your current version to Power SCADA Operation 9.0. The number of necessary steps will depend on whether you do an offline or online upgrade.

If you plan to perform an "Offline Upgrade" on page 107, you can upgrade your project from as early a version as 7.20 SR1, directly into Power SCADA Operation 9.0.

If you plan to perform an "Online Upgrade" on page 115, in which runtime and historical data are migrated and upgraded, you need to follow an upgrade path that will depend on your starting version:



Advanced Reporting and Dashboards Module version 9.0 must be used with Power SCADA Operation 9.0, so an upgrade may be required for the Advanced Reporting and Dashboards software (Power Monitoring Expert). The same is true for versions 8.0, 8.1, and 8.2: the versions of Power SCADA Operation and Power Monitoring Expert must be the same.

- **Prior to v7.20 SR1** - If your starting version is prior to v7.20 SR1, upgrade to v7.20 SR1. Compile and run your project in order to restore and convert your historic alarm data.
- **v7.30 or v7.30 SR1** - If this is your starting version, you need to restore your project to v7.40. Compile and run your project in order to restore and convert your historic alarm data.
- **v7.40** - If this is your starting version, there is no intermediate version upgrade required. Upgrade directly to 9.0.
- **v8.0 or v8.0 SR1** - If this is your starting version, there is no intermediate version upgrade required. Upgrade directly to 9.0.
- **v8.1** - If this is your starting version, there is no intermediate version upgrade required. Upgrade directly to 9.0.
- **v8.2** - If this is your starting version, there is no intermediate version upgrade required. Upgrade directly to 9.0.

Optional Enhancements

- Integration of Diagnostics feature. See "Diagnostics Overview" on page 502 for details.
- Integration of notifications solution; in Power SCADA Operation 9.0 you can open notifications from the Alarms page. See "Notifications" on page 325 for details.
- Integration of new Citect 2018 features
- In PowerSCADA Expert 8.2 there are software features or modified functionalities that are different or new compared with those of the old version of PowerSCADA Expert you are upgrading.
- Integration of Advanced Reports and Dashboards. For information on using Advanced Reports and Dashboards see "About Advanced Reporting and Dashboards" on page 66.
- After v7.20, the dynamic one line animation engine and related genies are different, so updates may need to be made to a v7.20 project you are upgrading to ensure correct operation of the dynamic one line animation in the project.
- If the existing project uses the ES_StartAdvOneLine() function, instead use PLS_StartAdvOneLine available in all Power SCADA Operation versions since v7.30.
- It is recommended to modify persistent memory devices currently using the DISKXML driver, by updating them to use the IEC61850N driver. As shown below set the following properties:
  - Protocol: "IEC61850N"
  - Startup Mode: "Primary" or "StandbyWrite" if configuring a redundant instance of the device.
  - Memory: "TRUE"
  - Priority: "1" or "2" if configuring a redundant instance of the device.
  - Persist (extended field enabled by pressing "F2"): "TRUE"
  - Persist Period (extended field enabled by pressing "F2"): Default is 10 minutes (00:10:00) or set to a different value based on how frequently this memory device's data is cached to disk.

Offline Upgrade in Test Environment

It is strongly recommended to perform the Offline Upgrade steps in a test environment prior to completing the Offline or Online upgrades in the production environment. It is also strongly recommended to perform the Offline Upgrade before traveling to the Production site.

Completing the following activities in a test environment, before traveling to the Production site, will save time and effort:

- Use the Citect and Power SCADA Migration Tools to migrate the existing project configuration to the next product version in the Upgrade Path and finally to Power SCADA Operation 9.0.
- Fix any compile errors and warnings that appear during project upgrade and migration.
- Validate the merge of the existing Citect.INI file into the upgraded version Citect.INI

- Discovery of hard-to-find files listed in Offline Upgrade > Backup your current project and relevant files.

To perform the Offline Upgrade steps in a test environment:

1. Complete all steps of the Offline Upgrade:
   i. In step 3 (Install next version) be sure to install Power SCADA Operation 9.0 and skip step 13 (Install Power SCADA Operation 9.0). This action upgrades the current version of the project directly to version 9.0.
   ii. Skip step 11 (Restore historical data files). In this step runtime data and historical data from the existing system are restored, but it is only necessary to do this later when completing the Offline or Online Upgrade procedures in the Production system (while located at the Production site).
   iii. By completing this step the project should now be upgraded to Power SCADA Operation 9.0.
2. Address any project compile issues.
3. Test the project's functionality, verifying that key features of the customer solution still function as expected.
4. (Optional) Add "Optional Enhancements" on page 106 from Power SCADA Operation 9.0 to the project, recompile and test.
5. Backup the upgraded 9.0 project, upgraded include projects, sub-directories, and configuration files.

Offline Upgrade

This is the basic upgrade process and you will need to perform these steps even if choose to use the Online Upgrade method.

Offline Upgrade to Power SCADA Operation 9.0 consists of the following steps:

1. Backup your current project and relevant files.

   Perform a backup of your project and other relevant files from all servers in the system. For the upgrade to complete smoothly without errors, you need to back up a number of files/folders from your system other than your project files. The number of files you need to back up depends on your system configuration. For more information about performing a backup, refer to the Backing Up a Project section in the online help of your current version.

   You may also need to inspect any include projects for some of the files listed below.

   The following files need to be backed up:

File	Description
Project backup (.ctz file)	This is the main file to back up. For information about backing up a project, refer to your current version's online help. You need to have the **Save Included Projects**, **Save sub-directories** and **Save configuration files** options selected in the Backup dialog.
Citect.ini	This file is located in the config folder. Gather these INI files from client machines in addition to the server machines in the system.
Data directory	This file is found on the path [CtEdit]Data
ALMSAV.DAT and ALMINDEXSAVE.DAT (For v7.20) **OR** Alarm Database (for v7.30SR1, v7.40, v7.40SR1, v8.0, v8.0SR1, v8.1 and v8.2)	<ProjectName>_<ClusterName>_ALMSAV.DAT and <ProjectName>_<ClusterName>_ALMINDEXSAVE.DAT. These files contain alarm configuration data as well as runtime data. Their path is defined in the Citect.INI file. The default path is same as the data directory path. The Alarm Database is located in the Data directory: **[Data]\<Project Name>\<ClusterName.AlarmServerName>**. For each alarm server you have in your system, a corresponding Alarm Database will exist. You need to backup all alarm databases.
Trend files: *.HST and *.00X	The path and names of these files are defined on the trend tag itself, and created in the Data directory defined in [CtEdit]Data. The files will be named after the trend name and number of files. For example, if the trend name is CPU, file names will be CPU.HST, CPU.001, CPU.002 and so on. There may also be archived trend history files that do not exist in the [CtEdit]Data directory. If you are required to maintain this trend history, these files will need to be backed up and incorporated into the upgraded system. This is especially true if the upgraded system is being installed on a new physical server machine rather than on the existing production machine. Be sure to test the loading of these trend history files in the upgraded system.
Report Files	These files contain the code that is executed on your reports, and are located in the [CtEdit]User\<Project Name> folder.

File	Description
Custom ActiveX Controls (.OCX)	Power SCADA Operation includes a number of ActiveX controls, which will be available with the 9.0 installation, but need to take a back up of your custom ActiveX controls. Check your ActiveX.dbf file in the [CtEdit]User\<Project Name> folder. This file contains a list of the ActiveX controls in your project and their GUID. Using the GUID, find the path of an ActiveX control using the Windows Registry key KEY_LOCAL_MACHINE\SOFTWARE\Classes\CLSID\"GUID"\InProcServer32\. The default value for this key is a path to the .DLL or .OCX file you need to back up. Check for these files in all include projects, as well.
Process Analyst files	Backup the main <Project Folder>\Analyst Views and <Project Folder>\Dictionary folders.
Device logs	These files contain any logging (alarm logs, report logs) you have configured in your project. You will find their location in the Devices dialog. Refer to your online help for more information. Check for these files in all include projects, as well.
Additional Files	Check your Citect.ini file or use the **Setup Editor | Paths** section as it could contain runtime files used by custom code in the project. It is also recommended to search "C:\" (or other volumes where multiple hard disks are installed) in the Power SCADA Studio > Find and Replace tool. These search results will display any paths in use by all project components.
Driver Hotfixes	If you are aware of any driver hotfix in your system, backup this driver DLL which is located in the Bin directory where Power SCADA Operation is installed. **Note:** The fixes contained in this hotfix may have been included in the drivers which ship with Power SCADA Operation 9.0. See the Power SCADA Operation Exchange Community for additional driver downloads or Citect Driver Web for additional driver downloads.

2. Upgrade your licenses.

   In order to do this, you will either need to have a valid support agreement or you will need to purchase a license upgrade. Upgrade your key or soft license using our online license generator. You can also check the support status at the same URL.

   If your license is out of support, contact your Schneider Electric account manager. If you are not sure who your account manager is, send an email to Orders.Software@schneider-electric.com with your license and site ID details. For more information about licensing in Power SCADA Operation 9.0, refer to "Licensing" on page 141.

3. Uninstall your current version and install the next version defined on your upgrade path

   If you need new hardware or need to upgrade to a new operating system to run Power SCADA Operation, it is unnecessary to uninstall.

   If upgrading in a production environment as part of an Offline or Online upgrade process:

   Uninstall the current version of Power SCADA Operation completely and install the next version specified in your upgrade path.

   If this is step is being done in a test environment:

   It is unnecessary to install the next software version in the upgrade path. Upgrade directly by installing Power SCADA Operation 9.0 and any recent hotfixes available at the Power SCADA Exchange Community > Downloads page: https://exchangecommunity.schneider-electric.com/docs/DOC-17373

   Proceed with upgrading and migrating the project configuration for later use in the production environment.

4. Restore your project

   Restore your project. Select all included projects if available. **Note**: "PLS_Include" will be restored from the Power SCADA Operation 9.0 install.

5. Upgrade your project

   As a default, when you restore your project from a previous version, Power SCADA Operation will force an update, and you will get a warning message. Click **Yes** to proceed with project upgrade.

   If this message is not displayed, you can force an update of all projects by setting the **[CtEdit]Upgrade** INI parameter to 1 and restarting Power SCADA Operation. Once you restart, you will get a warning message. After clicking **Yes** all projects will be upgraded.

   Pack all projects in the Power SCADA Studio > Projects screen and Pack Libraries in Active and Included projects in the Graphics Builder > Tools menu.

6. Migrate your project

   The automatic project upgrade does not fully upgrade your projects, and needs to be followed by the Migration tool. The Citect and Power SCADA Operation Migration Tools are separate applications that must be run manually after the project upgrade has been executed, and adds computers from the existing topology. You may need to run the Citect Migration tool separately for other components. Refer to the online help for more information about running the Citect Migration tool.

   Run the Citect and Power SCADA Operation Migration Tools.

   Ensure all IO devices in the project have been assigned Equipment names

7. Merge your .INI file

   In addition to the INI settings below, be sure to identify other INI settings that may be custom and necessary for the proper operation of the upgraded software project. The Computer Setup Editor tool is very useful for comparing the old and new INI files. Select "Compare INI Files" from the Computer Setup Editor > Tools menu.

When upgrading a standby server, first merge the standby server's existing .INI into the upgraded version .INI. Then compare this result to the upgraded, merged .INI from the primary server to ensure they are consistent; the two files should have consistent [Alarm], [Trend], [Report] and driver parameters. Other parameters that include <Server>, <Cluster> or <Device> names will have different parameter names but similar values.

If you have defined the following parameters in your Citect.INI file, merge them into the new version's INI file.

Parameter	Description
[General] **TagStartDigit=1**	Without this parameter, you will encounter the 'Tag not defined' compiler error. Setting this to 1 allows you to define tag names that begin with a number or a symbol.
[General] **CheckAddressBoundary=0**	Without this parameter, you could encounter the 'Bad Raw Data' or other tag address related errors. Setting this to 0 allows defining variable tags of the same data type in odd or even addresses. When this parameter is set to 1 all variable tags from the same data type need to be defined on odd OR even addresses.
[General] **ClusterReplication=1**	Without this parameter, compile will fail in a multi-cluster system. Setting this parameter to 1 will enable tag/tag reference replication in a multi-cluster system.
[CtDraw.RSC] **ListSystemPage=1**	This allows you to open popup pages from Graphics Builder.
[CtDraw.RSC] **AllowEditSuperGeniePage=1**	This allows you to edit super genie pages from Graphics Builder.
[CtEdit] **DbFiles=100**	This allows you to set the maximum number of .DBF files that can be open simultaneously. Allowable values are between 50 to 32767 with the default set to 100. Increase the value of this parameter for larger projects.

Merge any driver parameters from you old .INI file as they will most likely be necessary to interface with your I/O network. For a list of changes to .INI parameters, see "Upgrading Reference" on page 525.

8. Compile your project

After upgrading your project and running the Migration tool, compile your project (See the product help > Compile topic for details) to ascertain that runtime functionality works as expected. It is very likely that you may encounter errors when you compile your project. One of the most common sources of errors when upgrading is Cicode functions. This is because functions may have changed, deprecated or simply because the compiler code has been updated to prevent runtime errors.

After fixing any errors, do the following:

a. Use the Power SCADA Studio > Options menu to un-check "Incremental Compile".

b. Then Pack the project from the Power SCADA Studio > Projects screen.

c. Update Pages and Pack Libraries in the Active/Include projects from the Graphics Builder.

d. Compile the project again.

Refer to your online help for instructions on compiling your project.

9. Run the Setup Wizard

Before running your project, run the Setup Wizard (known as Computer Setup Wizard in previous versions) to configure the Runtime Manager and other settings that are relevant to the runtime process. The Setup Wizard will automatically determine the role of your computer based on the network addresses defined in your project. After finishing the Setup Wizard, restore your historic data and other files, and run your project.

Be sure to enter the Server Password obtained or created before the upgrade (see Server Authentication Password) on the Server Authentication screen of the wizard.

10. Restore runtime files

After compiling your project, place the files necessary for runtime in the correct directories. Refer to point 1 in this topic for the list of files you need to place in the corresponding directories as defined in your Citect.INI file and project configuration. If performing the Online upgrade or upgrading in a test environment, it is unnecessary to restore alarm database, alarm history and trend files. These files will be restored manually later in the production environment or automatically in an Online Upgrade through Primary-Standby server synchronization.

11. Restore historical data files (necessary if upgrading in the production environment)

Restore the historical data files before running your upgraded projects. It is not required to restore these files when performing the Online upgrade or if upgrading the project in a test environment. During an Online upgrade these files will be restored automatically through Primary-Standby server synchronization.

NOTE: Consideration should be given to the size of the alarm and trend files. Automatic Primary-Standby server synchronization can take a very long time, depending on the size of these files. If there are thousands of files or gigabytes of data it will be best to copy the existing backup files from the standby directly to the primary server using Windows File Explorer or a tool like Easy RoboCopy (Availbale tools)

Alarms (v7.20 SR1 and earlier)

Before you can upgrade to Power SCADA Operation 9.0, perform the following steps to convert your <Project Name>_<Cluster Name>_ALMSAV.DAT and <Project Name>_<Cluster Name>_ALMINDEXSAVE.DAT files to a format that can be read by the new alarm server architecture introduced in v7.30:

Make sure that the [Alarm]SavePrimary parameter points to the directory in which you have placed your backed-up ALMSAV.DAT and ALMINDEXSAVE.DAT

Alarms (v7.30SR1, v7.40, v7.40 SR1, v8.0, v8.0 SR1 and v8.1)

Convert your Alarm Database in the Data directory with the following steps:

a. Make sure to place your backed-up Alarm Database in the directory defined by the [CtEdit]Data parameter.

b. Before starting runtime, confirm that the directory [Alarm]SavePrimary does NOT contain ANY ALMSAV.DAT nor ALMINDEXSAVE.DAT files.

Trends

Follow these steps to convert the files:

a. Create the same file hierarchy on the new system.

b. Place the files in the same folders.

c. If you want to change the folder location, or you cannot replicate the same file hierarchy, use the trend renaming tool available at the Support Site.

12. Run your project

Run your project to check that the functionality works as intended:

- Check any Cicode that you needed to modify in order to compile your project.
- Test communications to your I/O devices, alarm triggering and trend capture

13. Install Power SCADA Operation 9.0

After you have completed all the steps in your "Upgrade Path" on page 105, install Power SCADA Operation 9.0 and repeat steps 4 through 12. Refer to the Installation section. Be sure to also install any recent hotfixes available at the Power SCADA Exchange Community > Downloads page: https://exchangecommunity.schneider-electric.com/docs/DOC-17373

14. Update settings in the Application Config Utility.

In particular, the authentication settings in the Citect Data Platform and settings of the One Line Engine screens need to be completed. Ensure that your redundancy parameters are set for the one line engine in a redundant system

15. Implement "Optional Enhancements" on page 106 available in Power SCADA Operation 9.0.

Add optional functionality or enhancements that are now available in Power SCADA Operation 9.0 that may not have been available in the older version of the software.

Migrating to Production

Review the following information to complete your Offline Upgrade process, and apply the changes to your production system.

Testing Considerations

After the upgrade and configuration changes to the project are complete, it is recommended to perform system testing of the new project version. This is to check that functionally and operation behaves as expected before applying the new project to the production environment.

Licensing

When changing to use a newer product version, the hardware/software key may need to be updated. To prepare the system, it is recommended to update the production machine keys before the project is updated on the production machines as the updated key will still license the previous version. The hardware key is a physical key that plugs into either the parallel port or USB port of your computer. The key update utility can be run from the Help menu of the product Explorer application. To upgrade the key a new authorization code is required which can be created by using the AuthCode Generator.

Prepare Configuration [INI] Files

Before beginning any changes to the production computers, it is recommended that you backup the configuration [INI] files for each machine as they may be required for reference.

The current configuration file can be used with the new product version after the path parameters have been updated to the new version file locations. Refer to the setup of the development environment section of the specific version for further parameter information.

The Setup Editor and Setup Wizard can be used to finalize the configuration of the computer setup.

Server Addresses

During a migration with an existing system, it may be useful to use a new set of IP addresses and computer names for the new version. This is typically done when there is a need to provide isolation between the system project versions to allow the two systems to individually co-exist on the network for a period of time.When isolated, the systems will be independent and not cross communicate or synchronize between the existing and new versions. This type of upgrade would have the new version start with a snapshot of the historical data from the previous system and then run in parallel.

Communication Drivers

The project may be using specialty drivers and if so, it is recommended to backup the driver files located in the product ‘bin’ directory. Existing specialty drivers that are used may be required to be installed for the new version. The driver web can be checked for availability and compatibility with the new version at the DriverWeb.

Specialty Software

The project may be using specialty software to provide certain system functionality. These applications may be required to be updated or re-installed during the upgrade process and considered in the context of the upgrade.

Format File

The project may be using custom configuration forms in the product. This configuration is located in the FRM file which may be required in the new installation. For more information, see KB1579

Trend and Alarm Data

A project upgrade may also require the trend and alarm data to be updated based on the new product features. It is recommended to keep a backup of the existing production trend data files and the alarm save data file from the original

Once the data files have been upgraded, the updated data files may not be compatible with the previous version.

It is not recommended to change the directory path of the trend data files during the project upgrade as this may affect the trend operation. The default data directory may be changed between product versions and may need to be considered in the context of the install and upgrade with regards to the trend file location.

Troubleshooting Offline Upgrade

This section lists common issues you might encounter during your Offline Upgrade, which may be compiling errors or any other pre-runtime issues.

Not able to upgrade license key

1. Make sure you have correctly installed the latest versions of CiUSafe and Sentinel Driver.
2. Make sure the Authorization code matches the Key you are trying to upgrade. If you still cannot upgrade your license, check KB article Q3672 for more information on the error codes.

Compiler errors and warnings not related to deprecated functions

As Power SCADA Operation evolves, the compiler feature becomes more strict in order to ensure project quality and runtime success. The fact that you are getting compiling errors that were not appearing before is because of stricter compilation, which will result in more predictable and stable runtime. Refer to the error code in the error message to resolve any errors and warnings. You can search the online help using the error code for more information about a specific error code.

Online Upgrade

NOTICE
LOSS OF DATA Backup your project and other relevant historical data files from all servers in the system. **Failure to follow these instructions can result in a loss of data.**

Carefully follow this guide. See also Backup your current project and relevant files for details on which files to back up.

An online upgrade takes advantage of Power SCADA Operation's native server redundancy to minimize or avoid loss of data or downtime on your production system, allowing for one server to take ownership while the other is being upgraded. An online upgrade is the only way to avoid loss of data where you perform an upgrade in parallel. This is the process in which the two SCADA systems (the old version and the newer one) are running side-by-side. The old version is decommissioned after the new version has been fully tested and validated.

Similar to the "Offline Upgrade" on page 107, you will need to follow the "Upgrade Path" on page 105, and repeat the process as many times as the number of steps in your upgrade path.

Validate Hardware and Software Requirements for Power SCADA Operation 9.0

Validate that the server hardware running the current Power SCADA Operation project on both the primary and standby servers meets the Power SCADA Operation 9.0 minimum requirements listed in "Hardware and software requirements" on page 51. Additionally, because this document is being prepared for more complex systems, the CPU and memory allocated to the machine should be validated against the project design (# of I/O Servers, tags per I/O Server, and so on) in "Hardware and software requirements" on page 51.

Depending upon your current version of Power SCADA Operation, refer to Citect Help.

Prerequisites for Online Upgrade

As previously mentioned, an online upgrade lets you avoid downtime and loss of data. It is important that you take into consideration the complexity and size of your project when planning for this upgrade.

Review the following prerequisites before you start an online upgrade:

1. "Before Upgrading" on page 103 and before traveling to the production site
2. Required Files:
   - Power SCADA Operation 9.0 ISO
   - From the primary and standby servers and client machines, the files as listed in the Offline Upgrade > Backup your current project and relevant files section.
   - Backup the alarm and trend database files from the standby server before syncing an upgraded primary to the standby still running an older software version, in case any unforeseen problems arise and modifications are unintentionally made to the databases on the standby server.
   - The Power SCADA Operation 9.0 project files that have been upgraded in a test environment.
3. At least one pair of redundant servers: This is to upgrade one server at the time while the redundant server assumes primary operation, avoiding downtime and loss of data.
4. Server Authentication Password: In order for the upgraded primary server to synchronize with the standby server the Server Password from the Server Authentication screen of the Computer Setup Wizard must be known. If it is not known, it must be reset to a known password on both servers using the Computer Setup Wizard before beginning the online upgrade process.
5. Upgraded project: Check that your project runs and works properly on Power SCADA Operation 9.0 before migrating to production and starting the online upgrade. If your project is complex or if you are upgrading from a version earlier than v7.20 SR1, it is recommended that you have a test environment as the offline upgrade could be complex and could involve a long server downtime if done on your production system.

6. Restore runtime files: Check that you have restored the necessary files for runtime onto the appropriate directories to avoid any disturbances on the upgraded live system.
7. Capture data files: To allow historic data to be restored into the new version, you need to assess and move data files to the required location during the upgrade process. This is described in detail in the online upgrade steps in the relevant sections.
8. Computer Setup Wizard Screens: It can be helpful to make screen capture images of the Computer Setup Wizard screens from servers the existing system. This will help later in the upgrade process if a mistake is made or if you would like to validate the settings when running through the Computer Setup Wizard in 9.0.
9. Configure your running system for Online upgrade: To allow this process to be as smooth as possible, we recommend leveraging of your current redundant system and adding the following Citect.INI parameters before the online upgrade
   - **[LAN] EarliestLegacyVersion**: Use values for this parameter according to the table below. For example, use 7200 for upgrades from v7.20, v7.20 SR1 and v7.30 SR1. This will allow your upgraded servers to accept connections from the older version.

Product Version	EarliestLegacyVersion
7.20	7200
7.20 SR1	7200
7.30 SR1	7300
7.40	7400
7.40 SR1	7400
8.0	7400
8.0 SR1	7500
8.1	7500
8.2	8000

   - [Alarm]EnableStateLogging: Set this parameter to 1 to allow logging the alarm synchronization messages into the syslog.
   - [Alarm.<ClusterName>.<AlarmServerName>]ArchiveAfter: This parameter is specific for an upgrade to v2015. If this parameter is not set to Citect 2015, the alarm server will not start up. This is configured for each Alarm Server instance. When configuring this parameter you need to decide what time period of data you wish to maintain during upgrade. For example, if you set this parameter to 1 week, it means that during the upgrade process you will lose any summary data that is older than 1 week. If you don't want to lose any data, you need to set this parameter to the earliest data in your summary (v7.20) or SOE (v7.30 and v7.40)
   - [Debug] Kernel = 1 (optional): Enable this to allow for monitoring the kernel during the upgrade.

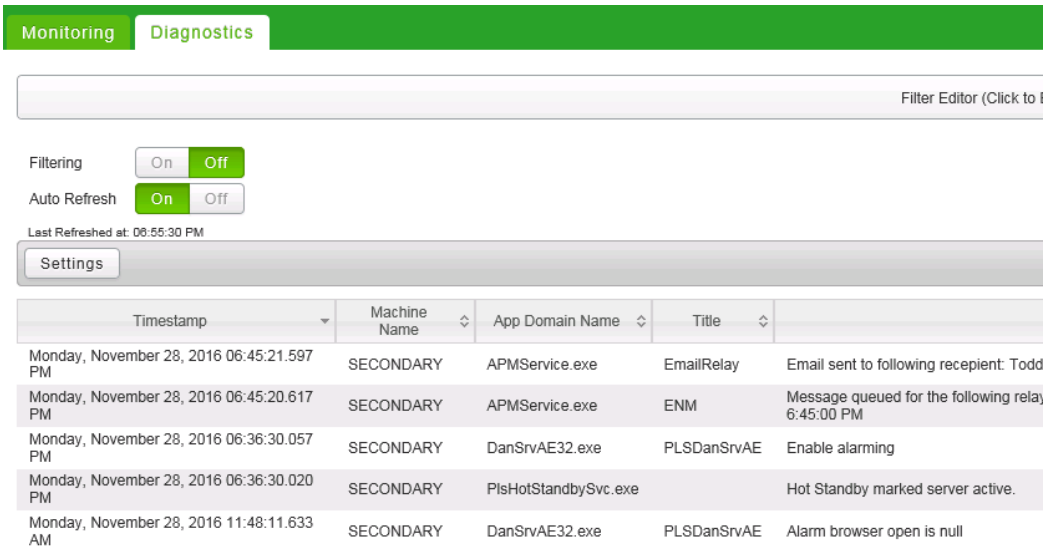
10. Disabled Alarms: If any alarms have been disabled in the project runtime capture screen shots of the Disabled Alarms page in the runtime. If there are problems with the Online upgrade it will be necessary to manually disable those alarms to put the system back in its original state.
11. Disabled IO Devices: If any IO devices have been disabled in the project runtime be sure to double check the [DisableIO]<Device name> or [DisableIO]<Server name> parameters to ensure the devices remain disabled after the upgrade.

Upgrading from v8.2

To upgrade from v8.2:

1. Check that you have added the following parameters on the .INI file to all your server nodes before you start the online upgrade:

   [LAN]EarliestLegacyVersion = 8000.
2. Restart the servers after adding the parameter for the changes to take effect.
3. On the primary server:
   a. Before stopping the primary runtime, validate dynamic one line pages, device communications, and Event Notification Module (ENM) operation if installed
   b. Shut down runtime on the primary server.
   c. Validate one line pages, device communications, and Event Notification operation on the standby server. You should see a messages similar to this in the ENM diagnostics tab (http://localhost:85) on the standby when it becomes active:



   d. If the standby server has not assumed ENM operations the primary server will have to be brought back online. You will have to troubleshoot the system redundancy.
4. Upgrade the primary server according to the "Offline Upgrade" on page 107.
5. Place the backed-up Alarm database in the [CtEdit]Data directory. This will allow a quicker synchronization of alarm servers.
6. Restart the primary server. It is now upgraded.

7. Check all functionality on the new Power SCADA Operation9.0 primary server:
   - Check the dynamic one-line operation, device communications, pop-up graphics, the alarm log, and any other critical functionality. Validate that the ENM emails are being sent through the ENM standby server (Diagnostics tab, "Email Sent..." messages). If possible, validate the emails from other alarms.
8. Power SCADA Operation 9.0 server will synchronize its alarm database with the running older version standby server.

   Wait for the synchronization process to finish; this will depend upon the size of your alarm database. The synchronization information is available from the main kernel window of the Alarm Process as well as the syslog.

   Check the status of the alarm server synchronization using the Alarm Server Kernel, on the Main Window:
   - When the Alarm Servers synchronization starts you should see the following message: **Alarm: Peer update request sent.**
   - Then you should see a number of messages with Update packets (number is dependent on your Alarm historic events and configuration). **Alarm: Update packet XXXX received.**
   - Finally, the following messages will indicate that the synchronization has been finalized successfully: **Alarm: Database objects state synchronization completed.** **Alarm: Database is initialized, preparing to Start the Alarm Engine.** **Alarm: Starting Alarm Engine** **Alarm: Server startup complete.**

   Trends from the Standby server will fill the time period the Primary server was offline. Monitor the Kernel pages `PAGE QUEUE TrnRdn.GapFillDelayQue` and `PAGE QUEUE TrnRdn.GapFillSentQue`. Wait for the queues to be empty before shutting down and upgrading the standby server, if possible. See Citect Knowledgebase Q3723 article: https://www.citect.schneider-electric.com/scada/citectscada/find-answers/knowledge-base?view=kbarticle&id=3723
9. On the newly-upgraded primary server, migrate the ENM configuration to Power SCADA Operation notifications. See Migrating notifications for more information.
10. Decommission ENM on the Primary server by uninstalling ENM 8.3.3 through the Control Panel > Programs and Features. Stop and uninstall SQL Server if it is no longer needed by other applications.
11. "Verify notifications" on page 140 functionality on the Primary Server.
12. Upgrade your client nodes one by one. On each client complete the steps 1 through 3 and 7 of the "Offline Upgrade" on page 107. In step 2, only the citect.ini file is relevant for client machines. When the newly upgraded v2016 server assumes the primary server role it will migrate the entire alarm database to the new format, and you should now be able to see Alarm Summary data on all migrated Clients.

It is helpful to leave one client on the existing version of the software in case there is anything not functioning properly in the new version. This is also helpful in order to verify if anything was negatively affected by the upgrade versus having been non-functional prior to the upgrade. Once both servers have been upgraded, these clients will need to be upgraded as well.

13. Shut down the standby server and confirm operation of the new Power SCADA Operation9.0 primary server. Validate one lines, device communications, and event notification operation on the primary server.
14. Upgrade Power SCADA Operation on the standby server according to the "Offline Upgrade" on page 107.
15. Now that the standby server is upgraded, restart it and check system functionality:
    a. Check for hardware alarms when it is connected to the primary server.
    b. Check dynamic one-line operation, device communications, popups, alarm log, etc. Validate that the heartbeat notifications are being sent from the primary server's event notifications system. If possible, validate emails from other alarms as well.
    c. If there are issues with the advanced one-line displays, begin troubleshooting with the *AdvOneLineStatusLog*, found in your project folder.
16. Restore and check event notifications on the Standby server:

    On the Primary Server launch the event notification settings and save the settings. Accept the prompt to automatically synchronize the configuration to the Standby Alarm Server. See Creating Notifications.

    "Verify notifications" on page 140 functionality on the Standby Server.
17. Check functionality of the system as a whole. It is a good idea to check the log files in the [Logs] folder on both servers. There may be errors about deprecated parameters being used, invalid file paths, logins from clients that weren't upgraded, untrusted connections (clients/servers with different Server Passwords), or other errors.
18. Finally, test redundancy by switching off the primary server and checking that the standby server takes over Event Notification and Power SCADA clients all switch over.
19. On both servers remove upgrade-related parameters that were set in "Prerequisites for Online Upgrade" on page 116 and parameters noted Troubleshooting > Remove Upgrade Parameters.

Special Considerations

Alarm Save Files

When doing an online upgrade from v8.0 SR1 or v8.1 to v9.0 check that any pre-7.20 Alarm Save files are removed from the v9.0 project folders (e.g. <project_cluster>_ALMSAVE.DAT and <project_cluster>_ALMINDEXSAVE.DAT).

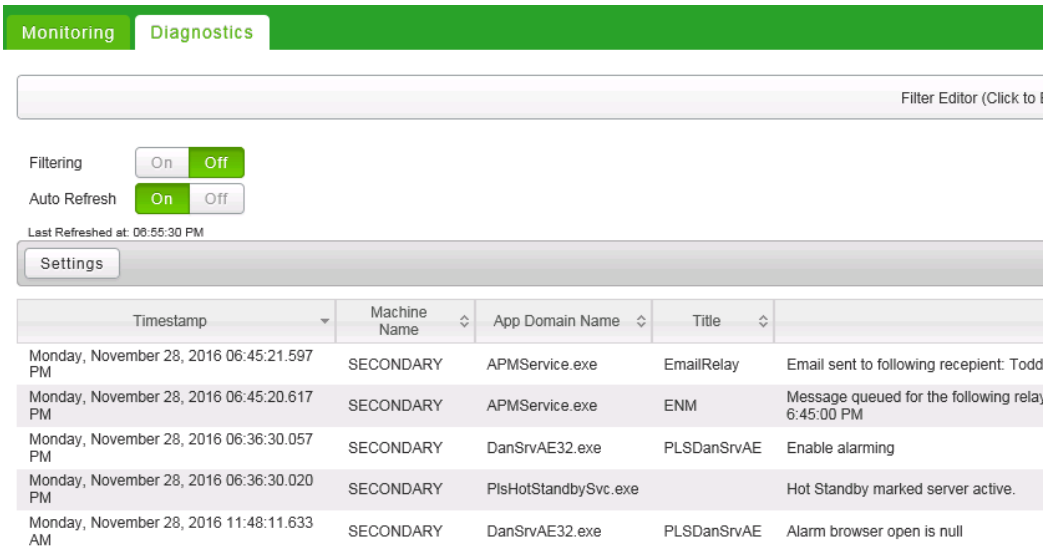
Upgrading from v8.1 and v8.0 SR1

To upgrade from v8.1 and v8.0 SR1:

1. Check that you have added the following parameters on the .INI file to all your server nodes before you start the online upgrade:

   [LAN]EarliestLegacyVersion = 7500.

2. Restart the servers after adding the parameter for the changes to take effect.
3. On the primary server:
   a. Before stopping the primary runtime, validate dynamic one line pages, device communications, and Event Notification Module (ENM) operation if installed
   b. Shut down runtime on the primary server.
   c. Validate one line pages, device communications, and Event Notification operation on the standby server. You should see a messages similar to this in the ENM diagnostics tab (http://localhost:85) on the standby when it becomes active:



   d. If the standby server has not assumed ENM operations the primary server will have to be brought back online. You will have to troubleshoot the system redundancy.
4. Upgrade the primary server according to the "Offline Upgrade" on page 107.
5. Place the backed-up Alarm database in the [CtEdit]Data directory. This will allow a quicker synchronization of alarm servers.
6. Restart the primary server. It is now upgraded.
7. Check all functionality on the new Power SCADA Operation9.0 primary server:
   - Check the dynamic one-line operation, device communications, pop-up graphics, the alarm log, and any other critical functionality. Validate that the ENM emails are being sent through the ENM standby server (Diagnostics tab, "Email Sent..." messages). If possible, validate the emails from other alarms.
8. Power SCADA Operation 9.0 server will synchronize its alarm database with the running older version standby server.

   Wait for the synchronization process to finish; this will depend upon the size of your alarm database. The synchronization information is available from the main kernel window of the Alarm Process as well as the syslog.

Check the status of the alarm server synchronization using the Alarm Server Kernel, on the Main Window:

- When the Alarm Servers synchronization starts you should see the following message:
  Alarm: Peer update request sent.
- Then you should see a number of messages with Update packets (number is dependent on your Alarm historic events and configuration).
  Alarm: Update packet XXXX received.
- Finally, the following messages will indicate that the synchronization has been finalized successfully:
  Alarm: Database objects state synchronization completed.
  Alarm: Database is initialized, preparing to Start the Alarm Engine.
  Alarm: Starting Alarm Engine
  Alarm: Server startup complete.

Trends from the Standby server will fill the time period the Primary server was offline. Monitor the Kernel pages `PAGE QUEUE TrnRdn.GapFillDelayQue` and `PAGE QUEUE TrnRdn.GapFillSentQue`. Wait for the queues to be empty before shutting down and upgrading the standby server, if possible. See Citect Knowledgebase Q3723 article: https://www.citect.schneider-electric.com/scada/citectscada/find-answers/knowledge-base?view=kbarticle&id=3723

9. Upgrade ENM to version 8.3.3 - Uninstall the current version of ENM through the Control Panel > Programs and Features. Install ENM 8.3.3 by running the install executable.
10. On the newly-upgraded primary server, migrate the ENM configuration to Power SCADA Operation notifications. See Migrating notifications for more information.
11. Decommission ENM on the Primary server by uninstalling ENM 8.3.3 through the Control Panel > Programs and Features. Stop and uninstall SQL Server if it is no longer needed by other applications.
12. "Verify notifications" on page 140
13. Upgrade your client nodes one by one. On each client complete the steps 1 through 3 and 7 of the "Offline Upgrade" on page 107. In step 2, only the citect.ini file is relevant for client machines. When the newly upgraded v2016 server assumes the primary server role it will migrate the entire alarm database to the new format, and you should now be able to see Alarm Summary data on all migrated Clients.

    It is helpful to leave one client on the existing version of the software in case there is anything not functioning properly in the new version. This is also helpful in order to verify if anything was negatively affected by the upgrade versus having been non-functional prior to the upgrade. Once both servers have been upgraded, these clients will need to be upgraded as well.
14. Shut down the standby server and confirm operation of the new Power SCADA Operation 9.0 primary server. Validate one lines, device communications, and event notification operation on the primary server.
15. Upgrade Power SCADA Operation on the standby server according to the "Offline Upgrade" on page 107.

16. Now that the standby server is upgraded, restart it and check system functionality:
    a. Check for hardware alarms when it is connected to the primary server.
    b. Check dynamic one-line operation, device communications, popups, alarm log, etc. Validate that the heartbeat notifications are being sent from the primary server's event notifications system. If possible, validate emails from other alarms as well.
    c. If there are issues with the advanced one-line displays, begin troubleshooting with the *AdvOneLineStatusLog*, found in your project folder.
17. Restore and check event notifications on the Standby server:

    On the Primary Server launch the event notification settings and save the settings. Accept the prompt to automatically synchronize the configuration to the Standby Alarm Server. See Creating Notifications.

    "Verify notifications" on page 140
18. Check functionality of the system as a whole. It is a good idea to check the log files in the [Logs] folder on both servers. There may be errors about deprecated parameters being used, invalid file paths, logins from clients that weren't upgraded, untrusted connections (clients/servers with different Server Passwords), or other errors.
19. Finally, test redundancy by switching off the primary server and checking that the standby server takes over Event Notification and Power SCADA clients all switch over.
20. On both servers remove upgrade-related parameters that were set in "Prerequisites for Online Upgrade" on page 116 and parameters noted Troubleshooting > Remove Upgrade Parameters.

Special Considerations

Alarm Save Files

When doing an online upgrade from v8.0 SR1 or v8.1 to v9.0 check that any pre-7.20 Alarm Save files are removed from the v9.0 project folders (e.g. <project_cluster>_ALMSAVE.DAT and <project_cluster>_ALMINDEXSAVE.DAT).

Upgrading from v7.30 SR1, v7.40, v7.40 SR1 and v8.0

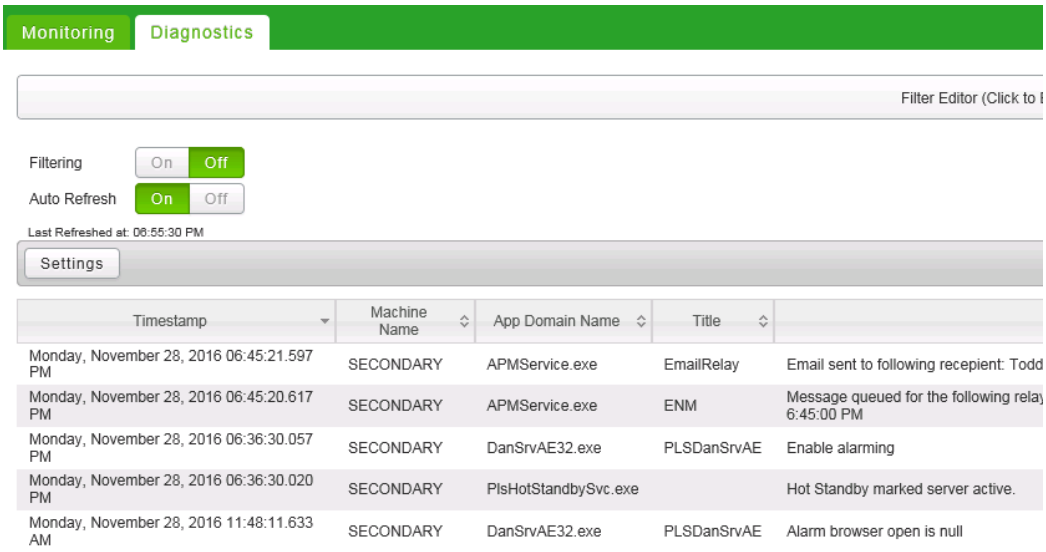
To upgrade from v7.30 SR1, 7.40, 7.40 SR1 and v8.0:

1. Check that you have added the following parameters on the .INI file to all your server nodes before you start the online upgrade:
2. Add the following parameter on the .INI file to all your server nodes before you start the online upgrade.

   For v7.30: [LAN]EarliestLegacyVersion = 7300.

   For the other versions: [LAN]EarliestLegacyVersion = 7400.
3. Restart the servers after adding the parameter for the changes to take effect.

4. On the primary server:
   a. Before stopping the primary runtime, validate dynamic one line pages, device communications, and Event Notification Module (ENM) operation if installed
   b. Shut down runtime on the primary server.
   c. Validate one line pages, device communications, and Event Notification operation on the standby server. You should see a messages similar to this in the ENM diagnostics tab (http://localhost:85) on the standby when it becomes active:



   d. If the standby server has not assumed ENM operations the primary server will have to be brought back online. You will have to troubleshoot the system redundancy.
5. Upgrade the primary server according to the "Offline Upgrade" on page 107.
6. Place the backed-up Alarm database in the [CtEdit]Data directory. This will allow a quicker synchronization of alarm servers.
7. Restart the primary server. It is now upgraded.
8. Check all functionality on the new Power SCADA Operation 9.0
   - Check the dynamic one-line operation, device communications, pop-up graphics, the alarm log, and any other critical functionality. Validate that the ENM emails are being sent through the ENM standby server (Diagnostics tab, "Email Sent..." messages). If possible, validate the emails from other alarms.
9. Power SCADA Operation 9.0 server will synchronize its alarm database with the running older version standby server.

   Wait for the synchronization process to finish; this will depend upon the size of your alarm database. The synchronization information is available from the main kernel window of the Alarm Process as well as the syslog.

   Check the status of the alarm server synchronization using the Alarm Server Kernel, on the Main Window:

   - When the Alarm Servers synchronization starts you should see the following message: **Alarm: Peer update request sent.**

- Then you should see a number of messages with Update packets (number is dependent on your Alarm historic events and configuration).
  Alarm: Update packet XXXX received.
- Finally, the following messages will indicate that the synchronization has been finalized successfully:
  Alarm: Database objects state synchronization completed.
  Alarm: Database is initialized, preparing to Start the Alarm Engine.
  Alarm: Starting Alarm Engine
  Alarm: Server startup complete.

Trends from the Standby server will fill the time period the Primary server was offline. Monitor the Kernel pages `PAGE QUEUE TrnRdn.GapFillDelayQue` and `PAGE QUEUE TrnRdn.GapFillSentQue`. Wait for the queues to be empty before shutting down and upgrading the standby server, if possible. See Citect Knowledgebase Q3723 article: https://www.citect.schneider-electric.com/scada/citectscada/find-answers/knowledge-base?view=kbarticle&id=3723

10. Upgrade ENM to version 8.3.3 - Uninstall the current version of ENM through the Control Panel > Programs and Features. Install ENM 8.3.3 by running the install executable.
11. On the newly-upgraded primary server, migrate the ENM configuration to Power SCADA Operation notifications. See Migrating notifications for more information.
12. Decommission ENM on the Primary server by uninstalling ENM 8.3.3 through the Control Panel > Programs and Features. Stop and uninstall SQL Server if it is no longer needed by other applications.
13. "Verify notifications" on page 140
14. Upgrade your client nodes one by one. On each client complete the steps 1 through 3 and 7 of the "Offline Upgrade" on page 107. In step 2, only the citect.ini file is relevant for client machines. When the newly upgraded v2018 server assumes the primary server role it will migrate the entire alarm database to the new format, and you should now be able to see Alarm Summary data on all migrated Clients.

    It is helpful to leave one client on the existing version of the software in case there is anything not functioning properly in the new version. This is also helpful in order to verify if anything was negatively affected by the upgrade versus having been non-functional prior to the upgrade. Once both servers have been upgraded, these clients will need to be upgraded as well.
15. After you are confident that synchronization of alarms, trends etc., is complete, and that your v9.0 clients are working correctly, shut down the standby server and confirm operation of the new Power SCADA Operation 9.0 primary server. Verify correct operation of dynamic one lines, device communications, and event notification operation on the primary server.
16. Now that the standby server is upgraded, restart it and check system functionality:
    a. Check for hardware alarms when it is connected to the primary server.
    b. Check dynamic one-line operation, device communications, popups, alarm log, etc. Validate that the heartbeat notifications are being sent from the primary server's event notifications system. If possible, validate emails from other alarms as well.

c. If there are issues with the advanced one-line displays, begin troubleshooting with the *AdvOneLineStatusLog*, found in your project folder.

17. Restore and check event notifications on the Standby server:

    On the Primary Server launch the event notification settings and save the settings. Accept the prompt to automatically synchronize the configuration to the Standby Alarm Server. See Creating Notifications.

    "Verify notifications" on page 140

18. Check functionality of the system as a whole. It is a good idea to check the log files in the [Logs] folder on both servers. There may be errors about deprecated parameters being used, invalid file paths, logins from clients that weren't upgraded, untrusted connections (clients/servers with different Server Passwords), or other errors.

19. Finally, test redundancy by switching off the primary server and checking that the standby server takes over Event Notification and Power SCADA clients all switch over.

20. On both servers remove upgrade-related parameters that were set in "Prerequisites for Online Upgrade" on page 116 and parameters noted Troubleshooting > Remove Upgrade Parameters.

Special Considerations

Alarm Summary

The v9.0 Summary feature will be disabled when connecting to a v7.30 server. You may still see summary records for active alarms.

Alarm Save Files

When doing an online upgrade from v7.30 to v9.0 check that any pre-7.20 Alarm Save files are removed from the v9.0 project folders (e.g. <project_cluster>_ALMSAVE.DAT and <project_cluster>_ALMINDEXSAVE.DAT)

Historical Alarm Events

Set the **[Alarm.<Cluster Name>.<Server Name>]ArchiveAfter** .INI parameter to a date prior to the earliest historical event date from which you want to migrate.

Upgrading from v7.20 and v7.20 SR1

When upgrading from v7.20, you will NOT need to restore the alarm data files (ALARMSAV.DAT and ALRMSAVEINDEX.DAT) under most circumstances. Power SCADA Operation 9.0 is equipped to read this information from the redundant v7.20 SR1 server that is still not upgraded.

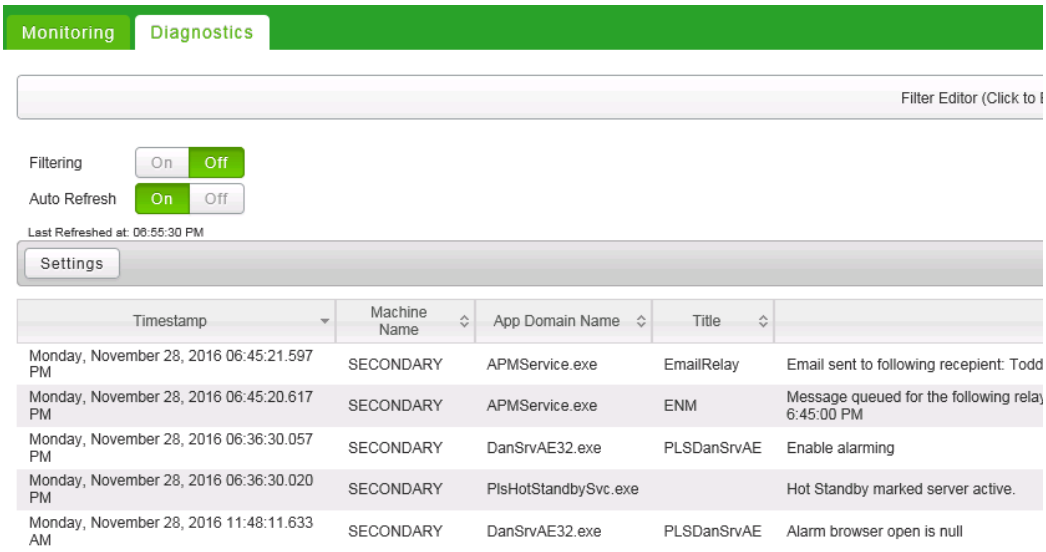
To upgrade from v7.20 or 7.20 SR1:

1. Add the following parameter on the .INI file to all your server nodes before you start the online upgrade.

   [LAN]EarliestLegacyVersion = 7200.

2. Restart the servers after adding the parameter for the changes to take effect.

3. On the primary server:
   a. Before stopping the primary runtime, validate dynamic one line pages, device communications, and Event Notification Module (ENM) operation if installed
   b. Shut down runtime on the primary server.
   c. Validate one line pages, device communications, and Event Notification operation on the standby server. You should see a messages similar to this in the ENM diagnostics tab (http://localhost:85) on the standby when it becomes active:



   d. If the standby server has not assumed ENM operations the primary server will have to be brought back online. You will have to troubleshoot the system redundancy.
4. Upgrade the primary server according to the "Offline Upgrade" on page 107
5. Restart the primary server. It is now upgraded.
6. Check all functionality on the new Power SCADA Operation 9.0 primary server:
   - Check the dynamic one-line operation, device communications, pop-up graphics, the alarm log, and any other critical functionality. Validate that the ENM emails are being sent through the ENM standby server (Diagnostics tab, "Email Sent..." messages). If possible, validate the emails from other alarms.
7. Now, the Power SCADA Operation 9.0 server will build the new alarm database, and will import the historic data from the Standby v7.20 server.
   - Trends from the Standby server will fill the time period the Primary server was offline. Monitor the Kernel pages `PAGE QUEUE TrnRdn.GapFillDelayQue` and `PAGE QUEUE TrnRdn.GapFillSentQue`. Wait for the queues to be empty before shutting down and upgrading the standby server, if possible. See Citect Knowledgebase Q3723 article: https://www.citect.schneider-electric.com/scada/citectscada/find-answers/knowledge-base?view=kbarticle&id=3723
8. Check the status of the alarm server synchronization using the Alarm Server Kernel, on the Main Window:
   - When the Alarm Servers synchronization starts you should see the following message: Alarm: Peer update request sent.

- Then you should see a number of messages with Update packets (number is dependent on your Alarm historic events and configuration).
  Alarm: Update packet XXXX received.
- Finally, the following messages will indicate that the synchronization has been finalized successfully:
  Alarm: Database objects state synchronization completed.
  Alarm: Database is initialized, preparing to Start the Alarm Engine.
  Alarm: Starting Alarm Engine
  Alarm: Server startup complete.

9. If you find that your Alarm Server synchronization is not completing successfully, place the ALARMSAV.DAT and ALRMSAVEINDEX.DAT on the [Alarm]SavePrimary directory.
10. Upgrade ENM to version 8.3.3 - Uninstall the current version of ENM through the Control Panel > Programs and Features. Install ENM 8.3.3 by running the install executable.
11. On the newly-upgraded primary server, migrate the ENM configuration to Power SCADA Operation notifications. See Migrating notifications for more information.
12. Decommission ENM on the Primary server by uninstalling ENM 8.3.3 through the Control Panel > Programs and Features. Stop and uninstall SQL Server if it is no longer needed by other applications.
13. "Verify notifications" on page 140
14. Upgrade your client nodes one by one. On each client complete the steps 1 through 3 of the "Offline Upgrade" on page 107. In step 2, only the citect.ini file is relevant for client machines.

    It is helpful to leave one client on the existing version of the software in case there is anything not functioning properly in the new version. This is also helpful in order to verify if anything was negatively affected by the upgrade versus having been non-functional prior to the upgrade. Once both servers have been upgraded, these clients will need to be upgraded as well.
15. After you are confident that synchronization of alarms, trends etc., is complete, and that your v9.0 clients are working correctly, shut down the standby server and confirm operation of the new Power SCADA Operation 9.0 primary server. Verify correct operation of dynamic one lines, device communications, and event notification operation on the primary server.
16. Now that the standby server is upgraded, restart it and check system functionality:
    a. Check for hardware alarms when it is connected to the primary server.
    b. Check dynamic one-line operation, device communications, popups, alarm log, etc. Validate that the heartbeat notifications are being sent from the primary server's event notifications system. If possible, validate emails from other alarms as well.
    c. If there are issues with the advanced one-line displays, begin troubleshooting with the *AdvOneLineStatusLog*, found in your project folder.
17. Restore and check event notifications on the Standby server:

    On the Primary Server launch the event notification settings and save the settings. Accept the prompt to automatically synchronize the configuration to the Standby Alarm Server. See Creating Notifications.

"Verify notifications" on page 140

18. Check functionality of the system as a whole. It is a good idea to check the log files in the [Logs] folder on both servers. There may be errors about deprecated parameters being used, invalid file paths, logins from clients that weren't upgraded, untrusted connections (clients/servers with different Server Passwords), or other errors.
19. Finally, test redundancy by switching off the primary server and checking that the standby server takes over Event Notification and PowerSCADA clients all switch over.
20. On both servers remove upgrade-related parameters that were set in "Prerequisites for Online Upgrade" on page 116 and parameters noted in Troubleshooting > Remove Upgrade Parameters.

Special Considerations

Custom Alarm Filtering

The AlarmSetQuery Cicode function was deprecated in v7.30. This means that if you are using custom alarm filtering code, you will most likely need to convert it.

Historical Alarm Events

Set the [Alarm.<Cluster Name>.<Server Name>]ArchiveAfter.INI parameter to a date prior to the earliest historical event date from which you want to migrate.

Alarm server synchronization during online upgrade

In the event that there is a disconnection or timeout during synchronization between the v9.0 and v7.20 alarm servers, follow these steps:

1. Shutdown your 9.0 server.
2. Delete the alarm database and re-start it.
3. Wait for the synchronization between servers to finish.

Also, you can increase the timeout using the [Alarm]StartTimeout .INI parameter. This will allow the v9.0 server to wait for connection from the v7.20 server.

If you find that the synchronization between the two servers is experiencing interruptions, delete the alarm database, and place your ALARMSAV.DAT and ALARMSAVINDEX.DAT in the [Alarm]SavePrimary directory and the v9.0 server will convert the data. However, we recommend always trying the peer synchronization first.

Changes during the upgrade process

Because of the differences between Power SCADA Operation 9.0 and v7.20, any actions that happen during the online upgrade process are subject to incompatibilities that are not reconcilable between versions. However, the scenarios are quite particular and should not have a great impact if any, on your SCADA system. Here is a list of such scenarios:

- UserLocation field: In Power SCADA Operation 9.0, a record of the **UserLocation**, that is the IP address, for alarm operations such as acknowledge is available. If an acknowledge occurs on the v7.20 server during the upgrade, the v9.0 server will be unable to record the UserLocation, which will be displayed as "0.0.0.0".

- Summary Comments during the upgrade: Comments that you add to an alarm summary record on the v7.20 server during the online upgrade will not be available in the upgraded version.

Troubleshooting Online Upgrade

This section lists common issues you might encounter during your Online Upgrade, which may be related to runtime issues and redundancy connectivity.

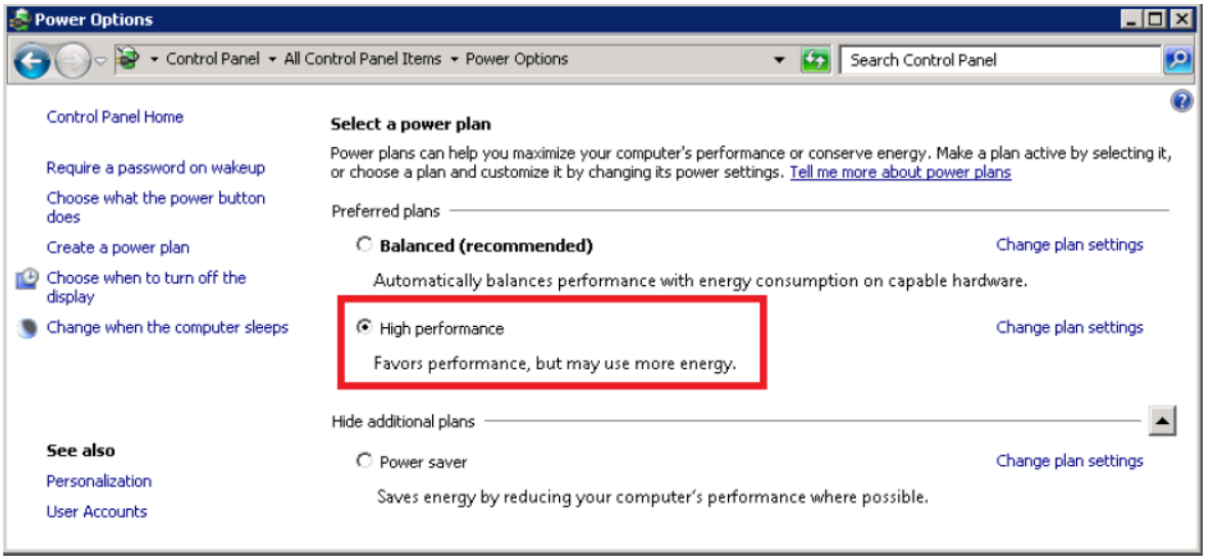
Redundant servers fail to communicate

I cannot make my redundant servers communicate and I keep getting the hardware alarm "Redundant Server not found"

1. Check that you have set your [LAN]EarliestLegacyVersion parameter correctly.
   - If upgrading from v7.20 use [LAN]EarliestLegacyVersion=7200.
   - If upgrading v8.0SR1 or v8.1 use [LAN]EarliestLegacyVersion=7500.
   - If upgrading v8.2 use [LAN]EarliestLegacyVersion=8000.
   - Check that you have run the Setup Wizard and set both servers to Networked mode.
2. Set the same server password on both servers in the Setup Wizard (see Configure Server Password in installed help).

My system is performing slowly even though Hardware and software requirements are met

Check your system's power options: Control Panel |All Control Panel Items | Power Options.



Remove Upgrade related parameters

After finalizing the upgrade process and confirming that runtime is fully functional, we recommend removing or updating the following .INI parameters. You will need to restart the servers after changing the parameters for the changes to take effect.

- [Alarm]SavePrimary: remove this parameter.
- [Alarm]SaveStandby: remove this parameter.

- [Debug]Kernel = 0: this is to enhance security and keep operators out of the kernel.
- [LAN]EarliestLegacyVersion: remove this parameter.

It is important to note that after removing the EarliestLegacyVersion parameter, the next time you change your user's passwords, you should change all the passwords on one server, and then roll out the updated project in the same order in which you conducted the online upgrade (primary server, clients and then standby server). Refer to KB article Q7865 for more information.

Upgrading Information

Refer to "Upgrading Reference" on page 525 for detailed information on the steps you may need to perform before and after the upgrade process. Review the information up to and including the version to which you are upgrading.

NOTE: Citect Help also contains detailed information on the Cicode functions and Citect INI settings changes with each release.

Migration Tools

The automatic update that occurs when you initially launch Power SCADA Operation 9.0 does not fully upgrade your projects, and needs to be followed by the use of the Migration Tools (if migrating from v7.x this is particularly noteworthy). The automatic update is a passive action which updates the database field definition for any database that has been changed between the two versions and copies new files that are necessary in 9.0.

The Migration Tools are two separate applications: one for Citect SCADA and one for Power SCADA Operation. Both have to be run manually after the automatic upgrade has been executed. You can do this after you have prepared the project for final migration.

⚠ WARNING

UPGRADE ALTERS COMMUNICATIONS CONFIGURATIONS

After upgrading, confirm and adjust the configuration of I/O devices in your project.

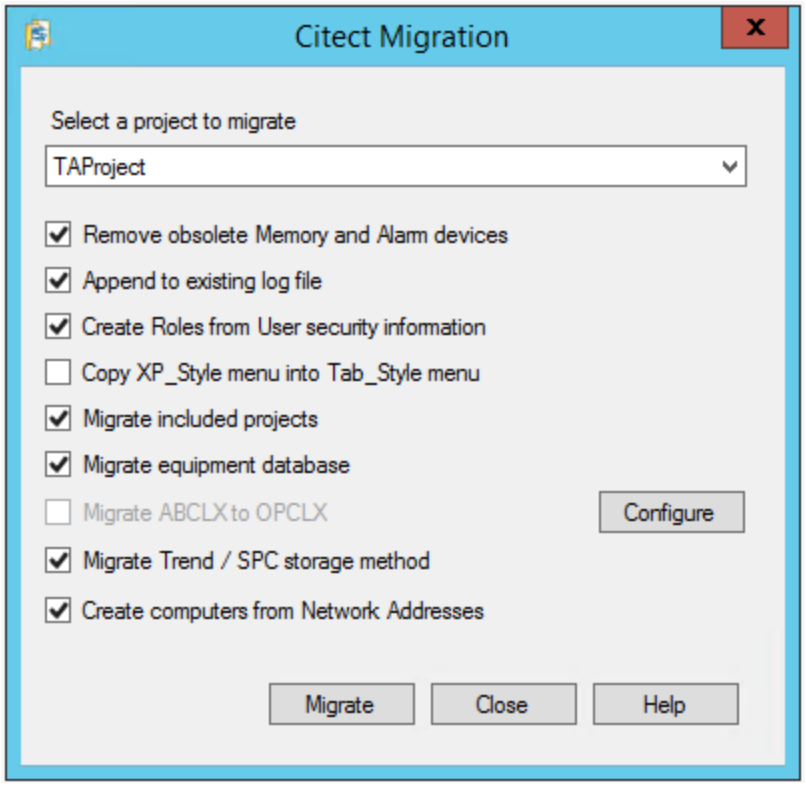
Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

Using the Citect Migration Tool

NOTE: Before you use the Citect Migration Tool, familiarize yourself with the process it performs, and the preparatory steps you need to carry out with your existing projects.

To run the Citect Migration Tool:

1. Backup the projects that you need to migrate.
2. In Power SCADA Studio, click **Project**, select **Home** | **Migration Tool** to display the Citect Migration Tool dialog.



3. Either accept the project displayed in the edit box, or browse for the project that you wish to upgrade.
4. Specify the changes you would like to implement during the migration process by selecting from the options described in the following table.

Option	Description
Remove obsolete Memory and Alarm devices	Select this check box if you wish to delete these types of devices after successful migration (see "Remove Obsolete Memory and Alarm Devices" on page 137). **Note:** Do not select this check box when you run the tool for the first time on a project that contains any included projects which are shared with more than one master project. If you want to delete obsolete devices under these circumstances, you can run the tool a second time using this option if the migration is successful after it is run the first time.
Append to existing log file	Use this option to append information about the migration process to the existing Migration Tool log file (located in Power SCADA Operation's User directory). If this option is not selected, a new log file will be created when migration is complete.

Option	Description
Create roles from User security information	Select this option if you wish to migrate the users database from an existing project (see "Creation of Roles for Existing Users" on page 139).
Copy XP_ Style menu into Tab_ Style menu	Select this option to convert legacy menu entries to the format necessary for the new menu configuration system. By default, this option is unchecked to avoid potential compile errors that may occur if the legacy menu.dbf contains functions which have been removed.
Migrate included projects	Select this option to migrate the included projects associated with the selected project (see "Migrate Included Projects" on page 139).
Migrate equipment database	Select this option if you have an existing database that you want to migrate into this version. When upgrading from an earlier version, and the "PARENT" field of the equipment table was used, you should select this check box. Otherwise existing data from the PARENT field will be ignored. If runtime browsing is used, the PARENT field will return the equipment parent (the substring of the equipment name without the last '.' and anything after that). To retrieve information that was stored in the previous "PARENT" field the "COMPOSITE" field should be used.
Migrate ABCLX to OPCLX	Select this option if you want to migrate devices that currently use the ABCLX driver to the OPCLX driver. Select the **Configure** button to indicate which I/O devices you would like to migrate. **Note:** You should confirm that the OPCLX driver is installed before you use this option.
Migrate Trend/SPC storage method	If you select this option, the storage method will be set to scaled (2-byte samples) for all trends that have no storage method defined. Use this option to stop the compiler error message "The Storage Method is not defined". In previous versions, a blank storage method would default to scaled. However, this is no longer supported, resulting in the compile error message.
Create computers from Network Addresses	If you select this option, computers will be created from the servers and network addresses that you have configured for a project and its include projects. This option distinguishes whether a computer has multiple IP addresses.

NOTE: If 'Copy XP Syle menu into Tab_Style Menu' and 'Migrate Included Projects' are both selected when the migration tool runs, the following message will be displayed: "Copying menus of included projects may lead to conflicts. Any conflicts will need to be manually

corrected". To avoid this from occurring, it is recommended you run the migration tool twice. In the first instance just select the option 'Copy XP_Style menu into Tab_Style Menu', and in the second instance just select the option 'Migrate Included Projects'.

5. Click **Migrate** to begin the migration process.

   A progress dialog will display indicating the stage of the conversion and the name of the project being migrated. If you wish to cancel the migration at this point click the **Abort** button.

   NOTE: Aborting a migration will stop the migration process, and any changes already completed will not be rolled back. You will have to restore your project from the backup created in the first step.

   When the migration is complete, an information window displays information indicating the number of variables converted and the number of I/O devices deleted (if device deletion was selected at the start of migration), and where the resulting log file is stored.

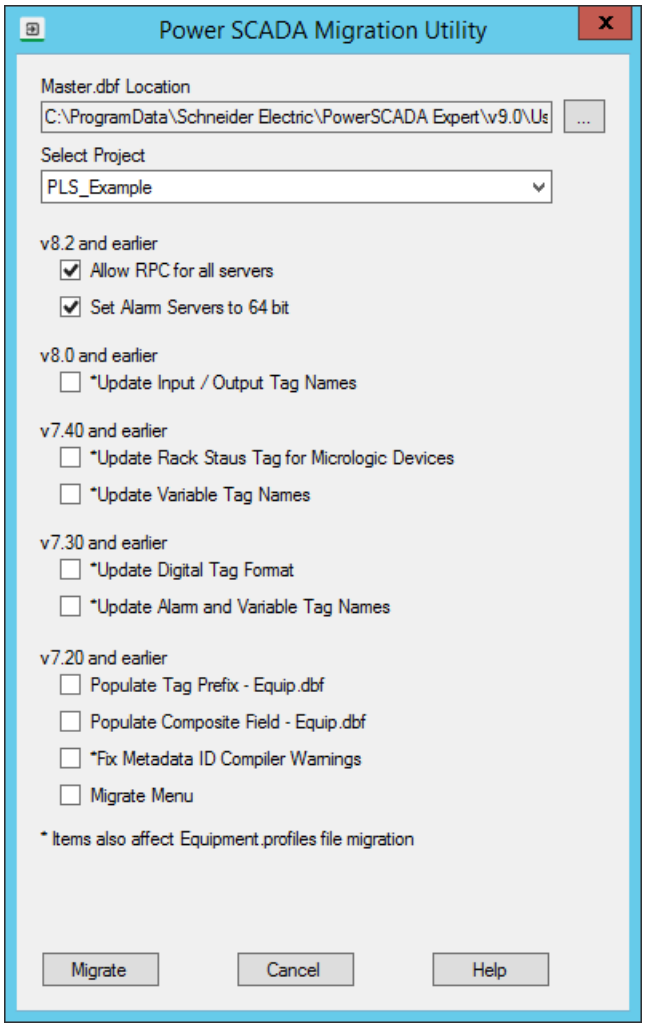
6. Click the **Close** button to close the dialog.

Use the Migration Utility

The Migration Utility lets you migrate previous versions of Power SCADA Operation to the current version. You only need to run this utility one time. Before you run the migration utility, back up your system.

To migrate your project:

1. Launch the Power SCADA Migration Tool: In Power SCADA Studio: Click **Projects**
2. Click the **Migration Tool** drop down and then click **Power SCADA Migration Tool**.



3. From the **Master.dbf Location**, choose the location for the Master.dbf.
4. From **Select Project**, choose the project that you are migrating.
5. In the bottom section, check the boxes for the elements you want to update in the runtime database (see table below for descriptions).

   NOTE: The items with asterisks will be updated in the Equipment.profiles file at every migration. When you check an item with an asterisk, it will also update the Profile so that future information added to it will be in sync with the current version. So, for example, if you run the migration and check "Update Variable Tag Names," future variable tags will be correctly formatted for the current version.

6. Click **Migrate**.
7. Verify that you have backed up the project, then click **Yes**.
8. If there is already a PageMenu.dbf file that is creating a menu for your graphics pages, you see a message telling you that the PageMenu.dbf is not empty. Click **Yes** to override this file, which overwrites the menu, leaving it blank. Click **No** to retain the menu for version 9.0

When the migration is complete, a summary screen lists the results of the migration, including updates and errors.

9. In Power SCADA Studio **Projects** activity, click **Pack** and then click **Compile**.
10. After you install version 7.3, you need to:
    a. Back up the project.
    b. Uninstall (if you are using the same computer to reinstall).
    c. Install the new version.
    d. Add the One-Line device (see "One-line memory device (zOL)" on page 274).
    e. Run the Advanced One Line tool (see "Reviewing Genie Configurations" on page 292).
11. If you are going to upgrade to a later version, you need to:
    a. Back up your project.
    b. Uninstall version 7.3.
    c. Install the new version.
    d. Restore your project.

The following table describes the changes that will be made:

Element	Description	Changes
v8.2 and earlier:		
Allow RPC for all servers	Allows performing remote MsgRPC and ServerRPC calls.	This causes the default Allow RPC value (FALSE) to be changed to TRUE.
Set Alarm servers to 64 bit	Extended memory on Alarm servers.	This setting is required for Notifications.
v8.0 and earlier:		
Update Digital Tag Format	Updates all DIGITAL tags in Variable.dbf to FORMAT "##."	Updates I/O descriptive names that were renamed to the latest standard name.
v7.40 and earlier:		
Update Rack Status	Updates the tag addresses of Micrologic rack status tags.	Corrects the tag addresses of Micrologic rack status tags.
Update variable tag names	Updates edited tag names to standard tag names.	Updates tag names to the latest standard name.
v7.30 and earlier:		

Element	Description	Changes
Update Digital Tag Format	This option will update all DIGITAL tags in Variable.dbf to FORMAT "##."	This causes digital tags in Power SCADA Operation to display without decimals ("1" or "0," but not "1.000."
Update Alarm and Variable Tag Names	Previously, some device-specific tags were renamed to fit a generic naming convention for version 7.40. Renames all existing tags to the new convention names.	Check this box to rename all new convention names. For example, the old "Sepam Not Reset" is now "Generic Not Reset."
v7.20 and earlier:		
Populate Tag Prefix-Equip.dbf	Equipment Name, which was used to build tag names, is now the equipment hierarchy name (can no longer be used to build tag names)	TagPrefix field added. It is now used to build tags. If the TagPrefix field is empty, IODevice name is used to populate Tag Prefix. If IODevice name is also empty (in a composite device), EquipmentName is used IF there are no periods in the name.
Populate Composite Field - Equip.dbf	The Parent field (previously used to determine the parent piece of equipment) has been removed from the .dbf file.	The Composite field replaces the Parent field. The Composite field will display the Parent field information, if applicable.
Fix Metadata ID Compiler Warnings	The Cicode function StrToLocal no longer allows partially translated text. For example, in *@(Protection),2*, "Protection" must be translated. Also, "2" is the metadata ID; in all custom fields (1-8) of all alarm tags, the ID part of the field must be removed.	All custom fields in alarm tags will remove the ID part (1-8) of the field, IF the translation identifier is present. Thus, in *@(Protection), 2* the "2" is removed; it will be changed to *@(Protection)*.

Remove Obsolete Memory and Alarm Devices

When you use Power SCADA Operation Migration Tool, the **Remove obsolete Memory and Alarm devices** option adjusts the following:

Memory tags to local variables: tags that are on an I/O device that are configured to use a 'memory' port.

NOTE: If there are real I/O devices in your project that have been set to use a 'memory' port during testing, these can be changed before running the migration tool to avoid those tags getting adjusted.

Alarm devices: can remove I/O devices that have a protocol set to 'Alarm', which was needed in earlier versions to enable alarm properties as tags. In version 7.x, the alarm properties are enabled via a setting on the alarm server configuration form.

Memory Devices

In previous versions of Power SCADA Operation, an I/O Device could be defined as a memory device by setting the port value to "Memory". This was generally done for one of the following purposes:

- To provide for future devices that were not currently connected to the system, but their points needed to be configured at this stage of project.
- For virtual devices where there was no corresponding physical I/O Device and you needed data storage with the entire functionality normally associated with I/O variables such as alarms.
- To act as a variable which was local to the process being used in place of Cicode global variables.

You can still use I/O Devices for future or virtual devices in version 7.0, but manually set the Port parameter to an unused value other than Memory, and set the Memory property of the device to True to indicate that it is an offline in-memory device before running the Migration Tool.

You need to review your project to identify which memory I/O Devices are local variable holders and which ones need to be changed to non-memory so that the Migration tool does not convert their variables.

The Migration Tool will set any I/O Device's port which is identified as a Memory device to the new Local Variable, and the original device record will be deleted

Alarm Devices

In previous versions of Power SCADA Operation, Alarm devices were defined as devices with their Protocol property set to "Alarm". In version 7.0 the function of configuring such a device is now replaced by setting the Publish Alarm Properties property to True on the Alarm Server.

Alarm devices with their Protocol property set to "Alarm" will be deleted from I/O Devices table by the Migration Tool.

The Migration tool can delete memory and alarm device records. If you want to delete the devices at a later time, deselect the "Remove obsolete Memory and Alarm Devices" option.

NOTE: Alarm devices with their Protocol property set to "Alarm" are no longer used and will be removed by the Migration Tool. All Alarm Servers will now publish Alarm Properties.

Converting Memory Variables

A memory variable is a variable with its I/O Device Port property set to either "Memory" or "MEM_PLC".

If there are multiple I/O Devices with the same name, possibly on different I/O Servers, the device would not be considered as a memory device regardless of its port value. In other words the Migration tool will not process the variables for memory devices with duplicate names.

Inserting New Local Variables

When the Migration Tool runs, a local variable record will be inserted for each identified memory variable, and the variable data will be copied into the new local variable.

Local variables have fewer fields than variables; the following table shows the mapping from variable to local variable when copying their data.

Variable Tag Parameter or Constant Value	Local Variable Parameter
Variable Tag name	Name
Data Type	Date Type
(Empty)	Array Size
Eng. Zero Scale	Zero Scale
Eng. Full Scale	Full Scale
Comment	Comment

With the exception of the Array Size, which has been introduced in version 7.0 exclusively for local variables, every field receives its value from the same or similar field.

Deleting Variable Tags

Once the Migration Tool has created the local variable records it will insert those variable tag records that have been converted in the previous step, and delete the original variable tag.

If an error is detected during the insertion of the local variables, the deletion of the variable tags will not be performed. If this occurs it is possible to have two records with same name and data, one in the local variable (the newly inserted record) and one in the variable tags (the original record that has not been deleted). You need to delete either of the variables manually, or restore the backed up project after removing the cause of the error then run the Migration Tool again.

Deleting Obsolete I/O Devices

Deleting obsolete I/O Devices is an optional step in the Migration Tool and will be performed after the memory variables are converted. If the delete option is chosen, obsolete Memory devices and Alarm devices will be deleted as the final step of the Migration Tool operation.

Creation of Roles for Existing Users

When upgrading an existing project using the Migration Tool, a new role will be created (if needed) for every existing user. The new role will have the same security settings that were defined for that user and be given a generic name such as Role_1, Role_2 etc. During the upgrade process, if a role exists with the same security settings as the user, then the existing role will be assigned to the user being upgraded. For example; If Role_1 exists and matches the security settings of the upgraded user then that user will be assigned Role_1 also.

If you do not want to migrate users from an existing project clear the option **Create Roles from User security information** from the migration tool dialog before running it.

Migrate Included Projects

Each project may contain multiple included projects. Additionally, any included project may contain its own included project, creating a cascading project.

The Migration Tool needs to process the original project and included projects in a single step. The reason for this is that variables can be defined in one project that refer to I/O Devices defined in another included project.

The Migration Tool performs this procedure sequentially on the "master" project then each included project.

In the case where two master projects share the same project as an included project, you should not click **Remove obsolete Memory and Alarm devices** when you process a project that contains shared included projects. This is because the removal is performed at the conclusion of the migration process on each master and included projects sequentially. This could cause the deletion of an I/O Device in the first master project which is referenced by a tag in a shared included project which is processed in a later step.

If two separate "master" projects contain the same included project, run the Migration Tool on each "master" project without selecting to delete obsolete devices.

⚠ WARNING

UPGRADE ALTERS COMMUNICATIONS CONFIGURATIONS

After upgrading, confirm and adjust the configuration of I/O devices in your project.

Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

To remove obsolete devices it is recommended that once the Migration Tool has completed successfully (without the check box being selected), run it a second time with the check box selected. This will safely remove the devices since every tag conversion were completed in the first pass of the Migration Tool.

Default Scale

The Scale properties in both variable tags and local variables are optional. If a Scale value is not specified the default value is indicated by a parameter in the Citect.ini file. The parameter name is "DefaultSliderScale" under the [General] section in the Citect.ini file. The default values for Scale is 0-32000, unless the default slider scale is true in which case the default value depends on the type, for example, Integer, String, or so on.

The Migration Tool will read this parameter and if it is not set, or set to false, then it will explicitly set any empty Scale property to a value in to the range of 0 to 32000. This will be done even if either of the Zero Scale or Full Scale parameters has a value, in which case the empty Scale parameter will receive the default value.

If the DefaultSliderScale in the Citect.ini file set to True, the Scale parameters will not be populated with a default value if they are empty, rather they will be interpreted at runtime.

Verify notifications

On a newly installed Power SCADA server:

1. Create, compile and run a simple project.
2. Open the Notification Settings. See Creating Notifications.
3. Add at least one recipient.
4. Add the settings for at least one delivery method. See Configure SMS Text Notification or Configure the Email Server.
5. Use the Test button to send a test message to the recipient. See Enable and Test Delivery.

Licensing

NOTICE
LOSS OF COMMUNICATION • Activate product and component licenses prior to the expiry of the trial license. • Activate sufficient licenses for the servers and devices in your system. **Failure to follow these instructions can result in loss of data.**

Power SCADA Operation supports two different software licensing models:

- Sentinel Licensing (using USB keys)

  Sentinel Licensing is a legacy licensing solution for Power SCADA Operation. It uses physical USB keys that plug in to each computer in your Power SCADA Operation system. The USB key contains details of your user license, such as its type and I/O point count.

  You are occasionally required to update your Sentinel keys, for example, when you upgrade to a new version of Power SCADA Operation. To do this, you need to retrieve an authorization code from Schneider Electric's online License Generator. See "Update a Sentinel Key with CiUSAFE" on page 142.
- FLEXERA Softkey Licensing

  The FLEXERA softkey solution stores license information on a FlexNet Enterprise License Server. The Power SCADA Operation client process will retrieve licenses from this server as required by the Power SCADA Operation system. To activate and administer licenses, you use the **Floating License Manager** (see "Activate Licenses Using the Floating License Manager" on page 143)

In both cases, Power SCADA Operation uses a "Dynamic Point Count" on page 144 to determine if your system is operating within the limitations of your license agreement. This process tallies the number of I/O device addresses being used by the runtime system.

A point limit is allocated to each type of license included in your license agreement. These license types include:

- Full Server Licenses
- Control Client Licenses
- View-only Licenses.

If required, you can specify how many points will be required by a particular computer. See "Specify the Required Point Count for a Computer" on page 145.

Notes:

- There is no distinction between a Control Client and an Internet Control Client.
- There is no distinction between a View-Only Client and an Internet View-Only Clien

Update a Sentinel Key with CiUSAFE

If your Power SCADA Operation system uses Sentinel Licensing, there may be times when you need to update your USB keys (for example, when you upgrade to a new version of Power SCADA Operation). To do this, you use the CiUSAFE dialog box.

To update a Sentinel USB key with CiUSAFE:

1. Plug the key you would like to update in a local USB port.
2. Open Power SCADA Studio.
3. On the Activity Bar, select **Licensing** from the menu.

   OR

   Click **Licensing**  .
4. On the **Sentinel Key Update** panel, click **Launch**.

   The CiUSAFE dialog box will appear. (See below for a description of the CiUSAFE dialog box fields.)
5. Retrieve the **Serial Number** for the key from CiUSAFE.
6. Visit www.citect.schneider-electric.com/license-generator, and enter the serial number in the **USB Key Serial Number** field.
7. Click **Submit**.

   If the key is validated, an authorization code will be generated.
8. In CiUSAFE, enter the generated code in the **Authorization Code** field.
9. Click **Update**.

   CiUSAFE will display a **Return Code** to confirm if the update was successful. See the table below for an explanation of the return code values.

CiUSAFE Dialog Box Fields

Serial Number

The serial number of the attached hardware key. If it does not appear automatically in CiUSAFE, you can read the number from the label on the hardware key. You need to enter the serial number into Schneider Electric's online License Generator to update the key.

KeyID

Each time you launch CiUSAFE, a Key ID will display in the **KEYID** field. You might need to provide the Key ID plus the serial number when updating the hardware key. This depends on the status of the key in the Power SCADA Operation license database, and you are prompted if the Key ID is required. Click **Save KeyID** to save the Key ID and serial number to a text file, which you can refer to when visiting the Schneider Electric web site.

Authorization Code

To update the hardware key, enter the 106-character authorization code. You are asked for this code once you have entered the Key ID and serial number, and your license and Customer Service agreement have been verified. Click **Update** to update your hardware key.

Return Code

The Return Code indicates the result of the key update:

0	The key was updated successfully.
1,3	Either the KeyID or the Authorization code you entered is invalid.
2	Either the KeyID or the Authorization code you entered has been corrupted.
4,16	Either the KeyID or the Authorization code you entered is invalid.
9	No hardware key could be found.

Activate Licenses Using the Floating License Manager

If your Power SCADA Operation system uses FLEXERA Softkey Licensing, you need to activate your licenses to allocate the computers in your system. To do this, you use the Schneider Electric Floating License Manager.

NOTE: If you purchased softkey licenses for your Power SCADA Operation system, the required activation codes will be emailed to your from scada.orders@schneider-electric.com.

To activate a license using Floating License Manager:

1. Obtain the required license activation code from the purchase confirmation email.
2. Open Power SCADA Studio.
3. On the Activity Bar, select **Licensing** from the menu.

   OR

   Click **Licensing**  .
4. On the **License Manager** panel, click **Launch**.

   The Schneider Electric Floating License Manger will appear. It will include a list of the floating licenses that are already available on the FlexNet Enterprise License Server.
5. Click **Activate**.
6. On the dialog that appears, select an **Activation Method**, then click **Next**.

7. Enter the **Activation ID** that was emailed to you, then click **Next**.

   The following steps will be determined by activation method you selected. If you require assistance, click the **Help** button for instructions.

8. To finalize the activation process, you will be prompted to restart the FlexNet License Administrator. Click **Yes**.

   The license you have activated will now appear in the list displayed in the Floating License Manager.

There are several other tasks you can perform with Floating License Manager. For more information on its supported functionality, see the documentation that is available from the **Help** menu.

Dynamic Point Count

Power SCADA Operation counts I/O device addresses dynamically at runtime.

The client process keeps track of the dynamic point count. This includes variable tags used by the following:

- Alarms
- Trends
- Reports
- Events
- OPC DA Server
- EWS Server
- Pages and Super Genies
- Cicode functions (TagRead, TagWrite, TagSubscribe, TagGetProperty and TagResolve)
- Any tag referenced by Cicode
- Reads or writes using DDE, ODBC, CTAPI or external OPC DA clients.

A particular variable tag is only counted towards your point count the first time it is requested. Even if you have configured a certain tag on a particular page in your project, the variable tag will not be counted towards your point count unless you navigate to that page and request the data.

You should also be aware of the following:

- A dynamic point count is tag based, not address based. For example, two tags that use the same PLC address will be counted twice.
- For the multi-process mode, each server component will accumulate its own point count which will add to the total of the client dynamic point count.

  If two trend tags use the same variable tag, it will be counted once. If two server components use the same tag(s) (say alarm and trend), the tags will not be counted twice when the point count gets totaled in the client process.
- For the multi-process mode, the client component will also accumulate its own point count, which will include all the variable tags that are used by the process.

- For the multi-process mode, the machine point count will be the point count of the client component, or the point count added up from each server component, depending on whichever is bigger. If the server point count is greater than 500, the client component point count is disregarded.
- Reading properties of a tag with TagGetProperty() or TagSubscribe() will cause that tag to be included in the point count, even if the value is not read.
- Persisted I/O (memory devices), local variables and disk I/O variable tags will not count towards the dynamic point count, unless they are written to by an external source (via OPC, DDE, ODBC, or CTAPI). For example, if you use an OPC client to write to a local variable, each local variable will be counted once the first time it is used.

  Notes:
  • You can use the CitectInfo() Cicode function or the General page in the Power SCADA Operation Kernel to determine the point count status of a client process.
  • You can specify the point count required by a client computer by using the **[Client]PointCountRequired** INI parameter.

Specify the Required Point Count for a Computer

The available point count for a Power SCADA Operation computer is determined by the type of license to which it is entitled. This is based on the role assigned to the computer by the [Client]ComputerRole parameter (which is typically set via the Computer Role page of the Setup Wizard).

Normally, the computer will get the first available matching point count. However, you can specify the point count required by a client computer by using the [Client]PointCountRequired INI parameter.

When any remote clients disconnect, the corresponding licenses that have been served to them can be reclaimed.

NOTE: A INI parameter is also available to control IP address aging. It is used to indicate how long to reserve a license for a given IP address in cases when a remote client connection is lost. This does not apply to full server licenses. The parameter is [General]LicenseReservationTimeout.

Run the software in demo mode

You can run Power SCADA Operation without the hardware key in demonstration (demo) mode. Demo mode lets you use all Power SCADA Operation features normally, but with restricted runtime and I/O.

The following demo modes are available:

- 15 minutes with a maximum of 50,000 real I/O.
- 10 hours with a maximum of 1 dynamic real I/O. This is useful for demonstrations using memory and disk I/O. Power SCADA Operation starts in this mode if no hardware key is avail-

able. If the system detects that you are using more than 1 real I/O point at runtime then it will swap to the 15 minutes demo mode.

NOTE: Writing to any tag through DDE, CTAPI, or ODBC will cause that tag to contribute to the dynamic point count even if it is a memory or disk I/O point. So if you write to more than 1 point through these interfaces it will swap to the 15 minute demo mode.

- 8 hours with a maximum of 42,000 real I/O. This is only available through special Power SCADA Operation Development keys.

Configuring

The Configuring chapter describes the different tools and tasks for configuring Power SCADA Operation

Use the information in the tables below to find the content you are looking for.

Section	Description
"Configuring prerequisites" on page 148	A list of things to consider to help you prepare for configuring a Power SCADA project
"Configuration tools" on page 150	An introduction to the Power SCADA Operation configuration tools
"Power SCADA Projects" on page 159	Creating a Power SCADA project using Project Setup, as well as compiling, backing up, an restoring a project.
"Devices" on page 174	Information and tasks on how to configure and work with: • Device profiles • Device types • Device tags • Profile Editor projects • Adding I/O devices to the project • Alarms
"Power SCADA Runtime " on page 263	Information and tasks on how to configure and work with: • Graphics pages • Animated one-lines • Menus and pages • Basic reports • LiveView • Notifications
"Assign and control user privileges" on page 465	Information and tasks on how to configure and manage user access
"Cybersecurity" on page 361	Cybersecurity considerations including: two-factor authentication (one-time password), McAfee white listing, and the Tofino Firewall.

"Customize default behaviors " on page 371	Information on using Cicode to customize a project, localizing a project, and running PSO as a Windows Service.
"System Startup and Validation Checks" on page 381	Procedures to help you validate your configured system on startup.
"Distributed systems" on page 386	Information and tasks on how to configure: • Advanced Reporting and Dashboards Module • Power SCADA Anywhere • EcoStruxure Web Services • Time synchronization and time zone settings • OFS time stamping • OPC-DA Server and Client
"Redundant systems" on page 459	Information and tasks on how to configure a redundant server.

Configuring prerequisites

- Review the system development process provided in this document.
- Gather the supporting documents that you may need. See "Resources" on page 24 for more information.
- Create a system architecture drawing, including the servers, devices and all connectivity. Define the IP addressing for each gateway and device.
- Order the appropriate equipment, including computers, software, and system devices. For help in determining what you need for your system, see the Planning section.
- Ensure that all devices that will communicate through this system are set up and properly addressed.
- Have a copy of the `Example.CSV` file for adding devices to the system. You will use this file if you need to manually add multiple devices to your project.
- Set up the Server and Client computers that you need for your system.
- Ensure that the IT team has opened the appropriate firewall ports. See the *Power SCADA Operation with Advanced Reporting and Dashboards – IT Guide* for details.
- Ensure that all license keys have been purchased and are ready to be installed.

Server CPU load balancing

Ensure that you are aware of how Power SCADA Server loading balancing works.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Do not exceed more than 50,000 tags or 200 devices per I/O Server.
- When tag and device counts indicate two different I/O server counts, use the larger number of I/O servers as your requirement.
- Assign and balance the tags or points that the Power SCADA Servers are managing across multiple CPU cores.

Failure to follow these instructions can result in death or serious injury.

While a Server machine may have sufficient overall CPU processing power, if all tags are being managed and processed by a single CPU core, the Power SCADA Server could become overloaded and could unexpectedly stop running. Important events and alarm notifications would not be received.

Configuration tools

NOTICE

INOPERABLE SYSTEM

Ensure that you have received Power SCADA training and understand the importance of the Power SCADA Operation productivity tools and workflows.

Failure to follow these instructions can result in overly complex projects, cost overruns, rework, and countless hours of support troubleshooting.

NOTE: Power SCADA Operation is build on Citect Studio and includes productivity tools that are designed and optimized to create the tags you need to configure power-based SCADA projects. If you have prior experience using Citect Studio, do not rely exclusively on Citect tools to build a power SCADA project.

Power SCADA Operation configuration tools consist of:

- **Profile Editor:** Use this tool to select tags to be used by device types (tags must be consistent with IEC 61850 naming conventions), create device profiles for individual devices, and create projects that include the device profiles to be used in a single installation. You can specify real-time tags, PC-based alarm tags, onboard alarm tags, trend tags, and reset tags to be generated for this device.
- **Application Configuration Utility:** Use this utility to configure many features that would require more time-consuming effort if performed by editing INI settings.
- **I/O Device - Wizard:** Using this wizard, you will import device profile information from the Profile Editor into a project. This tool is simply a means of moving device profile information into the project and converting it into formats that Power SCADA Operation can use.
- **Power SCADA Studio:** Use Power SCADA Studio for basic navigation. From here, you also choose the active project. Use the Power SCADA Studio for entering database-type information, such as adding clusters and servers, creating new users, and editing tags within projects.
- **Graphics Builder (design time):** Use the Graphics Builder to create one-line drawings that users can view in the runtime environment. These drawings are populated with interactive objects that are generated by genies. You can also use the graphics tool to set up system alarms and trends.
- **One-Line Configuration Utility:** You can review genie configurations, and then make necessary repairs before you compile your project.

When a Power SCADA system is deployed, the **Power SCADA Runtime** lets users view the one-line drawings, including alarms, events, and history data. With the appropriate degree of password-controlled authority, users can also perform advanced tasks, such as changing alarm setpoints and racking devices in and out.

Application Configuration Utility

Use the Application Configuration Utility to configure many features that would require more time-consuming effort if performed by editing INI settings. See "Add INI settings to AdvOneLine.ini.txt and Citect.ini" on page 277 for details about these settings.

Options that are available on every page are:

- **Project Name:** Located at the top of the page, this option allows you to choose the project. Unless you change it, this project will then remain selected for each window in the Application Configuration Utility.
- **Display Selected Settings:** Click this link to display the settings that have been entered in specific area of the Application Configuration Utility (Application Services, Application Services Host, Applications, Security) that you are viewing.
- **Display All Settings:** Click this link to view the settings that have been entered for the entire Application Configuration Utility.
- **Search:** Click this link to open a search window. Type the key word or phrase you want to search on, then click to view the list of screens on which the word or phrase are found. Click a screen name, and the screen displays. Click the 'x' in the upper right corner of the search results to close the search window.
- **Tooltips:** To view help for an individual field, point your mouse and hover over the field.

The Application Configuration Utility includes the following sections:

- **Application Services:** The Application Services section lets you configure services that connect with Citect and includes screens for setting up:
  - "Diagnostics Overview" on page 502
  - "EcoStruxure Web Services setup" on page 432
- **Application Services Host:**
  - "Application Services Host—Citect Data Platform" on page 152: four tabs contain settings to configure server/user name, ignored devices and topics, deadbands, and to display the license of the server.
- **Applications:**
  - "Basic Reports" on page 305: Use this screen to set up the delivery mode and email address from which Power SCADA Operation 9.0 basic reports will be sent.
  - "One Line Engine configuration" on page 275: Three tabs contain settings to configure one line engine behaviors.
- **Diagnostics:** Lets you set the application logging level and provides a quick view of the I/O device INI settings for all protocols, clusters, servers, ports, and devices. Use this information as the first step in troubleshooting device/communication issues in your system.
  - "Diagnostics Overview" on page 502
  - "Application Services Logging" on page 502

- **Security:**
  - "Two-Factor Authentication (One-Time Password)" on page 361
  - "Configure Single Sign-On (SSO)" on page 388

Application Services Host—Citect Data Platform

This section relates to how the Schneider Electric CoreServiceHost connects to Power SCADA Studio.

Use this page to link a Power SCADA Studio user name and password to be used when the Schneider Electric CoreServiceHost services connect with runtime.

Before you begin:

- Add the username/password to the Power SCADA Studio project.
- Have the project running in runtime mode.

Follow these steps:

1. In `Citect.ini`, set `[ctAPI] Remote = 1`.
2. Open Application Configuration Utility and then click **Application Services Host > Citect Data Platform**.
3. In **Citect I/O Server Address** choose the server address for the project that is running.
4. In **Citect User Name** enter the user name for this user.
5. In **Citect Password** enter the password for this user.
6. Click **Test Credentials** to verify these credentials. If you see an error, verify the name and password, and that runtime is running, and then try again.

   When your project is running and the credentials are valid, you see Connection Successful. The user name and password can be used to connect to Power SCADA Studio.

NOTE:

To provide extra security you can run as a CoreServiceHost as a service. Power SCADA Studio and services must both run on Session 0.

1. In `Citect.ini`, remove `[ctAPI] Remote` or set it to 0.
2. Leave **Citect I/O Server Address** blank.
3. Leave **Citect User Name** blank.
4. Leave **Citect Password** blank.

   When you click **Test Credentials**, the test will fail. However, you can verify that the service has started by viewing the Event Log.

Set up data acquisition parameters

Credentials configured in the Citect Data Platform allow applications to run externally and allow Citect to get data from basic reports, LiveView, the EWS Server, and the ETL.

This section relates to how the core service host connects to the live, running Power SCADA Operation project.

Before you begin:

- Add the username/password to the Power SCADA Studio project.
- Have Power SCADA Studio running in runtime mode.

To link a user name and password that will be used when the Schneider Electric CoreServiceHost services connect with runtime:

1. a. Open the Application Configuration Utility:
   - In Power SCADA Studio: click **Projects**  **> Power Applications > Application Config Utility**.

     OR
   - From the Start menu: Click **Schneider Electric > Application Config Utility**.

   b. In Application Configuration Utility, expand **Applications Services Host** and then click **Citect Data Platform**.
2. In **Citect I/O Server Address**, enter the server address for the project that is running.

   NOTE: This can be left blank if you are using a local connection and you are running Power SCADA Operation as a service.
3. In **Citect User Name**, enter the user name of a user configured in the project.
4. In **Citect Password**, enter the password for the Power SCADA Studio project user entered above.
5. Click **Test Credentials** to verify these credentials.

   If you see an error, verify the name and password, the Power SCADA Runtime is running, and try again.

   When your project is running and the credentials are valid, a Connection Successful message appears. The user name and password can be used to connect to Power SCADA Studio.

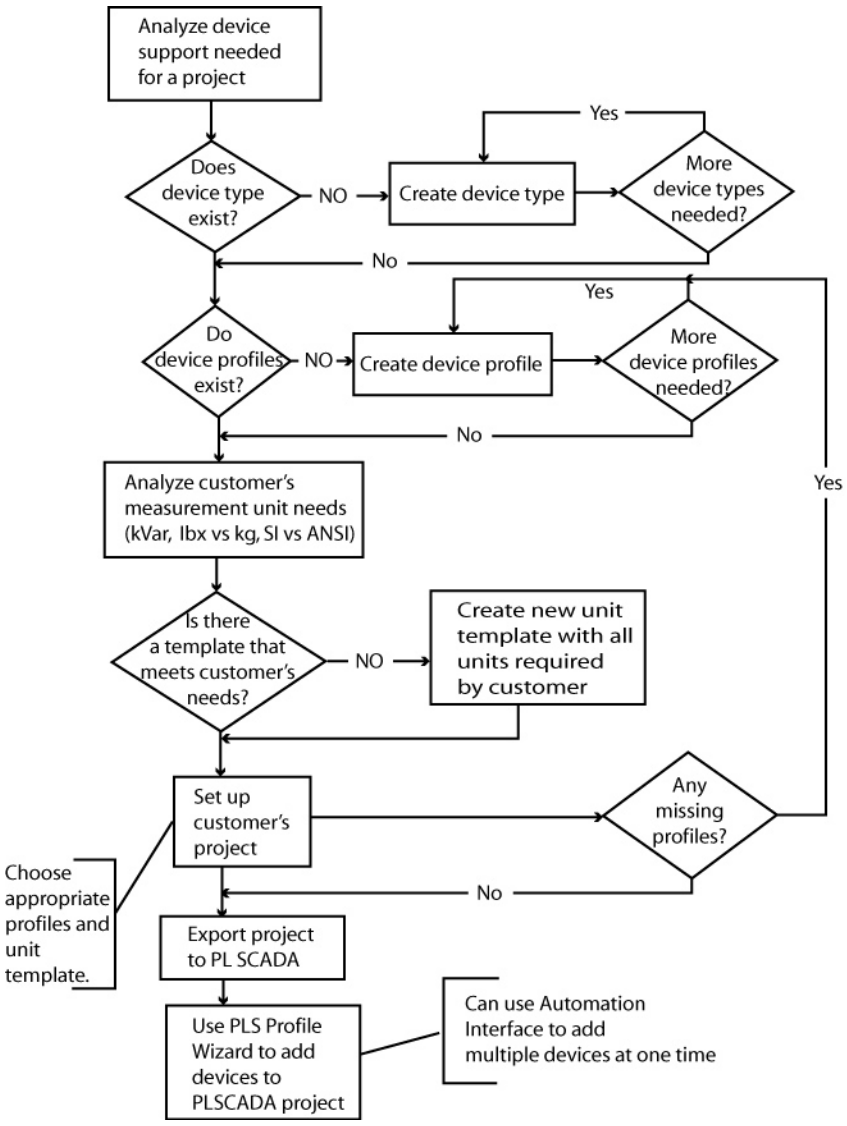
   Citect Licensing Details: This is a read-only field that displays the license key currently in use on the Power SCADA Studio server machine.

Profile Editor typical workflows

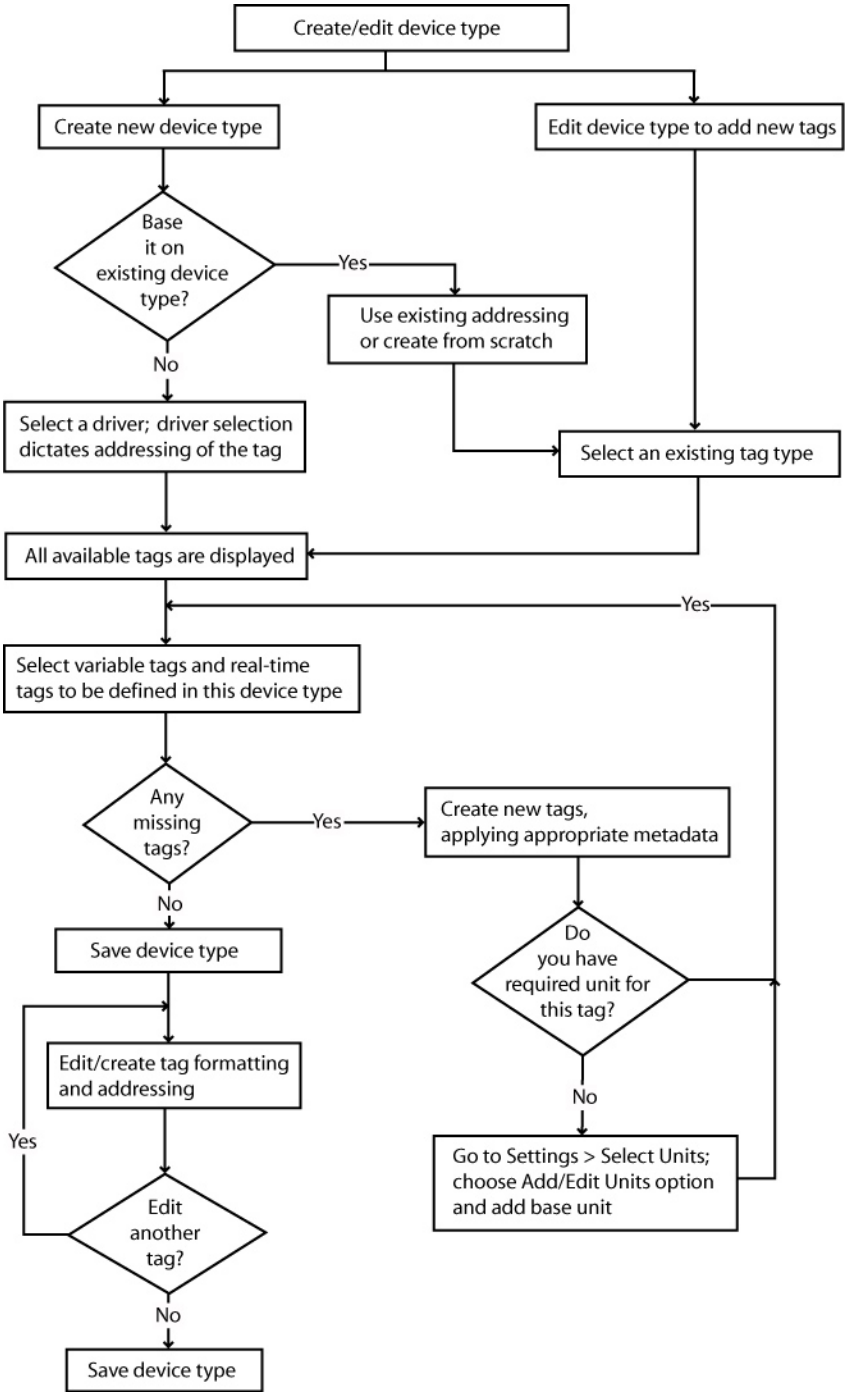
The following flow charts illustrate how to use the Profile Editor. The first illustration provides an overview, while the subsequent workflows show:

- Creating/editing a device type
- Creating/editing a device profile
- Creating/editing unit templates

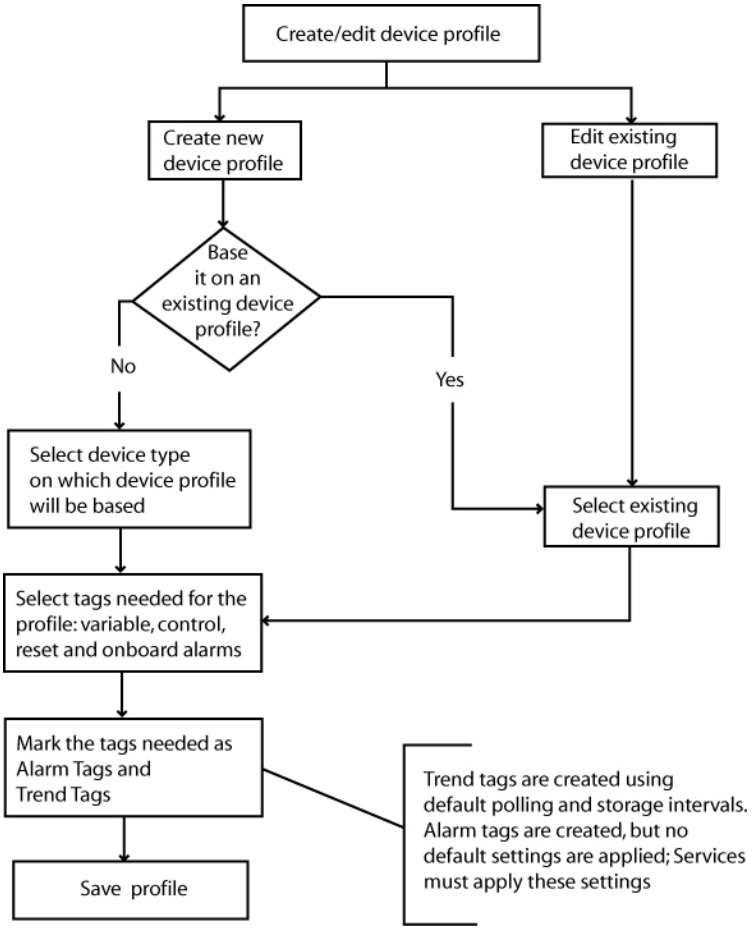
Workflow overview



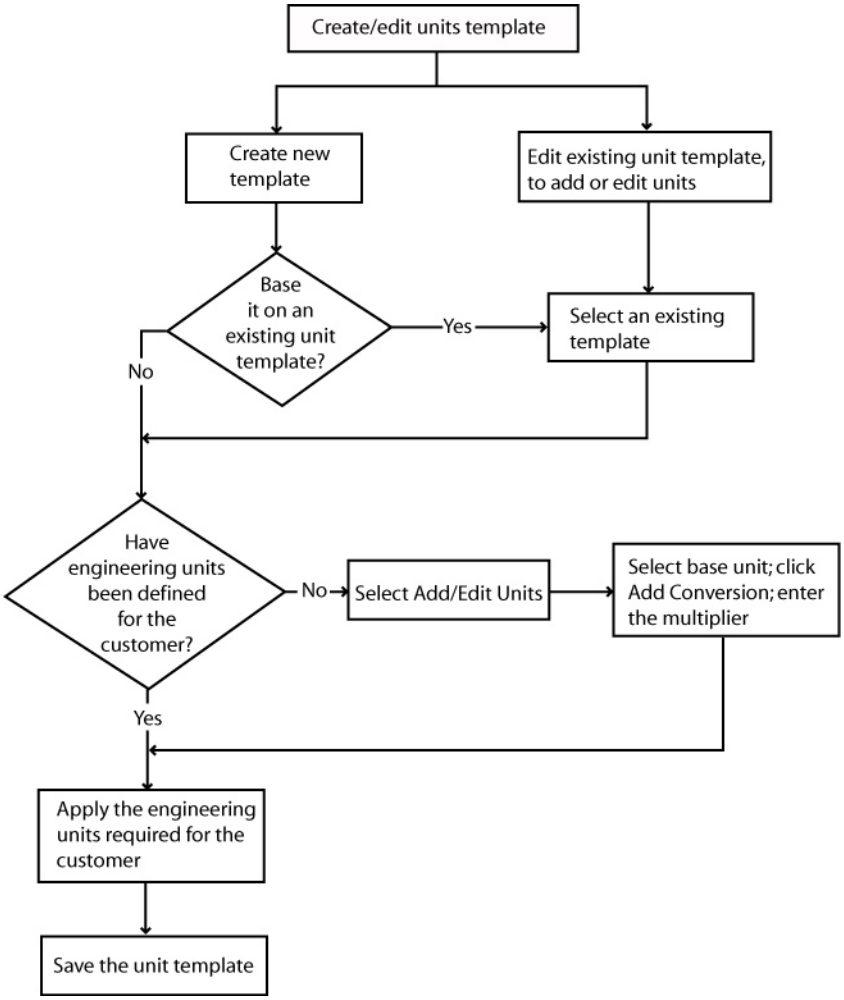
Create/edit device type



Create/edit device profile



Create/edit unit templates



Profile Editor main menu options

The main menu options (File and Settings) on each of the major tabs of the Profile Editor are described in the following table:

Field Name/Valid Entries	Comments
File > Save	Saves any current changes.
File > Create CSV file	Creates a CSV file of basic tag data. Store the file in a folder you designate. The file can be viewed in Excel.
File > Print Tag Selections	Displays a print preview of all of the tags for this device type. You can then print the spreadsheet.
File > Import	Import projects from other instances of the Profile Editor. These must be PLS or ICD files. For more information, see "Import and export project files" on page 231.

Field Name/Valid Entries	Comments
File > Export	Export a PLS or ICD file to be used in another instance of the Profile Editor, or to be used as a backup. For more information, see "Import and export project files" on page 231
Settings > Display Advanced Properties	Causes additional “advanced information” columns to display.
Settings > Remove Import Templates	Delete any import template that has been added to the project. To add import templates, see "Using import templates" on page 240.
Settings > Set Up Custom Tags	Displays the Add/Edit Custom Tags screen. See for a description of this screen.
Settings > Set Up Device Type Categories	Displays the Set Up Device Type Categories. See "Set Up Device Type Categories" on page 188 for a description of this screen.
Settings > Set Up Engineering Unit Templates	Displays the Set Up Engineering Unit Templates screen. Click "Set up engineering templates and select conversions" on page 639 for a description of this screen.
Settings > Set Up Trend Definitions	Displays the Set Up Trend Definitions screen. Click for more information.

Power SCADA Projects

Power SCADA projects are repositories that hold the configuration information for your system that includes information such as servers and other system components, I/O devices, tags, alarms and graphic pages that are used to build a runtime system, and Cicode/CitectVBA.

The configuration for a runtime system can be spread across multiple projects depending upon the scale of operations. Small, simple operations may require only a single project that houses all components required for runtime. For larger, complex operations or multi-site operations, several projects can be created based on specific plant areas, engineering processes or libraries, which are “included” together to form a single merged configuration used at runtime.

This section includes the following project-related topics:

- "Before you add a project" on page 159
- "Add a project using Project Setup" on page 159
- "Compile the Project" on page 300
- "Restore a project" on page 172
- "Backup a project" on page 172

In the Citect SCADA help file (...\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin\Help\Citect SCADA), see also:

- **Citect SCADA Projects** for information about the components that make up a project. This topic also discusses physical layout, requirements such as architecture and security, and project design.
- **Project Types** for information on preparing for a project.

Before you add a project

Before you start adding data in the project, make sure that you have:

- Used the Profile Editor to add all of the device types, device profiles, and projects
- Created a project; from the Power SCADA Studio, added clusters, network addresses, and servers
- Exported devices from the Profile Editor
- Added devices into the Power SCADA Operation project, using the Profile Wizard

Add a project using Project Setup

Project Setup lets you quickly set up a Power SCADA project. Using Project Setup, you can:

- Create and name a project
- Select screen resolution and contrast
- Specify primary and secondary server connections
- Specify the Advanced Reports and Dashboards connection
- Add users and link user roles to Windows authentication

- Add devices to a project
- Add default pages
- Add runtime menus
- Choose the landing page for each monitor in a multi-monitor project

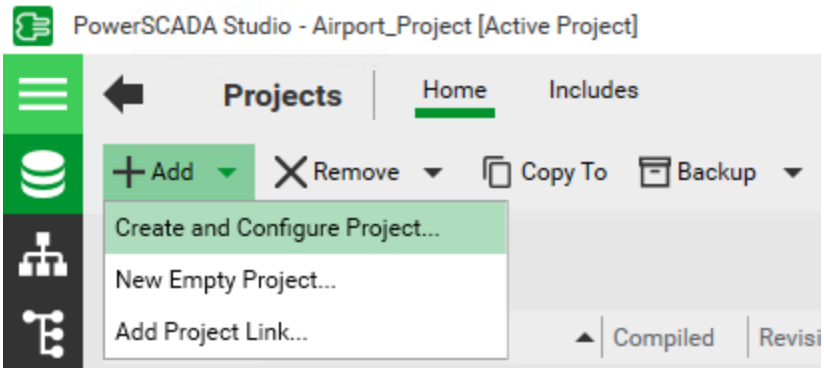
After you create the project and define its features, you can also use Project Setup to change other settings, such as devices in the CSV file, and to update your project.

For a list of project-related parameters that are created using Project Setup, see "Project Setup – Changed Parameters" on page 169

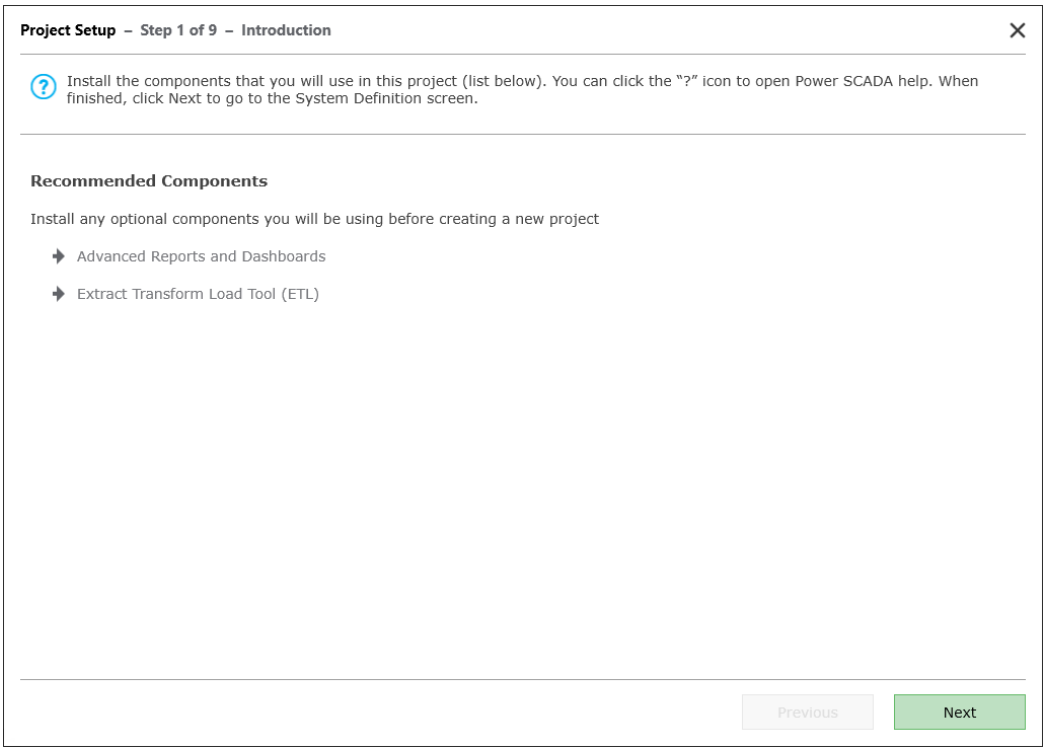
Launch Project Setup

To launch Project Setup:

1. Launch Power SCADA Studio.
2. Click **Projects**, click **Add > Create and Configure Project**.



The Introduction page appears.



The Introduction page lists optional components that you might want to include in your project. If you are using optional components, you need to install them separately. The install files are on the root of the Power SCADA Operation9.0 installation media.

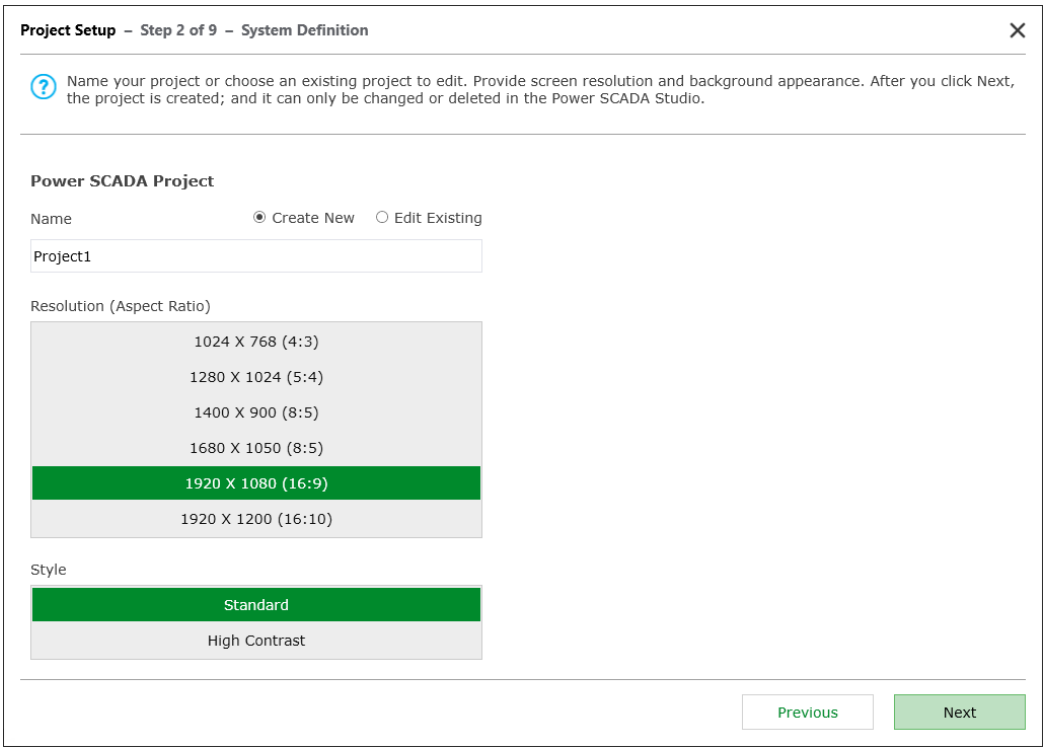
- **Advanced Reports and Dashboards** – Lets you view advanced reports and dashboards from Power Monitoring Expert. Install this component from the Power SCADA Operation installation media.
- **Extract Transform Load tool (ETL)** – Use this component to extract reporting information from Power SCADA Operation and transfer it to Power Monitoring Expert, for use in reports. For best performance during data load operations the ETL should be installed on an Advanced Reporting and Dashboards Module server.

To create a new Power SCADA project, or edit an existing project, click **Next**.

TIP: For help on any of the Project Setup pages, click "?" to the left of the top line to view the entire Power SCADA Operation help file or hover your mouse over fields to read tooltips

System Definition

Use System Definition to set the project display settings.



To set the project display settings:

1. For **Name**, click either **Create New** or **Edit Existing**.
   a. If you click **Create New**, enter a project name. Use only alphanumeric characters and underscores.
   b. If you click **Edit Existing**, choose a project from the list.

2. Under **Resolution**, choose the screen resolution that you want for the graphics pages in this project. This should match the resolution of the monitor that will display graphics pages.
3. Under **Style**, choose the contrast. Standard uses a white background. High Contrast uses a black background, which makes it easier to view graphics pages.

   NOTE: You can also set high contrast using the parameters in the Power SCADA Studio. Open your project in the Power SCADA Studio, then click **Settings > Parameters**. The parameter name is `IsHighContrast`. 0 = standard; 1 = high contrast.

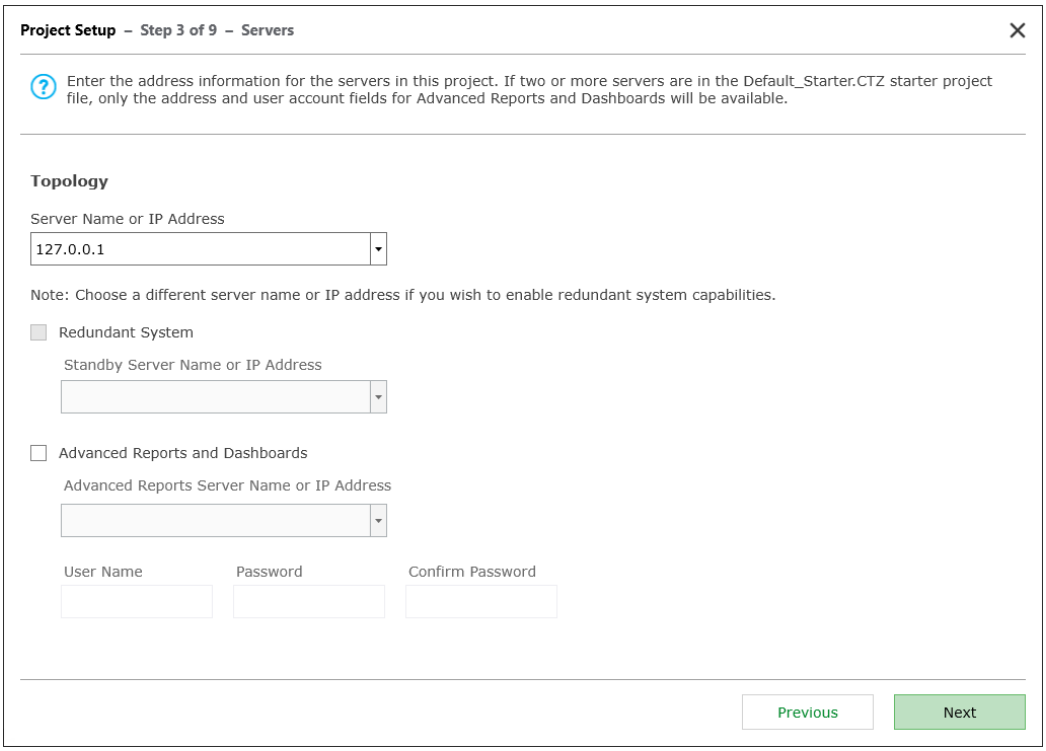
4. Click **Next**.

NOTE: After you click **Next**, the project is created. You cannot change or delete the project in Project Setup . To change or to delete it, use the Power SCADA Studio.

Servers and Web Client

Use Servers and Web Client to define the server information for your primary server, and for the Advanced Reports and Dashboards server.

Project Setup detects the number of servers that are in your starter project. If you only have one server—for example, using the loopback IP address—you see all the fields in the following image. If you are using a project that has two or more servers identified, you only see the bottom section, Advanced Reports and Dashboards.



To define the server information:

1. Enter the **Server Name or IP Address** for the project's primary server, or select it from the list.

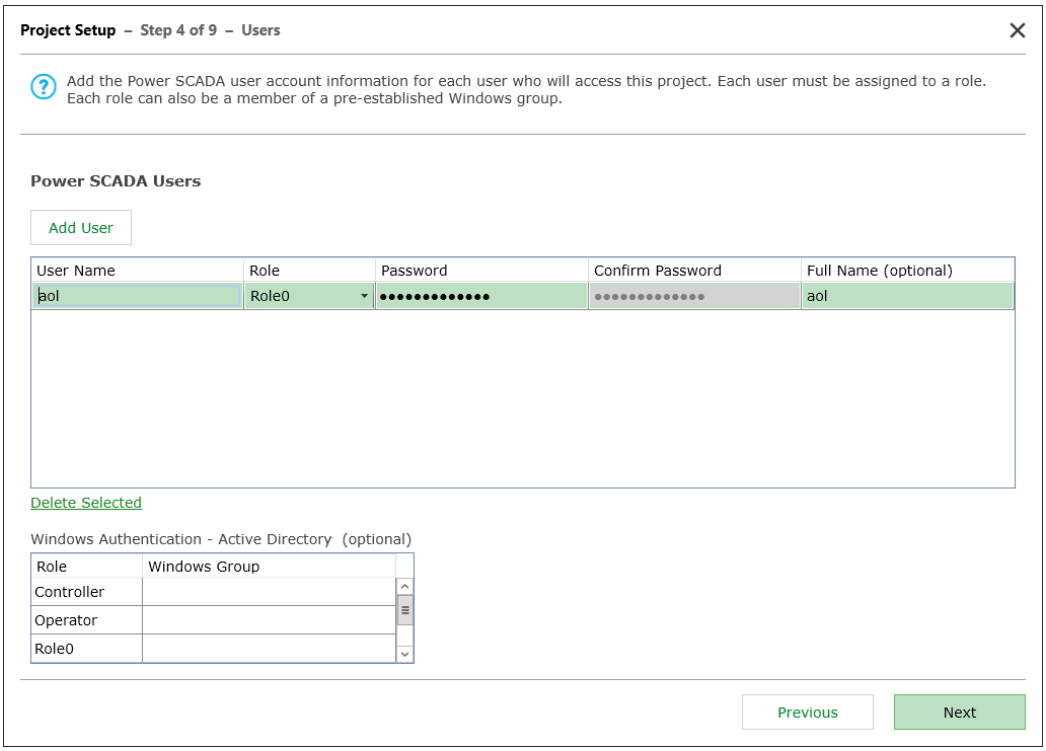
2. (Optional) If this is a redundant system:
   a. Click **Redundant System**.
   b. Enter the server name or IP address of the standby server, or select it from the drop-down list.
3. (Optional) If you installed the Advanced Reports and Dashboards module:
   a. Click **Advanced Reports and Dashboards**.
   b. Enter **the Advanced Reports Server Name or IP Address**, or select it from the list.
   c. In the **User Name/Password** fields, enter the user name and password used for the Advanced Reports and Dashboards Server. Re-enter the password in the **Confirm Password** field.

   NOTE: WebReach is also assumed to be on this server.
4. Click **Next**.

For more information on Power SCADA Operation with Advanced Reporting and Dashboards server configuration, see "Distributed systems" on page 386.

Users

Use Users to add the Power SCADA user information for each user who will access the runtime pages in this project.



To add a user account:

1. Click **Add User**. A blank row displays in the list of users.

   If you are editing a user, click the user name row.

2. Click the **Role** column for the user, and then select the appropriate role.

   NOTE: You must assign a role to each user.

3. In the **Password** and **Confirm Password** fields, enter and confirm the password to be used by this user.

4. (Optional) Enter a full name for the user. This field lets you enter a more descriptive user name; it is not used to log on to the system.

5. (Optional) Under **Windows Authentication**, assign a role to a Windows group.

   This provides central management of users through Windows. It also means that Windows users who are in the specified Windows group will have the privileges that are assigned to this role.

   For more information on Windows users, see the "Use Windows Integrated Users" section in "Add and modify user accounts" on page 359.

6. Click **Next**.

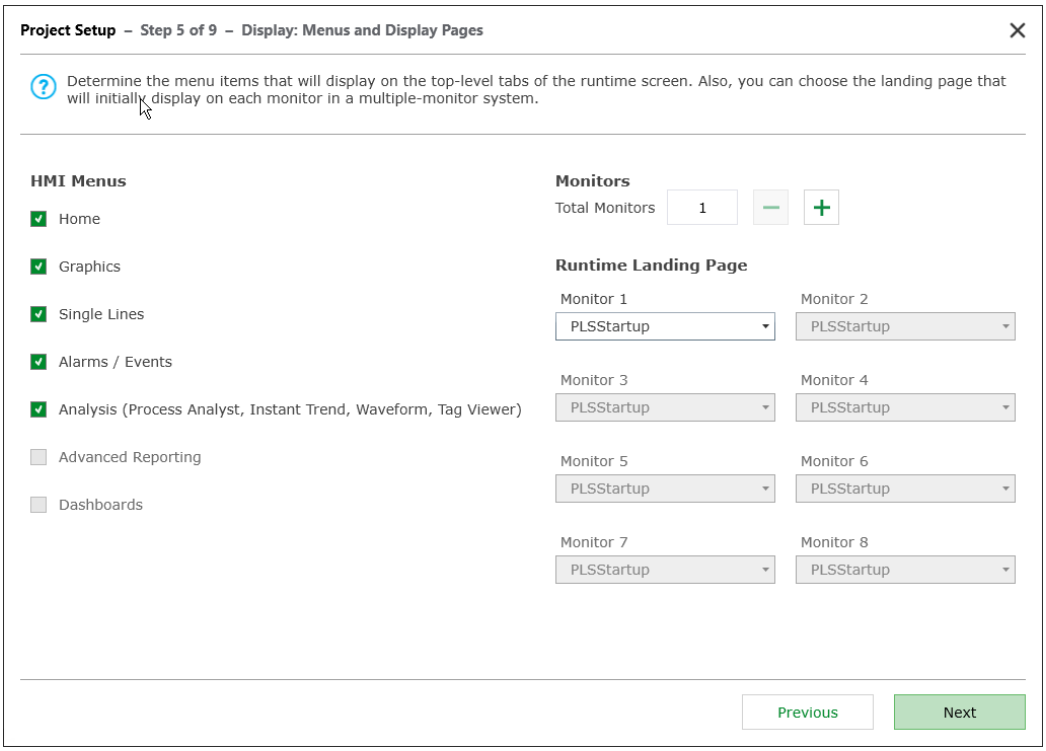
To delete a user that you previously added:

1. Highlight the user line and then click **Delete Selected**.

For more information on Power SCADA Operation user access configuration, see "Assign and control user privileges" on page 465.

Menus and Display Pages

Use **Menus and Display Pages** to add top-level menus that display in the runtime human-machine interface (HMI). The HMI is the view that users see. You can also define the default runtime page that will display on a monitor.



To add menus and landing pages:

1. Under **HMI Menus**, click the top-level menu items that you want to include in the HMI.

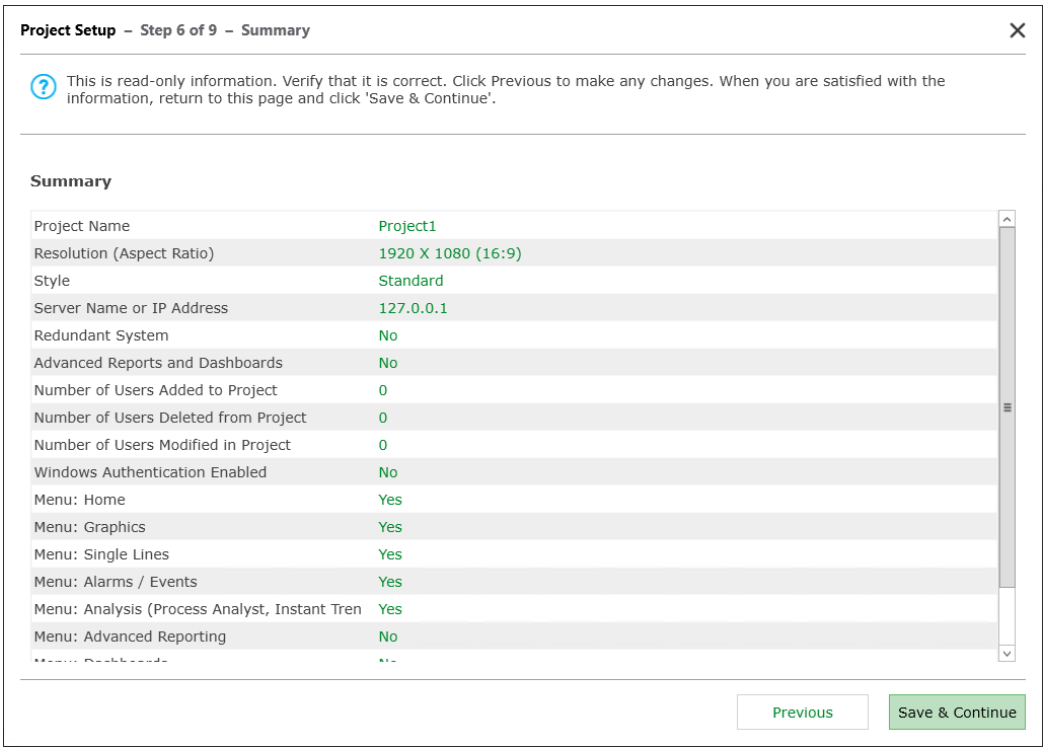
   NOTE: You can add more menu levels in the Power SCADA Studio Menu Configuration page: Visualization > Menu Configuration.

2. (Optional) If you have multiple monitors in your system:
   a. Under **Monitors**, enter the number (up to 8) of monitors in the Total Monitors field. You can also click the plus and minus buttons to increase or reduce the number.
   b. Under **Runtime Landing Page**, the corresponding number of monitors are enabled.
   c. For each monitor, select landing page you want to see when this monitor views Power SCADA Operation
3. Click **Next**.

For more information on Power SCADA Operation menu configuration, see "Power SCADA Runtime menus" on page 301.

Summary

Use **Summary** to verify that the project information is correct for your system.

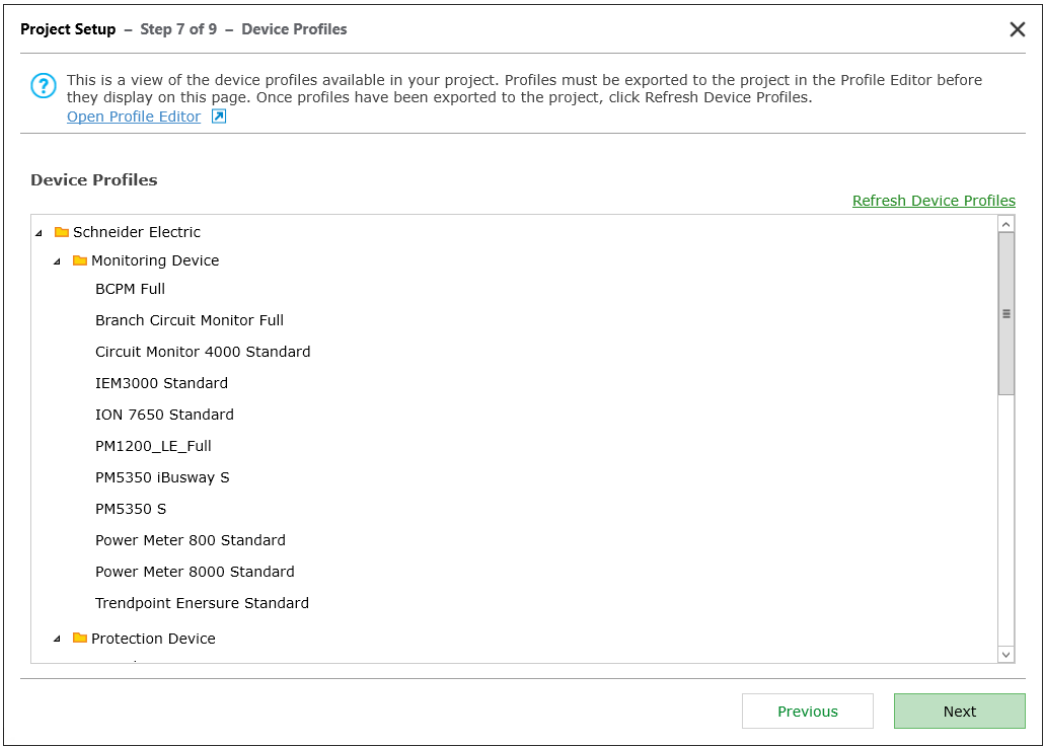


The Summary page is read-only. If you need to change something, click **Previous** to return to that screen.

When you are satisfied with the information, click **Save and Continue**.

Device Profiles

Use **Device Profiles** to add device profiles to the project.



NOTE: **Device Profiles** displays device profiles that are available to use in the project. Device profiles are displayed only if they exist in the project. If a device profile that you want to use is not listed here, you must optionally create it, add it to the project, and then export it to the project using the Profile Editor.

To add a device profile to your project that is missing from this list:

1. Click Open Profile Editor.
2. Click the **Set Up Projects** tab.
3. Under **Project**, select the project to which you want to export the device profiles, and then click **Add/Edit**.

   In the Add / Edit Project window:

   a. Add the device profiles you want to export to your project by selecting them in the Device Profile list, and then click the arrow button to move them into the Selected Device Profile list.

      NOTE: If the device profile you want to use is not in the Device Profiles list, you must create it. See for more information.

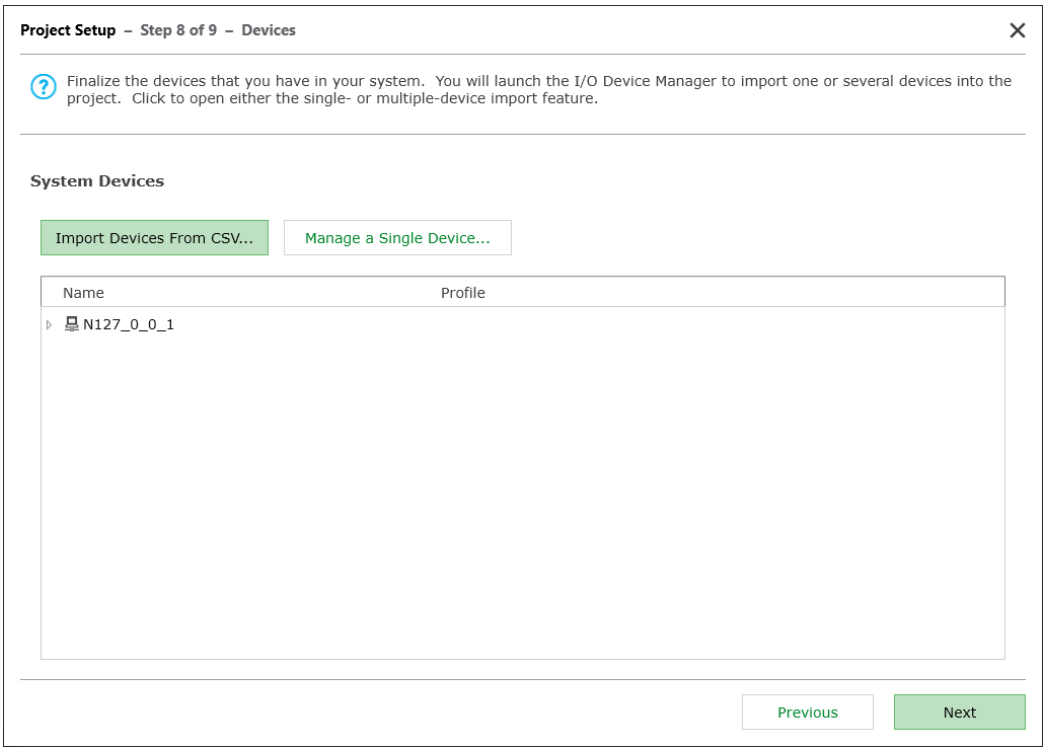
   b. Click **Save & Exit**.
4. In the Profile Editor, click **Export Project**.
5. Click **OK** to close the Export Summary window.
6. Close Profile Editor.
7. In Project Setup, click Refresh Device Profiles.

   The device profiles you added in the Profile Editor are now available to use in your project.
8. Click **Next**.

For more information on Power SCADA Operation with Advanced Reporting and Dashboards device profile configuration, see "Create Device Profiles" on page 208

Devices

Use **Devices** to add one or more devices from your system into the project.



To add one or more devices to your project:

1. Click either:
   a. **Import Devices From CSV** and then use Manage Multiple Devices to import multiple devices. For more information, see "Define multiple devices using a CSV file" on page 250

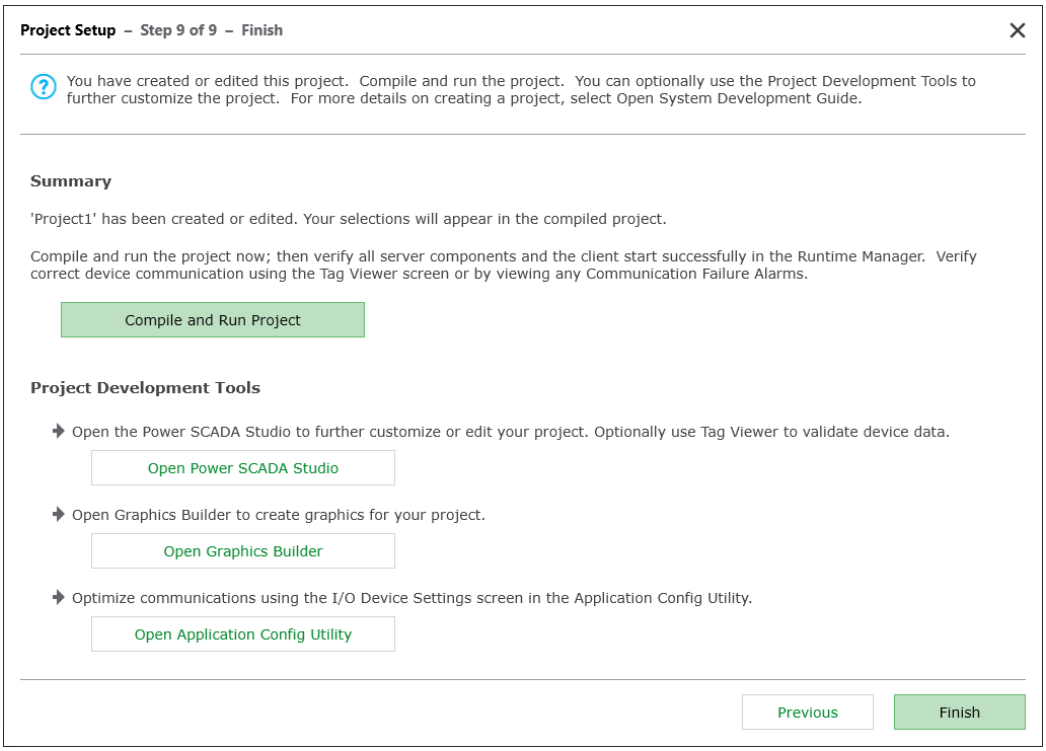
      OR

   b. **Manage a Single Device** and then create the device using the I/O Device Manager. For more information, see "Define one I/O device in a project" on page 244.
2. Click **Next**.

For more information on Power SCADA Operation device configuration, see "Managing I/O devices in a project" on page 241.

Finish

Use **Finish** to compile and run the project.



Click **Compile and Run Project** to view the project in the Power SCADA Runtime. In runtime, verify correct device communication using the Tag Viewer screen or by viewing any Communication Failure Alarms.

You can also use Finish to open the following Power SCADA Operation project development tools to further customize your project:

- **Open Power SCADA Studio** to make a variety of changes to the project.

  Many of the settings made by Project Setup are included in the Parameters file: Power SCADA Studio > Settings > Parameters. You can also change these parameters in that file.

  NOTE: If you cannot make the newly-added project active, close and then re-open Power SCADA Studio.

- **Open Graphics Builder** to create and edit the project graphics pages. For more information, see "Graphics pages" on page 265.
- **Open Application Config Utility** to edit or set up many project features. For more information, see "Application Configuration Utility" on page 151.

When you are finished, click **Finish** to close Project Setup.

Project Setup – Changed Parameters

Project Setup lets you quickly set up a variety of project information. The following parameters are organized according to the Project Setup page that lets you edit them.

System Definition screen

Project Setup Setting	Section	Parameter Name
Resolution	MultiMonitors	Resolution
Style	MultiMonitors	IsHighContrast

Servers

Project Setup Setting	Section	Parameter Name
Advanced Reports Server	Applications	Hostname

Menus in Project Setup

For each page selected in Project Setup (Step 5), the menu configuration items are added.

Project Setup Setting	Section	Parameter Name
Monitor Count	MultiMonitors	Monitors
Monitor 1 Landing Page	MultiMonitors	StartupPage1
Monitor 2 Landing Page	MultiMonitors	StartupPage2
Monitor 3 Landing Page	MultiMonitors	StartupPage3
Monitor 4 Landing Page	MultiMonitors	StartupPage4
Monitor 5 Landing Page	MultiMonitors	StartupPage5
Monitor 6 Landing Page	MultiMonitors	StartupPage6
Monitor 7 Landing Page	MultiMonitors	StartupPage7
Monitor 8 Landing Page	MultiMonitors	StartupPage8

In addition to parameters, you can do the following:

Servers, Network Addresses, and Computers

Project Setup Location	Item
Step 3: Servers	Add I/O, Alarm, Trend, and Report Servers, primary and redundant NOTE: Clusters are also added here.
Step 3: Servers	Add network addresses, primary and redundant
Step 5: Display: Menus and Display Pages	Create HMI menus: setup for graphics pages
Step 5: Display: Menus and Display Pages	Determine runtime landing pages at various monitors used in the project
Step 7: Device Profiles	Choose device profiles
Step 8: Devices	Add I/O devices; including equipment, ports, boards, I/O devices, variable tags, alarm tags, trend tags
Step 9: Finish	Compile and run the completed project

Final - Compile and Run Project

When you click **Compile and Run Project** on the final screen, the following changes are made to the citect.ini file:

Section	Parameter Name	Value
Lan	TCPIP	1
CTEDIT	Run	(Project's path)
CTEDIT	LASTDATABASE	(Project's name)
CTEDIT	LASTDATABASEPATH	(Project's path)
Client	ComputerRole	0
Client	FullLicense	0
Client	PartOfTrustedNetwork	1
Client	Clusters	(Comma separated list of available clusters for the project)
CtSetup	CustomSetup	0
Internet	Server	0
Alarm	SavePrimary	(Project's path)
Report	InhibitEvent	1
Report	RunStandby	1
Trend	InhibitEvent	1
Event	Server	0
Win	AltSpace	1
Server	AutoLoginMode	1
Server	EWSAllowAnonymousAccess	0
(ServerType.Cluster.ServerName)	StartupCode	PLS_StartAdvOneLine() *This is set on one IO server on each server machine in the project
(ServerType.Cluster.ServerName)	Clusters	(Comma separated list of available clusters for the project)

Compile the Project

After you install the software and create the project—along with clusters, network addresses, and servers—compile the project. You will also need to compile your project periodically during system setup.

Pack your project before you compile. In Power SCADA Studio, click the **Projects** activity, click **Pack**.

In Power SCADA Studio, click **Compile**. If you are promoted to save your changes, click **Save**.

If there are errors or warnings after the project is compiled:

1. At each error, click **GoTo**, which opens the location where the error occurred.
2. Using the information in the error message, correct the error.
3. After all errors are addressed, re-compile to verify that the errors are removed.

   For additional information, click Help at the error screen.

Restore a project

To restore a project, overwriting its current settings:

1. In Power SCADA Studio, click **Projects**.
2. Click the **Backup** drop down and then click **Restore**.
3. Beside the **Backup file** text field, click **Browse**, and then browse to the location of the project file you will use to restore.
4. (Optional) Click **Select all included projects**.
5. In the **To** area, click **Current Project**.
6. In the **Options** area:
   a. Click **Configuration files** to restore backed up INI files and the TimeSyncConfig.xml file (used to store time synchronization settings).
   b. If you backed up the sub-directories under the project, the directories will be listed under **Select all sub-directories** to restore. You can restore all or no sub-directories, or you can select specific sub-directories to restore.
7. Click **OK**.

Backup a project

To back up a Power SCADA Operation project file:

1. In Power SCADA Studio, click Projects.
2. Click **Backup**.
3. From the **Name** drop down, choose the project you want to back up.
4. (Optional) Click **Select all included projects**.
5. Click **Browse** and then browse to the location where you want to store the project backup file.
6. In the **Options** area, click **Save configuration files**. This saves the citect.ini file.
7. Click **OK**.

The backup CTZ file is written to the location that you choose during backup. This is a Citect Zip file; you can open it with WinZip.

NOTE: To back up a Profile Editor project file, see "Profile Editor export" on page 232.

Delete information from Power SCADA Operation

If you need to delete any data that you entered (clusters, servers, genies, and so on), see Citect SCADA Help for information on how to delete the data, then use the Pack command to completely delete it. To do this, in Power SCADA Studio, from the **Projects** tab, click **Pack**.

Devices

Use the Power SCADA Operation Profile Editor to create and manage device type tags and tag addresses, and use tags as building blocks for device types. You can also create device profiles for unique devices. Once all your device tags are created, you save them as a Profile Editor project which can then be exported for use in Power SCADA projects.

About device profiles and tags

By default, Power SCADA Operation includes a large number of device types and their associated tags. You can use these device types as is or as templates to create your own custom device types.

Before you create your own device types, review the topics in this section. The device types and tags that you want may already be created for you.

Reviewing default device types and tags

By default, Power SCADA Operation includes a large number of device types and their associated tags. Before you create custom device types and tags, verify that the device type does not already exist in Power SCADA Operation.

To review the default device types and tags:

1. Open the Profile Editor.
2. On the **Define Device Type Tags** tab, select a device type name from the **Device Type Name** drop-down list.

   The available tags display in the body of the page. There are several sub-tabs for real-time tags, onboard alarms, control tags, and reset tags. The tags that are selected for the device type display there.
3. If you do not find the device type or tags that you need, you can:
   - "Create custom device types" on page 187
   - "Create custom tags" on page 190

Supported device types and protocols

When you install Power SCADA Operation, you are prompted to choose the drivers that you will use. A certain number of generic drivers are installed by default (including PowerLogic device types), and you are not prompted for them. Device types and protocols supported in Power SCADA Operation are:

- Generic MODBUS (includes BCPM and any device, such as a PLC or UPS, that communicates via MODBUS). When adding a controllable device in the Profile Editor, such as a circuit breaker, use the “Controllable Device” driver; otherwise, use the “Generic Power Device” driver. For JBus devices, select Generic JBus Device.
- Sepam 20, 40, and 80 Range, 2000
- Masterpact MicroLogic 5P and 6P, A, H
- Compact NSX (MicrologicV)
- CM2000

- CM4000 series
- PM650
- PM800 series
- PM5000 series
- PM700 series
- ION protocol devices
- IEC 61850 protocol devices
- IEC 870-5-104
- DNP3
- BCPMA (branch circuit power meter, full feature support)
- CSI SER (Cyber Sciences SER)
- ProTime 100 SER (Monaghan Engineering)

The Profile Editor

The Profile Editor is a multiple-screen application that lets you create device types, create device profiles, and set up projects.

The Profile Editor consists of the following tabbed panes:

Define Device Type Tags – Use this pane and its screens to add and edit information for real-time, onboard alarm, control and reset tags and to create and edit device types. See "Define Device Type Tags" on page 178 for complete instructions.

Power SCADA Operation uses the IEC 61850 tag-naming convention to create tags that measure device quantities. Although most of the tags you will use are already entered into the system, you can add custom tags. For more information, see "About tags" on page 200.

NOTE: To avoid potential communication errors, use the Profile Editor to create all custom tags that will communicate with equipment.

Create Device Profiles – Use this pane and its screens to add and edit individual profiles for specific devices. A device profile is a subset of the possible variable tags, alarm tags, and trend tags for a particular device type. See "Create Device Profiles" on page 208 for complete instructions.

Set Up Project – Use this pane and its screens to bring together all of the system attributes for a single customer or installation.

For example, the customer installation will include a certain combination of device profiles (depending on the devices installed at the site). The project allows a specific unit template to be applied, converting units (such as watts) into units used by the customer (such as megawatts). This causes tags to display in the converted format. Projects also allow you to rename tags to suit a customer's needs (for example, Current A could be renamed to Current Phase A). See "Set Up Projects" on page 226 for details.

TIP: For more information on how to use the Profile Editor screens, click the help link (?) at the top of the page. The help file will open to instructions for the Profile Editor screen you are viewing.

Related reference topics:

- "Profile Editor typical workflows" on page 153
- "Profile Editor main menu options" on page 157

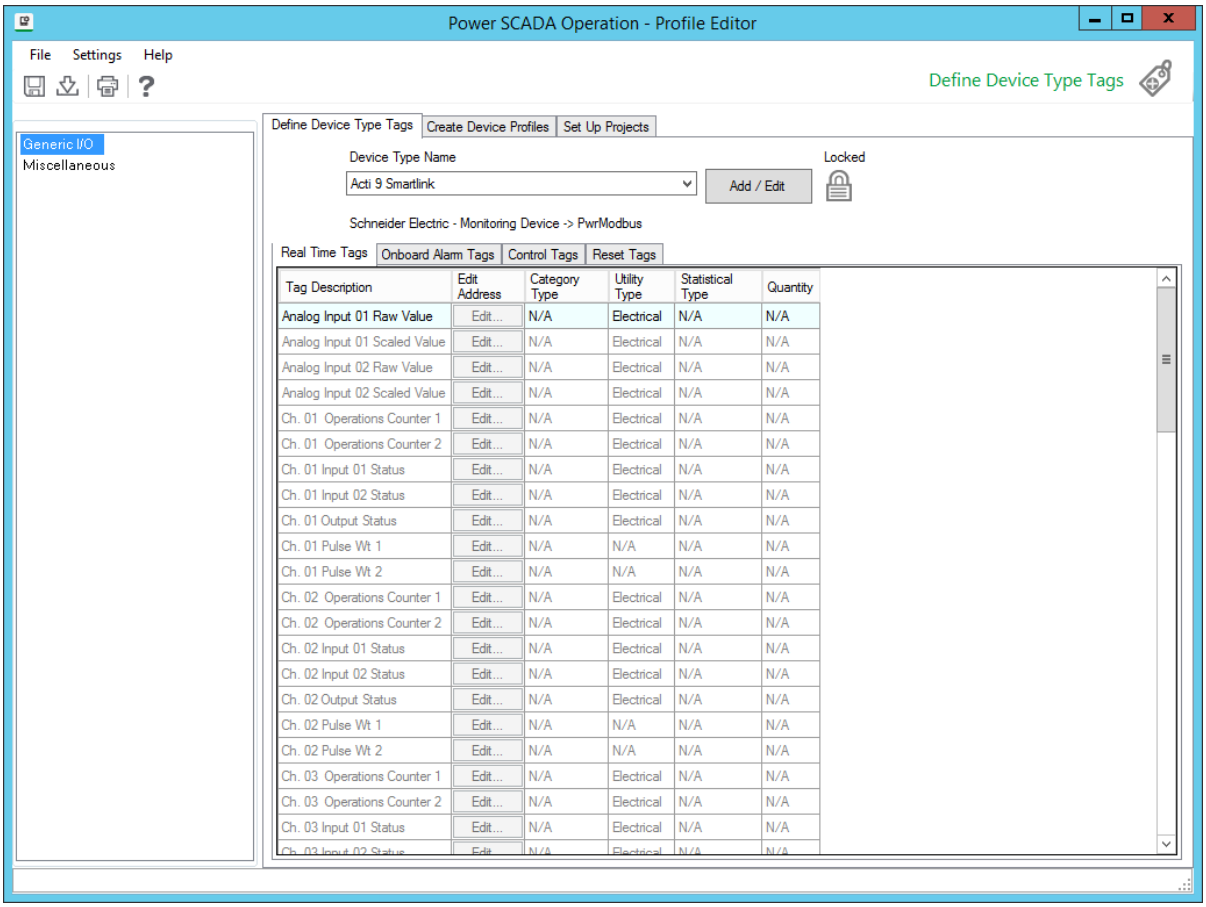
Launch the Profile Editor

NOTE: To avoid potential communication errors, use the Profile Editor to create all custom tags that will communicate with equipment.

There are several ways to open the Profile Editor:

- From the **Start** menu:
  - Start > All Programs > Schneider Electric > Power SCADA Operation > Config Tools > Profile Editor.
  - Start > Apps > Schneider Electric > Profile Editor.
- From the desktop: Double-click the **Profile Editor** shortcut.
- In Power SCADA Studio > **Topology** activity, click **I/O Devices > Device Profile Editor**.

The Profile Editor screen displays with the **Set Up Projects** tab selected. There are two other tabs, used to create device type tags and profiles.



Locked and custom icons

Two icons may appear to the right of the **Add / Edit** button on some screens: the locked icon and the custom icon.

The Locked Icon  :

This icon indicates that the selected file (e.g., device type, profile, or project) cannot be edited. All standard device types (for example, Circuit Monitor 4000, MicroLogic Type P, Power Meter 800) are automatically locked; they cannot be unlocked.

To lock a device type that you create:

1. From **Define Device Type Tags**, click **Add / Edit**.
2. In **Add / Edit Device Type**, you can choose **Create New**, **Create From**, **Edit Existing**, or **Delete Existing**.
3. Enter the new device information, or select the device to be edited.
4. To lock the device, click **Lock this Device Type**.
5. Click **Save & Exit** to save the lock and exit **Add / Edit Device Type**.

After you lock a device type, you cannot unlock it. However, you can restore by copying the locked device type (Create From on the Add/Edit Device Type screen), then saving the copy with a new name.

The Custom Icon  :

This icon indicates that a device type or profile is user-created. It may have been created new, created from an existing device type or profile, or created by editing an unlocked custom device type or profile.

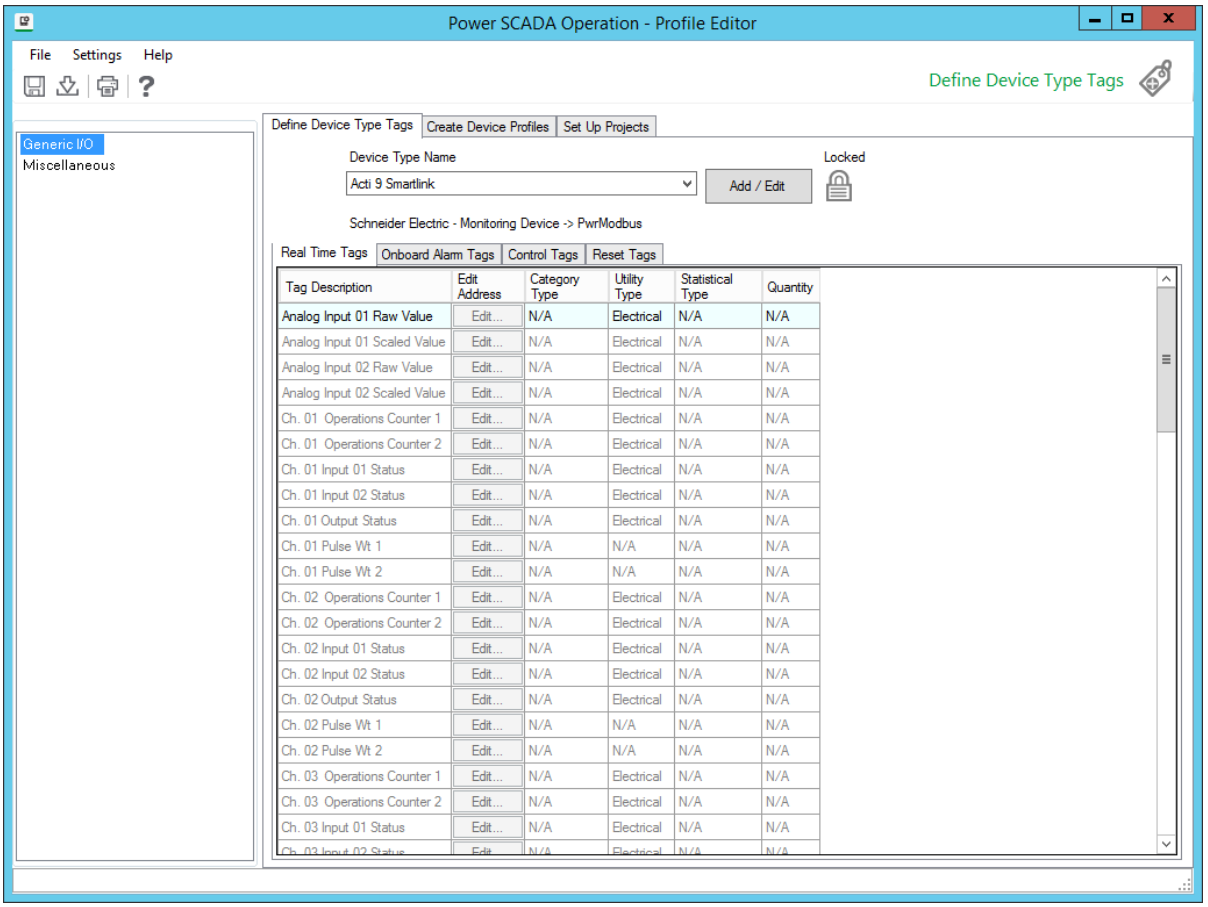
Set the screen resolution

Depending on the screen resolution you use, some of the Profile Editor screens may take up the entire viewing area. We recommend that you use at least 1024 x 768 resolution. You can also auto-hide the taskbar to provide more room.

TIP: For more information on how to use the Profile Editor screens, click the help link (?) at the top of the page. The help file will open to instructions for the Profile Editor screen you are viewing.

Define Device Type Tags

Define Device Type Tags and its related screens are used to define the following device-related data: custom tags, device types and device type categories, and base units/conversions:



On the Profile Editor > **Define Device Type Tags** tab, follow these general steps to add tags and devices to your system:

1. Manage the units and unit conversions that you will use (such as amperes into milliamperes), see "Add or edit a base engineering unit or conversion" on page 643.
2. Add and edit custom tags, see "Setting up custom tags" on page 196.
3. Add or edit device types, see "Managing device types" on page 182.
4. Establish device type categories and subcategories, used in reporting, see "Print the .CSV file" on page 187.
5. Edit tag addresses, see “"About tags" on page 200”.

Define Device Type Tags tab

The **Define Device Type Tags** tab displays device types and the tags that may be associated each device type. This includes real-time, onboard alarm, control, and reset tags. Most of the fields on this tab are read only (they can be changed on other screens). The following table describes this tab. The tags listed assume that Advanced Properties has been checked. Not all elements appear on every sub-tab.

Field Name/Valid Entries	Comments
Device Type Name (Select the device type)	Each device type includes a different number of tag categories, which also changes the list of tags that display. The device list includes the default device types, as well as any that have been created for this system.
Add / Edit button: Click to open the Add / Edit Device Type screen.	Provides a means of adding new device types and editing custom device types (user-created device types that are not locked). Also provides a means of adding new custom tags and editing existing tags.
Locked/Custom icons: 	Locked icon indicates that the list of selected tags cannot be edited. Custom icon indicates that the device type was created by a user. See "Launch the Profile Editor" on page 176 for complete information.
Tag groups (left-hand pane)	

Field Name/Valid Entries	Comments
Select a tag group; the tags included in that group display on the right.	Each tag belongs to a group. The group is determined when the device is added to the system. For custom tags, this is on the Add/Edit Custom Tags screen. Tags for standard device types are pre-determined and cannot be changed.) Note: If a tag group displays in red copy, there is at least one address that is not valid for the tag to which it is assigned. To correct this issue, click the tag group, ensure that Display Advanced Properties is selected, then scroll down through the tags in the right-hand column. The tags that have invalid addresses will have the “Edit...” displayed in red. Click this field to open the Edit Address page; correct the errors in the address.
Tag tabs: Real Time, OnBoard Alarm, Control, and Reset Click a tab to view the tags of that type that are included for the selected device type. If the device type is not locked, you can use the Add/Edit Device Type screen to edit the list of tags.	
Tag Description (all tag types)/Display only	This is the tag name, hard-coded for standard tags. For custom tags: The name is from the Tag Name field in the Add/Edit Custom Tags screen.
Units/Display only	Lists the abbreviation, added when creating the engineering unit template.
IEC Tag Name/Display only	Tag name that conforms to IEC61850 standard. See "About tags" on page 200 for more information.
Type (Real Time only)/Display only	Displays the data type chosen when the tag was created.
Address (not Control tags)/ To edit, click the Edit Address link.	Displays the address information for this tag, including elements such as type of register, number of registers, and scaling and bitmasking data. "About tags" on page 200 for a detailed description of address construction.

Field Name/Valid Entries	Comments
Normally Closed (Control tags only)/ Check the box to invert the functionality of the control. See description.	For a control with one command, writing a 1 to the tag will cause the command to occur. (This option is greyed out.) For a control with two commands that is either static or normally open, writing a 1 to the tag will cause the first command to occur; writing a 0 will cause the second to occur. (Check box not checked.) For a control with two commands that is normally closed, writing a 1 to the tag will cause the second command to occur; writing a 0 will cause the first command to occur. (Check box checked.)
Edit Addr/Click to display Edit Address screen. (Real Time and Onboard Alarm only)	Provides the means of changing the elements of an unlocked real-time tag address (for example, the number of registers, their numbers, and whether they are consecutive). See "Edit tag addresses" on page 190. for detailed information.
Register 1/Display only (Real Time tags only)	This field contains first register used to store this tag. If there are additional registers, they are indicated in the address. The total number of registers is listed in the Num Registers column. This field allows you to verify and/or change the value of Register 1 without having to open the Edit Address screen. Note: If you enter a number that is not compatible with other address settings, you are prompted to go to the Edit Address screen.
Num Registers/Display only (Real Time tags only)	Displays the number of registers used by this tag.
Formatting/Select the format type from the drop-down list (Real Time tags only)	After you change formatting for a tag and move the cursor to another field, you are asked whether you want to open the Address Editor. If you click No, the format is unchanged; if you click Yes, the Edit Address screen opens for you to enter the appropriate changes for this tag. See "Edit tag addresses" on page 190.
Scaling Register/View or enter the register number (Real Time tags only)	This is entered in the Edit Address screen, but it can be edited here. It is the register used to read the value for scaling. Note: If you enter a number that is not compatible with other address settings, you are prompted to go to the Edit Address screen.

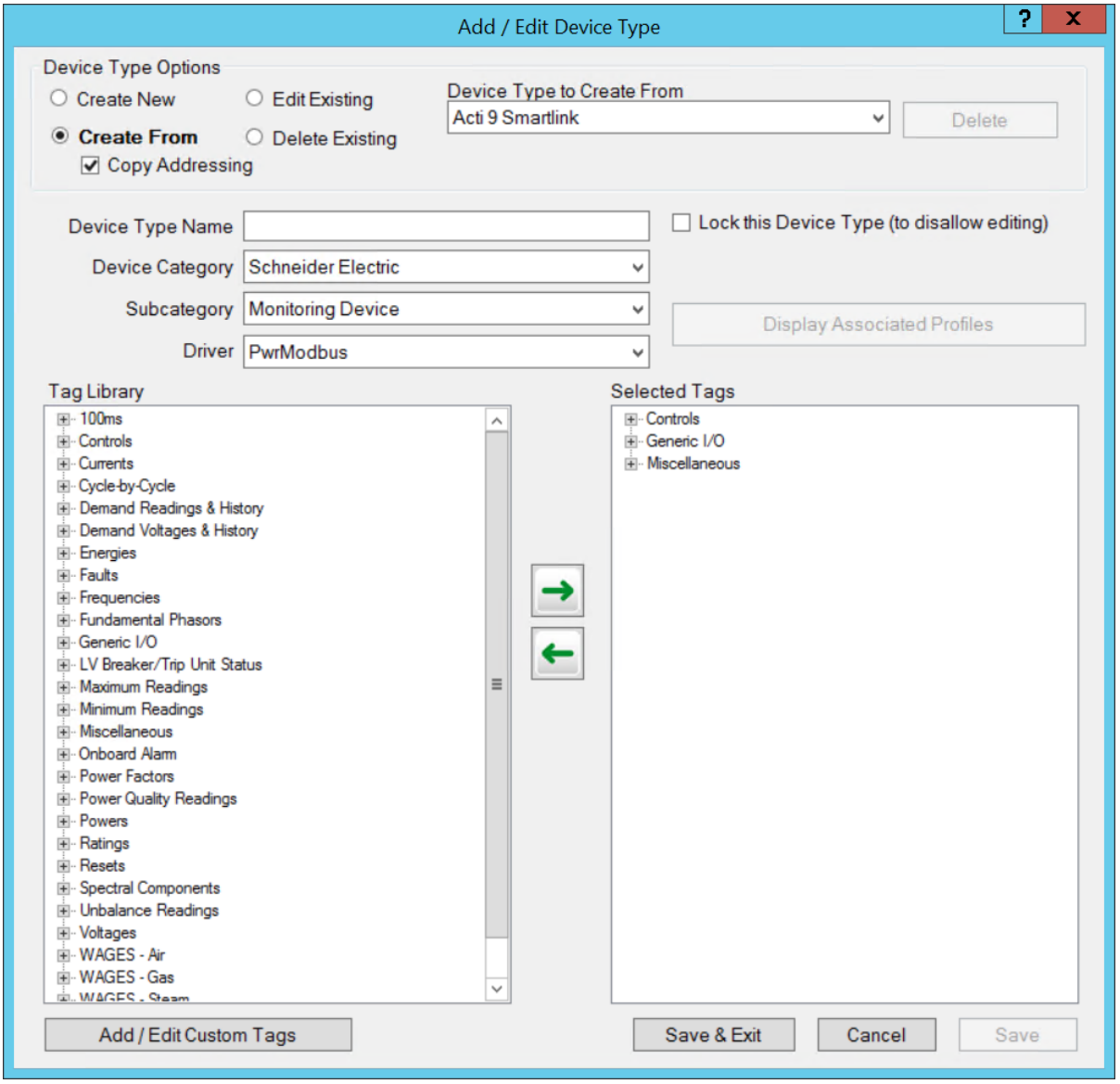
Field Name/Valid Entries	Comments
Functional Address/Display only (Real Time, Onboard Alarm, Control, and Reset tags)	If you have added a functional address for this tag, it displays here. To add or edit this address, use the Edit Functional Address field. Note: Functional addressing is described in "Glossary" on page 656.
Edit Functional Address/Add the code for the address	Typically used for data concentrators, the functional address is a means of entering the individual data points needed to define multiple addresses. Entered as a formula (must be in C#), it will contain the variables the user must enter when the block is instantiated by the Profile Wizard. A simple example: Address = "T:MV;m:" + (startingpoint + 1005).ToString() + ";L:P:22" You would then define "startingpoint" when instantiating the profile in the Profile Wizard.
Tag ID/Display only/Display only	Assigned by the system when the tag was created. If this is a custom tag, it will be a negative number.
Category Type (real-time only) Utility Type (real-time only) Statistical Type (real-time only) Quantity (real-time only)	Each of these types is a real-time filter, added when the tag was created. See for more information.
Categorization (onboard alarm only) Subcategorization (onboard alarm only) Alarm Type (onboard alarm only) Alarm Group (onboard alarm only) Alarm Level (onboard alarm only)	Each of these is an onboard alarm folder, added when the tag was created. See for more information.

Managing device types

Use the **Add / Edit Device Type** screen to begin adding, editing, or deleting a device type from the system. See "Edit a device type" on page 185 and "Delete a device type" on page 186 for instructions on editing or deleting device types.

To open **Add / Edit Device Type**:

In Profile Editor, click **Define Device Type Tags**, then click **Add / Edit** (to the right of the Device Type Name field.)



The following table describes the parts of the **Add / Edit Device Type** screen.

Field Name	Valid Entries	Comments
Create New	Click one of the radio buttons to select the action you want to take.	Click to add a device type that is not based on an existing type.
Create From		Click to copy an existing device type.
Edit Existing		Click to edit an unlocked device type.
Delete Existing		Click to delete an unlocked device type that is not associated with a profile.

Field Name	Valid Entries	Comments
Device Type (to Create From/to Edit/ to Delete)	select type	Select the device type that you want to create from, edit, or delete.
Copy Addressing		Active when you choose Create From. Check this box if to copy the addressing of the “from” device. This gives each tag in the new device type the same address string as the matching tag in the “from” device.
Device Type Name	Type or select the name: maximum 32 characters, do not use \ / : * ? < > |	If creating a device type, type the name. If editing a device type, the device type that was selected for editing displays here. You can change the name here.
Lock this Device Type	Check to lock device, so that it cannot be edited.	This action cannot be undone. You cannot edit a locked device type. If it is a standard device type, you cannot edit or delete it. If you added the device type, you can delete it but not edit it.
Device Category	Choose the category for this device.	To create categories, see "Print the .CSV file" on page 187 In addition to predefined categories, you can add custom categories. See "Set Up Device Type Categories" on page 188 for instructions. Categories are used in the Device Creation wizard, and are a means of shortening the list of devices you must view.
Subcategory	Choose the subcategory for this device, if needed.	Default options are Monitoring Device, PLC, or Protection Device. Depending on the device you select at the top of the page, this field is filled in for you. As with categories, subcategories are created in the "Print the .CSV file" on page 187 screen.
Driver	Select the driver for the device type.	Predefined drivers are created for all PowerLogic compatible devices, though you may need to use these drivers for multiple device types. For example, you would use the CM4000 driver for a CM3000. Use the Generic Power Device driver for third-party devices. The Controllable Device driver is currently not used. Use Generic JBus Device driver for JBus devices.

Field Name	Valid Entries	Comments
Display Associated Profiles	(Active only in Edit mode) Click to display a list of profiles that are associated with the selected device type.	This list is used when you want to delete a device type that is associated with a profile. In Edit mode, select the device type you want to delete; then click this button. Note the profile(s) associated with the device type. Before you can delete the device type, go to the Add / Edit Device Profile screen, locate each profile in the list. You cannot save an empty profile, so you must either select another device type for it or delete the profile. Then you can delete the device type.
IEC Tags	n/a	This list includes all tags that have been added to the system, standard tags as well as custom tags that you have added. Tags are listed in their groups (such as 100ms, Onboard Alarm, Power Factors).
Selected Tags	Select tags from Tags; click the right arrow to move them to this box.	You can move single tags or entire tag groups. They must be moved one at a time (cannot shift+click to select). Note: You cannot deselect tags for a device type if that device is associated with a device profile.
Add/Edit Custom Tags	Click to begin adding a custom tag.	Live when creating or editing a tag. Opens the Add/Edit Custom Tags screen. See for instructions. If you add a custom tag here, you are prompted to save the device type. After adding the tag, you have the option of adding that tag to the device type.

Edit a device type

If you want an edited version of a locked device type, you must create a new device type from it and then delete the original device type. Certain “standard” device types can be used to create new types, but they cannot be deleted. Examples: Circuit Monitor 4000, Power Meter 800, and Sepam S42.

NOTE: You cannot edit any locked device type. When a device type is locked, the locked icon displays on the Define Device Type Tags tab: 

To edit a device type:

1. Open **Add / Edit Device Type**: In Profile Editor, click **Define Device Type Tags**, then click **Add / Edit** (to the right of the Device Type Name field.)

There are two ways to edit device types:

1. In **Define Device Type Tags**, select a device type and then make the following changes:
   a. Edit the functional address (see "Edit functional addresses" on page 189).
   b. In **Real Time Tags** you can edit the address (see "Edit tag addresses" on page 190) and choose a different format.
2. In **Define Device Type Tags**, select the device type you want to edit, then click **Add / Edit**. Follow through the screens to edit additional information:
   a. In the **Device Type Options** box, click **Edit Existing**.
   b. Click the **Device Type to Edit** list to display the **Select Device** box. Select the device type that you want to edit.
   c. You can change the device type name, category, subcategory, and driver. You can also lock the device.
   d. Select tags and tag groups and move them into or out of the Selected Tags list.
   e. If a device type is associated with a device profile, you cannot deselect tags.
   f. After all of the appropriate changes are made, click **Save** to save your current settings.
   g. To create additional custom tags, click **Add / Edit Custom Tags**; otherwise, click **Save & Exit** to save your settings and close the window.

If you add a tag to a group that is already included in a device type, you must then individually add the tag to that device type.

Delete a device type

You cannot delete standard device types or custom device types that are associated with a device profile; those device types do not display in this option.

To view a list of profiles associated with a device type:

1. Switch to the Edit Existing view, then select the device type you want to delete.
2. Click List Profiles associated with this Device Type to display all associated profiles.
3. Before you can delete the device type, return to the Add/Edit Device Profile screen for each profile in the list; either select a new device type for the profile or delete the profile entirely.

To delete a device type:

1. In **Define Device Type Tags**, click **Add / Edit**.
2. In the **Device Type Options** box, click **Delete**.
3. From the drop-down list, select the device type you want to delete (the list includes only unlocked or custom device types; you cannot delete any of the “standard” device types or custom device types that are associated with a profile.
4. Click **Delete**. At the Confirm prompt, click **Yes**.

Assign tags to generic I/O points

Device types have default tags that have the appropriate formatting and addressing assigned for all the generic I/O points. It may be necessary to redefine a generic I/O point by assigning it to a tag that has a specific meaning.

Example 1: The Branch Circuit Monitor 42 has been configured to read 42 current channels. To assign channel 1 to Current A:

1. From the Branch Circuit Monitor 42 device type, choose the “Ch.01 Current tag.”
2. Note the addressing and formatting for the tag.
3. Locate and add the standard tag that you want to assign to this channel. In the example above, you would add “Current A.”
4. Edit the address of the Current A tag to match the address of Channel 1.

Example 2: If the Sepam I11 / I12 have been configured to represent circuit breaker position, you may choose to redefine the tag name:

1. From the Sepam 40 Series device type, choose tags “Input Status I11” / “Input Status I12.”
2. Note the addressing and formatting for each tag.
3. Locate and add the standard tag that you want to assign to these I/’O points. In the example above, you would add “Device Closed.”
4. Edit the address of the Device Closed tag. In order to create the “device closed” functionality, you must combine inputs 11 and 12 into an enumerated status (choose the Enumerated Status logic code for the indicated address for I11 and I12),

Create custom device types

A custom device type is any device type that is not included in the standard Power SCADA Operation set of devices. Typically, this is a third-party device type that communicates through a protocol such as IEC 61850 or DNP3. Each protocol requires a slightly different process.

The help file describes the process for each of these protocols:

- IEC 61850
- Modbus third party
- DNP3
- Composite device type

To create a new custom device type:

1. Open the Profile Editor.
2. In the **Define Device Type Tags** pane, click **Add / Edit**.
3. In **Add/Edit Device Type**, complete the information for the device, following instructions in the help file for the protocol the device uses.

Print the .CSV file

For each device type, device profile, or project, you can create a .CSV file that includes the following data:

Type of File	Data Included
Device Type	tag descriptions, IEC tag names, type, and address
Device Profile	tag descriptions, IEC tag names
Project	data profiles and custom tag names included in the project

To create the CSV file:

1. Display the device type, profile, or project for which you want the file. For example, to create a CSV file for the Sepam 42 Full device profile, select the Create Device Profiles tab and choose Sepam S42 Full from the drop-down list.
2. Click **File > Create CSV File**.
3. In the Save As window, choose a location for the file and optionally rename it. Click **Save**.

   The file is created in the location you specified.
4. View and print the file in Microsoft Excel.

Set Up Device Type Categories

Use **Set Up Device Type Categories** to add, edit, and delete categories. These categories are used in the Profile Wizard to logically group the list of profiles that display, and to make them easier to locate.

When you add device types in the Add/Edit Device Type screen, you associate a category and subcategory with each device.

To view the Set Up Device Type Categories screen, click Settings > Set Up Device Type Categories.

The following table describes the parts of this screen. Detailed instructions are after the table.

Field Name	Valid Entries	Comments
Categories Options box	Create New	Click to begin adding a new device type that is not based on an existing type.
	Edit Existing	Click to begin editing the category or subcategory name.
	Delete Existing	Click to begin deleting a category. You cannot delete a category that is associated with a device type.
	Category Name	If new: Type the name. If editing or deleting, select the name from the drop-down menu. Predefined categories do not display. Currently, there is one predefined category: Schneider Electric.
Subcategories Options box.	As with categories, you can create new, edit existing, or delete.	If new: Type the name. If editing or deleting, select the name from the drop-down menu. You cannot delete a subcategory that is associated with a device type. Predefined subcategories do not display. Currently, the predefined subcategories are: Protection Device, Monitoring Device, and PLC.

Add a category or subcategory

To add a category or subcategory:

1. Click **Create New** in the appropriate box (Categories or Subcategories).
2. In the **Name** field, type the name of the new category or subcategory.
3. Click **OK** to save the new entry and close the screen.

Edit a category or subcategory name

To change the name of a category or subcategory:

1. From the appropriate box, click **Edit Existing**.
2. From the dropdown menu, select the category or subcategory that you want to edit.
3. Type the new name for this category or subcategory.
4. Click **Save** to make the change, or click **Save & Exit** to save changes and close the screen.

Delete a category or subcategory

Predefined categories and subcategories, or those associated with a device type, do not display for deletion.

To delete a device type associated with a category or subcategory:

1. Change to the **Edit** view
2. Select the category or subcategory, then click **List Device Types**.
3. Note the device types and go to the **Add / Edit Device Types** screen.
4. Change the category or subcategory on that page.
5. Return to the **Set Up Device Type Categories** screen to delete the category/subcategory.

To delete a category or subcategory:

1. From the appropriate box, click **Delete Existing**.
2. From the dropdown menu, select the category or subcategory that you want to delete.
3. Click **Delete**.
4. Click **Yes** to confirm the deletion.
5. Click Save to save the change, or click Save & Exit to save changes and close the screen.

Edit functional addresses

Use this feature to add variables to addressing. You can re-use a variable by copying and pasting parts of it into other addresses, then making changes to the code for use in other tags. You will be prompted for these variables in the Profile Wizard.

To access the **Edit Functional Address** screen, click the **Edit Functional Address** button for a real time tag, onboard alarm tag, control tag, or reset tag. The fields on this screen are used in this way:

- **Tag Name and Original Address**: These fields display from the tag you selected; you cannot edit this information.

- **Device Variables**: Click New to begin adding new variable properties. The following fields become live:
- **Name**: This name must be in format %NNN%, where NNN includes only letters or underscores.
- **Description**: This required field is free-form. It displays in the Profile Wizard and will help you ensure that you have the correct information entered.
- **Regular Expression**: You can use one of the pre-defined expressions, or you can create your own
- **Test Value**: This will become the default in Citect; use it here for testing the new address.
- **Help**: Use this optional field to add more definition to this address. It displays in the Profile Wizard.
- **Code Body**: Enter the code in C# to define the action you want to take place.
- **Return**: Type the return statement that you want from C# code. It might look like:

```
string.FormatFormat("SomeString{0}SomeOtherString", someVariable)
```

- **Result**: Click Test in the lower right corner of the screen. If there is a compile error, check your C# code. Otherwise, the result displays. Verify that it is what you wanted.

Create custom tags

Power SCADA Operation comes with most of the tags that are needed for each device type. However, you can create custom tags to assign to device types and device profiles. A *custom tag* is a unique measurement that is assigned to a device type, or is an existing tag for which the tag address is changed. You can also edit address attributes for any tag.

NOTE: To avoid potential communication errors, use the Profile Editor to create all custom tags that will communicate with equipment.

To create a custom tag:

1. In Profile Editor > **Define Device Type Tags** pane, click **Add/Edit** and then click **Add/Edit Custom Tags**.
2. Enter the information for the new tag.

TIP: On the Add / Edit Custom Tags screen, click the help link (?) at the top right of the screen. The help leads you through adding, editing, or deleting custom tags.

For more information on adding custom tags, see:

- "Setting up custom tags" on page 196
- "Edit generic tag addresses" on page 196

Edit tag addresses

Use the Edit Address screen to edit the attributes of a single tag address. If a device type is locked, you cannot edit any of its tag addresses; they will be grayed out. A thorough discussion of IEC 61850 tags and their construction is included in "About tags" on page 200 and "About logic codes" on page 206.

NOTE: Case and order are critical in the tag address. Be careful to observe the exact address order. For address order, see "About logic codes" on page 206. Also, be sure you use the correct case. For example, use M for register numbers in hexadecimal, and use m for register numbers in decimal.

To view the Edit Address screen:

1. In the Profile Editor, click **Define Device Type Tags**.
2. Choose the device type, then click the Edit... field for the tag that you want to change.

The Edit Address screen is different for real-time and alarm tags.

Each type of tag (real-time, onboard alarm, reset, and control) is described separately in the following tables.

Real-time tag addresses

The following table describes the fields of the Edit Address screen for real-time tags.

Field Name	Entry	Comments
Data Type	For display only	You can edit this field in the Add/Edit Custom Tags screen.
Priority	High, Normal, or Low Logic Code:	You can edit this field either here or in the Add/Edit Tag screen.
Logic Code	Select the logic code for this tag.	The logic code list depends on the Data Type for this tag. For more information about logic codes, see "About logic codes" on page 206.
Display Registers in:	hexadecimal/decimal	Click the radio button for the way you want to view register information.
Module	Select module	Choose the type of module in which the tag is used. Used for Micrologic at this time.
Register Type	Select register type	Select the type of register that is to be written or read.
Number of Registers	Select the total number of registers for this address (1-10). Is Consecutive, check if the registers are to be consecutive (determined in the logic code).	Enables for editing the appropriate registers in the lines below.

Field Name	Entry	Comments
Fixed Scale/Register Scale	Click the radio button for the correct type of scale.	A fixed scale is the actual value of the scale. A register scale is the register address where the scale is held. The value will be scaled in this manner: Value x 10x where X = the scale. Scales can only be –10 to 10.
Conversion Factor	Enter the multiplier to convert the base units to the desired conversion.	Conversion factors are used for straight multiplication with the value. The conversion factor could also be changed in the Add/Edit Units screen (Settings > Select Units > Add/Edit Units. Conversion factors take this form: ####E##. For example, 123E-2 becomes 123x10-2 which becomes 1.23.
Offset	y = ,x + b	y = the final value reported by PLSCADA b = the offset m = the conversion factor x = the original value in the meter b = rarely used, mainly in temperature conversion The offset is added to the final value (after the conversion factor is applied).
Register 1-4	Enter the register number.	Be aware of whether you chose hexadecimal or decimal. Use the same format here.
Bitmask for Register 1-4	For digital input/output tags: Set the bits to 1 or 0 to match the pattern for “True” in the device register.	When all bits match exactly the pattern in the register, the status is True. When any one bit does not match the pattern in the register, the status is False. Note: On PM8s and CM4s, there is a device-specific format, DIgIn and DigOut. In each case, you must first specify the indicator register (which becomes the first register). The second register will have the mask.
Invert Result	Check this box to invert.	Will turn False to True or vice versa; typically used for Normally Open or Normally Closed.

Onboard alarm tag addresses

The following table describes the fields of the *Edit Address* screen for onboard alarm tags.

Field Name	Entry	Comments
Tag Name	For display only	This is the tag name, which cannot be changed.
File Number	Select the number.	This is the file number for the alarm file on the device.(Sepam has no file number; enter 0.)
Module	Select the module.	Choose the type of module in which the tag is used. Used for Micrologic at this time.
Unique ID	Choose the identifier.	This unique identifier must be used to ensure that alarms will annunciate correctly. For CM4, PM8, PM5000, and Micrologic, the unique ID must be decimal. For Sepam, the unique ID is the coil bit address that indicates the alarm; it must be in hexadecimal.
Hexadecimal	check box	Check this box if you want to display the ID in hexadecimal, rather than decimal.
Has Unique Sub ID	check box	Check if this tag has a unique sub-identifier (Micrologic, CM4000, PM800, and PM5000 devices).
Unique Sub ID	Enter the Sub ID.	Enter the unique sub-identifier. Active only if Unique Sub ID box is checked.

Reset tag addresses

NOTE: Once the tag is set up, writing a 1 to the tag will cause the “write” to occur.

Standard device types include some pre-defined resets. These pre-defined commands cause proprietary functions within the device. Do not edit these commands.

To add a custom reset that will operate by writing to a register, do the following:

1. From the **Add/Edit Custom Tags** screen, set the **Group** to **Resets** and the **Data Type** as **Digital**.
2. Save the tag.
3. Add the new tag(s) to the appropriate device type(s).
4. From the **Define Device Type Tags** tab, locate the tag and click **Edit**.

The following table describes the fields of the Edit Address screen for reset tags.

Box Name	Field Name	Comments
Tag Information	Command Type	The Command Type and Command to Edit are already selected.
	Command to Edit	

Box Name	Field Name	Comments
Data Information box	Data Type: for display only	You can edit this field in the Add/Edit Custom Tags screen.
	Priority: High (default)	Cannot be edited.
	Logic Code: Select the logic code for this tag.	Choose the appropriate logic code for this tag. See "About logic codes" on page 206.
Device Information box	Display Registers in: hexadecimal/decimal	Click the radio button for the way you want to view register information.
	Module	Choose the type of module in which the tag is used. Used for Micrologic at this time.
	Register Type	Select the type of register that is to be written or read.
Number of Registers	There is only one register for this address.	Enables for editing the appropriate registers in the lines below.
Fixed Scale/Register Scale	n/a	Not used for digital logic codes.
Conversion Factor	n/a	Not used for digital logic codes.
Register 1	Enter the register number.	Be aware of whether you chose hexadecimal or decimal. Use the same format here.
Bitmask for Register 1	For digital input/output tags: Set the bits to 1 or 0 to match the pattern for “True” in the device register.	When all bits match exactly the pattern in the register, the status is True. When any one bit does not match the pattern in the register, the status is False. Note: On PM8s and CM4s, there is a device-specific format, DIgIn and DigOut. In each case, you must first specific the indicator register (which becomes the first register). The second register will have the mask.
Invert Result	n/a	Not used for resets.

Control tag addresses

NOTE: For a control with one command, once the tag is set up, writing a 1 to the tag will cause the “write” to occur. For a control with two commands that is either static or normally open, writing a 1 to the tag will cause the first command (ON) to occur; writing a 0 will cause the second (OFF) to occur. For a control with two commands that is normally closed, writing a 1 to the tag will cause the second command (OFF) to occur; writing a 0 will cause the first command (ON) to occur.

Standard device types include some pre-defined controls. For example, Operate (ENERGIZE). These pre-defined commands cause proprietary functions within the device. Do not edit these commands.

To add a custom control that will operate by writing to a register, do the following:

1. From the **Add/Edit Custom Tags** screen, set the **Group** to **Controls** and the **Data Type** as **Digital**.
2. Save the tag.
3. Add the new tag(s) to the appropriate device type(s).
4. From the **Define Device Type Tags** tab, locate the tag and click **Edit**.

The following table describes the fields of the Edit Address screen for control tags.

Box Name	Field Name	Comments
Tag Information	Command Type	For commands that have an opposite (such as On and Off), choose Normally Open/Normally Closed or Static with Off Command. For commands with only one action, choose Static without Off Command.
	Command to Edit	If you are editing a command with two parts, use the Command to Edit drop-down menu to select the On Command.
Data Information box	Data Type: for display only	You can edit this field in the Add/Edit Custom Tags screen.
	Logic Code: Select the logic code for this tag.	Choose the appropriate logic code for this tag. See "About logic codes" on page 206.
Device Information box	Display Registers in: hexadecimal/decimal	Click the radio button for the way you want to view register information.
	Module	Choose the type of module in which the tag is used. Used for Micrologic at this time.
	Register Type	Select the type of register that is to be written or read.
Number of Registers (1)	n/a	Enables for editing the appropriate registers in the lines below.
Fixed Scale/Register Scale	Click the radio button for the correct type of scale.	A fixed scale is the actual value of the scale. A register scale is the register address where the scale is held. The value will be scaled in this manner: Value x 10x where X = the scale. Scales can only be –10 to 10.
Conversion Factor	n/a	Not used for digital controls.
Register 1	Enter the register number.	Be aware of whether you chose hexadecimal or decimal. Use the same format here.

Box Name	Field Name	Comments
Bitmask for Register 1	For digital input/output tags: Set the bits to 1 or 0 to match the pattern for "True" in the device register.	When all bits match exactly the pattern in the register, the status is True. When any one bit does not match the pattern in the register, the status is False. Note: On PM8s and CM4s, there is a device-specific forma, DIgIn and DigOut. In each case, you must first specific the indicator register (which becomes the first register). The second register will have the mask.
Invert Result	n/a	Not used for digital controls.

Edit address information

To edit address information for a real-time tag:

1. From the **Define Device Type Tags** tab, choose a device type (cannot be locked). From the **Real Time Tags** sub-tab, highlight the tag whose address you want to edit.
2. In the **Edit Address** column, click **Edit** for the address you want to edit.
3. The Edit Address screen displays.
4. You can change any of the tag address attributes. See the preceding table for descriptions of each field.
5. Click **OK** to save changes and close the screen.

Add a new tag address

You can also add a tag address, when none exists. As with editing addresses, click the **Edit Address** column for a tag; then follow instructions in the table above.

Edit generic tag addresses

This window displays when you click **Edit** for an address of a non-PowerLogic compatible device type, such IEC 61850 or DNP3.

The variable tag properties used in this screen are described in a topic in the Citect SCADA help file. For detailed information, see **Add a Variable Tag** in the Citect SCADA 2018 help file: ...\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin\Help\Citect SCADA

Setting up custom tags

Use the Add / Edit Custom Tags window to create, edit, and delete custom tags.

To create custom tags:

1. Open the Add / Edit Custom Tags window using one of the following methods:
   - At the bottom of the **Add / Edit Device Type** window, click **Add / Edit Custom Tags**.
   - In Profile Editor, click **Settings > Set Up Custom Tags**.
2. Set up the custom tag using the Add / Edit Custom Tag fields.

The following table describes the Add / Edit Custom Tag fields.

NOTE: See "Edit a custom tag" on page 199 and "Delete a custom tag" on page 200 for instructions on how to edit or delete custom tags.

Field Name	Valid Entries	Comments
Custom Tag Options	Create New	Click to begin adding a new tag.
	Create From	Click to begin adding a new tag that is based on an existing custom tag. For example, you might want to change metadata for another custom tag.
	Edit Existing	Click to edit the attributes of an existing tag.
	Delete Existing	Click to delete a tag (tag cannot be associated with a device type).
	Tag to Create From Tag to Edit Tag to Delete	From the drop-down menu, select the tag you want to create from, edit or delete.
	Delete button	Live only when Delete Existing is selected. Click to delete the tag. You can only delete custom tags not associated with a device type.
Display Associated Device Types	Click to display device types that are associated with this tag.	Live only when in Edit mode. Click to list device types that are associated with this custom tag. Note the device types so that you can delete the tag from them (in the Add/Edit Device Type screen) before you delete the tag. See "Delete a custom tag" on page 200 for instructions on using this button.
Tag Name	Type the new tag name; or type the changed name for a tag you are editing.	Maximum 32 characters; can include any alpha or numeric character, as well underscore (_) and backslash (\). Must begin with either an alpha character of underscore.
Display Name	Type the name that you want to display when selecting the tag and in other displays.	You can use this field for additional information on the Add/Edit Custom Tags screen. For example, you could describe the data that it logs. It does not display anywhere else in the system.
Group	Select the group.	Includes all the real-time groups (such as 100ms, controls, currents) plus onboard alarms, resets, and controls.

Field Name	Valid Entries	Comments
Data Type	Select the data type.	These are Power SCADA Operation tag data types. They affect the logic codes that are available for display in the Edit Address screen. See "About logic codes" on page 206 for the data type that matches each logic code.
Eng. Units	Select the base unit.	These are the base engineering units for tags; the values come from Engineering Unit Setup.
Ignore Unit Conversion	Check to cause the system to ignore any conversions that were added for this tag.	Causes reporting to be according to the base unit, rather than the conversion that was chosen for this tag in the template that is being used.
Add Eng Unit	Click to open the Add/Edit Units screen, to add a new engineering unit and/or conversion.	Provides a quicker means of adding an engineering unit that had been overlooked.
Citect Format	Select the numerical format.	This is used for display purposes in Power SCADA Operation graphics pages. It determines where the decimal displays. Choose the reporting format, to be used in Power SCADA Operation, from ## to #0.#####. For example, if you select #.##, the number 8.12579 would be displayed as 8.12.
Polling Priority	Low, Normal, or High	Indicates the level of priority Power SCADA Operation uses when reading data from devices. Note: In the address field, a priority of 1 = High, 2 = Medium, 3 = Low.
Alarm On Text	For onboard alarms only: enter the text for when the alarm is On.	This text displays on the Create Device Profiles tab for the onboard alarm tag, when it is selected for the device type in the profile. It also displays in the Alarm Log.
Alarm Off Text	For onboard alarms only: enter the text for when the alarm is Off.	

Field Name	Valid Entries	Comments
Display 'Advanced' filter selections	Check to display additional filter options in the Real Time Filter and Alarm Filter tabs	Displays several additional filter options on the two "Filter" tabs. These options will be useful in the future for reporting purposes.
You can include additional filters for either real time filters or alarm filters. Though not currently used, these filters will provide metadata for later reporting. Standard tags have some of these filters selected. A typical usage for these filters might be: when creating a custom tag from an already existing standard tag, you can create matching metadata by using the filters that have been built in to the standard tag.		
Real Time Filters tab (dropdown lists are expanded when "Display 'Advanced' filter selections" is checked)		
Category Type	Select a category for this tag.	This field provides metadata about the tag. It will be used in future reports.
Utility Type	Select a utility type.	Metadata for future use in reporting.
Statistical Type	Select a statistical type.	Metadata for future use in statistical reporting.
Quantity	Select a quantity.	Metadata for future use in statistical reporting.
Alarm Filters tab (dropdown lists are expanded when "Display 'Advanced' filter selections" is checked)		
Categorization	Select the alarm category	Used for filtering and sorting alarm data, and metadata for future use in statistical reporting.
Alarm Type	Select the alarm type.	Used for filtering and sorting alarm data, and metadata for future use in statistical reporting.
Alarm Group	Select the group.	Used for filtering and sorting alarm data, and metadata for future use in statistical reporting.
Subcategorization	Select a subcategory.	Used for filtering and sorting alarm data, and metadata for future use in statistical reporting.
Alarm Level	Select the severity level of the alarm.	Used for filtering and sorting alarm data, and metadata for future use in statistical reporting.

Edit a custom tag

You can edit any custom tag.

To edit a tag:

1. Open the **Add / Edit Custom Tags** screen: from the **Add / Edit Device Type** screen, click **Add / Edit Custom Tags**.
2. In the **Custom Tag Options** box, click **Edit Existing**.

3. You can change any of the tag attributes. (This does not change the tag's assignment status; if it is selected for a device type, it does not move back to the IEC Tags list.)
4. Click **Save** to save changes, or click **Save & Exit** to save changes and close the screen.

Delete a custom tag

You can delete any custom tag that is not associated with a device type.

1. If the tag is associated with a device type, you must first deselect the tag:
2. Change the option to **Edit Existing** and display the tag you want to delete.
3. Click **Display Associated Device Types** to display all device types that include this tag. Make a note of the device types.
4. Return to the **Add/Edit Device Type** screen. For each device type listed, deselect the tag that you want to delete.

Continue deleting the tag:

1. Open the **Add/Edit Custom Tags** screen.
2. In the **Custom Tag Options** box, click **Delete Existing**.
3. From the drop-down menu, choose the tag you want to delete.
4. Click **Delete**.
5. Click **Yes** to confirm the deletion.
6. Click **Save** to save the change, or click **Save & Exit** to save changes and close the screen.

About tags

Power SCADA Operation includes a variety of tag types: real-time, alarm, and trend. Most of the tags that you will need are already added. However, you can add custom tags to suit special needs. This section describes how tags are constructed and provides further specific information about the construction of format codes, logic codes, and addresses.

The Power SCADA Operation tag naming convention follows the IEC 61850 standard. IEC 61850 tags are flexible, which allows them to specify how functions are implemented in devices. The IEC 61850 tag was developed for medium-voltage and high-voltage applications, such as monitoring, control, and substation automation.

Some of our devices include data and functionality that are not yet covered by IEC 61850. For these devices, the general IEC 61850 formatting was followed when creating tags.

If you are writing Cicode (see "Customize a project using Cicode" on page 371). You will need to know the IEC 61850 tag name that you added to the device profile for that device. You can print the CSV file to view tag names (see "Print the .CSV file" on page 187). Apart from that, you would only need to add tags if you are installing a third-party device that is not standard to Power SCADA Operation. If you do need to add tags, create any category you wish, and follow the format shown below.

For detailed information on tag naming, see "Tag naming convention" on page 201.

Tag naming convention

Tag names cannot exceed 79 characters. Use a backslash as a separator between tag parts. Tags are constructed in this manner:

`EquipmentName\Logical_Node\Data Object\Data Attribute` (may have more than one)

For detailed information on tag syntax, see **Tag Name Syntax** in Citect SCADA Help.

The following table lists the main categories for the common IEC 61850 logical nodes. After the table, the most commonly used category (Mxxx: metering and measurement) is described.

Category Name	Description
Axxx	automatic control; e.g., ATCC (tap changer), AVCO (voltage control)
Cxxx	supervisory control; e.g., CILO (interlocking), CSWI (switch control)
Gxxx	generic functions; e.g., GGIO (generic I/O)
Ixxx	interfacing/archiving; e.g., IARC (archive), IHMI (HMI)
Lxxx	system logical nodes; e.g., LLNO (common), LPHD (physical device)
Mxxx	metering and measurement; e.g., MMXU (measurement), MMTR (metering), MSTA (metering statistics), MSQI (sequence and imbalance), MHAI (harmonics and interharmonics)
Pxxx	protection; e.g., PDIF (differential), PIOC (instantaneous overcurrent or rate of rise.), PDIS (distance), PTOV (time-overvoltage)
Rxxx	protection related; e.g., RREC (auto reclosing), RDRE (disturbance)
Sxxx	sensors, monitoring; e.g., SARC (arcs), SPDC (partial discharge)
Txxx	instrument transformer; e.g., TCTR (current), TVTR (voltage)
Xxxx	switchgear; e.g., XCBR (circuit breaker), XCSW (switch)
Zxxx	other equipment; e.g., ZCAP (cap control), ZMOT (motor)

The following example illustrates the IEC 61850 tag for current A:

`EquipmentName\MMXU1\A\PhsA`

where:

`M` = the category

`MXU` = measurement of currents, voltages, power, and impedances

`1` = the instance (there could be multiple MMXU tags)

`A` = the data object, current

`PhsA` = the attribute that further defines the data object, phase A

All of the tags that are currently used in the system can be viewed from the Profile Editor > **Define Device Type Tags** tab. Click **Settings > Display Advanced Properties** to display the full tag names.

Define an enumeration

An *enumeration* is a single value (0-15) that is used to define a condition that is determined by multiple-bit input. You will add enumerations to handle scenarios that are more complicated than simply true-false, to allow for dynamic contingencies. For example, when you need to use multiple bits to describe the position of a circuit breaker, you might do the following:

Bit *y* (closed) | Bit x (open). Note that the least significant bit is register 1.

Bit x | Bit y	Status	Circuit Breaker Position	Returned Value
0 | 0	Indeterminate	Circuit breaker is neither open nor closed	0
0 | 1	Open	Circuit breaker is open.	1
1 | 0	Closed	Circuit breaker is closed.	2
1 | 1	Error	Circuit breaker is reporting both open and closed condition. Possible device/wiring error	3

Using the enumerated status, we place the register and bitmask for the open position in register 1 (least significant) and the register and bitmask for the closed position in register 2 (most significant).

Use special tags to control circuit breaker status

When you want to include a device that does not have a pre-defined device profile (such as a third-party circuit breaker), you must identify the registers that the device uses for the operations you want, then choose the correct tags and tag addresses to write to these registers. Finally, when creating the one-line on the graphics page, you will choose the appropriate genie:

1. Determine the device registers used for the open and close operations on the circuit breaker.
2. In the Profile Editor, choose the tag needed for each operation.
3. Ensure that tag address references the correct action and register(s). See "Edit tag addresses" on page 190 for instructions on editing the address,
4. When adding a genie for the circuit breaker on the graphics page, choose from the default library (see "Default Genie Library" on page 629), or create a custom genie (see "Create a new genie" on page 285).

Format code definitions

The address field is part of the tag. It includes a variety of attributes, some of which are required, and some optional. The following tables list the attributes, whether they are required, and their possible modifiers. All parts of a tag are case sensitive. The order of the fields is fixed; and all fields are separated by semi-colons. See "About logic codes" on page 206 for templates of constructed tags.

Real-Time Format Code Definitions

Attributes	Modifiers	Comments
T (type) Required	SS = single status	
	DS = double status enumeration	
	ST = string	
	UT = UTC time	
	MV = measured value (float)	
	CM = complex measured value (float)	Temporarily, this may return a string; when Power SCADA Operation is upgraded to handle large integers, this will change.
	BC = binary counter (integer)	
D (module—Micrologic devices)	B = BCM	
	P = PM	
	M = MM	
	C = CCM	

Attributes	Modifiers	Comments
M/m/S/s/C/c/I/i (register type)	M = holding registers in hexadecimal m = holding registers in decimal S = input coil (status register) in hexadecimal s = input coil (status register) in decimal C = output coil (writable only) in hexadecimal c = output coil (writable only) in decimal I = input register (read only) in hexadecimal i = input register (read only) in decimal	
Register Number Modifiers (register number from 1–4)	u## = ## registers are unsigned, ## is a decimal s## = ## registers are signed; ## is a decimal	After the modifier, there may be a number indicating scaling factor. See “V,” below in this table. Used for conversion to SI units, this number will be: RegisterValue x scale For SS and DS: there must be a 1U default; the modifier will be a bitmask: - The mask must use hex only, 16 bits/register - Attach the ones, then the zero mask, to the register; if you only have ones masks, just attach them - Only one register cases can be inverted. Add **:I** after the masks for inversion.
N (scale)	numerical entries; range is -10 to 10	N defines a constant scale; the logic code knows how to use it.

Attributes	Modifiers	Comments
R (scale register)	the register number in decimal	R defines the holding register where the scale is held; the logic code knows how to use it.
E (priority)	single digit: 1, 2, or 3; default 2 is used if this is not included (1 = high, 2 = normal, 3 = low)	Defines the priority Power SCADA Operation uses in processing data.
V (conversion factor)	Use scientific notation without the decimal.	Examples: 354E-3 = 0.354 354E1 = 3540 Will be multiplied before the value is returned.
L:P (logic code) Required	The number that is used comes from the Logic Codes table.	L:P is the logic code for PowerLogic. Other codes may follow, such a L:I for ION. For logic code descriptions, see "About logic codes" on page 206

Alarm format code definitions

Attributes	Modifiers	Comments
T (type) Required	ALM = alarm	
D (module—optional for Micrologic devices)	B = BCM P = PM M = MM C = CCM	BCM is straight addressing, and therefore, optional.
F (file) Required	File number will be in decimal, up to 5 digits	
Q (unique ID) Required	Unique ID will be in decimal.	This number can be huge.

Control format code definitions rules of operation

These rules are true for predefined and custom codes:

Address structure	Result
C:N;(action)	If 1, perform action. If 0, undefined.
C:N;(action1);(action2) C:NO;(action1);(action2)	If 1, perform action1. If 0, perform action 2.

Address structure	Result
C:NC;(action1);(action2)	If 1, perform action2. If 0, perform action1

Predefined control format codes

Attributes	Modifiers	Comments
C (command) Required	NO = normally open NC = normally closed N = normal operation	Normal operation does not have a closed/open status.
OPERATE (command word)	n/a	Two required for NO and NC.

Predefined reset format codes

Attributes	Modifiers	Comments
Reset (command word)	n/a	Entering a one to this tag causes the reset to take place.

Custom control and reset format codes

Attributes	Modifiers	Comments
C (command) Required	NO = normally open NC = normal closed N = normal operation	Normal operation does not have a closed/open status.
Followed by one or two entire “write” addresses; used only for logic codes 101, 102, 103. For logic code descriptions, see "About logic codes" on page 206. Write Address format: T:SS;m:##:#;L.:P:101 Example: C:NO;T:SS;m:1234:1;L:P101;T:SS;m:3456:1;L:P101		

About logic codes

Logic codes tell Power SCADA Operation how to mathematically operate on the values in device registers to give users the desired values. For detailed information on each logic code and its related information, see "Logic code definitions" on page 600.

Block writes

Block writes represent blocks of registers that are updated in a single write operation. There are two types of block writes:

- Fixed: fully specified and compiled before run time. Writing the value of '1' to such a variable tag causes the specified fixed values to be written to the specified registers.
- Variable: specified on the fly. The registers and the values to be written are not fixed; they are specified during run time by the user.

Fixed block writes have the following format:

`T:BWF;[D:{B|C|M|P};]S:<start_register>,<values>`

where

`B, C, M,` or `P` are applicable only to Micrologic devices (otherwise the D: section is omitted) and is the module (manager) identifier (Circuit Breaker, Chassis, Metering, Protection).

`<start_register>` is the first register number for a contiguous block of registers.

`<values>` is a comma-separated list of up to 10 values that will be written to the registers starting from <start_register>.

For example:

`T:BWF;S:100,1,2,3,4,-5`

Variable block writes have the following format:

where

`B,` or `C,` or `M,` or `P` is applicable only to Micrologic devices (otherwise the D: section is omitted altogether) and is the module (manager) identifier (Circuit Breaker, Chassis, Metering, Protection)

For example:

`T:BWV;`

The start register and the values to be written follow exactly the same rules and syntax as the definition for the Fixed Block Write, however, these are specified at the time the write operation is performed. For example, specifying "`S:100,1,2,3,4,-5`" as the write value for the tag "`T:BWV;`" would write values 1,2,3,4, and -5 to the registers 100, 101, 102, 103, and 104.

How do drivers work?

For each unique tag request made, the I/O server adds one point to the point count. Tag subscriptions are limited based on the point count in the license. Exceeding the subscribed point count will ultimately cause the I/O server to shut down.

Two subscription types

There are two subscription types one used between the graphics level and I/O Server, and one for polling devices and cache refreshing. The subscription between drivers and polling devices does not increase point count. Only the subscription that begins at a client system and ends up in the I/O server will increase point count. Via this subscription, requests are sent to the drivers with value changes propagating all the way back to the client system. The client system could be the display client, alarm server, trend server and so on. What a driver then chooses to do with the requests—in terms of coupling this to a physical request to a field device—can differ, depending on the protocol. Some simple protocols propagate the request straight through to the field device; others have their own polling scheme to the field device and merely service the driver requests from a cache.

Subscription expirations

If a tag is no longer being read, the cache refreshes in this manner: Graphics client subscriptions are immediately unsubscribed when the graphics page is closed. Although most drivers release subscriptions if no client is requesting them, the I/O Server is capable of background polling (configurable on a per-device basis). These tag subscriptions are not released, and the driver still polls them. However, they are not counted anywhere, because nothing is consuming the data for those tags on the I/O Server. On the other hand, once a subscription goes against the point count, it remains in the count as long as the project is running.

Expiration is immediate if no clients are subscribed to the tag. An "expiration time-out value" is not configurable.

Create Device Profiles

Use the **Create Device Profiles** screens to view and edit profiles for individual devices. Profiles are predefined for the standard devices; you will mostly use this feature to add third-party device profiles.

After device types are added to the project, use the **Create Device Profiles** windows to view and edit profiles for individual devices. Because profiles are defined for the standard devices, use this feature to add third-party device profiles. On these windows, you can make changes to a standard device type, and then save the device as a profile that is included in your project.

Before you create profiles, you need to be sure that all of the tags and device types that you need are created (see "Define Device Type Tags" on page 178). Also make sure that you have added any new units or conversions and device type categories and subcategories that are needed.

Create Device Profiles tab

The **Create Device Profiles** tab displays all of the tags that are included in each device type profile. It is the starting point for creating/editing device profiles for individual devices. Most of the data on this screen displays for information only; however, to enable waveforms, you need to check the Waveform box (see "Enable Waveforms" on page 210 for more information).

The following table describes the fields on this tab. The tags listed assume that **Advanced Properties** has been checked. Not all elements appear on every sub-tab. Detailed instructions are after the table.

Field Name	Valid Entry	Comments
Tag Groups (left-hand pane)	Click a group to display the groups of tags that have been selected for the chosen device profile.	To associate tags and tag groups with a device type (thus creating a device profile), click Add/Edit.
Device Profile	Choose the device for which you want to view profile details.	Device Profiles are created on the Add/Edit Device Profile screen (click Add/Edit).
Add/Edit button	Click to display the Add/Edit Device Profile screen.	Use that screen to add device profiles and to associate PC-based alarms and trends.

Field Name	Valid Entry	Comments
Tag type sub-tabs	Click to display the selected tags for each type of tag: real-time, trend, PC-based alarm, onboard alarm, control, or reset.	Organized according to tag groups.
Tag Description	n/a	This is the tag name used when adding the tag.
IEC Tag Name	n/a	This is the IEC 61850-compatible name created when the tag was added.
Waveform (Onboard Alarm)	Check this box as part of the process of enabling waveform viewing.	You must also set up the alarm and waveform capture in the onboard files of the device. Waveforms will then be viewable in the runtime environment.
Category Type (Real Time)	n/a	These are real-time filters. They provide metadata to be used in future reporting.
Utility Type (Real Time)	n/a	
Statistical Type (Real Time)	n/a	
Quantity (Real Time)	n/a	
Categorization (PC Based and Onboard Alarm)	n/a	These are alarm filters. They can be used for filtering and sorting alarm data in the runtime environment. They also provide metadata to be used in future reporting.
Subcategorization (PC Based and Onboard Alarm)	n/a	
Alarm Type (PC Based and Onboard Alarm)	n/a	
Alarm Group (PC Based and Onboard Alarm)	n/a	
Alarm Level (PC Based and Onboard Alarm)	From the drop-down list, you can edit the alarm level.	If PC-based and/or onboard alarms are set for this profile, you can change their levels here.

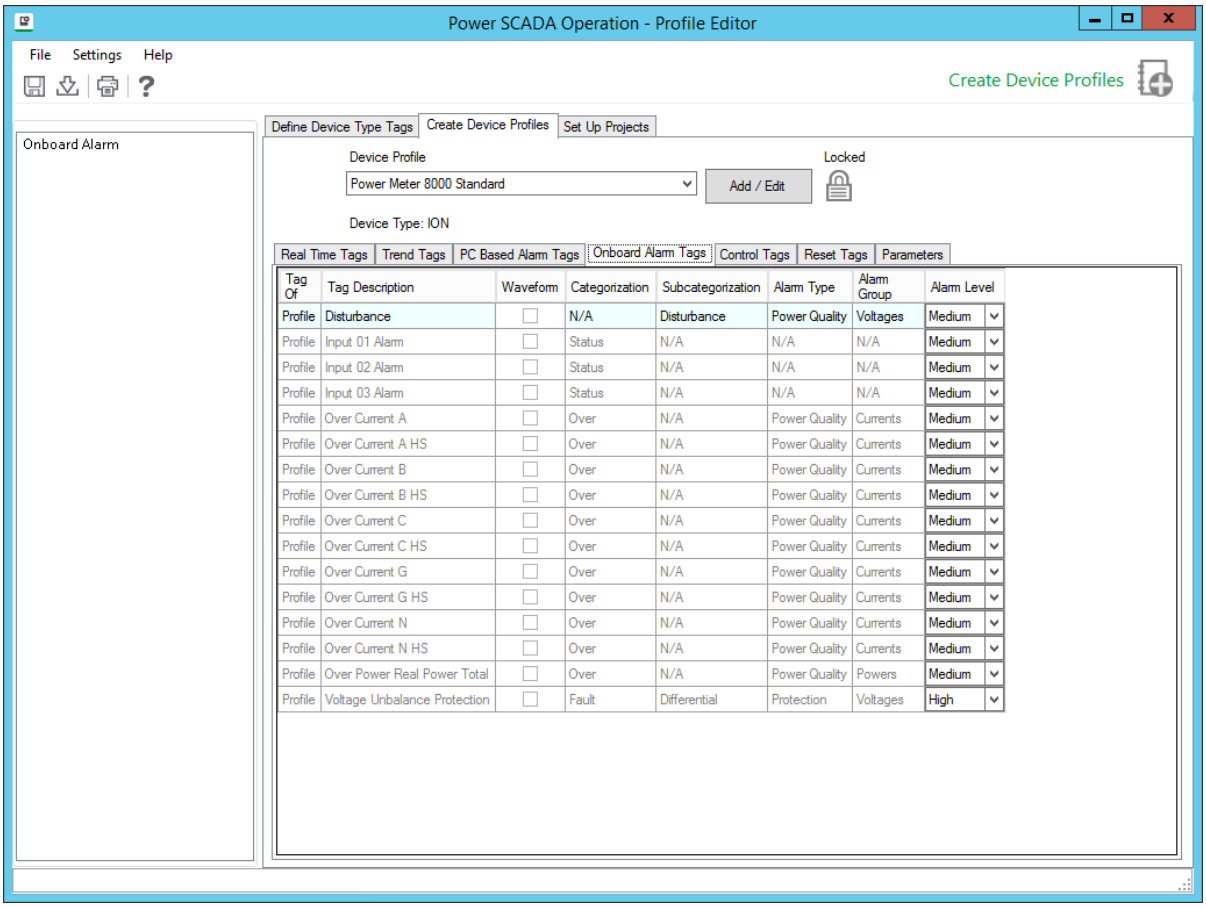
To view profile information:

1. Select the device profile from the drop-down menu.
2. Use the tag tabs (such as real-time, trend) to view the tag groups included in this device profile.

To begin adding, editing, or deleting a profile, click **Add/Edit**.

Enable Waveforms

On the **Create Device Profiles** tab, in the **Onboard Alarm Tags** sub-tab, there is a **Waveform** check box. Check the box for each alarm tag for which you want to be able to view waveforms. On the device, the alarm must also be set up for the waveform to be captured on event and stored in one of the device's data logs.



To acquire waveforms for Sepam, use the CET manual. For PowerLogic devices, refer to the PMCU help file.

As device information is polled and received by Power SCADA Operation, the waveform becomes available for viewing. See *The Alarm Log* in "View the Alarms/Events Page" on page 470 for information on viewing waveforms in the Power SCADA Runtime.

Enable waveforms for onboard alarms

Enabling waveforms for onboard alarms makes them available for viewing in the Power SCADA Runtime.

When an onboard alarm occurs at the device, the waveform is captured. The are transmitted to Power SCADA Operation and are available for viewing. The amount of time this takes depends on the number of I/O Servers you have and the number of serial devices on a chain. On a very large system with numerous serial devices, this could take as long as an hour.

To enable waveforms for onboard alarms:

1. At the device, or via the meter configuration software (PMCU), add the alarm and enable the automatic capture of a waveform when the alarm occurs.
2. In the Profile Editor, on the **Create Device Profiles** tab, for the same alarm you added in PMCU, check the **Waveform** box.

You can view the waveform from the Alarm Log in the runtime environment.

Add an onboard alarm tag

When a device onboard alarm has not been included in Power SCADA Operation, you can add it using Profile Editor. You need to follow these steps to include the device's unique identifier. Otherwise, the alarm will not annunciate in the Graphics page.

You can only add onboard alarms for devices using the CM4, PM8, Micrologic, or Sepam drivers. CM4, PM8, and Micrologic unique IDs must be decimal; SEPAM unique IDs must be hexadecimal.

To add an onboard alarm tag:

1. From the device, obtain the following information:
   a. The unique identifier for this alarm. Additionally, for MicroLogic, you need to include the unique sub-identifier.
   b. The file number in which alarms on stored on the device.
2. From the Profile Editor, add the onboard alarm.

Add edit or delete device profile

Use the **Add/Edit Device Profile** screen to add device profiles to the system. To view this screen, go to the Create Device Profiles tab.

Add a device profile

1. Open the Create Device Profiles tab: from the **Create Device Profiles** tab, click **Add / Edit**.
2. In the **Profile Options** box, click **Create New** or **Create From**.

   If you are creating from another device profile, choose it from the Device Profile to Create From drop-down menu.
3. Click **Next** to make the name and description fields live.
4. Type a unique **Device Profile Name** using a maximum of 32 characters; do not use \ / : * ? < > |
5. If you want to lock this profile, preventing anyone from editing it, check the **Lock this Device Profile** box. This action cannot be undone. If you wish to edit a locked profile, you must use the Create From option to add a new one, then delete the locked one.
6. (Optional) Type a device description. This will display as a tool tip in later screens.
7. Click **Next** again to make the remaining fields live.

8. From the **Available Devices** list, highlight the first device or device group (Protection, Monitoring, Composite) to be included in this profile. Click the **right arrow** button to move it to the **Selected Devices** box. You must select and move devices or device groups one at a time (no shift+click to select multiples).
9. If you will want to import this project into another instance of the Profile Editor, see "Add project parameters" on page 230.
10. When you have all of the devices you want, click **Next**.
11. From the **Device Type Tags** list on the left, select the tags you want to include in this profile. You can select entire tag groups or individual tags from a group; but you must select them one at a time.
12. After each addition, the tag or tag group displays in the Selected tags box. You can override any tag name (typically for generic I/O devices with multiple tags, such as inputs, for which names alone would not be intuitive in runtime. To override a tag, select it, then click Override Tag Name. Choose the tag you want. Click **OK**. The new tag will correctly display the value of the original tag, but will take the appearance of the override tag (such as description, metadata).
13. The final column, **Is Device Tag**, displays only for composite devices. Check this box to tie a tag back to its actual physical device. For example, if the same tag is in three devices, and you set PC-based alarms for each device, you need to be able to determine which device has a problem in runtime. To prevent confusion, check Is Device Tag to cause Power SCADA Operation to report the tag for its physical device, rather than the composite device.
14. When you have selected all tags, click **Next**.

    NOTE: If you have duplicate tags from multiple devices, you need to resolve this by using an override for one of the tags.

15. On the next page, choose whether each tag will have a PC-based alarm and/or trend associated with it. Click **Finish**.

    When the project is added to the project, PC based alarms are added to the Analog Alarms or Digital Alarms file. When the project is added to the project, historical trends are added to the Trend Tags file. Logging will automatically begin when the tag is added to the project.

    By default, there are two different intervals for scanning trend tags. All selected tags are scanned every 15 minutes with FIFO storage of 12 months. For the following tags, there is an additional “short” scanning interval of 5-seconds, with FIFO storage of two weeks:

    Current A, Current B, Current C, Voltage A-B, Voltage B-C, Voltage C-A, Power Factor Total, Apparent Power Total, Reactive Power Total, Real Power Total, and Frequency.

    For instructions on changing the “short” scan interval settings, see "Trend tag scan intervals" on page 220.

    To change a trend interval for a tag, see . To add additional trend tags, see
16. The **Driver Parameters** box contains options that you can check for IEC 61850 devices. If a device includes datasets and report control blocks, you can edit the information on the "Working with IEC 61850 datasets" on page 216 and "Edit IEC 61850 Report control blocks" on page

217 screens.

17. Check the **Close Wizard** box, and click **Finish** to return to **Create Device Profiles** tab. Or, leave it unchecked, and click **Finish** to return to the **Add/Edit Device Profile** screen.

Edit a device profile

Only unlocked profiles are available for editing.

1. Open the **Create Device Profiles** tab: from the **Create Device Profiles** tab, click **Add/Edit**.
2. In the **Profile Options** box, click **Edit Existing**.
3. From the drop-down menu, choose the profile you want to edit.
4. You can change any of the attributes that have been selected for this profile.
5. Click **Save** to save the change, or click **Save & Exit** to save changes and close the screen.

There are two ways to edit tags:

1. From this first screen, you can select a profile and then:
   - **Trend Tags** sub-tab: choose trend intervals (to create or edit intervals, see).
   - **PC Based Alarms or Onboard Alarms** sub-tabs: change alarm levels (this will override the default that is set in).
   - **Onboard Alarms** sub-tab: enable waveform capture for on-board alarms (see "Enable waveforms for onboard alarms" on page 260 for complete instructions on enabling these waveform captures).
   - **Onboard Alarm**s sub-tab: add Alarm On and Alarm Off text. What you enter here will override the default setting that comes from the custom tag (see for more information).
   - **Parameters** sub-tab: Edit parameters for IEC 61850 driver parameters (see "Edit driver parameters" on page 219 for more information).
2. Click **Add/Edit** to progress through several screens to edit all aspects of the profile. See the tables below for detailed instructions.

Delete a device profile

You cannot delete standard profiles or custom profiles that have been associated with projects. To delete a custom profile that is associated with a project, you need to go to the Set Up Project tab.

1. Open the **Create Device Profiles** tab; from the **Create Device Profiles** tab, click **Add/Edit**.
2. In the **Profile Options** box, click **Delete Existing**.
3. From the drop-down menu, highlight the profile you want to delete.
4. Click **Delete**.
5. Click **Yes** to confirm the deletion.
6. Exit the screen.

IEC 61850 system setup workflow

These are the basic steps you need to follow to set up an IEC 61850 device in your project.

1. List all of the SCL files (ICD, CID) for the IEC 61850 devices in your installation. ICD files are preferred. Pay special attention to data concentrated devices (for example, the G3200 with multiple devices communicating through it; see "In the Profile Editor" on page 224).
2. Import the first ICD file into the Profile Editor (see "Import Filter screen" on page 236).
   a. Create the device type.
   b. Match or verify tags for Power SCADA Operation.
   c. Complete the import.
3. Create a device profile for the IEC 61850 device type (see "Adding an IEC 61850 device" on page 248).
   a. If needed, add/edit datasets and report control blocks (see "Working with IEC 61850 data-sets" on page 216 and "Edit IEC 61850 Report control blocks" on page 217).
   b. Select the appropriate tags for Power SCADA Operation to monitor for this device.
4. Repeat steps 2 and 3 for additional ICD files.
5. Create a Profile Editor project, adding the device profiles. Configure as needed.
6. Export to Power SCADA Operation, and to SCL.

   Power SCADA Operation creates the equipment.profiles file for the Profile Wizard or Automation Interface.

   SCL will create an IID file for the profile. If newly added datasets and/or report control blocks are to be used, this IID file is required for step 7. Otherwise, you can use the original ICD file.
7. Use the appropriate IEC 61850 configuration tool for the device to configure a CID file from the ICD/IID file. Then download it to the device.
8. Create the project:
   a. From within Power SCADA Operation, add a new project.
   b. Add the appropriate clusters, networks, and servers.
9. Using the I/O Device Wizard, add your devices to Power SCADA Operation.
10. When you are prompted for the SCL file, use the CID file you created in step 7. For more information, see "Adding an IEC 61850 device" on page 248.
11. Compile and run the project.

Create IEC 61850 Device Type

The first step in creating an IEC 61850 device type is to import the device SCL files, after which you can make any necessary changes.

Import the SCL File

You can only import SCL files that meet the schema requirements for Ed 1.4 of IEC 61850. If an SCL file does not meet these requirements, an error message will display, telling you that the scheme must validate against the scheme of Ed. 1.4. The Profile Editor will accept SCL files that use either Ed. 1 or Ed. 2 data structures; but it will apply data structures only as defined in Ed. 2.

During this import, you need to reconcile mismatches; and data will be available for creating device types, device profiles, and projects. If you import an SCL for a PM700, note that all tags for date and time are excluded by default.

You can save the information in one of two ways:

- IID file: This IID file will maintain all of the configuration and communication information that comes from its device. The only items you can change are:
  - You can delete datasets and control blocks, and add new ones.
  - You can edit buffered and unbuffered control blocks (provided you have created them in the Profile Editor).
- Power SCADA Operation profile: The data will then follow the normal rules for the profiles in this project.

The Import Filter Screen

This screen displays after you choose an IEC 61850 file to import (.ICD, .CID, or .IID extension) and click Start Import. Use this screen to begin filtering data for import. You choose whether to filter on functional constraints or report control blocks. We recommend that you use report control blocks:

Report Control Blocks

Click the Report Control Block button.

The list of devices and their related report control blocks that are included in the import file displays in the middle column.

Check the devices and/or related report control blocks that you want to include in the import. If you check a device, all of the report control blocks under it are included.

The right-hand column displays the IEDs/report control blocks that you have selected.

NOTE: Use the filter above the middle pane to search. You can enter partial names separated by dots to further shorten the list.

When you have selected either the functional constraints or report control blocks, click Continue. The data is filtered on the last filter option that you chose (you cannot combine filters). The Import Reconciliation screen displays.

Use the Reconcile Import Screen to find matches for the items you are importing and to filter import tags to determine whether items are matched or not matched.

Edit IEC 61850 Datasets

To add and edit IEC 61850 tag datasets to a profile, display the Create Device Profiles tab for a device that includes ICD files. Click the Parameters sub-tab, then click Edit on the DataSets line.

NOTE: Not all ICD files allow you to add, edit, or delete datasets. If all fields are greyed out, you will not be able to change the set.

In the upper left corner are the device profile name and device type names that come from an imported ICD file. All of the entry fields are initially greyed out. The device type datasets (upper box) are resident in the ICD. The device profile datasets (lower box) have been created or copied from other datasets in the device type or device profile.

Create and Edit DataSets

If you need to create or edit IEC61850 datasets to a profile, see "Working with IEC 61850 datasets" on page 216.

Working with IEC 61850 datasets

Use this screen to add and edit IEC 61850 tag datasets to a profile.

To access this screen:

1. Display the **Create Device Profiles** tab for a device that includes ICD files.
2. Click the **Parameters** sub-tab, then click **Edit on the DataSets** line.

NOTE: Not all ICD files allow you to add, edit, or delete datasets. If all fields are greyed out, you will not be able to change the set.

In the upper left corner are the device profile name and device type names that come from an imported ICD file. All of the entry fields are initially greyed-out. The device type datasets (upper box) are resident in the ICD. The device profile datasets (lower box) have been created or copied from other datasets in the device type or device profile.

Create a new dataset

1. Click **Create New** beside the Device Profile DataSets box.

   The fields on the right side of the screen are enabled.
2. Type a name and description for the new dataset. These are free-form fields, but they must comply with IEC 61850 standards.
3. Choose the appropriate logical device, then choose the logical node for that device.
4. Choose the functional constraint for the content. This will filter the display of device type objects/topics in the box below.

   When you choose **All**, you must then choose an object that already has a functional constraint in it. If you choose a specific constraint, the list of available objects is filtered to display only those that include that constraint.
5. From the **Device Type Objects**, choose the appropriate objects for this profile.
6. Click **OK**.

The new dataset is added in the lower left, to the Device Profile list..

Create a dataset from an existing dataset

You can create a new dataset either from one that resides in the ICD (from the device type) or from the device profile.

To create a dataset from another block:

1. Click the dataset (either device type or device profile) to be used as the starting point for the new dataset.
2. Click **Create From**.
3. Make the appropriate changes. You must change the name. All datasets in a single profile must have unique names.
4. Click **OK**.

The new name displays under the **Device Profile List**.

Copy a dataset to a Device Type

This feature will not typically be used. If, however, you delete a dataset from the device type, but later decide you want to add it back, follow this procedure. (You cannot delete datasets that are used by a report control block.)

1. From the **Device Type DataSets** box, highlight the dataset you want to add back.
2. Click **Copy To**.

   The dataset

The dataset displays under the **Device Type** list in the **Device Profile DataSets**.

Edit and delete datasets

You cannot edit or delete datasets that are being used by a report control block or those that belong to the device type.

To edit a dataset, highlight its name, then click **Edit**. Make the desired changes, then click **OK**.

To delete a dataset, highlight its name. Click **Delete**, then click **OK**.

Edit IEC 61850 Report control blocks

Use this screen to edit report control blocks for device type information that comes from imported ICD files.

To access this screen:

1. Display the **Create Device Profiles** tab for a device that includes ICD files.
2. Click the **Parameters** sub-tab, then click **Edit on the Report Control Blocks** line.

NOTE: Not all ICD files allow you to add, edit, or delete report control blocks. If all fields are grayed out, you will not be able to change the set.

In the upper left corner are the device profile name and device type names that come from an imported ICD file. All of the entry fields are initially grayed out. The device type report control blocks (upper box) are resident in the imported ICD file. The device profile report control blocks (lower box) have been created or are copied from report control blocks in the device type or device profile.

Create a New Report Control Block

To begin creating a new report control block:

1. Click **Create New** beside the **Device Profile Report Control Blocks** box.

   The fields on the right side of the screen are enabled

2. Type a name and description for the new report control block, conforming to the IEC 61850 naming conventions.
3. Choose the appropriate dataset for this block. Datasets are added/edited in the Add/Edit DataSets screen, accessed from the Parameters sub-tab on the Create Device Profiles tab.
4. Type a report ID, again conforming to the IEC 61850 convention.
5. ConfRev determines the version number of the report control block.
6. If this is a buffered block (BRCB), check Buffered and enter the time and integrity period. (Indexing is currently unavailable in Power SCADA Operation).
7. Check the appropriate boxes for trigger conditions and report content.
8. Click **OK**.

The new report control block is added in the lower left, to the **Device Profile** list.

Create a Report Control Block from an Existing Report Control Block

You can create a new report control block either from a block that resides in the ICD (from the device type) or from the device profile.

To begin creating a block from another block:

1. Click the report control block (either device type or device profile) to be used as the starting point for the new block. **Click Create From**.
2. Make the appropriate changes. You must change the name. All report control blocks in a single profile must have unique names.
3. Click **OK**.

The new name displays under the **Device Profile List**.

Copy a Report Control Block to a Device Type

This feature will not typically be used. If, however, you delete a report control block from the device type, but later decide you want to add it back, follow this procedure.

1. From the **Device Type Report Control Blocks** box, highlight the block you want to add back.
2. Click **Copy To**.

The report control block displays under the **Device Type** list in the **Device Profile Report Control Blocks**.

Edit and Delete Report Control Blocks

You cannot edit or delete datasets that belong to the device type.

To edit a report control block, highlight its name, then click **Edit**. Make the desired changes, then click **OK**.

To delete a report control block, highlight its name. Click **Delete**, then click **OK**.

Edit driver parameters

Certain IEC 61850 devices may have driver parameters associated with them. You can edit the datasets and report control blocks that will then be exported to Power SCADA Operation.

To begin editing driver parameters: from the Create Device Profiles tab, click the Parameters sub-tab.

To begin editing datasets, click Edit in the DataSets line. Follow instructions in "Working with IEC 61850 datasets" on page 216 for help.

To begin editing report control blocks, click Edit in the Report Control Blocks line. Follow instructions in "Edit IEC 61850 Report control blocks" on page 217 for help.

Set Up Trend Intervals

For any of the trend definitions that are in the system, you can add, edit, or delete trend intervals.

To add a trend interval:

1. In Profile Editor, click **Settings > Set Up Trend Definitions**.
2. From the Set Up Trend Definitions screen:
   a. Click **New** to begin adding a new trend
   b. Select a trend, then click **Copy** to create a new trend from an existing trend.
3. Enter a **Name**: must begin with either an alpha character (A-Z or a-z) or the underscore character (_). Any following characters must be either alpha characters (A-Z or a-z), digit characters (0 - 9), backslash characters (\), or underscore characters (_).
4. Type the appropriate information in the following fields. For detailed information, see **Trend Tag Properties** in the Citect SCADA help.

To edit a trend interval

1. From the Set Up Trend Definitions screen, select the trend name, then click **Edit**.
2. You can edit any of the fields except the trend name.

To delete a trend interval

1. From the Set Up Trend Definitions screen, highlight the name of the trend to be deleted.
2. Click **Delete**, then click **Yes** when you are asked to confirm.

Select Trend Intervals

Use the Select Trend Intervals screen to edit settings for existing trends for specific device profile/tag combinations. To create new trends, see "Set Up Trend Intervals" on page 219.

To change a trend interval, follow these steps:

1. On the **Create Device Profiles** tab, choose the device profile, then click the **Trend Tags** sub-tab.
2. Locate the tag for which you want to change the trend. Click **Edit**.
3. In **Select Trend Intervals** screen, you can select one or all of the interval options.
4. Click **OK**.

Trend tag scan intervals

When you select a trend tag for a device profile (**Add / Edit Device Profile** screen), the tag will be scanned at the "long" interval" (every 15 minutes, with FIFO storage of 12 months); but certain trend tags have an additional "short" scan interval. This interval is set by default at 5 seconds, with FIFO storage of two weeks.

The default tags are: Current A, Current B, Current C, Voltage A-B, Voltage B-C, Voltage C-A, Power Factor Total, Apparent Power Total, Reactive Power Total, Real Power Total, and Frequency. When you choose one of these tags for trending, you will get both long and short interval trending. The long interval trend will use the trend tag name from the Profile Editor. The short interval trend tag will have the same name as the long tag with an "s" appended to it.

You can edit the *Profile Editor.exe.config* file to add or delete tags that will have short scan intervals, and to change the short scan interval for all of the tags that are listed.

To edit short scan interval settings:

1. In Notepad, open `Profile Editor.exe.config`. It is located in: [Project Drive]\Program Data\ Schneider Electric\Power SCADA Operation\v9.0\Applications\Profile Editor
2. To change the short scan interval:
   a. Scroll to the "TrendShortIntervalSamplePeriod" setting. The default value is 00:00:05, or 5 seconds (HH:MM:SS). Changing this rate will change the interval for all of the tags that are listed in the setting in step 3.
3. To change the tags that are included in the short scan interval:
   a. Scroll to the "TrendShortIntervalTags" setting. The numbers listed (defaults: 1003,1004,1005,1050,1046,1042,1014,1015,1016,1001,1034) are the tag IDs. You can add or delete tags. Tag IDs are listed on the Define Device Type Tags tab (when the Advanced Properties option checked).

   NOTE: If you choose a device that includes the tags in this list, you will always have these short scan interval tags included.

   For example, if you wanted to change the scan interval to ten seconds and add Overcurrent A for a CM4000, you would edit these two lines in this way:

   "TrendShortIntervalSamplePeriod" value="00:00:10"

   "TrendShortIntervalTags"
   value="1003,1004,1005,1050,1046,1042,1014,1015,1016,1001,1034,19"

Disk storage calculation for trends

There are two methods of calculating disk space usage for trends: scaled and floating point. The Profile Editor uses floating point by default. For more information on these calculations, see **Calculating Disk Storage** in the Citect SCADA help file (...\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin\Help\Citect SCADA).

Create a Composite Device Profile

A composite device profile includes more than one device type. Each device type can use its own protocol for communication.

With the composite device type, the user can use two devices for a single monitoring point. For example, a circuit breaker and a monitoring device can provide data to this single point. Because Power SCADA Operation combines the functionality of the multiple devices, end users only need to consider a single device when analyzing a location in their system.

The following links provide instructions for specific device types:

- "Create Third Party Modbus Device Type" on page 221
- "Create a composite device type" on page 222
- "Create Data Concentrator" on page 224
- "In the Profile Editor" on page 224

Create Third Party Modbus Device Type

To create a third party Modbus device and add it to your Power SCADA Operation project:

1. Find the Device Modbus Reference. This should be included in a document from the manufacturer for the device you want to add.
2. Familiarize yourself with the manner that the Modbus device specification.
3. Verify the Power SCADA Operation supports the device:

   The following table lists allowed values for each data type:

Data Type	Variable	Size	Allowed Values
String	string	256 bytes (maximum)	ASCII (null terminated)
Digital	digital	1 bit or 1 byte	0 or 1
Long	long integer	4 bytes	–2,147,483,648 to 2,147,483,647
Real	floating point	4 bytes	–3.4E38 to 3.4E38

4. Verify that the tags you want to use are compliant with Power SCADA Operation. To ensure that data is reported for reporting, LiveView tables, and breaker graphics. Refer to the Common Data Model (CDM), which is located in C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\9.0\Applications\AppServices\bin..
5. Find the best fit tags: From the Profile Editor tag library, find the tag that comes closest to the quantity you want to measure.

6. Verify the tag you have chosen by comparing it with the CDM. The DisplayName
7. Create the device type in the Profile Editor: Use PwrModbus as the driver.
8. Select the appropriate tags (CDM)
9. Configure the Modbus tags: Continuing on the Define Device Type Tags tab, edit the tag addresses to map them to the Modbus register of the device (these tags will be red). You can locate instructions on editing addresses in the Power SCADA Operation help file.
10. Create the device profile: Click Add/Edit to launch the Add/Edit Device Profile window. Create the new profile and choose the device(s) that you want.
11. On the next screen, move the tags into the Selected Tags pane. Select Trend for all tags that require it.
12. Continue with setting up the project and exporting as you do with other device profiles.

Create a composite device type

A *composite device* is a device profile that includes more than one device type. Each device type can use its own protocol for communication.

With the composite device type, the user can use two devices for a single monitoring point. For example, a circuit breaker and a monitoring device can provide data to this single point. Because Power SCADA Operation combines the functionality of the multiple devices, end users only need to consider a single device when analyzing a location in their system.

NOTE: For instructions on setting up and using Cyber Sciences Sequence of Events Recorder (SER), refer to the system technical note (STN) entitled *How can I Use Cyber Sciences SERs with PowerSCADA Expert?*

To create the composite device type:

1. From the **Create Device Profiles** tab, click **Add/Edit**.
2. At the **Add/Edit Device Profile** screen, choose whether you are creating a new device or creating from an existing device. If you are creating from a device type, select it. Click **Next**.
3. Still on the **Add/Edit Device Profile** screen, give the composite device type a name. Optionally, add a description (which will become a tool tip display in later screens). Click **Next**.
4. Choose the device types to be in the composite. Click **Next**.

   The **Add/Edit Device Profile** displays with only device type tags available for selection.
5. Add the tags you need for each device type listed on the left. To add all of the tags for a device type, highlight the device type name and click the right green arrow.

   The **Add/Edit Device Profile** displays with only device type tags available for selection.

   You may find, especially when dealing with generic I/O, that the tag name is not descriptive enough to determine what it is when reading data in runtime mode. Thus, you may want to override the generic name with something more meaningful.

For example, a device may have ten inputs: Ind1, Ind2, Ind3, and so on. Using those names, you have no idea what each input is reading. If you override the tag, the tag's value will still come from the original tag (it still keeps the addressing from the device); however the tag's appearance (name, metadata, display name) will be taken from the new tag.

6. To override a tag:
   a. Highlight the tag, then click Override Tag Name.
   b. From the Select Tag window, choose the tag you want. If necessary, enter a search term, then click Search to display related tags.
   c. Choose the tag, then click **OK**.

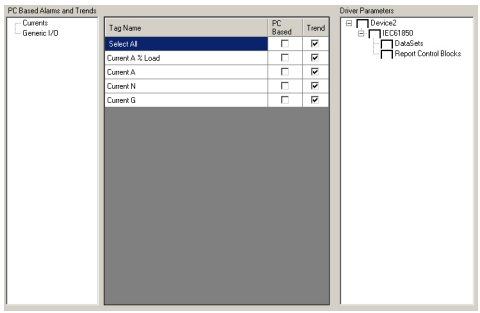
   Only or composite devices, the *Is Device Tag* check box displays. Use this box to tie a tag back to its actual physical device. For example, you might have the same tag in each of three devices, and you want to set PC-based alarms for each one. Normally, the composite device would generate a single alarm, but you would not be able to specify which physical device has the problem. To prevent confusion, you would check the *Is Device Tag*, which will cause Power SCADA Operation to report this tag for its physical device.
7. Check **Is Device Tag** to read this tag as specific to the physical device, not the entire profile..
8. Click **Next** to begin selecting tags for PC-based alarms and trends.
9. For each tag in the profile, determine whether it should have a PC-based alarm and/or trend associated with it. Check the boxes as appropriate.

   When the profile is added to the project, PC based alarms are added to the Analog Alarms or Digital Alarms file.

   When the profile is added to the project, historical trends are added to the Trend Tags file. Logging will automatically begin when the tag is added to the project.

   There are two different intervals for scanning trend tags. All selected tags are scanned every 15 minutes with FIFO storage of 12 months. For the following tags, there is an additional “short” scanning interval of 5-seconds, with FIFO storage of two weeks:
   Current A, Current B, Current C, Voltage A-B, Voltage B-C, Voltage C-A, Power Factor Total, Apparent Power Total, Reactive Power Total, Real Power Total, and Frequency.

   For instructions on changing the “short” scan interval settings, see "Trend tag scan intervals" on page 220.
10. The Driver Parameters box allows you to specify certain parameters to be attached to device profiles. Currently used in IEC 61850 devices, the available parameters will automatically populate this box. See the illustration below for an example.



In this example, Device 2 has two parameters, DataSets and Report Control Blocks.

11. Check the parameter(s) that you want to include in this profile.
12. To edit, this parameter, return to the **Create Device Profiles** tab, and click the **Parameters** sub-tab. See "Working with IEC 61850 datasets" on page 216 and "Edit IEC 61850 Report control blocks" on page 217 for information on editing these two parameters.
13. Check the **Close Wizard** box, and click **Finish** to return to **Create Device Profiles** tab. Or, leave it unchecked, and click **Finish** to return to the **Add/Edit Device Profile** screen.

Create Data Concentrator

When you use the Profile Editor to create data concentrator or a data-concentrated devices, all of the related devices must use the same protocol. Examples of data concentrators are PLCs that use inputs from various devices or an RTU that concentrates data from multiple devices.

To add a data concentrator to your project, follow these steps in the Profile Editor:

1. In the Profile Editor, click **Define Device Type Tags**.
2. Add a custom device type for the data concentrator. Use the Generic Power Device driver.
3. Add the tags that are specific to the data concentrator (such as device date and time).
4. Add addresses for any custom tags you created.
5. Add the data-concentrated device. Use the Generic Power Device driver, as you did for the data concentrator.
6. Add the tags for the data-concentrated device (such as currents, voltages, and breaker status).
7. Add addresses for these tags (or add functional addressing for them).
8. Repeat steps 5 through 7 for additional data-concentrated devices.
9. Click the **Create Device Profiles** tab
10. Add a device profile for each data-concentrated device type you included.
11. Click the **Set Up Projects** tab and then add the profiles to a project.

G3200 device setup

Use these instructions to set up a G3200 gateway in Power SCADA Operation.

For use with multiple devices

Before you begin, create the ICD files for each unique device type that will communicate via the G3200.

In the Profile Editor

1. Import each unique ICD file.
2. Create the profiles for each device.
3. Modify existing profiles, as needed (adding/modifying tags, and so on).
4. Create the project that will include the G3200 (mark profiles under the G3200 as data concentrated devices).

a. Ensure that **Add As Default** is not checked for the project.

b. Add the first device profile.

c. At **the Select Profiles** screen, enter the Configured Name, and check **Data Is Concentrated**.

d. Continue with steps "b" and "c" for additional device profiles.

5. Run Power SCADA Operation and SCL exports.

In CET850

Create the CID file for the G3200 gateway.

In Power SCADA Operation

1. Open the Profile Wizard.
2. To the System Devices, add an IEC 61850 data concentrator for the G3200:
   a. Enter instance information screen, select the BRCBs that you need.
   b. Select the CID file you created in CET850.
   c. Complete the remaining steps in the Profile Wizard.
3. Add a new device for each device under the G3200.
   a. From the Enter instance information screen, change the logical device as needed. Select the unit name of the G3200 device for the data concentrator.

For use with a single device

Although we recommend that you add individual G3200 devices as described in the section above, you can also do it this way:

Before you begin, create the ICD files for the device type that will communicate via the G3200.

In the Profile Editor

1. Import the ICD file.
2. Create the device profile.
3. Modify the profile as needed
4. Create the project, but do not mark it as Data Concentrated Devices.

In CET850

Create the CID file for the G3200 gateway.

In Power SCADA Operation

1. Open the Profile Wizard.
2. Add the device:
   a. From the Enter instance information screen, select the BRCBs that you need.
   b. Change the logical device as needed.

c. Select the CID file you created in CET850.

d. Complete the remaining steps in the I/O Device Manager.

DNP3 protocol support

You can create device types and profiles that use the DNP3 protocol. See "Adding a DNP3 TCP device" on page 247 or "Adding a serial device" on page 246 for more information.

You will then be able to enter DNP3 addresses, although the Profile Editor will not verify that the address has an allowed format.

The Profile Editor includes device types, and includes profiles for ION 7650, which natively supports DNP3. The Profile Editor includes device types for Sepam 20, 40, and 80 that have the ACE969TP module (which supports DNP3).

Set Up Projects

Use the Profile Editor > **Set Up Project** tab to begin adding, editing, or deleting projects. A project includes all of the tags that belong to the device profiles that you have created and added to the project. From this screen you also export individual projects to the .XML file format, which you can add via the Profile Wizard.

The Set Up Projects tab and screen are used to create separate projects for each customer or installation. This tab makes it easy to select only the devices that are used at that site. Project data is exported to Power SCADA Operation for use in the Device Creation Wizard.

This screen includes three tabs:

- **Selected Device Profiles**: (read only) You can view all of the profiles that are included in each project in the system. Profiles are listed with their descriptions.
- **Customize Tag Names**: You can customize tag names (for example, instead of Current A, you might need to use Current Phase A) within a single project. See "Customize tag names" on page 230.
- **Project Parameters**: You can add optional information to be associated with the export. This information can help you identify the correct project when you are importing. See "Add project parameters" on page 230.

To add or edit project information, click **Add / Edit**. The Add / Edit Project screen displays.

To view the most recently exported project, click the folder button to the right of the **Export Project** button: 

Set Up Project screens and workflow

The **Set Up Project** tab has three sub-tabs:

- **Selected Device Profiles** – Displays all of profiles that are included in the Project that is displayed in the drop-down menu.
- **Customize Tag Names** – Lets you customize individual tag names.

- **Project Parameters** – Lets you to add optional lines of information about this project. This information will be exported and can be used for verification or identification when you want to import the project (for example, you might add a version number or creator name).

You can click **Export Project** to create an XML file that contains all of the project data necessary for use in the Profile Wizard. If Power SCADA Operation is installed and the corresponding Power SCADA Operation project has been created, this also copies the file that is used by the Device Creation Wizard to the Power SCADA Operation project.

On the **Add / Edit Project** window, you can add, edit, or delete projects.

Typical workflow

To create a project file, you must first have established tags, device types, and device profiles. Additionally, you need to set up at least one base unit/conversion template. After these files are created, complete the following steps:

1. In **Set Up Projects**, click **Add / Edit**.
2. Add a new project, or copy and edit an existing project.
3. Select the device profiles that you want to use for this installation.
4. If a device profile has multiple drivers, choose the driver, and determine whether the individual device types will use functional addresses and act as data concentrators.
5. Save the project and close Add / Edit Project.
6. Customize tag names:
   a. From the **Set Up Project** tab, click the **Customize Tag Names** sub-tab.
   b. Change the name of any tag.

      For example, the customer might need “Current A” to read “Current Phase A.” The customized tag name will be used in all device profiles in the project for which you have created the customized tag.
   c. For this change to be in the Power SCADA Operation project: you need to delete the device profile from that project and then re-export it.
7. Add optional project information:
   a. From the Set Up Project tab, click the Project Parameters sub-tab.
   b. You can add optional information that will help verify or identify this project later. You could, for example, add the version or the creator's name. This information will be available when you import this project at a later date.
8. Refresh tags:
   a. From the **Set Up Project** tab, click the **Selected Device Profiles** sub-tab.
   b. Click the **Refresh Tags** button for any profile.
   c. You are prompted to confirm that you want to update changes you have made to this tag for the selected profile.
   d. For this change to be in the Power SCADA Operation project: you need to remove the device profile from that project and then re-add it.

9. Click **Export Project** to create an Equipment.Profiles file of all of the profiles included in the project.
10. View Equipment.Profiles by clicking the folder button, to the right of the Export button:

About project files

You create a project file to include the tags and devices you add in the Profile Editor. The project file is then exported from the Profile Editor.

By default, the project is exported to:

```
C:\ProgramData\Schneider Electric\Power
SCADA Operation\v9.0\Applications\Profile
Editor\WizardProfiles\"project name"\ProfileWizard
```

Where "project name" is the name used when you created the project.

After you export the profile, add the included I/O devices into your final project.

Add, edit, or delete a project

Use the **Add / Edit Project** window to begin adding, editing, or deleting projects. A project includes all of the tags that belong to the device profiles that you have created and added to the project. From this screen you also export individual projects to the format that can be added to Power SCADA Operation (using the Profile Wizard).

Adding a project

To add a project:

1. First ensure that you have set up the tags, device types, and device profiles that you want to include. Also, add at least one unit template.
2. Click the **Set Up Projects** tab, and then click **Add / Edit**.
3. In the **Project Options** section, click **Create New** or **Create From**.

   NOTE: If you are creating a project from an existing project, from the **Project to Create From** drop-down list select the project.
4. Type a **Project Name**: The name must be alpha-numeric only, beginning with an alpha character, and can be up to 32 characters long. Do not use:

   \ / : * ? < > |
5. To view a list of projects that have already been added to Power SCADA Operation, click the **Display Projects** button:

   A list displays with the projects that have been added (grayed-out if there are no projects yet or if the Profile Editor is not on the same computer as the server). To open a project for editing, select it and click **OK**.
6. (Optional) To prevent someone from editing the project it, click **Lock this Project**.

NOTE: This action cannot be undone. If you want to edit a locked project, you must use the Create From feature to add a new one, then delete the locked one.

7. Type a **Description** for the project. This description displays as a tool tip when you hover over the project name on the main Set Up Project tab.
8. Select a **Unit Template** from the drop-down list. Unit templates are created on the Units screens. See "Set up engineering templates and select conversions" on page 639 for instructions on creating templates.
9. (Optional) To add a new unit template, click **Set Up Eng. Unit Templates**. The Set Up Engineering Unit Templates page displays. See "Add or edit a base engineering unit or conversion" on page 643 for help.
10. In **Device Profiles**, select the first profile you want to include in this project and then click  to move the device profile to **Selected Device Profiles**.

    If this device profile will NOT have functional addressing or data concentration, check the "Add As Default" box at the bottom of the screen. (For a description of functional addressing, see the Functional Addressing entry in "Glossary" on page 656.)

    If the Select Profile Drivers screen displays, one of the following is true.

    - You did not click **Add As Default** for a device type, so the system does not know how to use the functional address/data concentrator option. Check the appropriate box to turn the related option "on."
    - At least one of the device types in this profile includes multiple drivers. For each multiple-driver device type listed, choose the driver that you want to use in this project. Additionally, you can click either **Functional Address** or **Data Is Concentrated** to enable those features.
11. Give the device type a Configured Name. This name might indicate its status (which driver it uses, whether it has a functional address, and so on) in future project references.
12. When all profiles are added, click **Save** to save the changes, or click **Save & Exit** to save changes and close the screen.

Edit a project

You can only edit projects that are unlocked.

To edit a project:

1. Click the **Set Up Project** tab, then click **Add / Edit** to open the **Add / Edit Project** window.
2. In the **Project Options** section: click **Edit Existing**, then from the **Project to Edit** drop down select the project to be edited.
3. You can change any attribute of the project.
4. Click **Save** to save the change, or click **Save & Exit** to save changes and close the screen.

Delete a project

You can only delete unlocked projects.

To delete a project:

1. Click the **Set Up Project** tab, then click **Add / Edit** to open the **Add / Edit Project** window.
2. In the **Project Options** section: click **Delete Existing**, then from the **Project to Delete** drop down select the project to be deleted.
3. Click **Delete**.

Customize tag names

From the **Set Up Project** tab, click the **Customize Tag Names** sub-tab.

You can add a custom name for any tag in the system, predefined and custom tags. The customized name will be used anywhere the original name would be used, but only for the project that is selected in the drop-down menu. When you use the Export option, it will be used by the Profile Wizard.

Add project parameters

The **Project Parameters** sub-tab allows you to add optional lines of information about this project. This information can be used for verification or identification when you want to import the project.

To add project parameters:

1. From the **Set Up Projects** tab, click the **Project Parameters** sub-tab.
2. On the first available line, type a name and value for this information. Example: If you want to track versions, in the Name field, you might type "Version." Then, in the Value field, type the appropriate version for this project.

The new parameter is added. It will help you identify the project when you want to import it into another instance of the Profile Editor.

Export a project

Exporting a project copies all project data (device tags, device types, and device profiles) from the project in Profile Editor to the project in Power SCADA Operation.

When the Profile Editor is on the same computer as Power SCADA Operation, and if you have created a matching project in the Power SCADA Operation project, this process will copy all project data (device tags, device types, and device profiles) into that project.

NOTE: If the Profile Editor is not on a computer with Power SCADA Operation, you need to manually move the exported file to the Power SCADA Operation server. See "Moving files when the Profile Editor is not on the server " on page 233.

To export a Profile Editor project to the Power SCADA Operation project:

1. In Profile Editor, click **Set Up Projects** tab.
2. From the **Project** list, select the project to be exported.
3. Click **File > Export**, then check the Power SCADA Operation Export option. (The selected export(s) are displayed beneath the **Export Project** button.)
4. Click **Export Project**.

NOTE: If you have added custom tags to devices, but the tag addressing is incomplete, a message displays with the device profile names that contain the tags. Return to the **Define Device Type Tags** tab. Locate any tags for which "Edit..." is red. Click **Edit** to open the Edit Address screen. Make the necessary changes. From the **Set Up Projects** tab, refresh the tags for those profiles. Then try exporting again.

A progress bar displays while the various profiles are saved. The resulting files are exported to these locations in the Profile Editor (assuming that you accepted the default locations during installation):

- Each Project file, used by the Profile Editor, is stored in Documents and Settings\All Users\Application Data\Schneider Electric\Profile Editor\Power SCADA Operation\Projects.
- Each Profile Wizard profile file is stored in Documents and Settings\All Users\Application Data\Schneider Electric\Profile Editor\Power SCADA Operation\WizardProfiles\[project name]. A single file for each included profile.
- The Equipment.Profiles file (contains all of the Profile Wizard profile information and the base profile information used by the Profile Wizard) is stored in Program Files\Schneider Electric\Profile Editor\Power SCADA Operation\WizardProfiles\[project name]\Profile Wizard.

In Power SCADA Operation, files are located in the following folders:

- DeviceProfiles contains .XML files for every profile (these are used by the Profile Editor).
- DeviceTypes contains .XML files for all device types (these are used by the Profile Editor).
- Projects contains all .XML files for all projects (these are used by the Profile Editor)

DeviceWizardProfiles contains the exported device profiles and equipment profiles files, organized by project (these are used by the Profile Wizard).

5. On the Project Editor window, use the Profile Editor to add device information.

Edit and delete information in a project

After you exported a project to a Power SCADA Operation project, you still need to use the Device Creation Wizard to add system information to the Power SCADA Operation project. See "Before adding I/O devices" on page 242 for information about this process.

Import and export project files

In the Profile Editor, you can import and export the following files:

- Export all of the tags and devices from a Profile Editor project into a project; see "Export a project" on page 230.
- Export SCL files, which allows you to export IID files that have been previously imported from an SCL file. The IID file can then be imported into other instances of the Profile Editor. See "SCL export" on page 233.
- Export a Profile Editor project. This makes a backup copy, which you can later import into a different instance of the Profile Editor. This is useful when you want to share custom tags and devices. See "Profile Editor export" on page 232.

- Import a project from another instance of the Profile Editor or from an IEC 61850 file.
- Import SCL files. You can import from the profile data of IEC 61850-compliant devices and create device types. These files can be exported as an IID profile or as a Power SCADA Operation profile.
- Import ICD files. You can import either functional constraints or report control blocks.

The import process works the same for each type of import. The only exception is that you cannot import profiles when you are importing SCL files. See "Import files into the Profile Editor" on page 234.

When importing data, you will need to reconcile the import information with the information that exists in the Profile Editor.

You can also use templates, both in exporting and importing. See "Using import templates" on page 240.

Before you export a project

If you are exporting a project for the first time to the Power SCADA Operation project, you need to create a matching project in Power SCADA Operation. To do this:

1. In Power SCADA Studio: Click **Projects**, add a new project. Be sure that the Template Resolution is SXGA.

   If you have questions about any of the fields, click **Help**.
2. Add your project to the Profile Editor, ensuring that the name matches exactly the one that you added in Power SCADA Studio (to ensure that it correctly exports to its matching project).

Profile Editor export

Export a Profile Editor project when you want to back up a Profile Editor project for re-use in another instance of the Profile Editor. This is useful when you have custom tags and custom devices that you want to share in other projects. After you export a project, you can import it to another Profile Editor project.

To back up a project file, see "Backup a project" on page 172.

To export a Profile Editor project:

1. In Profile Editor, click the **Set Up Projects** tab.
2. From the **Project** drop down box, select the project you want to export.
3. Click **File > Export > Profile Editor Export**.
4. See "Customize tag names" on page 230 and "Add project parameters" on page 230 for the information you need to make the changes that you want.
5. Click **Export Project**.

In addition to the project data, exported projects include:

- A unique project name, the date of the export
- The name of the computer to which it was saved
- (Optional) The description added when the source project was created.

The project – which will be named YOUR PROJECT.pls – is exported to the following location:

C:\ProgramData\Schneider Electric\Power SCADA Operation\v9.0\Applications\Profile Editor\Projects\YourProjectName.pls

Moving files when the Profile Editor is not on the server

If the Profile Editor is not on the same computer as the Power SCADA Operation server, you need to move the export file to the server computer.

To move the export file to a different server:

1. Export the project from the Profile Editor:
   a. Click the folder icon beside the **Export Project** button link: 
   b. Copy the file Equipment.profiles that displays and move it to a portable drive.
2. On the Power SCADA Operation server computer, paste Equipment.profiles to the following location:

   [Drive Letter]:\Documents and Settings\All Users\Application Data\Schneider Electric\Power SCADA Operation 9.0\User\[Project]

   Where:

   [Drive Letter]: is the drive on which you installed the Power SCADA Operation server

   the Application Data and ProgramData folders cannot be hidden (set the folder view for “view hidden folders”)

   [Project] is the name of the project you are creating; you must have already added this project to Power SCADA Operation (see **Before you export**, above).
3. Use I/O Device Manager to begin adding device information to the Power SCADA project.

SCL export

SCL export lets you export IID files (previously imported from an SCL file). The IID file can then be imported into other instances of the Profile Editor.

This process does not correct any errors in the files. If the imported file was an IID file from a different instance of the Profile Editor, it will contain the same configuration and communication information as the original. If the imported file was a Gateway SCL file with multiple devices, you can export each device as a separate IID file (the configuration and communication information is taken directly from the Gateway SCL file).

The only way you can edit these files are:

- You can delete data sets, and then add new ones.
- You can edit report control blocks (buffered or unbuffered).

Perform these edits in the device profile before you export, and they will be exported to the IID file.

Exporting the file

To export IID files:

1. From the **Set Up Projects** tab, select the project from which you want to export. (The project must have devices that include ICD files.)
2. Click **File > Export > SCL Export**.

   The export(s) that you select display beneath the **Export Project** button, on the right side of the screen.
3. Click **Export Project**.

The Export Summary displays with the results of the export. When the export displays under the Success topic, the listed files were exported. When the export displays under the Warnings topic, the reason that the export did not succeed is listed for the device types shown.

The exported files, listed according to their device types, will be saved in:

[Project Drive]\Program Data\ SchneiderElectric\Power SCADA Operation\9.0\Applications\Profile Editor\WizardProfiles\<project name>\SCL Export\sclFileName.iid).

Reuse projects created in the Profile Editor

You can create a project that can subsequently be reused for different installations.

To save and then reuse projects:

1. Export the project from the **Set Up Project** tab of the Profile Editor.
2. Click the folder icon beside the Export link: 
3. Copy the file (Equipment.profiles) that displays. If you need to use this file to another computer, you can move it to a portable drive.
4. On the server computer, paste Equipment.profiles to the location, where:

   [Drive Letter]: The drive on which you installed the server

   The Application Data and ProgramData folders are not hidden (set the folder view for “view hidden folders”)

   [Project]: The name of the project you are creating; you must have already added this project to Power SCADA Operation.
5. Be sure you have created the files described in "Before you add a project" on page 159.

Import files into the Profile Editor

Use this feature to import either an existing project or SCL files into the Profile Editor. This is commonly used to share project information by importing custom tags and devices from another instance of the Profile Editor; but you can also import SCL files from an IEC 61850 device.

For Profile Editor projects, you can import tags, device types, and profiles. For SCL imports, you cannot import profiles.

Before you begin, consider the source of the information you want to import. We strongly recommend that you use a master PC from which you draw this information. This will ensure that you are using a single source. Also, back up your data folder before you start. This gives you data to revert to, in case you accidentally lose data.

NOTE: You cannot complete the import until you match, merge, or reject every item.

To import data into the Profile Editor:

1. Note the location of the project file or other file (SCL, CID) that you want to import.
2. From the Profile Editor, click File > Import.
3. At the Import File Selection window, click browse, then navigate to the location of the file you want to import.
4. The Import Properties box displays the _ProjectName, _Description, _DateTime, and _ComputerName information. These lines were automatically generated for this file. Any additional lines will be information that you added on the Project Parameters sub-tab when you created or exported the project.
5. Use this information to verify that you are about to import the files that you want.
6. If desired, select an import template from the drop down list. (To create a template, see Creating a New Template During Import in "Using import templates" on page 240.) If you select a template, the import will accept default properties from the template. For example, if the template has alarm settings from a device, and you are importing tags for that device, the import will use those alarm settings.
7. When you locate the desired file, click Open to choose it. Then click Start Import.
8. The system analyzes the import and attempts to match imported items with existing items on the local machine.
9. If you are importing a PLS file, skip to step 8.
10. If you are importing IEC 61850 data, the Import Filter screen displays. Use this screen to perform an initial filter on functional constraints or on report control blocks. See "Import Reconciliation screen" on page 237 for more information.
11. Make your selections, then click Continue.
12. When the Import Reconciliation screen displays, you can begin the process of matching or rejecting individual tags. See "Import Reconciliation screen" on page 237 for a description of the parts of this screen.
13. On the Import Reconciliation screen, click an item in middle pane. Respond to the item according to your preference for it. You must set the status first for units, then tags, and finally the device type.
14. After you match or ignore all items in the import list, the Complete Import button becomes live. Click Complete Import.
15. After the import is saved, the Save Import Template dialog displays. See "Using import templates" on page 240 for instructions on creating, using, and deleting import templates.

Import SCL Files

You can import SCL files from individual devices, provided the files conform to IEC 61850 specifications. You can also import an individual device from a Gateway SCL file that contains multiple devices.

NOTE: You can only import SCL files that meet the schema requirements for Ed 1.4 of IEC 61850. If an SCL file does not meet these requirements, an error message will display, telling you that the scheme must validate against the scheme of Ed. 1.4. The Profile Editor will accept SCL files that use either Ed. 1 or Ed. 2 data structures; but it will apply data structures only as defined in Ed. 2.

During this import, you need to reconcile mismatches; and data will be available for creating device types, device profiles, and projects. If you import an SCL for a PM700, note that all tags for date and time are excluded by default.

You can save the information in one of two ways:

1. **IID file**: This IID file will maintain all of the configuration and communication information that comes from its device. The only items you can change are:
   - You can delete datasets and control blocks, and add new ones.
   - You can edit buffered and unbuffered control blocks (provided you have created them in the Profile Editor).
2. **Power SCADA Operation profile**: The data will then follow the normal rules for the profiles in this project.

Import Filter screen

This screen displays after you choose an IEC 61850 file to import (.ICD, .CID, or .IID extension) and click Start Import. Use this screen to begin filtering data for import. You choose whether to filter on functional constraints or report control blocks.

Functional Constraints

1. Click the Functional Constraint button.
2. Choose the functional constraints that you want to include.
3. The filters the list of devices for which you will import data to those that contain one or more of the selected functional constraints.
4. Check the device(s) that you want to include.

Report Control Blocks

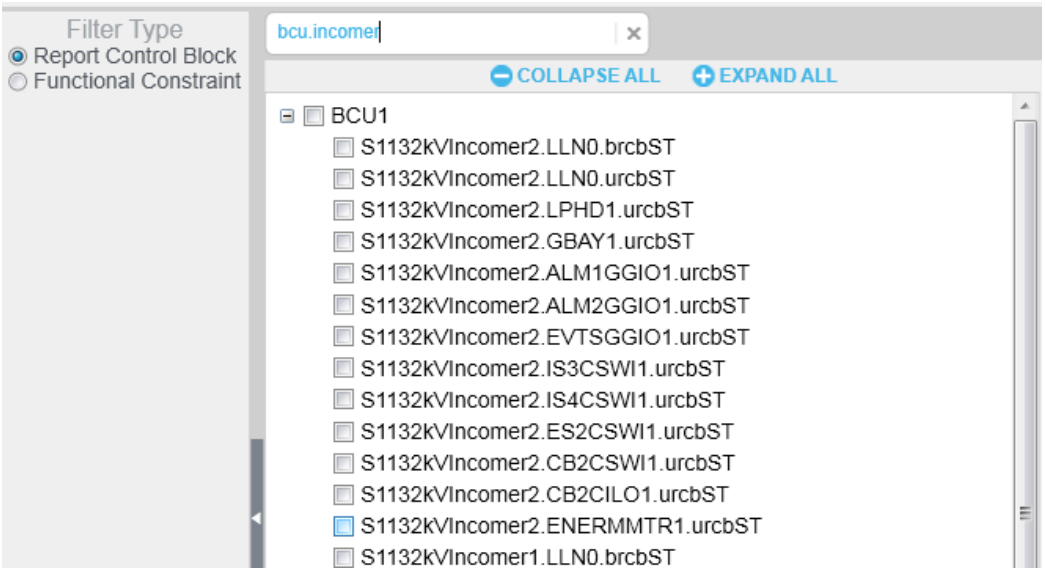
1. Click the Report Control Block button.

   The list of devices and their related report control blocks that are included in the import file displays in the middle column.
2. Check the devices and/or related report control blocks that you want to include in the import. If you check a device, all of the report control blocks under it are included.

   The right-hand column displays the IEDs/report control blocks that you have selected.

   Use the filter above the middle pane to search. You can enter partial names separated by dots to further shorten the list.

   The following image illustrates an example in which a search was done first on "bcu" and then on "incomer" (note that entries are not case sensitive). The search string would be: *bcu.incomer*

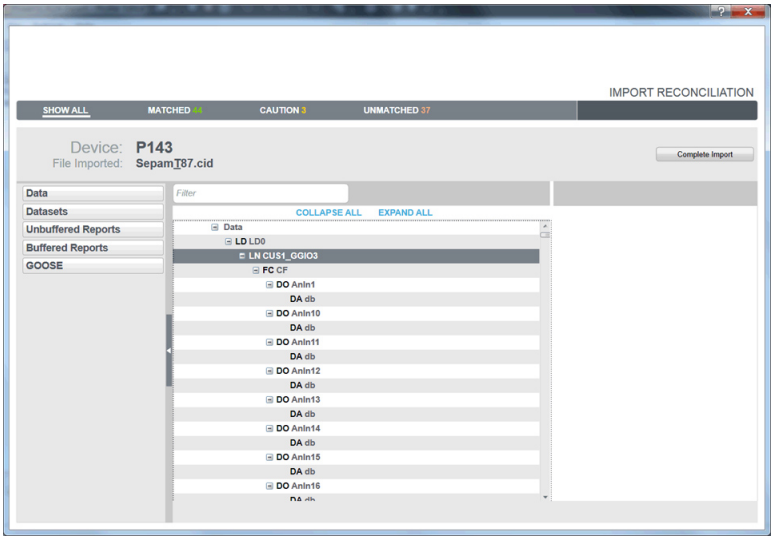


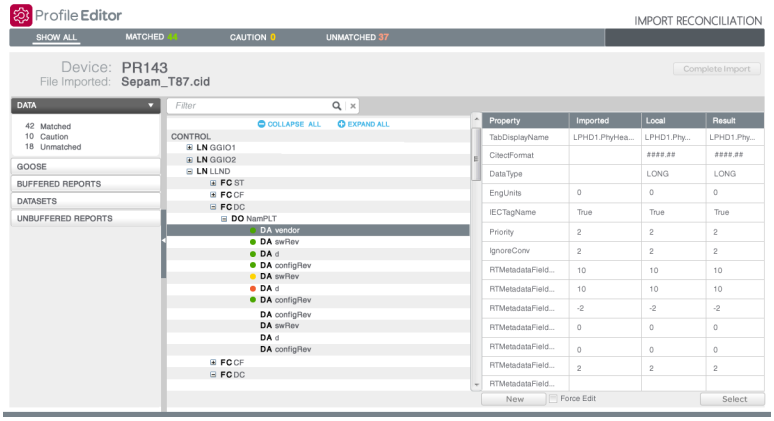
When you have selected either the functional constraints or report control blocks, click Continue. The data is filtered on the last filter option that you chose (you cannot combine filters).

The Import Reconciliation screen displays. See "Import Reconciliation screen" on page 237 for help finishing the import.

Import Reconciliation screen

Use the Reconcile Import Screen to find matches for the items you are importing and to filter import tags to determine whether items are matched or not matched. The first figure below shows the screen before import is complete. The second one shows the results after import has been completed.





The screen is divided into three panes:

Left pane:

The selections made in this pane provide an initial filter for what you view in the middle pane (see below). The tree view at the top shows the imported file data categories:

For .pls files imported from the Profile Editor, the categories are: device profiles, device types, tags, and units.

For IEC61850 files, the categories are: Data, Datasets, Unbuffered Reports, Buffered Reports, and GOOSE.

Select a category to filter the list in the middle pane to only the items belonging to that category.

To further filter the middle pane, click one of the matched status lines (matched, partially matched, unmatched) to view only items of that status. The number of items in that status also displays.

Middle pane:

This pane shows a tree view with data.

Filter: To filter on a specific item, type the name (such as phsA for phase A current). The entry can be the exact name, or you can enter a partial name or even a wildcard (*). The filter is not case sensitive.

The data in the middle pane is filtered to include only the items for the tag you specify. To clear the filter so you can enter a new one, click the "x" beside the filter box.

Collapse All/Expand All: Click *Collapse All* to collapse all nodes on the screen. Only the top-level nodes will display. Conversely, click *Expand All* to open all nodes, displaying all of the information on all nodes.

The bottom section of the middle pane displays, in tree form, the data that you selected in the left-hand pane:

- For files imported from the Profile Editor (.pls files), you can view: Show All, Device Profiles, Device Types, Tags, and Units.
- For IEC 61850 files, you can view: Data, Datasets, Unbuffered Reports, Buffered Reports, and GOOSE.

The bullets indicate:

: exact match; item is either a perfect match to a local item, or you accepted a merge for it

: unverified match; item is a partial match to a local item

: no match; item does not match any local item

Items that have no icon beside them are ignored during the import.

Re-match Items within a Logical Node

Because IEC 61850 tags are often imported with prepended information (logical node: LN) that prevents the import from matching them, you may find several unmatched items. You can use the re-match feature to enable matching for them.

To do this, right-click the logical node where the unmatched items are found, and choose Re-match.

The import feature will then exclude the logical node, and use the remaining information in the item name to find matches. In the screen shown above, it would include functional constraint (FC) **ST**, data objects (DO) **PhyHealth** and **Proxy**, and five data attributes below them:

ST.PhyHealth.q

ST.PhyHealth.t

ST.Proxy.q

ST.Proxy.stVal

ST.Proxy.t

Right pane:

This pane illustrates the status of each of the tags. Click a tag and read the information for it:

- Property: The property for which the other columns provide definitions.
- Imported: The value of the item in the import file.
- Local: The closest local match for the imported item.
- Result: The item as it will be added in this import; by default, this item is inherited from the local status.

New: At the bottom of the list, click this button to add an item as a custom topic. The Add/Edit Custom Tag screen displays for you to create the tag.

Force Edit: Check Force Edit to display a screen that lets you edit the item's information. You can make changes to an item, even though it may be an exact match with a local item. This new information will be applied to the item after you complete the import.

Select: After importing, you can manually match an unmatched item. To do this, highlight the tag in the middle pane, the type matching information in the Search field in the upper right corner of the screen. Choose the matching item and click Select. This yields an unverified match (yellow bullet). To confirm the match, click Match on the right.

Complete the import

1. in the middle pane, right-click the first item that you want to review or change, and then select the option for how you want the import to handle this item.

Each item's status controls the options you will see:

Item Status	Right-Click Options	Description
Ignored	New	The custom tag screen opens for you to add the attributes for a new tag.
Unmatched	Ignore, New	Ignore: Changes the status so that the import will exclude this tag. New: The custom tag screen opens for you to add the attributes for a new tag.
Matched	Ignore, Set to Unmatched	Ignore: The import will exclude this tag. Set to Unmatched: The tag is no longer matched; but the import will not succeed. All unmatched tags must be matched or ignored before you complete the import.
Partial Match	Ignore, Set to Unmatched, Match	Ignore: The import will exclude this tag. Set to Unmatched: The tag is no longer matched; but the import will not succeed. All unmatched tags must be matched or ignored before you complete the import. Match: The tag attributes will change to the information you see in the Results column.

2. Continue through all of the items until you have set the match status for each one.
3. Click **Complete Import**.

Using import templates

You can create, edit, and apply templates when you import files. You can also delete import templates. A template will include tags that you import into the Profile Editor from a project in another instance of the Profile Editor, or it will contain ICD files from an IEC 61850 device.

You will want to create a template for custom situations, like when you are importing SCL files or adding custom tags and devices.

Creating a new template during import

To create a new template:

1. From the either type of import (Profile Editor or SCL), choose the file (.pls or .icd) that you want to import.
2. Complete the matching for the items.
3. Click **Complete Import**.
4. At the Save Import Template prompt, click **Yes**.
5. Click the **New** radio button, then type a name for the new template. The name must begin with a letter. It can contain alpha-numeric characters, as well as dashes and spaces. Click **OK** to save it.

In future imports, you will be able to apply this template. When you do, the system will automatically match, where appropriate, the import items with the local items.

Applying a template during import

In this procedure, you will import files, and you will either create a new import template, or you will edit an existing one.

NOTE: Be careful when applying a template; you will overwrite an existing template on the local computer with the information that you choose during matching. Once completed, you cannot undo this.

To apply an existing template:

1. From the **Set Up Projects** tab, select a project for which you want to import data.
2. Click **File > Import** and then choose the file (.pls or .icd) that you want to import.
3. From the Import Template drop down list, choose the template you want to use. This is just a starting point for this import to make it quicker to match items. You will apply the template in step 7.
4. Click **Start Import**.

   After the import completes, the Import Reconciliation screen displays. The list in the left-hand pane should have some exact and partial matches.
5. As you work through the items, you must either designate that each a match or ignored.
6. When all items are completed, click **Complete Import**.
7. At the Save Import Template dialog, click **No** to import without applying a template. Or click **Yes** to either save a new template or edit the one you chose in step 3:
   - To create a new template for this import, click **New**, then type an Import Template Name.
   - To edit a template, click **Edit**, then select the template from the drop down menu. This will edit the template by adding the changes you made during matching. This cannot be undone after you click **OK**.
8. Click **OK**.

The import is completed, and the new template is created, or the existing template is edited to include the changes you made during matching.

Deleting a template

You can delete any import template, even if it was applied during a previous import.

To delete a template:

1. Click **Settings > Remove Import Templates**.
2. At the Import Templates dialog, select the template you want to delete and then click **Delete**.

The template is deleted.

Managing I/O devices in a project

Use the I/O Device Manager to create, remove, or update devices.

The first three options send you to a wizard that walks you through creating, removing, or updating.

Click one of the following links a link for what you want to do:

- "Define one I/O device in a project" on page 244
- "Remove an I/O device from a project" on page 249
- "Define multiple devices using a CSV file" on page 250
- "Update devices in a project" on page 255

For information on how to translate device information, see "Translating device information" on page 377.

Before adding I/O devices

Have a copy of each device's communication protocol and IP address. You will need to enter this information when you add the devices.

NOTE: You can use IPv6 IP addresses – including IPv6 shorthand – for TCP/IP level drivers. However, the ION protocol does not support IPv6.

For each cluster and the appropriate servers for this project (see the Citect SCADA help file for details)

For each cluster :

1. From the I/O Device Manager, under System Devices, click **Cluster Setup** and then click **Next**.
2. At the Enter Instance Information screen, a cluster name displays. Click **Next**.

   If there are multiple clusters, the Select cluster screen displays
3. Choose the cluster you want to set up and then click **Next**.

   If there are multiple I/O servers, the Select I/O Servers screen displays.
4. Select an I/O Server. (Optional) If you are developing a redundant system:
   a. Check **Supports Redundancy** and select the I/O servers to which you want to add the device.
5. Click **Next**.
6. At the Ready to perform action screen, click **Next**.
7. If you have more than one cluster to add, repeat steps 3 through 6 for each cluster.
8. When you are finished adding clusters and I/O servers, you return to the I/O Device Manager welcome screen.

Port names

The Profile Wizard does not take into account that multiple projects might be 'linked together' via a global include project. For instance, it does not allow you to specify a unique port name and port number, such that they will not conflict with other projects.

There are three possibilities:

- Protocols that support port name changes: includes Generic TCP and MODBUS TCP
- Protocols that support re-use of ports only: see the table below for protocols and settings that need to match
- Protocols that do not support port name changes: all protocols not mentioned above

The following table shows the settings that must match between the protocols for that column. For example, if you combine two generic serial protocols or a generic serial with a DNP3 via serial, all of the checked items need to match between them.

	Generic Serial, DNP3 via Serial	**MODBUS RTU via Serial**	**DNP3 via TCP/IP, IEC 60870-5-104 via TCP/IP, MODBUS RTU via Gateway**
Board Type	X	X	X
I/O Server Name	X	X	X
Port Number	X	X	
Baud Rate	X	X	
Data Bits	X	X	
Stop Bits	X	X	
Parity	X	X	
IP Address			X
Network Port Number			X
All attached I/O devices must use the same protocol.		X	X

Using the Port Settings page in the Profile Wizard, you can name ports. See "Define one I/O device in a project" on page 244 for more information.

Add Redundant NetworkTagsDev and zOL Devices

For systems with redundant I/O devices, you will need to create redundant NetworkTagsDev and a zOL device.

1. Open the I/O Device Manager.
2. Select **Create an I/O Device** in the project.
3. Under **System Devices**, choose **Cluster Setup**.
4. Accept the default device/equipment names.
5. Check **Supports Redundancy**.
6. Set the primary server to one of the available I/O servers.
7. Set the standby server to one of the I/O servers on a different network address.
8. Allow to finish and select Add/update/remove more devices.
9. Select Create an I/O Device in the project.

10. From System Devices choose OneLine Device Setup.
11. Accept the default device/equipment names.
12. Finish and close I/O Device Manager.

Define one I/O device in a project

Use the I/O Device Manager Wizard to add one device at a time.

Throughout the I/O Device Manager , there are fields that will only accept a valid entry. They are marked with a red exclamation point (!). The exclamation point remains there until you enter a response that is of the correct length or includes only the acceptable characters. The asterisk disappears after you enter a valid response.

Opening the I/O Device Manager Wizard

To open the I/O Device Manager Wizard:

1. In Power SCADA Studio: Click **Projects > Home** and verify that the project to which you want to add the devices is active.
2. Click **Topology > I/O Devices >**I/O Device Manager.

   The I/O Device Manager displays.
3. Click **Manage a Single Device**.

   The I/O Device Manager Wizard displays.

The steps to add a device vary by protocol. Click one of these links to display instructions to add each type of protocol:

- "Adding a TCP device" on page 244
- "Adding a serial device" on page 246
- "Adding a DNP3 TCP device" on page 247
- "Adding an IEC 61850 device" on page 248

For each device added using the I/O Device Manager wizard, follow the same redundancy steps outlined in "Add Redundant NetworkTagsDev and zOL Devices" on page 243. Be sure to select a primary I/O Server and a standby I/O Server, each from a different Network Address.

Adding a TCP device

NOTE: These instructions assume that you have two I/O Servers, and that you will be renaming ports.

To add a TCP device to a project:

1. In Power SCADA Studio: Click **Projects > Home** and verify that the project to which you want to add the devices is active.
2. Click **Topology > I/O Devices >**I/O Device Manager.

   The I/O Device Manager displays.
3. Click **Manage a Single Device**.

   The I/O Device Manager Wizard displays.

4. Click **Create an I/O Device** in the project and then click **Next**.
5. At the Choose profile screen, select the first device profile that you want to use to add a device to the project. Click **Next**.

   NOTE: To ensure that the Alarm Log displays properly with the PM5000 series devices, use the correct PM5000S or PM5000S1 driver for devices.

   Use the PM5000S driver (for the most recent Alarm Log implementation) with:
   - PM51XX
   - PM53XX
   - PM55XX
   - PM5350PB
   - PM5350IB with FW version 3.00 and higher

   Use the PM5000S1 driver (for previous Alarm Log implementation) with:
   - PM5350 with FW prior to version 3.00
6. At the Enter instance information screen, type a descriptive profile name, for example: `CM4Bay1Circuit1` (no spaces or punctuation; to allow space in Power SCADA Operation, the preferred limit is 16 characters). The Comment field is stored in the `equipment.dbf` file. Click **Next**.
7. At the Select I/O servers screen, choose the primary and standby servers. You can only set the standby server if you click **Supports Redundancy**. Click **Next**.
8. If you choose to add an optional sub-profile: At the Configure Sub-Profile Communications Method screen, choose the communications method used for the first sub-profile in this project. Click **Next**.
9. At the Communications Settings screen, type the gateway address and station address for each of the servers. If you click **Same as Primary** for standby, you will use the same addresses for the primary and standby. Click **Next**.
10. At the Port Settings screen, you can rename each of the ports. A new port will be generated for each new name. Click **Next**.
11. At the Ready to perform action screen, click **Next**.

    After the devices are added, a screen displays telling you that the project was updated successfully.
    - To view a detailed list of all the devices and all operations performed in the project, click **View audit log**. The list displays after the device is added.
    - To continue adding or removing devices, click **Next**. Repeat steps 3 through 10.
12. When you have finished adding devices, uncheck **Add/remove more equipment**, then click **Finish**.

    If you clicked **View audit**, the list displays.

If you did not click **Add/remove**, the I/O Device Manager closes. If you clicked **Add/remove**, the Welcome screen displays again.

13. "Compile the project" on page 256. Correct any compile errors and then compile the project again.
14. Click **Run** to view the runtime environment.

Adding a serial device

NOTE: These instructions assume that you have two I/O Servers, and that you will be renaming ports.

To add a serial device to a project:

1. From the Power SCADA Studio screen, display the project to which you want to add the device.
2. Click Topology > I/O Devices >I/O Device Manager.

   The I/O Device Manager welcome screen displays.
3. Click **Create an I/O Device** in the project, then click **Next**.
4. At the Choose profile screen, select the first device profile that you want to use to add a device to the project. Click **Next**.
5. At the Enter instance information screen, type a descriptive profile name, for example: *CM4Bay1Circuit1* (no spaces or punctuation; to allow space in Power SCADA Operation, the preferred limit is 16 characters). The Comment field is stored in the equipment.dbf file. Click **Next**.
6. At the Select I/O servers screen, choose the primary and standby servers. You can add information for the standby server if you click **Supports Redundancy**. Click **Next**.
7. If you choose to add an optional sub-profile: At the Configure Sub-Profile Communications Method screen, choose the communications method used for the first sub-profile in this project. Click **Next**.
8. At the Communications Settings screen, enter the information for each server (com port, baud rate, and so on). If you click **Same as Primary** for standby, you will use the same addresses for the primary and standby. Click **Next**.
9. At the Port Settings screen, you can rename each of the ports.
10. When you finish adding the last sub-profile, the Ready to perform action screen displays. Click **Next**.

    After the devices are added, a screen displays telling you that the project was updated successfully.

    - To view a detailed list of all the devices and all operations performed in the project, click **View audit log**. The list displays after the device is added.
    - To continue adding or removing devices, click **Next**. Repeat steps 3 through 10.
11. When you finish adding devices, click Finish at the Project updated successfully screen.

    If you clicked **View audit**, the list displays.

If you did not click **Add/remove**, the I/O Device Manager closes.

12. "Compile the project" on page 256. Correct any compile errors and then compile the project again.
13. Click **Run** to view the runtime environment.

Adding a DNP3 TCP device

NOTE: These instructions assume that you have two I/O Servers, and that you will be renaming ports.

To add a DNP3 TCP device to a project:

1. From the Power SCADA Studio screen, display the project to which you want to add the devices.
2. Click Topology > I/O Devices >I/O Device Manager.

   The I/O Device Manager welcome screen displays.
3. Click Create an I/O Device in the project, then click **Next**.
4. At the Choose profile screen, select the first device profile that you want to use to add a device to the project. Click **Next**.
5. At the Enter instance information screen, type a descriptive profile name, for example: *CM4Bay1Circuit1* (no spaces or punctuation; to allow space in Power SCADA Operation, the preferred limit is 16 characters). The Comment field is stored in the equipment.dbf file. Click **Next**.
6. At the Select I/O servers screen, choose the primary and standby servers. You can only set the standby server if you click **Supports Redundancy**. Click **Next**.
7. If you choose to add an optional sub-profile: At the Configure Sub-Profile Communications Method screen, choose: At the Configure Sub-Profile Communications Method screen, choose the communications method used for the first sub-profile in this project. Click **Next**.
8. At the Communications Settings screen, type the IP address, port number, and device address for each of the servers.

   NOTE: You can use IPv6 IP addresses for TCP/IP level drivers. However, the ION protocol does not support IPv6.

   NOTE: The DNP3 port number is 20000. You must type 20000 here for communications to work correctly.

   If you click **Same as Primary** for standby, you will use the same addresses for the primary and standby. Click **Next**.
9. At the Port Settings screen, you can rename each of the ports. A new port will be generated for each new name. Click **Next**.
10. At the Ready to perform action screen, click **Next**.

    After the devices are added, a screen displays telling you that the project was updated successfully.

- To view a detailed list of all the devices and all operations performed in the project, click **View audit log**. The list displays after the device is added.
- To continue adding or removing devices, click **Next**. Repeat steps 3 through 10.

11. When you have finished adding devices, uncheck **Add/remove more equipment**, then click **Finish**.

    If you clicked **View audit**, the list displays.

    If you did not click **Add/remove**, the I/O Device Manager closes. If you clicked **Add/remove**, the Welcome screen displays again.

12. "Compile the project" on page 256. Correct any compile errors and then compile the project again.
13. Click **Run** to view the runtime environment.

Adding an IEC 61850 device

NOTE: These instructions assume that you have two I/O Servers, and that you will be renaming ports.

To add an IEC 61850 device to a project:

1. From the Power SCADA Studio screen, display the project to which you want to add the devices: In the upper left corner of the screen, choose the project from the drop-down menu.
2. Click Topology > I/O Devices >I/O Device Manager.

   The I/O Device Manager welcome screen displays.
3. Click **Create an I/O Device** in the project, then click **Next**.
4. At the Choose profile screen, select the first device profile that you want to use to add a device to the project. Click Next.
5. At the Enter instance information screen, enter a descriptive profile name, for example: *Bay1Circuit1* (no spaces or punctuation; to allow space in Power SCADA Operation, the preferred limit is 16 characters). The Comment field is stored in the equipment.dbf file.

   LDName

   In the Additional Information section at the bottom, you can change the original logical device names for the IED. This is required only if the logical device name was changed in the SCL file that was imported into the Profile Editor.

   BRCBs and URCBs

   In the Additional Information, you can also enter BRCB or URCB information. BRCBs (buffered report control blocks) and URCBs (unbuffered report control blocks) can be used to return data in blocks rather than in individual tags. To enter either one, you need to have downloaded an SCL file for the device in question. When you click the line to add data, you must browse to the SCL file and select the BRCB/URCB you want. You will need the logical device, logical node, and RCB names. The Help column gives examples of the formatting that is required.

   Click **Next**.

6. At the Communications Settings screen, browse to the location where you have saved the SCL file. If there is only one IED, it displays automatically; otherwise, choose the correct device. Click **Next**.
7. At the Ready to perform action screen, click **Next**.

   After the devices are added, a screen displays telling you that the project was updated successfully.

   - To view a detailed list of all the devices and all operations performed in the project, click **View audit log**. The list displays after the device is added.
   - To continue adding or removing devices, click **Next**. Repeat steps 3 through 7.
8. When you have finished adding devices, uncheck **Add/remove more equipment**, then click **Finish**.

   If you clicked **View audit**, the list displays.

   If you did not click **Add/remove**, the I/O Device Manager closes. If you clicked **Add/remove**, the Welcome screen displays again.
9. "Compile the project" on page 256. Correct any compile errors and then compile the project again.
10. Click **Run** to view the runtime environment.

Remove an I/O device from a project

To remove an I/O device:

1. Open the I/O Device Manager application.
2. Click Remove one I/O device, then click **Next**.
3. At the Remove a device screen:
   a. Click the drop down menu to display the equipment names that were used when the device profiles were previously added to the project.
   b. From this list, select the device that you want to remove. Click **Next**.
4. At the Ready to perform action screen:
   a. (Optional) Compress the project files after removing this profile, click **Pack databases**.
   b. Click **Next**.

      After the device is deleted, a screen displays telling you that the project was updated successfully.

      - To view a detailed list of the devices that you added or deleted, click **View audit log**. The list shows all the device data that was added, as well as the data that was removed in this session. (The list displays after you click **Finish**.)
      - To remove additional devices, click **Add/remove more devices**, then click **Next**.
      - Repeat steps 3 and 4.
5. When you have finished removing devices, uncheck **Add/remove more equipment**, then click **Finish**.

If you clicked **View audit**, the list displays.

If you did not click **Add/remove**, the I/O Device Manager closes. If you clicked **Add/remove**, the Welcome screen displays again.

6. "Compile the project" on page 256. Correct any compile errors and then compile the project again.

Define multiple devices using a CSV file

I/O Device Manager makes it easy to create a Power SCADA Operation project. Use this tool to make either single or bulk additions, updates, and deletions to the Power SCADA Operation device database.

Valid communication protocols are:

- DNP3 Serial
- DNP3 Ethernet
- Modbus/RTU Gateway
- Modbus TCP
- ION
- ION/Gateway
- IEC60870-5-104 TCP
- IEC61850

You first need to create the CSV file that you will use to add the devices. See "Adding a comment" on page 253 for details.

After you create the CSV file, you use it to add multiple devices to the project. See "Status Options" on page 253 for details.

Create a CSV file to add multiple devices

You can create a CSV file to add multiple devices to the project.

Use the sample CSV files as templates to create your own CSV file. For more information the sample CSV device files, see "CSV file samples" on page 255.

TIP: You can edit the CSV file to remove unused columns, or to drag and drop columns to position them where they are easy to read.

Before you begin

- For existing projects: Make a backup copy of your project.
- For a new project: In the Power SCADA Studio, add a new project, define a cluster; and add alarm, trend, and I/O servers. See "Before adding I/O devices" on page 242 for details.

To create a CSV file to add multiple devices:

1. In the Profile Editor, create and export a project that includes the device types and profiles included in this installation.

2. In Excel, Open Office, or other .CSV file editor, open the example CSV file for your device type. The files are named "exampleXX," where XX is the device type, such as ION or Modbus TCP. These files are in the Windows Program Data file:
3. Program Data > Schneider Electric\Power SCADA Operation\9.0\Examples.
4. In the sample CSV worksheet, for each device that you want to add enter the following information:
   a. ProfileName: the name of the profile that has been exported from the Profile Editor into the target Power SCADA Operation project. Type the names of the profiles that have been selected for this project. To view names, open the Profile Editor utility.
   b. Name: Enter the device name, limit of 32 characters; include only letters, numbers, and underscores (_). The first character cannot be a number or underscore. This field becomes the "Name" on the I/O Devices screen and the "I/O Device" name on the Equipment screen.
   c. Cluster: The name of the cluster to which the device will be added. If there is only one cluster in the project, this column is not required.
   d. Equip: Enter the equipment name, limit of 40 characters; include only letters, numbers, and periods (.). The first character cannot be a number or period. This field becomes the "Name" on the Equipment screen. You will use this when adding genies to drawings.
   e. Primary IO Server Name: The name of the primary I/O Server for the device. If there is only one I/O Server in the project, this field is not required.
   f. CommsMethod: Type the communications protocol being used, e.g., MODBUS/RTU via Gateway. See list below for alternate communication connections. When using a composite device, do not use this field. You must enter a "SubProfile1Description" (and a "SubProfile2Description" for the second part of the composite device).

      NOTES: If the CommsMethod column is missing and you define more than one CommsMethod in the project:
      - If one of them is Modbus/RTU via Gateway, it will be used.
      - If one of them is ION it will be used (if there is no Modbus/RTU via Gateway).
      - If the CommsMethod column is missing and you define only one CommsMethod for the project, it will be used.

      DNP3 Serial
      DNP3 TCP
      Modbus/RTU via Gateway
      Modbus/TCP
      ION
      ION/EtherGate
      IEC60870-5-104 TCP
      IEC61850 Native
   g. PrimaryIPAddress: Type the IP address for the primary server (required only for MODBUS/RTU and MODBUS/RTU via Gateway).

NOTE: You can use IPv6 IP addresses – including IPv6 shorthand – for TCP/IP level drivers. However, the ION protocol does not support IPv6.

h. PrimaryEquipmentAddress: Type the device address (required only for MODBUS/RTU and MODBUS/RTU via Gateway).

i. PrimaryPortName: Type the port name of the primary server (required only for MODBUS/RTU and MODBUS/RTU via Gateway).

j. Standby IO Server Name: If you have a redundant I/O server, type the name here.

k. StandbyIPAddress: If you have a redundant I/O server, type its IP address.

l. StandbyEquipmentAddress: If you have a redundant I/O server, type the device address (required only for MODBUS/RTU and MODBUS/RTU via Gateway).

m. StandbyPortName: If you have a redundant I/O server, type the device port name (required only for MODBUS/RTU and MODBUS/RTU via Gateway).

n. Columns that begin with "SubProfile" followed by a number (e.g., SubProfile1, SubProfile2, SubProfile3, etc.) are used to provide the same information as the Primary and Standby columns for composite devices where each SubProfile is a specific device which is part of the larger composite device.

o. PrimaryPortNumber: Type the port number of the primary server (required only for MODBUS/RTU and MODBUS/RTU via Gateway).

p. PrimaryComPort: zzzzzzzz

q. PrimaryBaudRate: xxxxxxxx

r. PrimaryDataBits: xxxxxx

s. PrimaryStopBits: xxxxxx

t. PrimaryParity: asdaafds

u. Primary SclFileName: For IEC 61850 Native, the address where the CID (SCL) file is stored.

v. Primary IedName: For IEC 61850 Native, the name of the IED in the CID file. This was created when the profile was added in the Profile Editor.

w. FTPHost: For IEC 61850 Native, the on-board FTP. Not currently used in Power SCADA Operation.

x. FTPUserName: For IEC 61850 Native, the username for FTP on the device.

y. FTPPassword: For IEC 61850 Native, the password for FTP on the device.

z. BRCBS/URCBS: For IEC 61850 Native, buffered report control blocks (BRCBs) and unbuffered report control blocks (URCBs) can be used to return data in blocks, rather than in tags. These two fields provide the instruction used for each. The two examples in the example are:

BRCB: CFG/LLN0BRBRep01,CFG/LLN0BRBRep06
and
URCB: CFG/LLN0BRBRep01,CFG/LLN0BRBRep06

aa. Optional Parameters: Used for composite devices only.

ab. Parameter Values: This is optional, and is used in functional addressing. This column includes pipe ("|") delimited values for each of the Optional Parameters.

5. Comment: This is an optional description of the device; maximum 254 characters.
6. Close the example CSV file, if it is open.

See "Status Options" on page 253 for information on how to add the devices from this .CSV file to your Power SCADA Operation project.

Adding a comment

You can add a comment row that will be ignored during processing. To create a comment, begin the row with a double forward slash (//). Power SCADA Operation skips this line as it processes the device information. See the example below. In the example, lines 5 and 10 will be skipped.

	A	B	C	D	E	F	G	H	I	J
1	ProfileNa	Name	Cluster	Equip	Primary IC	CommsMe	PrimaryIP	PrimaryEc	PrimaryPc	Primar
2	// These are the meters for Plant 1									
3	PM870 Ful	test	c1	equip1	IOServer1	Modbus/F	10.10.10.1	3		P9
4	PM870 Ful	test2	c1	equip2	IOServer1	Modbus/F	10.10.10.1	3		P9
5	PM870 Ful	PM870_1	c1	Abcdefghi	IOServer1	Modbus/F	10.10.10.1	3		P9
6	Circuit Mc	CM4000_1	c1	ABCDEFGH	IOServer1	Modbus/F	10.10.10.1	3		P2
7	PM870 Ful	PM870_2	c1	powerme	IOServer1	Modbus/F	10.10.10.1	1		P21
8	Circuit Mc	CM4000_2	c1	monitor2	IOServer1	Modbus/F	10.10.10.1	1		P3
9	PM870 Ful	PM870_3	c1	powerme	IOServer1	Modbus/F	10.10.10.1	2		P21
10	// These are the meters for Plant 2									
11	Circuit Mc	CM4000_3	c1	monitor3	IOServer1	Modbus/F	10.10.10.1	25		P4
12	PM870 Ful	PM870_4	c1	powerme	IOServer1	Modbus/F	10.10.10.1	3		P21
13	Circuit Mc	CM4000_4	c1	monitor4	IOServer1	Modbus/F	10.10.10.1	26		P4
14	PM870 Ful	PM870_5	c1	powerme	IOServer1	Modbus/F	10.10.10.1	5		P21
15	Circuit Mc	CM4000_5	c1	monitor5	IOServer1	Modbus/F	10.10.10.1	27		P4
16										

Add multiple devices to a project using a CSV file

To use a CSV file to add multiple devices to a project, you need to be on the same computer as the Power SCADA Operation server, and you must have created and exported your project from the Profile Editor. You also need the CSV file that you previously created (see "Adding a comment" on page 253). Do not have your project running in runtime. You will need access to the following files:

- INI file for your project
- Equipment.Profile file for your project
- CSV file from which you want to add/update/remove data

Status Options

In the upper right corner of the screen, you see the following:

- Display options: Click any of the boxes to cause the corresponding message types (such as error and warning information) to display during and after the automation process.

- Automation status: In blue copy, the most recent activity displays, such as "Validation Complete: data is valid".
- Clear button: Click to clear the message lines from the right pane.

The Automation Process

To run batch changes related to a specific CSV file:

1. Open Manage I/O Devices tool: From the Power SCADA Studio, click Topology > I/O Devices.

   On the new screen, the Project Name field displays your project name. If there are multiple projects, it displays the first one in alphabetic order.

2. Choose the correct project.

   The Citect INI file and Equipment profile are automatically selected, based on the project.

3. Input CSV defaults to the current directory. If you stored the CSV elsewhere, browse to the location where it is saved.

4. Choose the action you want to perform:

Action	Description
Adding Devices	Use to add devices that you have defined in the CSV file.
Removing Devices	Use to remove devices from the project You only need the ProfileName and Equip columns for this action.
Updating Devices	Use to update tag associations for a device if the device profile has changed. You only need the ProfileName and Equip columns for this action. (Note: This action does not update the IP address or other device information. If these attributes are not correct, you need to remove, and then re-add, the device.).
Updating Profiles	Use to update the tag associations for all the devices in the specified profile (s). You only need the ProfileName and Equip columns for this action.

In this case, the action chosen is **Adding Devices**.

5. Click **Validate**.
6. On the new screen, in the right-hand pane, note that the data is valid.
7. If there are errors or warnings, they display in the Messages pane, and a specific line number is indicated.
8. After you validate, you can perform the action that you just validated. The following steps use adding devices as an example.
9. Do not change the project name or file locations. Click the appropriate action (in this case, Add Devices).

   NOTE: Before any action is performed, a validate is performed. If issues are detected, you will be prompted to choose whether you want to continue the action. If you continue, lines with issues will not be processed.

After the action is processed, you see a screen that indicates that you successfully added two devices.

If you are unable to validate or perform the desired action, read the right-hand pane. Errors and warnings will help you troubleshoot the issue.

10. Compile and then run your project. Verify communication for all the devices listed in the spreadsheet.

Exporting CSV Files

You can export information from the project file such as variable tags, clusters, and equipment.

To export information from the project file:

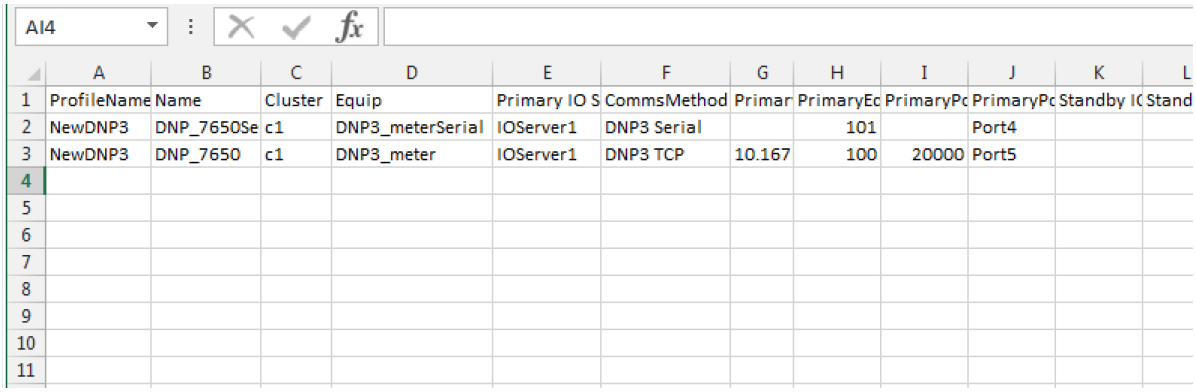
1. At the bottom left part of the window click **Export**.
2. Choose the location at which you want to store the files, and then click **OK**.

CSV file samples

Create CSV files to add multiple devices at once. The following files are samples of files that can be used for some of the various communication protocols.

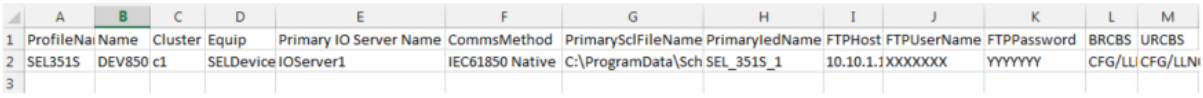
For more information defining multiple devices, see "Define multiple devices using a CSV file" on page 250.

DNP3 for Serial and Ethernet



	A	B	C	D	E	F	G	H	I	J	K	L
1	ProfileName	Name	Cluster	Equip	Primary IO S	CommsMethod	Primar	PrimaryEc	PrimaryPc	PrimaryPc	Standby IC	Stand
2	NewDNP3	DNP_7650Se	c1	DNP3_meterSerial	IOServer1	DNP3 Serial		101		Port4		
3	NewDNP3	DNP_7650	c1	DNP3_meter	IOServer1	DNP3 TCP	10.167	100	20000	Port5		
4												
5												
6												
7												
8												
9												
10												
11												

IEC104.2



	A	B	C	D	E	F	G	H	I	J	K	L	M
1	ProfileNa	Name	Cluster	Equip	Primary IO Server Name	CommsMethod	PrimarySclFileName	PrimaryIedName	FTPHost	FTPUserName	FTPPassword	BRCBS	URCBS
2	SEL351S	DEV850	c1	SELDevice	IOServer1	IEC61850 Native	C:\ProgramData\Sch	SEL_351S_1	10.10.1.1	XXXXXXX	YYYYYYY	CFG/LL	CFG/LLN
3													

IEC61850

Update devices in a project

Update a profile and add it back to the project

This feature works only if the device was added in version 7.20 or later.

After you add devices to the project, and you make changes to the device in the Profile Editor (for example, you add a large number of tags), you can use the I/O Device Manager to bring the changes in the project.

NOTE: If you have made manual changes to the profile in Power SCADA Studio, do not use this process: you could corrupt your data. You must delete the device from the project, re-export it from the Profile Editor, and add it back to the project using the I/O Device Manager.

To use this method of importing changes:

1. Make the changes in the Profile Editor. Make sure you refresh the tags before you continue.
2. Click **Set Up Projects** and then export the project.
3. Open I/O Device Manager.
4. Click **Update one or all I/O devices** and then click **Next**.
5. At the choose update type screen, check whether you want to update all instances in a profile, or just one instance. Click **Next**.
6. Note the two possibilities:
   a. If you selected all instances, choose the profile, and click **Next**.
   b. If you selected one instance, the Update profile instance screen displays. From the drop down list, choose the instance you want to update.
7. At the Ready to perform action screen, note the instance(s) you are about to update. If you want to change your choice, click **Back**.
8. (Optional) To compress the project files in Power SCADA Operation, click **Pack databases after update**.
9. When you have made the update choice you want, click **Next**.
10. When the update is finished, the *Project updated successfully* screen displays. You can view an audit log of changes that have been made, process more changes, or click **Finish** to close the I/O Device Manager.

Edit a device in Power SCADA Operation Only

If you entered incorrect information when you added the device to the project:

1. Delete the device from the project: Use the “Remove a device from the project” feature in the I/O Device Manager.
2. In the I/O Device Manager, add the device back to the project.

Add device data in Power SCADA Operation only

If you need to add a small amount of data to a device that is in the project (e.g., add a single tag), add it directly in Power SCADA Operation. Be sure that you also add it to the device in the Profile Editor so that it is available for other devices in the future.

Compile the project

In Power SCADA Studio, click **Compile**  . If you are promoted to save your changes, click **Save**.

If there are errors or warnings after the project is compiled:

1. At each error, click **GoTo**, which opens the location where the error occurred.
2. Using the information in the error message, correct the error.
3. After all errors are addressed, re-compile to verify that the errors are removed.

   For additional information, click Help at the error screen.

Work with alarms

In this section, you will find these topics:

Alarms overview

This section discusses two alarm types: time-stamped analog and time-stamped digital. To access the alarms, from Power SCADA Studio, select the project folder, then click Alarms. In the right-hand pane, the alarm types display. Double-click the one you want to view/edit.

PC-based alarms

PC-based alarm tags are added in the Profile Editor, when adding each device profile. See "Add edit or delete device profile" on page 211 for instructions. For instructions on entering setpoints and delays, see "Add setpoints and delays" on page 257.

Onboard alarms

If onboard alarms have been configured in a supported device, you can use the Profile Editor to map these alarms to digital time-stamped alarms in Power SCADA Operation.

You cannot configure new onboard alarms from Power SCADA Operation. You must add the alarm at the device, then you can create the alarm tag for it here. See "Add an onboard alarm tag" on page 261.

Add setpoints and delays

Any time you change setpoints, you should immediately restart the project. Otherwise, setpoints will not be properly read (they will be truncated and either rounded down or up to a whole integer).

NOTE: Before you enter setpoints and delays, ensure that you have configured the Alarm Server so that Publish Alarm Properties is set to TRUE.

There are 2 ways to add setpoints and delays for analog alarms:

- From the Analog Alarms window (accessible from the Project Explorer or Project Editor screens), you can type the setpoint and delay values for each alarm.
- In Power SCADA Runtime, you can edit setpoints/delays that were set by the method above.

Also, set the following parameter to allow persisting of alarm parameters at runtime:

[Alarm] UseConfigLimits = 1

Set up an alarm based on an enumeration

To define an enumeration in the Profile Editor, see "Define an enumeration" on page 202.

An example of an enumeration alarm is:

- 0 = unknown
- 1 = good
- 2 = warning
- 3 = alarm

To add an alarm that is based on an enumeration:

1. Open the analog alarm in Power SCADA Operation.
2. To alarm on states 0, 2, and 3:
   - Set Low = 1 (if the value < 1, the alarm indicates an unknown state)
   - Set High = 1 (if the value > 1, the alarm indicates a warning)
   - Set High High = 2 (if the value > 2, the alarm indicates an alarm)
3. In the Category field, ensure that the correct alarm level is entered (_PLSALM_HIGH, _ PLSALM_MEDIUM, _PLSALM_LOW, _PLSALM_EVENT).
4. Replace the alarm.

Change an alarm severity

To change the severity of an alarm:

1. Open the analog alarm in Power SCADA Operation.
2. In the Category field, ensure that the correct alarm level is entered (_PLSALM_HIGH, _ PLSALM_MEDIUM, _PLSALM_LOW, _PLSALM_EVENT).
3. Replace the alarm.

Waveform management

This chapter discusses how waveforms are stored and associated with alarms. In this section, you will find these topics:

Waveform storage

Waveform records are organized within devices into files. These files are periodically checked for and downloaded as they appear on the device. When downloaded, the files are converted into a Comtrade format on the Power SCADA Operation I/O Server and then stored in a hierarchical fashion.

A single waveform will be stored as follows:

```
<Waveform DB root>\<Cluster-
Name>\<IODeviceName>\Waveforms\<UTCTimestamp>.CFG
<Waveform DBroot>\<Cluster-
Name>\<IODeviceName>\Waveforms\<UTCTimestamp>.DAT
```

Where:

`Waveform DB root` path is configured in the WaveformDB configuration section.

For example,

```
C:\Data\Cluster1\Sepam_IODev\Waveforms\
DST_00000000001203566197_0000000511_utc.CFG
DST_00000000001203566197_0000000511_utc.DAT
```

NOTE: In case of redundant I/O devices, only the name of the primary I/O device will be used when waveform storage path is constructed.

The CFG file is a Comtrade configuration file, and the DAT file is the Comtrade data file. Within the CFG file is a timestamp that reflects the device time start time of the waveform. This time is not adjusted to the I/O Server time zone or daylight saving, but it is stored per the device configuration. The file name has the UTC time in seconds since 1970 of the waveform.

The prefix of waveform file name reflects the type of the waveform. Currently, waveforms of the following types are supported:

DST_	Disturbance waveform
ADT_	Adaptive waveform
SST_	Steady state waveform

If it is detected that the waveform data file has changed while it is being downloaded, the file gets discarded and is not stored on the I/O Server.

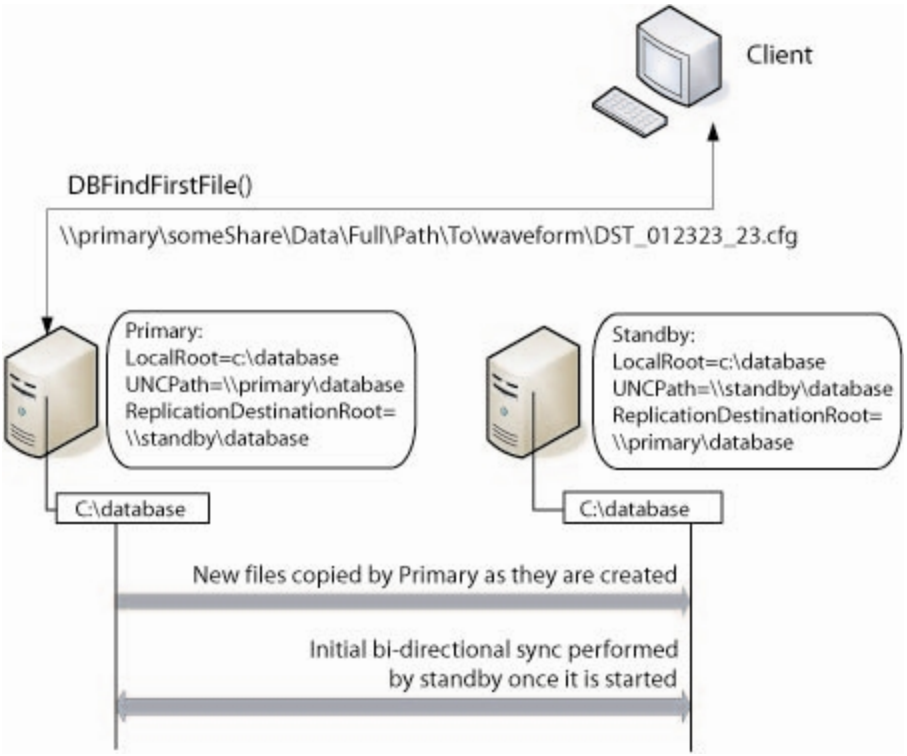
Waveform database and special waveform tags

Power SCADA Operation allows you to browse the waveform database for specific I/O devices. Search for all waveforms within certain time frame is also supported, allowing you to search for all waveforms that could be linked with a given alarm. When you perform this search, a list of all matching waveforms displays. If there are multiple waveforms in the list, you can select the waveform you want to view.

In addition, there are two special digital waveform tags defined (0 = FALSE, 1 = TRUE):

- WaveformDownloading: indicates whether a waveform file is currently being downloaded
- WaveformCollectionEnabled: indicates whether the waveform collection is enabled at all

The following image illustrates a configuration example and replication and linkage processes:



Enable waveforms for onboard alarms

Enabling waveforms for onboard alarms makes them available for viewing in the Power SCADA Runtime.

When an onboard alarm occurs at the device, the waveform is captured. The are transmitted to Power SCADA Operation and are available for viewing. The amount of time this takes depends on the number of I/O Servers you have and the number of serial devices on a chain. On a very large system with numerous serial devices, this could take as long as an hour.

To enable waveforms for onboard alarms:

1. At the device, or via the meter configuration software (PMCU), add the alarm and enable the automatic capture of a waveform when the alarm occurs.
2. In the Profile Editor, on the **Create Device Profiles** tab, for the same alarm you added in PMCU, check the **Waveform** box.

You can view the waveform from the Alarm Log in the runtime environment.

Set parameters for event log length and historical logging of events

You can use two parameters to determine the maximum number of entries in the Event Log and whether you want to log entries after they are FIFO'd out of the Event Log.

Event storage: [Alarm]SummaryLength parameter

The maximum number of alarms that can be stored is controlled by the Alarm Summary length parameter, which defines the maximum number of alarm summary entries (Event Log entries) that can be held in memory. You can view these alarm summary entries on the Alarm Log page. Each

event requires 256 bytes of memory, plus the length of the comment. 32,000 entries will require at least 8 MB of memory. If you have many events, you should ensure that there is enough memory to store them in RAM.

The default value is 5000.

When the value is set to a number greater than 1000 for a multiple-cluster system, the alarm log might not display correctly. The list of alarm history that displays on a client might be shorter than the actual history stored on the alarm server. To avoid this problem, do one or more of the following:

- Set alarm filtering in the alarm viewer to reduce the number of alarms that are returned by the server.
- Only support a one-cluster system.
- If a multiple-cluster system is necessary, display a separate alarm page for each cluster.

Add an onboard alarm tag

When a device onboard alarm has not been included in Power SCADA Operation, you can add it using Profile Editor. You need to follow these steps to include the device's unique identifier. Otherwise, the alarm will not annunciate in the Graphics page.

You can only add onboard alarms for devices using the CM4, PM8, Micrologic, or Sepam drivers. CM4, PM8, and Micrologic unique IDs must be decimal; SEPAM unique IDs must be hexadecimal.

To add an onboard alarm tag:

1. From the device, obtain the following information:
   a. The unique identifier for this alarm. Additionally, for MicroLogic, you need to include the unique sub-identifier.
   b. The file number in which alarms on stored on the device.
2. From the Profile Editor, add the onboard alarm.

Set up audible alarms

You can use a variety of Windows wave files for audible alarms.

To set up audible alarms:

1. Define the alarm sound to be used and the repeat interval for each priority in the alarm you want to be audible. Enter the following information, either in the project parameters (Power SCADA Studio > Setup tab > Parameters) or in the Citect.ini file:
   a. [Alarm]
   b. Sound<priority>=<wave file name>
   c. Sound<priority>Interval=<repeating interval in milliseconds>

   If you add the device using the I/O Device Manager, the alarm priority will be 1, 2, and 3 for _ PLSALM_HIGH, _PLSALM_MEDIUM, _PLSALM_LOW alarms respectively.

   You can define specific wave files for the sounds. The following Windows operating system sounds are supported:
   - SystemAsterisk
   - SystemExclamation

- SystemQuestion
- SystemDefault
- SystemHand
- SystemExit
- SystemStart

After audible alarms are set up

When an alarm occurs, its specified alarm sound will play continuously according to the specified interval. The alarm sound will stop when either:

- The user clicks **Silence Alarm** on the alarm page
- The alarm is acknowledged.

Power SCADA Runtime

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Do not use the software or devices for critical control or protection applications where human or equipment safety relies on the operation of the control action.
- Do not use the software to control time-critical functions.
- Do not use the software to control remote equipment without proper access control and status feedback.

Failure to follow these instructions can result in death or serious injury.

Because Power SCADA Operation lets you set user permissions on runtime graphical objects, thoroughly test the deployed project to ensure that permissions are applied as intended.

The Power SCADA Runtime is where the end user views system information that is added in the design-time pages. The Power SCADA Runtime can include:

- One-line diagram pages with interactive objects
- Alarm and event pages
- Analysis pages (trends and waveforms)
- Basic reports

If Power SCADA Operation includes Advanced Reporting and Dashboards Module, you can configure the Power SCADS Runtime to includes dashboards and advanced reports.

To customize the runtime environment, you can:

"Add a new graphics page" on page 268

"Create a one-line on a graphics page" on page 280

Open firewall ports for Power SCADA Runtime

For the system to properly run, you need to ensure that the following ports are properly set.

Before you begin, define the primary and standby Alarm Servers, Trend Servers, and I/O Servers. Then, to enable communication for runtime operations, use the information in the following tables. Each server has a unique default port assigned to it. Use this default port only with that type of server. If you attempt to use a default port on another type of server, you will see a compilation error:

Invalid port number (2073-2082,20222,21) are reserved.

Default Port Numbers and Associated Server Types

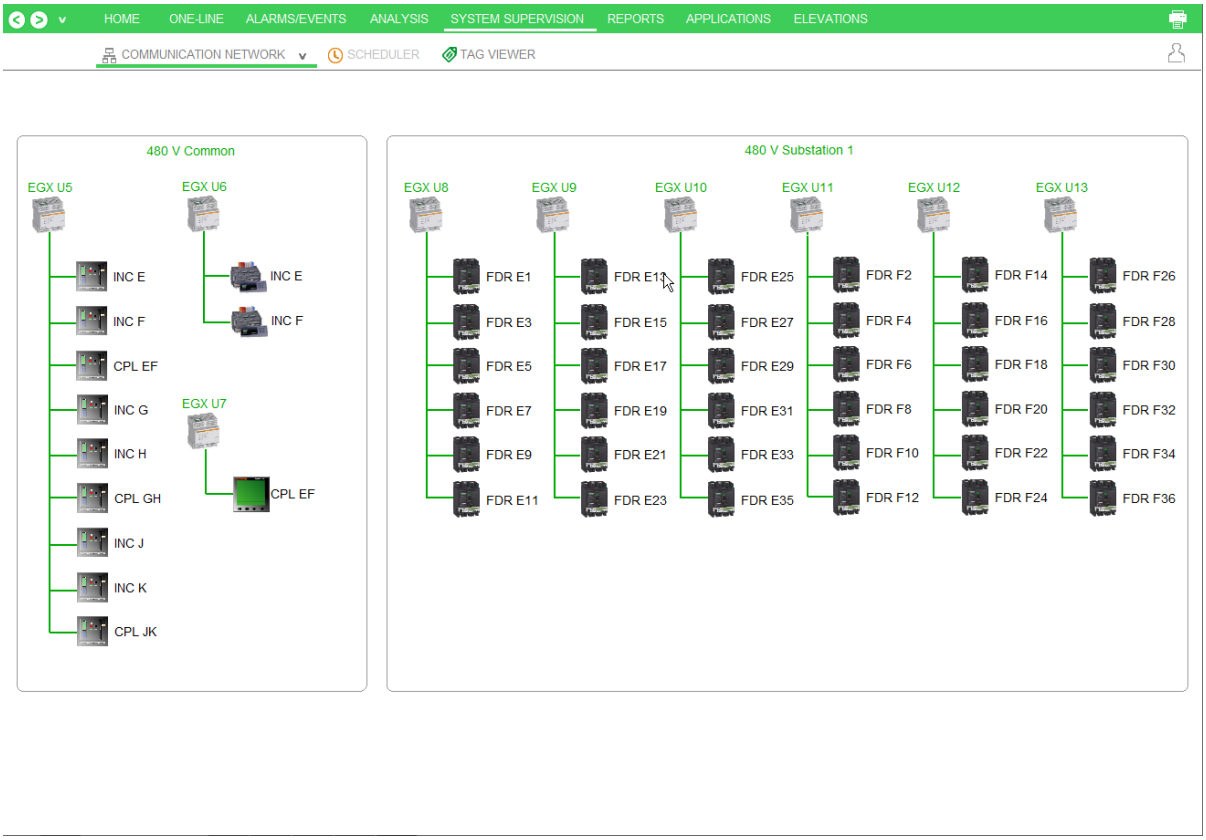
Default Port	Server Type	Server Role
21	FTP Server	Page downloads for IDC
	IDC	Internet Display Server/Client communications
2073	CTAPI	CTAPI communications
2074	Client	Cicode debugging
2084	Reports Server	Reports Server communications
2080	Alarm Server	Alarm Server communications
2085	Trends Server	Trends Server communications
2078	I/O Server	Legacy I/O communications (version 6 or earlier)
2080	Alarm Server	Alarm properties connector
2082	I/O Server	Publish, subscribe I/O Server communications
20222	ODBC	ODBC server
5482	Alarm Server	Database port

SCADA Web Server/Web Client Configuration

Default Port	Server Type	Server Role
80	Web Server	Project files for web client
2084	Reports Server	Reports Server communications
2080	Alarm Server	Alarm Server communications
2085	Trends Server	Trends Server communications
2078	I/O Server	Legacy I/O communications (version 6 or earlier)
2080	Alarm Server	Alarm properties connector
2082	I/O Server	Publish, subscribe I/O server communications
5482	Alarm Server	Database port
5500–5509	Web Client/ Web Server	Range of ports for server advise between web server and web client, for alarm notifications. Inbound on client; outbound on server.

Graphics pages

A *graphics page* provides an interactive view of an installation's power system in the Power SCADA Runtime. A Power SCADA project can consist of one or more graphics pages. Graphics pages are created using Graphics Builder.



Configured graphics pages can include symbols, images, and genies.. A *genie* contains logic such as control outputs and rack-in/rack-out.

Power SCADA Operation includes a standard library of genies. The genies contain logic such as control outputs and rack-in/rack-out. See "Default Genie Library" on page 629 for detailed information on the Power SCADA Operation genies.

This section includes the following topics:

See Citect SCADA help for more information on graphics pages.

Graphics pages prerequisites

Before you create a graphics page make sure you:

- Created a project in the Profile Editor
- Added a project with the same name to Power SCADA Operation; added at least one cluster, network address, and server.
- Exported the project from the Profile Editor
- Used the I/O Device Manager to add devices to the project.

Creating a graphics pages workflow

These instructions assume that you are using the default template. However, if you create your own template, use the Menu Configuration tool to add menu headings that will display on the one-line in runtime. See "Add Pages to Project Menu Configuration" on page 301 for more information about changing the appearance of the graphics page.

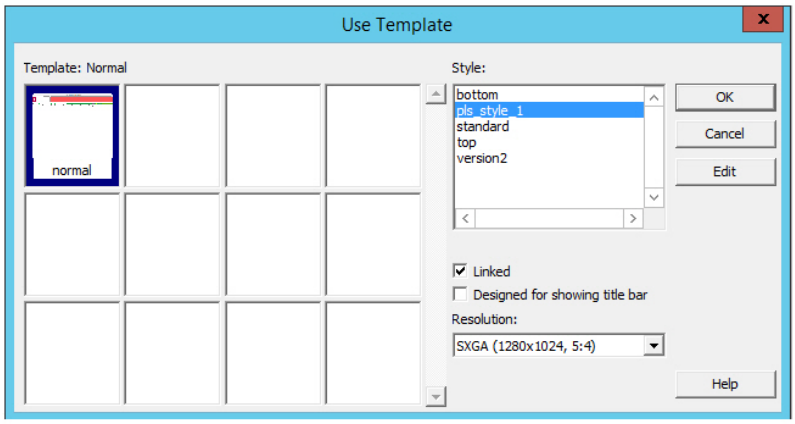
When creating a graphics page, these are the steps you will usually follow:

1. In the Graphics Builder, select style and page template to add a new page.
2. Add genies to the page.
3. Compile and run the project. Note any errors and warnings; correct all errors. Review warnings for problems such as missing tags.
4. Run the project to view the graphics page in the Power SCADA Runtime.

Create a graphics page using a template

To create a graphics page based on the default template in Power SCADA Operation:

1. In the Citect Graphics Builder, click **File > New**.
2. From the New window, click the top option, Page, which displays the Use Template page:



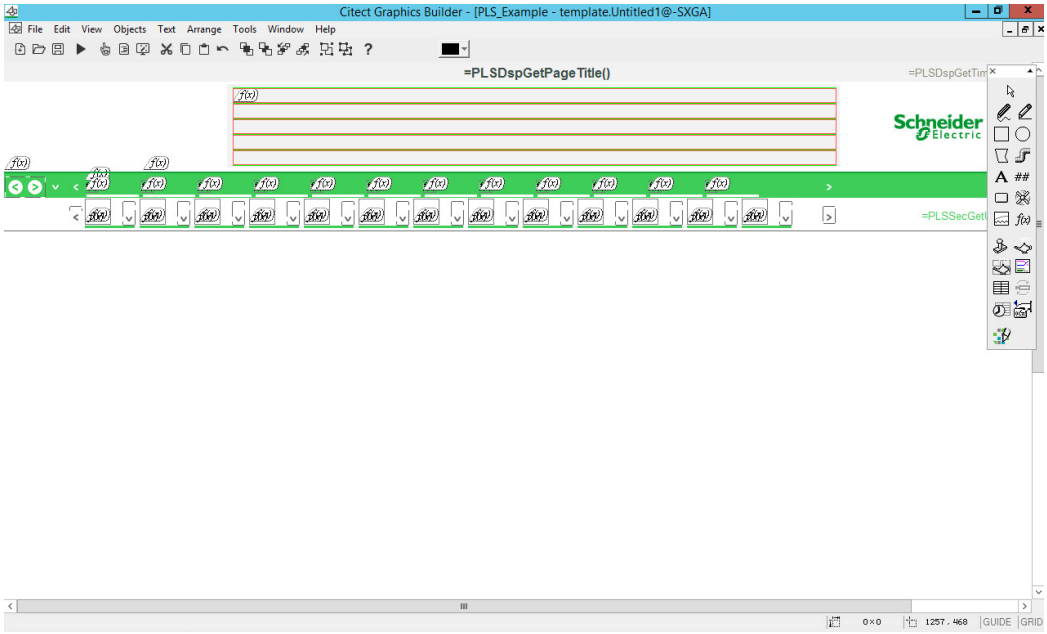
3. Choose the Normal template (shown above).

   Normal includes buttons for basic page control (such as alarms displays and menu pages). There is a large open area to display one-line drawings. (See the table below for a description of other templates that are provided in different styles.)

4. Select the pls_style_1 template from the Style list. It is the most feature-rich template that is designed for the complex projects created in Power SCADA Operation.
5. Select the screen resolution for the graphics page. We recommend that you use SXGA, or at least 1024 x 768. Options are:

Setting	Resolution (height x width)
VGA	640 x 480
SVGA	800 x 600
XGA	1024 x 768
SXGA	1280 x 1024
WUXGA	1920 x 1200
User	user-defined height and width

6. Click **Linked** to maintain the link for the graphics page with the original template. Then, if you change the template, the page will reflect the change.
7. (Optional) Click **Designed for showing title bar** if you want to display the graphics page with the Windows title bar visible. The Windows title bar lets you maximize, minimize and close the window.
8. Click **OK**.



9. Save the page (File > Save As): Type a name, to be used for the page in the runtime view; and choose the project to which you want to add it. Click OK.

Later, when you need to access this page, open it from the File > Open option in the Graphics Builder.

Add a new graphics page

To add a new graphics page:

1. From Power SCADA Studio, choose the project for which you will create a graphic.
2. On the Graphics Builder screen, click **File > New > Page**.
3. On the Use Template popup, set the resolution to SXGA (or default), and uncheck the Title Bar option.
4. Select the page template (for example, normal). Click **OK**.

   NOTE: If you use any style that is not preceded by "pls," the Power SCADA Operation graphics features will not be included. If you want to change the attributes of the default style, copy pls_style_1 and paste it in your project; rename it, and make the desired changes. The new template can then be used for your pages.

   The graphics page displays in design-time mode.
5. To make changes to the menus and tabs that display on the screen, use Menu Configuration tool. See "Use menu configuration to edit pagemenu.dbf (Change the graphics page appearance)" on page 270 for instructions on making these changes.
6. Save the graphics page, giving it the desired name and selecting the project that you have just added.

Set a new page as the project startup page

After you add a new page, follow these steps to make it be the system starting page.

If you do not make these settings correctly, the system will not properly execute the startup routine, and the browser navigation buttons will not work.

1. Open the Parameters dialog (Power SCADA Studio > Setup activity > Parameters).
   a. In **Section Name**, type `MultiMonitors`.
   b. In **Name**, type `StartupPage`.
   c. In **Value**, type the name of the new page.

      If you intend to use multiple monitors, and want to use a different startup page for a particular monitor, you can specify it by adding another parameter with the name of StartupPage<*n*>, where n is the monitor number.

      For example, `StartupPage2` for startup page on the second monitor.
2. In the Computer Setup Wizard (General Options Page), leave the ini settings as <default>.

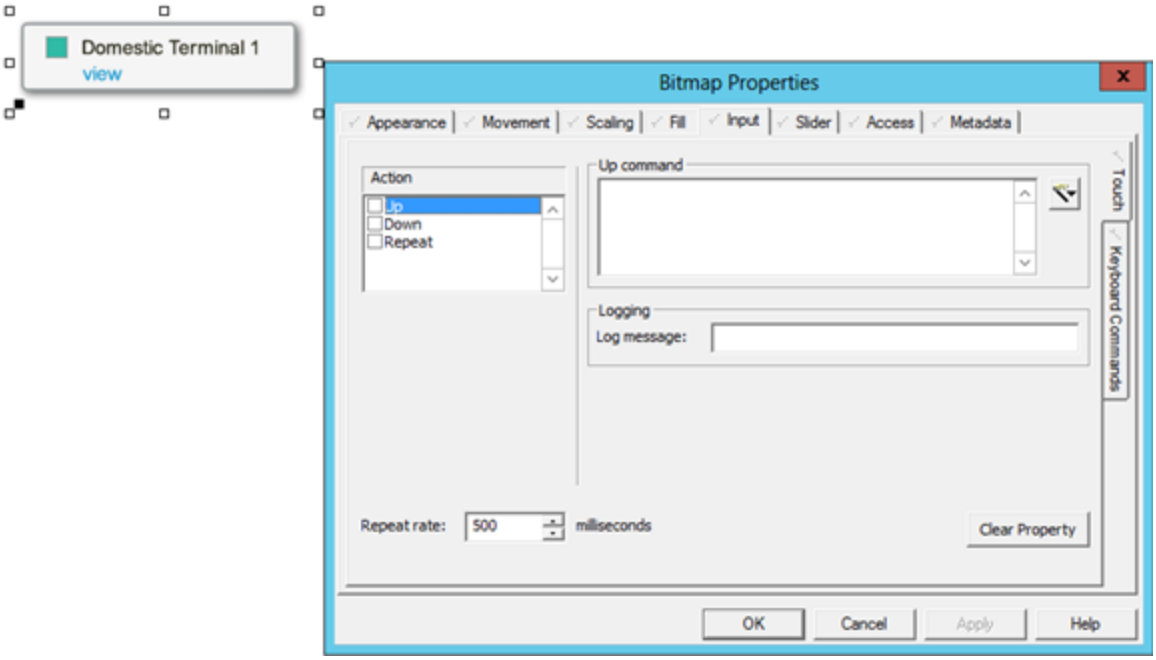
Add custom images to graphics pages

You can add custom images to a graphics page that are developed in external imaging software. For example, a facility floor plan image, or an image depicting a physical piece of switchgear, such as an elevation drawing.

NOTE: Power SCADA Operation supports several image file types, however, bitmaps (BMP) tend to display better than other image file types.

To add a custom image to a graphics page:

1. Create your image in an imaging editor.
2. In the Power SCADA Operation Graphics Builder, select File >Import and then select your image.
3. Re-size the image as necessary.
4. Overlay the image with any genies or data display objects that need to be displayed.
5. (Optional) If the image is intended to function as a button, it can be configured to do so by right-clicking and configuring properties similar to configuring any buttons or images that are native to Power SCADA Operation. As an example see the imported bitmap below and the properties that can be configured such that the bitmap acts as a button in the runtime environment.



(Optional) Change the page background color

To change the background color of individual pages in your project:

1. Create a new project, using "high contrast" as the starter project.

   This project will already include all of the pages that you will use. There will also be a new parameter, called *PageColorExt*. This parameter includes the extension that allows you to color backgrounds.

2. To designate background color for each page, pick the color in the Graphics Editor.

You can also change the genie font colors on your project pages. For details, see " (Optional) Change the genie color in project pages" on page 269.

(Optional) Change the genie color in project pages

To change the color of genies in pages of your project:

1. In your project, add a new parameter to the Graphics section, called *GenieFont*.
2. Enter the value designation for the color you want to use.
3. Make sure that there is sufficient contrast between the two colors so that the genie colors will be visible.

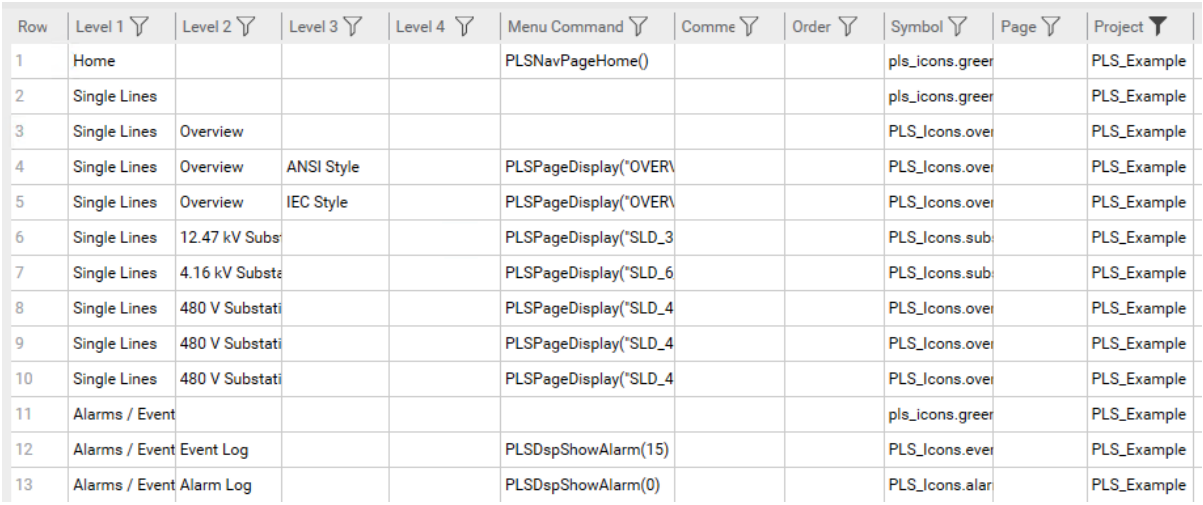
NOTE: If you use the "high contrast" starter project to create your project, this parameter is automatically added, as are all of the pages that you will use. For instructions on changing the background color of your project pages, see "(Optional) Change the page background color" on page 269.

Use menu configuration to edit pagemenu.dbf (Change the graphics page appearance)

The Menu Configuration form (Power SCADA Studio: **Visualization > Menu Configuration**) edits Pagemenu.dbf in your project. This controls the Power SCADA Runtime screen appearance: tabs and menus on the screen. By the entries you add there, you can also specify actions to be taken when an option is selected.

TIP: Copy and paste the menu settings from the PLS_Example project into a new project's menu configuration file.

The following illustrates a Menu Configuration page for the PLS_Example project (see the table below for descriptions of the columns).



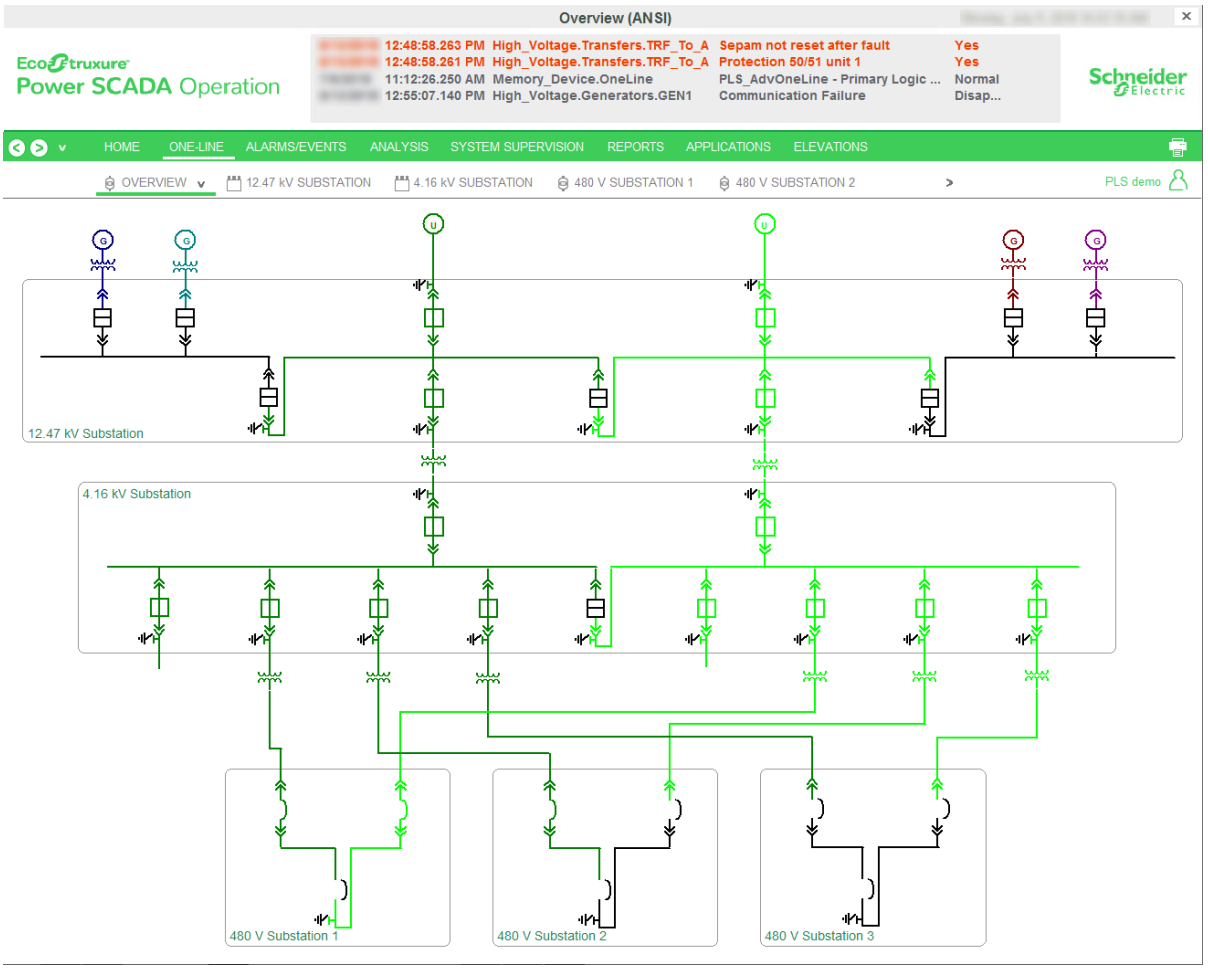
Row	Level 1	Level 2	Level 3	Level 4	Menu Command	Comme	Order	Symbol	Page	Project
1	Home				PLSNavPageHome()			pls_icons.greer		PLS_Example
2	Single Lines							pls_icons.greer		PLS_Example
3	Single Lines	Overview						PLS_Icons.ove		PLS_Example
4	Single Lines	Overview	ANSI Style		PLSPageDisplay("OVER\			PLS_Icons.ove		PLS_Example
5	Single Lines	Overview	IEC Style		PLSPageDisplay("OVER\			PLS_Icons.ove		PLS_Example
6	Single Lines	12.47 kV Subs			PLSPageDisplay("SLD_3			PLS_Icons.sub		PLS_Example
7	Single Lines	4.16 kV Substa			PLSPageDisplay("SLD_6			PLS_Icons.sub		PLS_Example
8	Single Lines	480 V Substati			PLSPageDisplay("SLD_4			PLS_Icons.ove		PLS_Example
9	Single Lines	480 V Substati			PLSPageDisplay("SLD_4			PLS_Icons.ove		PLS_Example
10	Single Lines	480 V Substati			PLSPageDisplay("SLD_4			PLS_Icons.ove		PLS_Example
11	Alarms / Event							pls_icons.greer		PLS_Example
12	Alarms / Event	Event Log			PLSDspShowAlarm(15)			PLS_Icons.eve		PLS_Example
13	Alarms / Event	Alarm Log			PLSDspShowAlarm(0)			PLS_Icons.alar		PLS_Example

Menu Item	Description
Levels 1 through 4	These items establish the menu levels that will display. For example, you might use "One-Lines" for level 1, followed by the substation for level 2, and the graphic name for level 3. (Each line: 256 characters maximum)

Menu Item	Description
Menu Command	The Cicode expression that you want to execute. Typically, you will use the "page display" command followed by the actual page you want to see. For example: PLSPageDisplay("CB_IEC_1") which displays the page CB_IEC_1.
Order	The relative position within the final graphics page. If you leave this field blank, the default value 0 is used. (64 characters maximum)
Symbol	Displays a defined image along with the description for that level. Images must already be defined in the project/include project. They are specified in the format <library name>,<symbol name>. For example, in PLS_Example, the symbol used for the level 2 of One-Lines is Substation3, entered as PLS_Icons.Substation3. Different menu levels are designed to be used with different symbol sizes for optimal display. For Level 1 items (tab), the recommended symbol size is 16 x 16 pixels. For Level 2 items, (buttons), the recommended symbol size is 32 x 32 pixels. Symbols are not displayed for menu items of Level 3 or beyond.
Page	The page on which this entry will display. If this is left blank, the entry will display on every page.
Comment	You can use up to 128 characters to add a comment (will not display on screen).

Animated one-line diagrams

Animated one-line diagrams provide built-in support for power flow diagrams. One-line color animates based on the source that is feeding the circuit. For example:

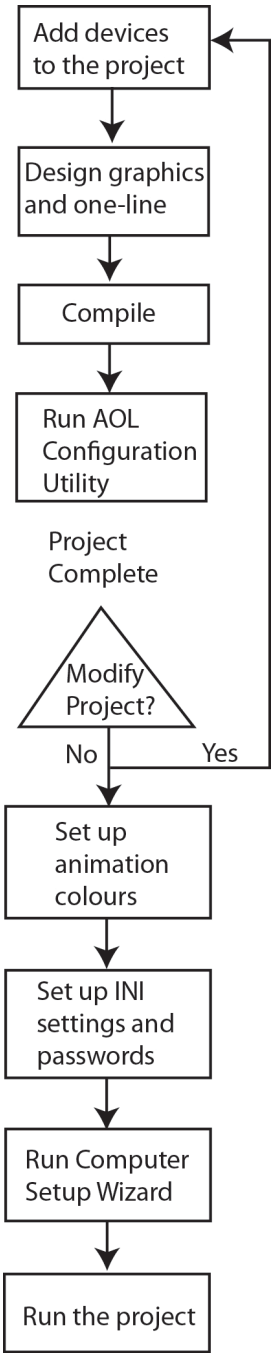


This section includes the following topics:

- "One line prerequisites " on page 274
- "Create a one-line on a graphics page" on page 280
- "Create a new genie" on page 285
- "Reviewing Genie Configurations" on page 292
- "Repair One-Line Diagrams" on page 290

One-line flow chart

The following flow chart provides an overview of the process to follow when setting up and using animation in one-line diagrams.



For detailed information on one-line diagrams see:

Running the Advanced One-Line Configuration Utility:

- "Reviewing Genie Configurations" on page 292
- "Repair One-Line Diagrams" on page 290

Setting Up Animation Colors:

- "Assign One-Line Colors" on page 290

INI Settings and Passwords:

- "Add INI settings to AdvOneLine.ini.txt and Citect.ini" on page 277

After you run the project, ensure that the password is encrypted (see *IsEncrypted* in "Add INI settings to AdvOneLine.ini.txt and Citect.ini" on page 277).

One-Line Configuration Utility

To access this screen from the Power SCADA Studio. From the left hand panel, click '*Launch the Single Line Configuration Utility*' (). There are two tabs.

On the **Genie Configuration tab**, you can:

- View genie types, along with their states (normal, warning, error) and their properties
- Repair genies that are part of a version 7.30 or later project:
  - Corrects incorrect breaker, source, meter, transformer, and Sim source numbers
  - Corrects invalid line active when a connected busbar has a valid line active
- Repair and upgrade genies that are part of a project from a version earlier than 7.30
  - Renumbers ALL breaker, source, meter, and Sim source numbers
  - Corrects invalid line active when a connected busbar has a valid line active
  - Reassigns ALL valid busbars

See "Reviewing Genie Configurations" on page 292 for information about using this tab.

On the **Color Configuration tab**, you can assign colors to sources. see "Assign One-Line Colors" on page 290 for information about using this tab.

Modify AdvOneLine.csv

After you run the One-Line Configuration Utility, open the project to verify that the animation is working correctly. If the animation is not correct, repeat the process of running the utility and verifying out animation until all errors are corrected.

One line prerequisites

Before you can create one lines, verify that the following tasks are completed:

- "Set up data acquisition parameters" on page 152
- "One-line memory device (zOL)" on page 274
- "One Line Engine configuration" on page 275
- "Add INI settings to AdvOneLine.ini.txt and Citect.ini" on page 277
- "Start and stop one-lines" on page 279

One-line memory device (zOL)

To use one-line graphics, your project must include a memory device named zOL. One-line graphics use the zOL device to drive animation. You must have at least one zOL device per project. If your project does not include this memory device, you must create it.

You can optionally edit the default zOL device support parameters .

NOTE: When you use Project Setup to create your project, a zOL device is added automatically to the project

To create the zOL device and add it to your project:

1. Open the Profile Wizard.
2. Click **Create an I/O Device** in the project. Click **Next**.
3. Select the device called **OneLine Device Setup**. Click **Next**.
4. Follow the device creation remaining steps to add the device.

By default, this device will support 100 sources, 1000 buses, 1000 meters, and 1000 breakers. You can modify this in the Profile Editor:

1. On the **Setup Projects** tab, choose the project.
2. Click the **Project Parameters** sub-tab.
3. Enter the optional project parameters (MaxBreakers, MaxBuses, MaxMeters, MaxSources). Valid entries are from 1 to 9999 (only 200 for MaxSources).
4. On the Selected Device Profiles sub-tab, click **Refresh Tags**.
5. Export the project.
6. In the Profile Wizard, click **Update one or more I/O Device(s) in the project** option. This updates the zOL I/O device parameters entered in step 3.

The new one-line device is ready to be used in the selected project.

One Line Engine configuration

To open the One Line Engine:

1. Open the Application Configuration Utility:
   - In Power SCADA Studio, click **Projects**  **> Power Applications > Application Config Utility**.

     OR
   - From the Start menu, click **Schneider Electric > Application Config Utility**.
2. In Application Configuration Utility, expand **Applications** and then click **One Line Engine**.

There are 3 tabs in the One-Line Engine module. On all 3 tabs, 2 buttons at the bottom allow you to:

1. **Restart AOL:** Restarts the Advanced One-Line Engine.
2. **Save:** Saves the settings you entered.

NOTE: When running the Power SCADA Studio project as a Windows service, `Advancedoneline.exe` must run on session 0. To achieve this, execute your advanced one line startup code from an I/O Server rather than from a client.

The three tabs are:

1. **Citect User**

   After you add a user to your Power SCADA Studio project, use this tab to test whether the user ID can be used by the One-Line engine to connect with runtime. Type the Power SCADA Studio user ID and password, and then click **Test Credentials**.

   The test will attempt to log in with this user information. A message displays, telling you whether the user information passed. If it does not pass, you see a message telling you that the connection failed because the user name/password are incorrect or Power SCADA Operation is not running. Make sure that Power SCADA Operation is running and that the user name/password have been set up in Power SCADA Studio, then try again.

2. **General**

   You can edit the following parameters that enable one lines to run properly. For more complete descriptions of the parameters, see "Add INI settings to AdvOneLine.ini.txt and Citect.ini" on page 277. If you are not setting up a redundant system, the default settings should be sufficient.

   - **Update Interval:** Interval in seconds at which the system tries to solve the system one-line
   - **Max Startup Delay:** Sets the amount of time in seconds the AdvOneLine.exe has to start up
   - **Health Timeout:** Performance parameter; dictates the amount of time in seconds that must elapse before the one-line engine is considered to be non-functioning
   - **Log File Length:** Suggests the log file length in number of lines
   - **Debug Level:** Selects the level of logging for AdvOneLine.exe

3. **Redundancy**

   - **Primary Server IP:** Used in redundant configurations to specify the IP address of the primary I/O Server. Click **Clear** to clear the current address, then type the correct address for the primary server.
   - **Standby Server IP:** Used in redundant configurations to specify the IP address of the standby I/O Server. Click **Clear** to clear the current address, then type the correct address for the secondary server.

     Use the **Export Key** and **Import Key** buttons to save an encryption key and export it to another computer as an AES file. This allows you to move an INI file from one computer to another and to have its contents unencrypted for use by that computer.

> **⚠ WARNING**
>
> **POTENTIAL COMPROMISE OF SYSTEM AVAILABILITY, INTEGRITY, AND CONFIDENTIALITY**
>
> - Store system keys, AES encryption files, or other files containing passwords to a secure site.
>
> **Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.**

Cybersecurity policies that govern how sensitive system files are securely stored vary from site to site. Work with the facility IT System Administrator to ensure that such files are properly secured.

- **Export Key:** After making or verifying changes here or in the AdvOneLine.ini.txt file, click to save a configuration that you can use on another computer.

  A Save As window displays, allowing you to browse to the preferred location. Save the AES file to a secure location, such as a secure network drive or a flash drive.
- **Import Key:** After you save the AES file to the secure drive, ensure that the drive is accessible to new computer. At the new computer, click this button to access the AES file.

After you access the AES file at the new computer, copy the INI file to the new computer. You will be able to access and use it. Remove the AES files from the source computer.

Add INI settings to AdvOneLine.ini.txt and Citect.ini

All INI settings are grouped in the OneLineEng section of the .INI files.

You must have correct .INI settings in order for the one-lines to run properly. Ensure that the following .INI parameters are properly set:

AdvOneLine.ini.txt Settings

NOTE: The following parameters are set in the Application Configuration Utility ("One Line Engine configuration" on page 275): UpdateInterval, PrimaryServerIP, StandbyServerIP, HealthTimeout, MaxStartupDelay, LoginUserName, LoginPassword, and LogFileLength.

Parameter	Description	Default Value
UpdateInterval	The interval at which the system tries to solve the system one-line. This interval can be changed to slow down the rate at which the animation is solved. Specifying a rate faster than possible will force the engine to solve the system as quickly as possible.	1000 msec

Parameter	Description	Default Value
PrimaryServerIP	Used by redundant configurations to specify the IP address of the server on which the primary I/O Server resides. This parameter is required for a redundant configuration. If either the primary or standby IP addresses are not specified, the logic engine will assume that the system is not redundant.	N/A
StandbyServerIP	Used by redundant configurations to specify the IP address of the server on which the standby I/O Server resides. This parameter is required for a redundant configuration. If either the primary of standby IP addresses are not specified, the logic engine will assume that the system is not redundant.	N/A
HealthTimeout	This is a performance parameter that dictates that amount of time that must elapse before the one-line engine is considered non-functioning, and a PC-based alarm is raised in Power SCADA.	[UpdateInterval] + [TagSubscribeWait] * 5 milliseconds Minimum value: 1000 msec
DefaultColor	This parameter tells the engine the default color to be assigned to objects on the screen at system startup. This is useful for identifying components that have been left out of the CSV or simply as a means of having the engine set all currently unused objects to a color that indicates that they are not being monitored. If an invalid color is specified, the engine will default to black.	250
MaxStartupDelay	Sets the amount of time the AdvOneLine.exe has to start up. If this time is exceeded, initial tag subscriptions will not succeed, and the EXE will report an exception.	60 sec
StartupDelay	Sets the amount of time after AdvOneLine.exe has started for the system to be online and all initializations complete.	[Updateinterval] + [TagSubscribeWait] * 5 milliseconds Minimum value: 1000 msec
LoginUserName	This is the Power SCADA Studio user name to be used for the ctAPI connection in AdvOneLine.	aol

Parameter	Description	Default Value
LoginPassword	This is the Power SCADA Studio user password to be used for the ctAPI connection in AdvOneLine.	aol
IsEncrypted	Determines if the password is encrypted. The first time the project is opened in run time, the password is automatically encrypted, and this will be set to True.	False (changed to True after the first run and successful password encryption)
CitectIniPath32	Provides the path to the global Citect.ini file for a 32-bit operating system install. This setting must be changed if SCADA is not installed on the C: drive, or if the Citect.ini file is moved/installed in another directory.	Default value: C:\Documents and Settings\All Users\Application Data\Schneider Electric\Power SCADA Operation 9.0\Config
LogFileLength	Suggests the log file length in number of lines. After surpassing this limit, the log file is saved with suffix ".bak," and a new file is created.	Default value: 5000 Allowed values: 10–10000
DebugLevel	Sets the level of logging for AdvOneLine.exe. Multiple values are separated by | (e.g., Error|Warn).	N/A Allowed values: All, Error, Warn, Debug

Citect.ini Settings

Parameter	Description	Default Value
AutoRestart	Indicates whether the one-line will restart itself when the logic engine is not responding.	0 (disabled) Allowed values: 0, 1
ServerRole	Informs the local instance of Power SCADA where it is (primary or standby server). This parameter is controlled by the AdvOneLine.exe application. The user does not need to create or modify this parameter. It is set based on the PrimaryServer IP and StandbyServerIP parameter settings.	Primary
StartupDelay	This is a performance parameter that dictates that amount of time that must elapse before the one-line engine is considered non-functioning, and a PC-based alarm is raised in Power SCADA.	15 sec

Start and stop one-lines

Use the following Cicode functions if you need to start or stop AdvOneLine.exe.

To stop AdvOneLine.exe, call `PLS_StopAdvOneLine(STRING sIOServer="",STRING sCluster="")`

To start AdvOneLine.exe, call `PLS_StartAdvOneLine(STRING sIOServer="",STRING sCluster="")`

NOTE: Call these functions only on an I/O Server that is communicating with a "One-line memory device (zOL)" on page 274.

NOTES:

- If the default parameters are used, the functions will run on the local machine.
- If you call the function from a remote server, enter the I/O Server name and cluster to run the function on that server. You must be logged in to perform this action.

Create a one-line on a graphics page

You will build a one-line by adding genies to the new page. Each genie has a defined list of information that displays by default. You can also expand this list to include everything that is known about that genie type. For more information about genies, see **Genies and Super Genies** in the Citect SCADA help file (C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin).

To begin creating a one-line, start adding genies to the page:

1. Click the “genie” icon on the toolbar: 
2. From the Paste Genie screen, choose a library that begins with PLS. PLS libraries are Power SCADA Operation libraries that include Power SCADA Operation features. To edit a genie or create a new one, see "Create a new genie" on page 285 for instructions.
3. From the available icons in the selected library, select a genie and then click **OK** to paste the genie on the graphics page.

   A popup window displays for you to enter genie properties.
4. Set the various properties so that they communicate properly on the one-line graphic. These properties include such information as its label, type of equipment, how each genie communicates within the one-line, and source and destination.
5. Enter the requested information for the appropriate object:
   - "Configure a meter" on page 281
   - "Configure a Source" on page 282
   - "Configure a Circuit Breaker or Switch" on page 282
   - "Configure a Busbar" on page 283
   - "Configure a Transformer" on page 284
   - "Configure an Automatic Transfer Switch (ATS)" on page 284

   NOTE: An asterisk in any of the windows indicates that it is a required field.
6. If a message displays indicating that variable tags are not found, and asks if you want to add the unknown tags, click **No**. The genie is expecting to see a variable, but one may not exist in the equipment. For example, the genie could be looking for a “racked out” variable in a Sepam.

NOTE: To disable this message, go to Power SCADA Studio. Click **Options** on the left hand side. From the Options dialog box, de-select **Prompt on tag not exist**. Click **OK**.

The genie is added to the page.

7. Continue adding the genies to make up the graphics page.

   Do not drag genies off of the page. They will disappear.

8. After you create the graphics page, click **Compile** .

   While the page is being compiled, the Compiler displays each file name as it is being compiled. A message then displays, telling you that compilation was successful.

   If there are errors and/or warnings after the page is compiled, a message box displays, telling you the number of errors and warnings. You must understand all warnings and correct all of the errors.

   To correct the errors:

   a. Click **OK** at the Compiler message.

   b. The Compile Errors window displays the first error.

   c. Note the error message. For more information on each error, click Help. Note the Description for that Error Message.

   d. Correct the error by clicking the **GoTo** link.

      The appropriate window displays for you to correct that error.

   e. Correct each error. Warnings do not need to be corrected in order to run the project; but they should be checked to see if they could impact the project (such as a tag that is not defined).

   f. If you delete any entries, click **Projects > Pack** to clean up the files.

   g. Compile the project again to verify that all of the errors have been addressed.

9. After all errors are corrected, run the project.

   The Graphics page displays in the runtime environment.

NOTE: Until you edit the Menu Configuration file, only basic tabs will display on the Graphics page.

Before you can view your one-line, you need to edit the Menu Configuration file, which controls the appearance of the graphics page in the Power SCADA Runtime. See "Use menu configuration to edit pagemenu.dbf (Change the graphics page appearance)" on page 270 for details.

Configure a meter

To configure meters that you add to a one-line in the graphics builder:

1. Display the meter window by pasting the meter genie onto the graphics builder page.
2. **Equipment**: From the drop-down list, choose the name of the equipment represented by the genie. This is the equipment name that you entered in the I/O Device Manager, for

example: *CM4Main*. To view the equipment that is available for this page, you must have the project selected in the Power SCADA Studio.

3. **Meter Number**: Enter the number you want to use for this meter. Valid entries: 1–1000. This number must be unique within this one-line.
4. **Busbar Number**: Enter the number for the busbar that connects to the meter. Valid entries: 1–1000.
5. **Line Active**: Enter the Cicode expression, such as *MyTag1 > 0*, to determine when the meter detects power on the busbar.
6. **Label 1 and Label 2**: Type the information that you want to appear in the upper left corner of the genie in the runtime environment.
7. **Value Type**: From the drop-down list, choose the tag you want to use for this genie. The related information displays in the lower left corner of the genie in the runtime environment.

   This tag causes real-time data, such as *currents*, to display on the genie status page in the runtime environment. If you do not choose a tag here, the status page will not display the real-time data.
8. **Units**: From the drop-down list, choose the unit that you would like to display on the genie in the runtime environment. Units that display here were added in the Profile Editor.
9. **Multiplier**: Enter the multiplier that is to be applied to the units chosen in step 8.
10. Click **OK** to save the genie to the page and to return to the graphics builder page.

See "Meter Information" on page 294 for more information on meters, and for resolving errors.

Configure a Source

To configure sources—such as utilities and generators—that you add to a one-line in the graphics builder:

1. Display the source window by pasting the generator or utility genie on to the graphics builder page.
2. **Source Number**: Enter the number you want to use for the source. Valid entries: 1–100. This number must be unique within this one-line.

   The source number is used when you determine coloring for the one-line.
3. **Busbar Number**: Enter the number for the busbar that connects to the source. Valid entries: 1–1000.
4. **Line Active**: Enter the Cicode expression, such as *MyTag1 > 0*, to dictate when the source powers the busbar.
5. Click **OK** to save the genie to the page and to return to the graphics builder page.

See "Source information" on page 295 for more information on sources, and for resolving errors.

Configure a Circuit Breaker or Switch

To configure circuit breakers or switches that you add to a one-line in the graphics builder:

1. Display the circuit breaker or switch window by pasting the appropriate genie on to the graphics builder page.
2. **Equipment**: From the drop-down list, choose the name of the equipment represented by the genie (this is the equipment name that you entered in the I/O Device Manager, e.g., *CM4Main*). To view the equipment that is available for this page, you must have the project selected in the Power SCADA Studio.
3. **Breaker/Switch Number**: Enter the number you want to use for this breaker or switch. Valid entries: 1–1000. This number must be unique within this one-line.
4. **Source and Destination**: When you connect to busbars, enter the numbers for the connection source and destination busbars. Valid entries: 1–1000.
5. **Label 1 and Label 2**: Enter the information that you want to appear in the upper left corner of the genie in the runtime environment.
6. **Value Type**: From the drop-down list, choose the tag you want to use for this genie. The related information displays in the lower left corner of the genie in the runtime environment.

   This tag causes real-time data (such as *currents*) to display on the genie status page in the runtime environment. If you do not choose a tag here, the status page will not display the real-time data.
7. **Units**: From the drop-down list, choose the unit to display on the genie in the runtime environment. Units that display here were added in the Profile Editor.
8. **Multiplier**: Enter the multiplier that will be applied to the units chosen in step 7.
9. Click **OK** to save the genie to the page and to return to the graphics builder page.

NOTE: If you choose to re-size a circuit breaker genie after you paste it into a page, you must keep the relative dimensions (proportions) the same. Otherwise, the racked in/racked out animation will not display correctly.

See "Breaker and Switch Information" on page 296 for more information on breakers and switches, and for resolving errors.

Configure a Busbar

Assign a busbar number to a busbar and use busbars to connect genies on a one-line.

Busbar numbers associate devices within drawings, and they help you set up animation for genies. The busbar entered here is also used for the associated device.

To configure busbars that you add to a one-line in the graphics builder:

1. Display the busbar window by pasting the busbar genie on to the graphics builder page.
2. **Busbar Number**: Enter the number you want to use for this busbar. Valid entries: 1–1000.
3. **Line Active**: This field is no longer used.
4. Click **OK** to save the genie to the page and to return to the graphics builder page.

See "Busbar Information" on page 297 for more information on busbars, and for resolving errors.

Configure an Automatic Transfer Switch (ATS)

To configure automatic transfer switches (ATS) that you add to a one-line in the graphics builder:

1. Display the Transfer Switch window by pasting the appropriate genie on to the graphics builder page.
2. Configure the transfer switch information (Left/Right/Bottom):
   a. **Breaker Number:** Enter the ID numbers you want to use for the left side and right side of this ATS. Valid entries: 1–1000. Each number must be unique within this one-line.
   b. **Busbar:** Enter the numbers of the left and right source busbars, and for the destination (bottom) busbar. Valid entries: 1–1000.
   c. **Label:** For each ATS side, enter the information that you want to appear on the switch in the runtime environment. For example: *Pri* and *Emer*).
   d. **Closed Expression:** For each side of the switch, type the information that you want to appear when that side is closed. For example: *Tag1* <> *1*). Do not use "NOT" in the expression.
3. Configure the display information:
   a. **Label 1/Label 2:** Enter the descriptive information that you want to appear in the upper left corner of the genie in the runtime environment.
4. Click **OK** to save the genie to the page and to return to the graphics builder page.

See "Automatic transfer switch (ATS) information" on page 298 for more information on ATSs, and for resolving errors.

Configure a Transformer

To configure transformers that you add to a one-line in the graphics builder:

1. Display the transformer window by pasting the appropriate genie on to the graphics builder page.
2. **Top and Bottom Source Number**: For each source, whether top or bottom, enter a source number to control voltage-level coloring for the secondary side busbar. Valid entries: 1–100.
3. There are two possible configurations:
   - **Pass-through coloring**: If you leave these fields blank, the transformer will transfer the color that is assigned to the primary side (source) to the secondary side (destination) busbar. For example, if Source 3 feeds the source busbar of a transformer, and you leave this field blank, then Source 3 will also feed the destination busbar (and the Source 3 color will be used).
   - **Voltage-level coloring**: When you enter top and bottom source numbers, the transformer colors the one line based on this number. For example, if Source 3 feeds the top of the transformer, but you enter 5 for the bottom source, the transformer feeds the color from Source 5 to the destination (bottom) busbar.
4. **Source and Destination**: When you connect to busbars, type the numbers for the connection source and destination busbars. Valid entries: 1–1000.

5. **Label 1 and Label 2**: Enter the information that you want to appear in the upper left corner of the genie in the runtime environment.
6. Click **OK** to save the genie to the page and to return to the graphics builder page.

See "Transformer Information" on page 299 for more information on transformers, and for resolving errors.

SupportedGenies.xml file

Use SupportedGenies.xml to define genies that support one-line coloring.

This file links genies in a library to a genie type. In this file, you need to define the project name, library name, and genie name. The genie name may be "*": which will select all genies that library. You can exclude individual genies.

Supported Genies XML file example

See also: GenieConfiguration.xml File

GenieConfiguration.xml file

Use GenieConfiguration.xml to define completely new (unique) genies and those that have been copied and modified from an existing genie.

This file defines each genie in detail. It links fields with genie parameters names, defines validation, and defines how to export each genie for the one-line.

Some fields have restrictions. See the comments for each part of the XML file.

Genie Configuration XML file example

See also: "SupportedGenies.xml file" on page 285

Create a new genie

Create a new one-line genie using one of the following methods:

1. **Copy a genie:** Create a genie that is completely compatible with an existing genie , having the same genie parameters and functionality. To do this, you can copy an existing genie and change attributes. This type of genie must be added to the SupportedGenies.xml File.
2. **Create a unique genie:** Create a genie that has unique parameters, validation requirements, and output types. See instructions below for this. This type of genie must be added to both the SupportedGenies.xml File.

The easiest way to create a new genie is to make a copy of a similar genie from the standard library and edit it in your project. When adding layers, keep the dimensions of the new layers the same as the original. For a list of all of the standard Power SCADA Operation genies, including all of the smaller parts that you could use to create genies, see "Default Genie Library" on page 629.

NOTE: If you modify a genie, the modifications will effect all instances of that genie in the project.

Create a copy of a genie

1. Open the genie that you want to use as a template: From Citect Graphics Builder, click **File > Open**.
2. Click the **Genie** tab, then select the library and the template genie. Click **OK**.

3. Save the genie with a new name in your local project.
4. To separate the genie into layers, click the genie and drag a layer to the bottom right.
5. Repeat the process to pull all of the layers apart.

   Each layer is a sub-genie that controls a different aspect of the overall genie.
6. Make the changes that you wish to the genie.

   NOTE: Be careful to maintain the same dimensions for any new layers that you create.
7. To re-assemble the genie:
   a. Draw a marquis around all of the parts.
   b. Click **Arrange > Align**, and choose Top and Left.
   c. Click **OK**.

The newly created genie, when applied to a page, will display with the generic input form. To create a customized form similar to those found with the default genies, you must create a new FRM file. Examples are found in the PLS_Include project directory.

You can rotate any of the genies.

Create a custom symbol for a custom genie

Power SCADA Operation genies are complex genies with the added ability to animate color during runtime (showing changing states). These genies are comprised of multiple symbols that have the same attributes (all call the same Cicode function).

1. Create the new symbol.
2. Configure the symbol to display the customized colors in the Power SCADA Runtime by calling the *PLS_GetBusColorIndex(INT nColorIndex)* function, which is provided in Power SCADA Operation. This function must be called from the Array Condition area of the Symbol properties window.

   Each Array symbol can display up to 255 colors. Ensure that the default color palette matches the palette used in PLS_Include. The parameter for *PLS_GetBusColorIndex(INT nColorIndex)* is the busbar number that determines the animated color. Always use the %variable% notation. Using this notation permits you to reuse genies.

   See the default symbols in the Power SCADA Operation libraries that begin with PLS_.
3. Save the symbol and use it in a custom genie.

Reviewing Genie Configurations

Use the One-Line Configuration Utility to review genie configurations before you compile your project.

If you are upgrading from an earlier version:

- Run Update Pages in the Graphics Builder.
- Create the pages.

Errors ( ) and warnings ( ). You must correct errors; otherwise, you may not be able to compile, and the animation will not work. Although you might not need to correct warnings, you should review them to ensure that their settings are correct.

TIP: When you hover over an error or warning icon, a tooltip tells you what is wrong with the genie.

To launch the One-Line Configuration Utility:

1. Make sure you are viewing the system for which you want to view information.
2. In Power SCADA Studio, click **Launch the Single Line Configuration Utility** .
3. Click OK to the message that displays.

   The first time you load the utility, a large system could take a couple of minutes to load. After that, it should load within a minute.

   There are 2 tabs:

   - Use **Genie Configuration** to:
     - View genie types, along with their states (normal, warning, error) and their properties
     - Repair genies that are part of a version 7.30 or later project:
       - Corrects incorrect breaker, source, meter, transformer, and Sim source numbers
       - Corrects invalid line active when a connected busbar has a valid line active
     - Repair and upgrade genies that are part of a project from a version earlier than 7.30
       - Renumbers ALL breaker, source, meter, and Sim source numbers
       - Corrects invalid line active when a connected busbar has a valid line active
       - Reassigns ALL valid busbars
   - Use **Color Configuration** to assign colors to sources.

   The **Genie Configuration** pane contains the following information:

Field	Description
Projects	Default: the project selected in the Power SCADA Studio
Show By	**Type:** Information is sorted first by genie type, then by page. This option is useful when you want to see all genies of a certain type together, regardless of where they are in the drawing pages. **Page:** Information is sorted first by page, then by genie type. This option is useful when you want to see all genies on a certain page.
Advanced Properties	Check this box to view the basic information plus any additional information relevant to that genie type.

Field	Description
State Filters	Check the individual boxes for how you want to view information. For example, you might only be interested in viewing genies that have error states. This option controls only the genie information in the right-hand pane.
Genie Types tree	Types are: breakers, busbars, meters, sources, and transformers
Genie Information grid	Columns of information display: In the Basic (default) view: the most used information If you click **Advanced Properties**, you see the basic information, followed by all the information known about the genie(s) you are viewing.
Repair—Upgrade Project	Check this box to cause the repair feature to repair the entire project. Use this feature only to upgrade projects that are earlier than Power SCADA Operation 9.0. This option repairs the entire project, renumbering all busbars, breakers, meters, duplicate Sim sources, and sources. Additionally, busbar line active states are used to determine meter and source line active states. DO NOT perform Repair—Upgrade Project more than once, and do not perform it on a Power SCADA Operation with Advanced Reporting and Dashboards project.
Repair	This feature attempts to repair errors and warnings.

For specific information about each type of genie, click a link below:

- "Breaker and Switch Information" on page 296
- "Busbar Information" on page 297
- "Meter Information" on page 294
- "Source information" on page 295
- "Transformer Information" on page 299

Enable lockout/tagout

⚠ DANGER

EQUIPMENT ELECTRIC SHOCK, EXPLOSION, OR ARC FLASH

- Do not rely solely on the display of the graphic on the one-line.
- Verify that the device is physically locked out/tagged out before you work on the equipment or any downstream equipment.
- Ensure that all safety regulations and procedures have been followed before you work on the equipment.

Failure to follow these instructions will result in death or serious injury.

NOTE: Do not incorrectly configure the tag, as this can lead to unexpected equipment operation. Also consider the possibility of communications loss that could yield false readings.

With this feature, you can cause the locked out icon (shown above) to display on your graphics page in the Power SCADA Runtime. The icon displays when a tag attribute for a device reaches a specified value. For example, you might set a PLC tag to 0 when the equipment is in lockout/tagout (the door is open), and to 1 when the equipment status indicates that the door is closed.

This is a read-only feature; but it does not prevent controls to the device or area. This feature is not available in PLS_Example.

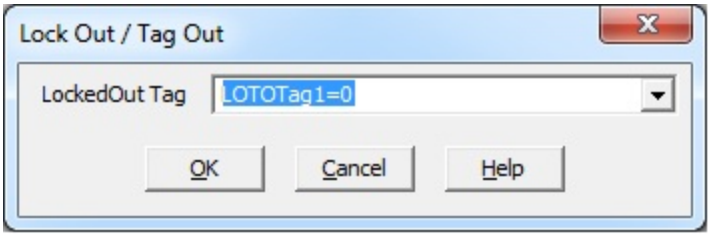
To enable the locked out icon for a device:

1. From the pls_gen_eq_2 genie library, add the lockout/tagout genie to the graphics page. Position it beside the equipment that is being monitored.
2. From Variable Tags, choose the device tag (or create a custom tag) that you want, and associate it with the device that will read the lockout/tagout status.
3. Define what indicates the status (for example, *door status open*, 0 = false and 1 = true).
4. Edit the lockout/tagout genie to read the tag on the device. Make the lockout/tagout genie visible when the device tag indicates that the device is in lockout/tagout status; and make the

icon hidden when the device is not in lockout/tagout. (By default, lockout/tagout status is hidden when the tag property is "NOT [variable tag].")

To continue the example, if the tag is reading *door status open* and the value = 0, then the door is open and in lockout/tagout; if the value = 1, the door is closed and not in lockout/tagout.

The following graphic illustrates how it might look in when adding the genie:



Assign One-Line Colors

Line coloring is based on the source and meter line active states. Sources dictate the colors for each genie. Meters can only determine if a bus is active. When the bus is live, the meter then colors based on the source that is connected to the bus. If there is no source, the default color is used.

NOTE: Depending on how you configure transformers, you can either use this "pass-through" coloring, or you can use "voltage-level" coloring. For more information configuring transformers, see "Configure a Transformer" on page 284.

To assign a color to a source:

1. Open the One Line Configuration Utility: In Power SCADA Studio click Launch Single Line
2. Click the **Color Configuration** tab.
3. From the Projects drop down, select the project for which you want to assign colors.
4. Choose the Project Color Palette. Select the project in which the project genies are defined; this is usually PLS_Include.
5. For each source or transformer, choose the desired color:
   a. Click the color cell for that source/transformer.
   b. Select a color from the drop down list.

      NOTE: You can also select a color for unknown sources, off, and error. To indicate a flashing color, select two colors.
6. When all colors are assigned, click **Save**.

Repair One-Line Diagrams

Before you repair your project one-line diagrams, back it up.

NOTICE
LOSS OF DATA Backup your project before you perform a repair. **Failure to follow these instructions can result in corruption of your project.**

For more information on correcting one-line errors, see "One-Line Errors and Warnings" on page 507

To repair one-lines:

1. In Power SCADA Studio: click **Launch the Single Line Configuration Utility**  .
2. Click **OK**.

   Genie information for the selected project displays. For descriptions of the fields on this page, see "Reviewing Genie Configurations" on page 292.
3. Choose the type of repair you want to perform:
   - **Repair option alone** (**Upgrade Project** not checked) attempts to fix errors and warnings in a project (used for Power SCADA Operation).
   - **Repair option with Upgrade Project checked** is used to upgrade projects from previous versions of the product. This option renumbers all genies in the project. Do not perform this option on a project more than once, and do not perform it on Power SCADA Operation 9.0 projects.

   NOTE: When two busbars have the same line active, they are assigned the same busbar number.

   The following table describes the repairs made in each option.

Genie Type	Repair	Repair— Upgrade Project
Breaker	Breaker Number	ALL Breaker Numbers
Sources	Source Number	ALL Source Numbers
	Line Active	Line Active
Meters	Meter Number	ALL Meter Numbers
	Line Active	Line Active
Transformers	Sim Source Numbers (top and bottom)	Sim Source Numbers (top and bottom)
Busbars	----	All valid busbars will be reassigned, including destination and source busbars for breakers and transformers

4. Click **Repair**.

   A message describes the degree of repair that is about to take place.

Each message states that graphics pages will not be modified by the repair process. This means that the repairs will not be applied to your project graphics pages until you click Save.

5. Click **Yes** to initiate the repair option that you have selected.

   A Repair Summary window displays, listing the repairs that have been initiated.

6. To save a CSV copy of this summary:
   a. Click **Export**.
   b. At the Save As window, enter a file name and choose the location to save the file.
   c. Click **OK**.

      The genie information changes, indicating that the repairs have been made.

7. Click **Save**.

   The Save window appears. This is where the changes are saved to your project.

8. Either click **Yes** to save the changes to the graphics pages of the project, or click **No** to cancel the changes.

   If you click **Yes**: The changes are saved to the project. For a large project, this might take several minutes. When the repairs are saved to the project, a Save Summary window appears listing the repairs that were made and saved.

   If you click **No**: Click **Close**, then click **No** when you are asked whether you want to save the modified project.

9. Click **Export** to save a CSV file of these changes.
10. Click **OK** to exit the summary window and return to the One-Line Configuration Utility window.

    It is possible that some errors and warnings will not be repaired, for example, missing busbar numbers or missing equipment. Click individual errors or warnings to view them (note that the warning and error icons include a tooltip to tell you what is wrong). Note the missing information, then go to the graphics builder and make the necessary changes.

11. Compile the project and then run it.

Reviewing Genie Configurations

Use the One-Line Configuration Utility to review genie configurations before you compile your project.

If you are upgrading from an earlier version:

- Run Update Pages in the Graphics Builder.
- Create the pages.

Errors ( ) and warnings ( ). You must correct errors; otherwise, you may not be able to compile, and the animation will not work. Although you might not need to correct warnings, you should review them to ensure that their settings are correct.

TIP: When you hover over an error or warning icon, a tooltip tells you what is wrong with the genie.

To launch the One-Line Configuration Utility:

1. Make sure you are viewing the system for which you want to view information.
2. In Power SCADA Studio, click **Launch the Single Line Configuration Utility**  .
3. Click OK to the message that displays.

   The first time you load the utility, a large system could take a couple of minutes to load. After that, it should load within a minute.

   There are 2 tabs:

   - Use **Genie Configuration** to:
     - View genie types, along with their states (normal, warning, error) and their properties
     - Repair genies that are part of a version 7.30 or later project:
       - Corrects incorrect breaker, source, meter, transformer, and Sim source numbers
       - Corrects invalid line active when a connected busbar has a valid line active
     - Repair and upgrade genies that are part of a project from a version earlier than 7.30
       - Renumbers ALL breaker, source, meter, and Sim source numbers
       - Corrects invalid line active when a connected busbar has a valid line active
       - Reassigns ALL valid busbars
   - Use **Color Configuration** to assign colors to sources.

The **Genie Configuration** pane contains the following information:

Field	Description
Projects	Default: the project selected in the Power SCADA Studio
Show By	**Type:** Information is sorted first by genie type, then by page. This option is useful when you want to see all genies of a certain type together, regardless of where they are in the drawing pages. **Page:** Information is sorted first by page, then by genie type. This option is useful when you want to see all genies on a certain page.
Advanced Properties	Check this box to view the basic information plus any additional information relevant to that genie type.
State Filters	Check the individual boxes for how you want to view information. For example, you might only be interested in viewing genies that have error states. This option controls only the genie information in the right-hand pane.
Genie Types tree	Types are: breakers, busbars, meters, sources, and transformers

Field	Description
Genie Information grid	Columns of information display: In the Basic (default) view: the most used information If you click **Advanced Properties**, you see the basic information, followed by all the information known about the genie(s) you are viewing.
Repair—Upgrade Project	Check this box to cause the repair feature to repair the entire project. Use this feature only to upgrade projects that are earlier than Power SCADA Operation 9.0. This option repairs the entire project, renumbering all busbars, breakers, meters, duplicate Sim sources, and sources. Additionally, busbar line active states are used to determine meter and source line active states. DO NOT perform Repair—Upgrade Project more than once, and do not perform it on a Power SCADA Operation with Advanced Reporting and Dashboards project.
Repair	This feature attempts to repair errors and warnings.

For specific information about each type of genie, click a link below:

- "Breaker and Switch Information" on page 296
- "Busbar Information" on page 297
- "Meter Information" on page 294
- "Source information" on page 295
- "Transformer Information" on page 299

Meter Information

The most commonly used information about the meter genie displays by default.

When the **Advanced Properties** box is selected, the table expands to include everything that is known about the selected breaker(s).

Basic meter information includes:

Column	Description
State	Normal ( ), Warnings ( ), or Errors ( ). See the following table for explanations of errors.
Page	Name of the page on which the genie is found (displays only from the folder level).
ID	This is the meter number, assigned when adding it to a page of a one-line.
Equipment	The equipment name entered when adding the genie via the I/O Device Manager.

Column	Description
Source Busbar	The number of the incoming busbar.
Line Active	The Cicode expression (such as MyTag1 > 0) that determines when the meter detects power on the busbar.

Meter Errors and Warnings

Before you use the drawing, correct all errors; otherwise the project might not compile and the animation will not work.

Warnings indicate settings that might be incorrect. Verify that the settings indicated by the warnings are what you want.

Errors and warnings that you might see for meters are:

State	Solution
Errors (⊗)	
Meter number must be a number greater than 0 and unique.	The meter number is missing, or it is less than or equal to 0. Add or change the meter number.
Busbar number must be a number greater than 0.	The busbar number is missing, or it is less than or equal to 0. Add or change the busbar number.
Equipment must be present.	There is no equipment attached to the meter. Add the appropriate equipment.
Busbar number must exist (busbar may link to a Busbar, transformer, meter, source, or breaker)	At least one busbar must be linked to this meter.
Warnings (⚠)	
Line Active should be present.	Line Active should be entered to determine when the meter detects power.
Busbars across all meters should be unique.	Verify that all busbars connected to this meter have the correct, unique, numbers.

Source information

The most commonly used information about the source genie displays by default.

When the Advanced Properties box is checked, the table expands to include everything that is known about the selected source(s).

Basic source information includes:

Column	Description
State	Normal (✓), Warnings (⚠), or Errors (⊗). See the following table for explanations of errors and warnings.

Column	Description
Page	Name of the page on which the genie is found (displays only from the folder level).
ID	This is the meter number, assigned when adding it to a page of a one-line.
Busbar	The number of the source that powers the connected busbar.
Line Active	The Cicode expression (such as MyTag1 > 0) that determines when the source detects power on the busbar.

Source Errors and Warnings

Before you use the drawing, correct all errors; otherwise the project might not compile and the animation will not work.

Warnings indicate settings that might be incorrect. Verify that the settings indicated by the warnings are what you want.

Errors and warnings that you might see for sources are:

State	Solution
Errors (⊗)	
Source number must be a number greater than 0 and unique.	The source number is missing, or it is less than or equal to 0. Add or change the source number.
Busbar number must be a number greater than 0 and unique across sources.	The busbar number is missing, or it is less than or equal to 0. Add or change the busbar number.
Busbar number must exist (busbar may link to a Busbar, transformer, or breaker)	At least one busbar must be linked to this source.
Warnings (⚠)	
Line Active should be present.	Line Active should be entered so the source can detect power on the busbar.

Breaker and Switch Information

The most commonly used information about the breaker genie displays by default.

When the **Advanced Properties** box is selected, the table expands to include everything that is known about the selected breaker(s).

Basic breaker information includes:

Column	Description
State	Normal (), Warnings (), or Errors (). See the following table for explanations of errors.
Page	Name of the page on which the genie is found (displays only from the folder level).
ID	This is the breaker number, assigned when adding it to a page of a one-line.
Equipment	The equipment name entered when adding the genie via the Profile Wizard or Automation Interface.
Source Busbar	The number of the source busbar.
Dest. Busbar	The number of the destination busbar.

Breaker and Switch Errors

Before you use the drawing, correct all errors; otherwise the project might not compile and the animation will not work.

Errors that you might see for breakers are:

State	Solution
Errors ()	
Breaker number must be a number greater than 0 and unique.	The breaker number is missing, or it is less than or equal to 0. Add or change the breaker number.
Source busbar number must be a number greater than 0.	The source busbar number is missing, or it is less than or equal to 0. Add or change the source busbar number.
Destination busbar number must be a number greater than 0.	The destination busbar number is missing, or it is less than or equal to 0. Add or change the destination busbar number.
Source and Destination busbars must not be equal.	The source and destination busbars have the same number; change one number.
Equipment must be present.	There is no equipment attached to the breaker. Add the appropriate equipment.
Either the Source or Destination Busbar number must exist (busbar may link to a Busbar, transformer, meter, source, or another breaker)	At least one busbar must be linked to this breaker.

Busbar Information

The most commonly used information about the busbar genie displays by default.

When the Advanced Properties box is selected, the table expands to include everything that is known about the selected busbar(s).

Basic busbar information includes:

Column	Description
State	Normal (), Warnings (), or Errors (). See the following table for explanations of errors.
Page	Name of the page on which the genie is found (displays only from the folder level).
ID	This is the busbar number, assigned when adding it to a page of a one-line.

Busbar Errors

Before you use the drawing, correct all errors; otherwise the project might not compile and the animation will not work.

Errors that you might see for busbars are:

State	Solution
Errors ()	
Busbar number must be a number greater than 0.	The busbar number is missing, or it is less than or equal to 0. Add or change the busbar number.

Automatic transfer switch (ATS) information

ATS Information

The most commonly used information about the ATS genie displays by default.

When the **Advanced Properties** box is selected, the table expands to include everything that is known about the selected ATS.

Basic ATS information includes:

Column	Description
State	Normal (), Warnings (), or Errors (). See the following table for explanations of errors.
Page	Name of the page on which the genie is found (displays only from the folder level).
ID	This is the breaker number for the left side, assigned when adding it to a page of a one-line.
ID2	This is the breaker number for the right side, assigned when adding it to a page of a one-line.
Source Busbar1	The number of the source busbar for the left side.
Source Busbar2	The number of the source busbar for the right side.
Dest. Busbar	The number of the destination busbar.

ATS Errors

Before you use the drawing, correct all errors; otherwise the project might not compile and the animation will not work.

Errors that you might see for ATSs are:

State	Solution
Errors (⊗)	
Breaker numbers must be a number greater than 0 and unique.	The breaker numbers are missing, or they are less than or equal to 0. Add or change the breaker numbers.
Source busbar numbers must be a number greater than 0.	The source busbar numbers are missing, or they are less than or equal to 0. Add or change the source busbar numbers.
Destination busbar number must be a number greater than 0.	The destination busbar number is missing, or it is less than or equal to 0. Add or change the destination busbar number.
Source and Destination busbars must not be equal.	The source and destination busbars have the same number; change one number.
Either the Source or Destination Busbar number must exist (busbar may link to a Busbar, transformer, meter, source, or another breaker)	At least one busbar must be linked to this ATS.

Transformer Information

The most commonly used information about the transformer genie displays by default.

When the **Advanced Properties** box is checked, the table expands to include everything that is known about the selected transformer(s).

Basic transformer information includes:

Column	Description
State	Normal (✓), Warnings (!), or Errors (⊗). See the following table for explanations of errors.
Page	Name of the page on which the genie is found (displays only from the folder level).
ID	This is the breaker number, assigned when adding it to a page of a one-line.
Source Busbar	The number of the source busbar.
Dest. Busbar	The number of the destination busbar.

Column	Description
Sim. Source	This is the top source number used when adding the transformer.
Sim. Source 2	This is the bottom source number used when adding the transformer.

Transformer Errors

Before you use the drawing, correct all errors; otherwise the project might not compile and the animation will not work.

Errors that you might see for transformers are:

State	Solution
Errors (⊗)	
Source busbar number must be a number greater than 0.	The source busbar number is missing, or it is less than or equal to 0. Add or change the source busbar number.
Destination busbar number must be a number greater than 0.	The destination busbar number is missing, or it is less than or equal to 0. Add or change the source busbar number.
Source and Destination busbars must not be equal.	The source and destination busbars have the same number; change one number.
Either the Source or Destination Busbar number must exist (busbar may link to a Busbar, transformer, meter, source, or another breaker)	At least one busbar must be linked to this transformer.
If a top or bottom source is identified, it must be greater than 0.	The number for the top or bottom source for this transformer must be greater than zero (for voltage-level transformers) or must be left blank (for pass-through transformers).

Compile the Project

After you install the software and create the project—along with clusters, network addresses, and servers—compile the project. You will also need to compile your project periodically during system setup.

Pack your project before you compile. In Power SCADA Studio, click the **Projects** activity, click **Pack**.

In Power SCADA Studio, click **Compile**  . If you are promoted to save your changes, click **Save**.

If there are errors or warnings after the project is compiled:

1. At each error, click **GoTo**, which opens the location where the error occurred.
2. Using the information in the error message, correct the error.
3. After all errors are addressed, re-compile to verify that the errors are removed.

   For additional information, click Help at the error screen.

Power SCADA Runtime menus

Content in the graphics pages is controlled in the `pagemenu.dbf` file. Use `pagemenu.dbf` to create the tabs and sub-tabs that will display on each graphics page. An example of a `pagemenu.dbf` file, for the PLS_Example project, is in:

C:\ProgramData\Schneider Electric\Power SCADA Operation\v9.0\User\PLS_Example.

The `pagemenu.dbf` file for your project is in the same User directory, in the folder that matches your project name.

For instructions on editing the pagemenu.dbf file, see "Add Pages to Project Menu Configuration" on page 301

To create new genies for a project, see "Create a one-line on a graphics page" on page 280..

Add Pages to Project Menu Configuration

The Menu Configuration form (in Power SCADA Studio, click **Visualization**  **> Menu Configuration**) edits `Pagemenu.dbf` in your project. This controls the Power SCADA Runtime tabs and menus on the screen. You can also use menu configuration to specify actions that will be taken when an option is selected.

TIP: Copy and paste the menu settings from the PLS_Example project settings and use them as a template for your new project's menu configuration file.

The following image illustrates a blank Menu page for the PLS_Example project (see the table below for descriptions of the columns):

Row	Level 1	Level 2	Level 3	Level 4	Menu Command	Comme	Order	Symbol	Page	Project
1	Home				PLSNavPageHome()			pls_icons.greer		PLS_Example
2	Single Lines							pls_icons.greer		PLS_Example
3	Single Lines	Overview						PLS_Icons.over		PLS_Example
4	Single Lines	Overview	ANSI Style		PLSPageDisplay("OVER\			PLS_Icons.over		PLS_Example
5	Single Lines	Overview	IEC Style		PLSPageDisplay("OVER\			PLS_Icons.over		PLS_Example
6	Single Lines	12.47 kV Subs			PLSPageDisplay("SLD_3			PLS_Icons.sub		PLS_Example
7	Single Lines	4.16 kV Substa			PLSPageDisplay("SLD_6			PLS_Icons.sub		PLS_Example
8	Single Lines	480 V Substati			PLSPageDisplay("SLD_4			PLS_Icons.over		PLS_Example
9	Single Lines	480 V Substati			PLSPageDisplay("SLD_4			PLS_Icons.over		PLS_Example
10	Single Lines	480 V Substati			PLSPageDisplay("SLD_4			PLS_Icons.over		PLS_Example
11	Alarms / Event							pls_icons.greer		PLS_Example
12	Alarms / Event	Event Log			PLSDspShowAlarm(15)			PLS_Icons.ever		PLS_Example
13	Alarms / Event	Alarm Log			PLSDspShowAlarm(0)			PLS_Icons.alar		PLS_Example

Menu Item	Description
Levels 1 through 4	These items establish the menu levels that will display. For example, you might use "One-Lines" for level 1, followed by the substation for level 2, and the graphic name for level 3. (Each line: 256 characters maximum)
Menu Command	The Cicode expression that you want to execute. Typically, you will use the "page display" command followed by the actual page you want to see. For example: PLSPageDisplay("CB_IEC_1") which displays the page CB_IEC_1.
Order	The relative position within the final graphics page. If you leave this field blank, the default value 0 is used. (64 characters maximum)
Symbol	Displays a defined image along with the description for that level. Images must already be defined in the project/include project. They are specified in the format <library name>,<symbol name>. For example, in PLS_Example, the symbol used for the level 2 of One-Lines is Substation3, entered as PLS_Icons.Substation3. Different menu levels are designed to be used with different symbol sizes for optimal display. For Level 1 items (tab), the recommended symbol size is 16 x 16 pixels. For Level 2 items, (buttons), the recommended symbol size is 32 x 32 pixels. Symbols are not displayed for menu items of Level 3 or beyond.
Page	The page on which this entry will display. If this is left blank, the entry will display on every page.
Comment	You can use up to 128 characters to add a comment (will not display on screen).

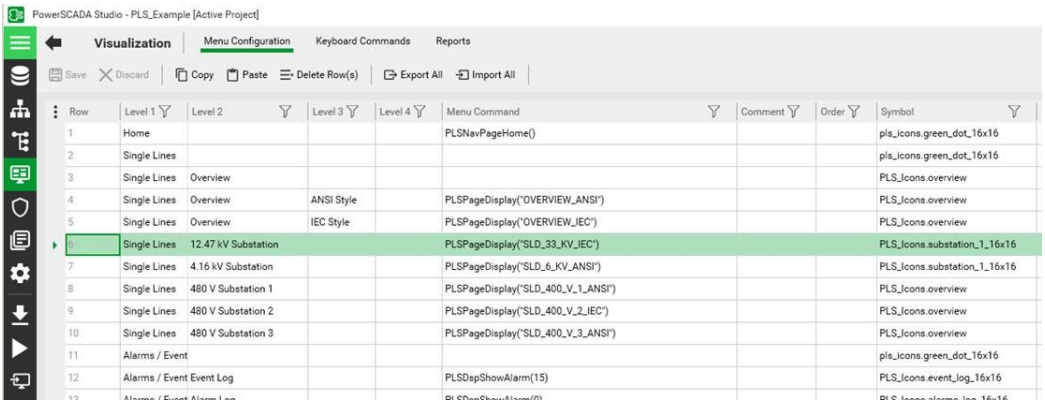
Add One-Line Pages

As indicated in "Add Pages to Project Menu Configuration" on page 301, you can easily add menu items for your one-line diagram pages by providing Level 1 - Level 4 menu item names and then using the PLSPageDisplay function in the Menu Command column to display your one-line pages by name. Do this for each one-line page you want to add to your project navigation.

For each one line page you want to add to your project navigation:

1. In Power SCADA Studio, click **Visualization**  **> Menu Configuration**.
2. In the **Menu Command** column, add the Cicode method that will open the page: `PLSPageDisplay("SLD_33_KV_IEC")`
3. In the **Symbol** column, type the appropriate symbol/size information. See "Add Pages to Pro-

ject Menu Configuration" on page 301 for information on this field.



Add Alarm Pages

To create separate alarm pages for each alarm type in the project:

1. In Power SCADA Studio, click**Visualization** **> Menu Configuration**.
2. In the Menu Command line, add the Cicode method that will open the page: `PLSDspShowAlarm (INT nType)`

   Where:

   nType = the type of alarm (e.g., 1=unacknowledged, 3=disabled)

   Example (for disabled alarms): `PLSDspShowAlarm(3)`

For more information on alarm types, see *AlarmDsp* in the Cicode Programming Reference help file.

TIP: The PLS_Example project also has several examples on how to add each alarm page to your project. With the PLS_Example project active in Power SCADA Studio, click **Visualization** **> Menu Configuration**. You will see all active alarms in a page named "Alarm Log" with AlarmType=0.

Add the Tag Viewer page menu item

The Tag Viewer displays in the graphics page during runtime. Use the Tag Viewer to view details about equipment. This screen provides the status of project tags.

To add the Tag Viewer to a project graphics page:

1. In Power SCADA Studio, click **Visualization** **> Menu Configuration**.
2. In the Menu Command line, add the Cicode method that will open the page:

   `PLSPageDisplay("PLSTagView")`

When viewing the Tag Viewer in runtime, as long as the screen resolution is one that Power SCADA Operation supports, the view will be correct.

For information about viewing tags, see "View the Tag Viewer" on page 487.

Add Menu Items for LiveView Data Tables

Using the names of real-time data table views that you saved earlier (see "Create Real-Time Data Views" on page 313), you need to add a Menu Configuration item for each saved view.

In Power SCADA Studio, click **Visualization** > **Menu Configuration**.

The following would save a view named "BasicReadingsSummary," with "localhost" used to indicate that LiveView is running on the Power SCADA server. Use the `PLS_LiveViewDsp` cicode function to display your saved view in the operator HMI.

- Level1: Applications
- Level 2: LiveView
- Level 3: Basic Readings
- Menu Command: PLS_LiveViewDsp("localhost", "BasicReadingsSummary", "BasicReadings")
- Symbol: PLS_Icons.Reports_16x16

Add the corresponding information for each saved real-time data table view you wish to see in the Power SCADA Runtime.

Add a Page menu item to Launch a WebDiagram

The following procedure describes how to access a WebDiagram by invoking Cicode from your project menu, however later procedures here describe how to alternatively add a WebDiagram view in your genie equipment popup. For more information see "Add Web Diagrams to Equipment Popups" on page 428.

To add a new page to the project that will display a given WebDiagram:

1. Create a new menu configuration item that calls the `PLS_WebReachDsp` Cicode explained below:
   a. In Power SCADA Studio, click **Visualization** > **Menu Configuration**.
   b. Enter the call to the PLS_WebReachDsp function (found in the PLS_Applications.ci file), with the slideshow (if desired), and the page title.

About the WebReachDsp Cicode

In the following step, you will call the `WebReachDsp` function from a button. This function is part of the Cicode in the `PLS_Include.ci` file, which is packaged with this document. The code is shown here for reference:

```
FUNCTION PLS_WebReachDsp(STRING sDeviceName, STRING sTitle = "")
STRING sPage = PLS_GetWebReachURL(sDeviceName);
IF ("" = sPage) THEN RETURN; END

IF ("" = sTitle) THEN sTitle = sDeviceName; END
PLS_WebDsp(sPage, sTitle);
END
```

There are some important things to note about this code:

- `sDeviceName` is the name of the device, determined in the previous topic.
- `sTitle` is the title of the page

If the diagram does not display, try the following troubleshooting steps:

- Enter the URL of the diagram directly into a browser window; verify that it launches

  The URL is: http://<*servername*>/ION/default.aspx?dgm=OPEN_TEMPLATE_DIAGRAM&node=<*device name*>

  If this does not work, verify that the WebReachServer is correct in `citect.ini`, and the diagram appears correctly in WebReach.
- The steps above should resolve most issues. One last option is to test by putting the Web browser in a window on the calling page.

Basic Reports

The Power SCADA Operation reporting feature is an Internet Information Services (IIS) Web application that is typically hosted on the same server as the Power SCADA services. The PLS_Inlude project defines a *PLS_ReportPage*, along with its screen resolution-specific variant pages. *PLS_ReportPage* contains a Microsoft Web Browser ActiveX control in which the reporting Web pages are displayed.

Power SCADA Operation with Advanced Reporting and Dashboards includes two different types of reports, basic and advanced.

Basic reports include the following:

- single-device usage reports
- mutli-device usage reports
- tabular reports
- trend reports

Advanced Reports and Dashboards are available when the Advanced Reporting and Dashboards Module is purchased and installed with Power SCADA Operation. See the *Power Monitoring Expert 9.0 - System Guide* for information on advanced reports.

Prerequisites

Before you can set up basic reports to generate and view reports, you must:

- Set up data acquisition parameters. To do this, use the Application Configuration Utility. See "Set up data acquisition parameters" on page 152 and (for receiving reports via email) "Configure basic reports for email" on page 309 for instructions.
- In Power SCADA Studio > System > Menu Configuration, menu tabs are configured to use the new "PLS_ReportDsp()" Cicode function to send URLs to the Web browser control at runtime. The control then browses to the available reporting Web pages. See the PLS_Example project for examples of this functionality.
- When switching between Power SCADA Operation projects in runtime, you must restart the Schneider Electric Service Host (CoreServiceHost) service before you run the reporting application. This allows the reporting application to load data from the currently running Power SCADA Operation project.

To get started setting up a report, see "Set up the Power SCADA Runtime for basic reports" on page 306.

For descriptions of each report type, see "Basic reports" on page 488.

NOTE: If you install Matrikon Explorer on the same computer as Power SCADA Operation, the LiveView and reporting features will not launch. To prevent this, install Matrikon before you install Power SCADA Operation. If you install Matrikon after you install Power SCADA Operation, you need fix the issue in this way: Go to IIS > ISAPI Filters, and then reset the DLL that is already selected (click browse and re-select *v4.0.30319 aspnet_filter.dll*). Click OK.).

Set up the Power SCADA Runtime for basic reports

Follow these steps to add new items to the project, add the necessary INI parameters for CtAPI and basic report security, and create the CtAPI connection for reporting.

For a complete discussion of reporting web application URLs, see "Create and view basic reports" on page 492.

Create the menu items for report page

The following steps describe how to interact with the reporting web application via the runtime environment.

1. In Power SCADA Studio, click **Visualization**  **> Menu Configuration**.
2. Add the new menu item that you want for each of your basic reports.
3. In each of these menu items, in the Menu Command line, add the Cicode method that will display a report tab. You can create your own custom method or use the default:

   ```
   PLS_ReportDsp(STRING sIPAddress, STRING sName, STRING sOptions =
   "", STRING sTitle="")
   ```

   Examples:

   ```
   PLS_ReportDsp(“10.10.10.10", "SingleDeviceReport”,
   "ShowConfiguration/MyConfiguration", “Single Device Usage
   Report”);
   ```

   or

   ```
   PLS_ReportDsp(“10.10.10.10", "SingleDeviceReport”, "", “Single
   Device Usage Report”);
   ```

   which opens an unconfigured single device usage report at the parameters entry page.

Add the following INI parameters

To allow trend queries that yield the desired amount of historical data:

```
[Trend]MaxRequestLength =100000000,
allowable range: 1-100000000
(example: a value of 70080 would yield two years of data for one
device/one topic, assuming 15-minute trends)
```

To allow CtAPI to connect remotely:

```
[CtAPI]Remote = 1
```

To define a privilege level for users to view reports:

```
[Reporting]PrivLevel - Default = 0
```

To define an area for users to view reports:

```
[Reporting]Area - Default = 0
```

See also:

"Localizing Power SCADA Operation" on page 374

Set up a display client for basic report viewing

To properly interact with the basic reporting Web application at a display client, you must set a registry key to force the Microsoft Web Browser ActiveX control to use Internet Explorer 9 emulation.

NOTICE
IRREVERSIBLE OPERATING SYSTEM DAMAGE OR DATA CORRUPTION Before making any changes, back up your Windows Registry in a network folder or other remote location. **Failure to follow these instructions can result in irreparable damage to your computer's operating system and all existing data.**

NOTE: Registry edits must be performed only by qualified and experienced personnel.

Create the following DWORD value at the following registry key path:

Path: HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\Microsoft\Internet Explorer\MAIN\FeatureControl\FEATURE_BROWSER_EMULATION

Value Name: Citect32.exe

Value: 9999 (decimal)

NOTE: This registry setting affects the Citect32.exe process only. It has no effect on other applications that use the Microsoft Web Browser ActiveX control.

Enable Windows Authentication for basic reporting

You can use Windows Authentication for logging in to the basic reports application. This could be to authenticate from Active Directory or to provide a single-sign-on.

NOTE: These steps are specific to Windows 7; they may be different for other operating systems. For further assistance, view Microsoft's documentation on this topic at: http://technet.microsoft.com/en-us/library/cc754628(v=ws.10).aspx

To enable Windows Authentication for basic reporting:

1. Turn on Windows Authentication:
   a. From the control panel, click Programs and Features > Turn Windows features on or off.
   b. Check Windows Authentication.
2. Enable Windows Authentication in IIS:
   a. From the Control Panel > Administrative Tools, choose Internet Information Services (IIS).
   b. Select the root node from the tree on the left (or the Reporting website, if this server hosts multiple sites).
   c. From the right pane, in the IIS section, click Authentication.
   d. Enable Windows Authentication.
3. Modify web.config to specify Windows Authentication:
   a. From the root of the Reporting direction, locate web.config.
   b. Change the line:

   ```
   <authentication mode="Forms">
   ```
   to
   ```
   <authentication mode="Windows">
   ```
4. Add roles to web.config to allow access to the basic reporting application. For example, to allow the role (group) Administrators, add the following to the web.config file:

   ```
   <authorization>
   <allow roles="Administrators"/>
   <deny users="?" />
   </authorization>
   ```

Modifying the web.config file is an advanced topic that is covered on the Microsoft Web site. See the following link for instructions provided on the Microsoft Web site:

http://www.iis.net/configreference/system.webserver/security/authentication/windowsauthentication

Additional information may be available in the following knowledge base article:

http://support.microsoft.com, and search on kb/815179.

Configure email settings to send basic reports

You can send Power SCADA Operation basic reports to multiple email addresses.

NOTE: You must configure the SMTP server and email list(s) before you email reports. See "Email basic reports" on page 496 for instructions on sending these emails.

SMTP Server and From Address

For instructions on setting up the SMTP server, see "Configure basic reports for email" on page 309.

Email Lists

Before you can send email via the URL or ReportMailer method, you must create at least one email list:

1. In a text editor; enter one or more email addresses (one per line, no commas).
2. Save this text file in the `Reporting\ReportConfigurations\` directory, located on the application root install directory (which is also the physical directory behind the reporting web application's virtual path in IIS).

   Example (64 bit):
   `C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\Power SCADA Operation Reporting\Reporting/ReportConfigurations\`

   The file name must be in the following format:

   `Email_<EmailListName>.cfg`

   Where:

   `<EmailListName>` = an alphanumeric (no spaces) name for the email list (for example, Administration)

Email Body

The email body that you send is contained in a resource (.resx) file in the `Reporting\bin\Resources\Reporting.en-US.resx\` directory, located on the application root install directory (which is also the physical directory behind the reporting web application's virtual path in IIS).

Example (64 bit):
`C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\Power SCADA Operation Reporting\bin\Resources\Reporting.en-US.resx\`

The email body is the same for all Report Configurations and Email Lists, but you can modify the entry for ReportEmailBody to change the body of the email that is sent.

Configure basic reports for email

Use this screen to set up the delivery method and email address from which Power SCADA Operation 9.0 basic reports will be sent. These settings specify the SMTP server for emailing basic reports.

NOTE: This screen is not used for configuring the SMTP server to send notifications.

Define the following:

- **Timeout:** The number of seconds Power SCADA Operation will attempt to deliver an email before no longer attempting
- **Delivery Method:** Network (default), Pickup Delivery from IIS, or Specified Pickup Directory. This is an SMTP-specific setting. In most cases, use Network. For more information on SMTP, see the Micrologic Developer Network website.
- **'From' Address:** the address from which reports will be sent.
- **Host:** The IP or network address of the SMTP server.

- **Port:** The network port to be used; default for SMTP is 25.
- **Use Default Credentials:** If required by the SMTP server being used, uncheck the box and enter the appropriate user name and password. If not required, check the box and enter the SMTP user name and password used for reporting.

Email basic reports

Before you can email Power SCADA Operation basic reports, configure the SMTP server and email list(s). See "Configure email settings to send basic reports" on page 308 for details.

There are 3 ways to email basic reports:

1. The Report Viewer email button
2. Visit a Specific URL
3. Use Cicode via ReportMailer

Report Viewer email button

Use this method to send a customized one-time email to an individual or group of email addresses.

1. Run the report as normal.
2. In the Report Viewer, click  (**Email**) .
3. Enter the requested information in the pop-up dialog.
4. Click **Send**.

Visit a Specific URL

NOTE: Each visit to a URL causes the email to be sent. Be sure that you have the correct report and email list before you visit this URL/send the email. Also, you should secure this URL using the web.config file. For information on modifying/using the web.config file, see http://support.microsoft.com, and search on kb 815179.

To send a basic report to an existing email list, visit the following URL:

```
http://<
ServerName
>/Reporting/Report/<
ReportName>/<ReportConfiguration>/Email/<EmailList>
```

where:

- <ServerName> = the name or IP of the reporting server
- <ReportName> = the name of the report you wish to view
- <ReportConfiguration> = the name of the saved configuration to use
- <EmailList> = the name of the email list you wish to use

You must use a saved configuration (see "Create and view basic reports" on page 492 for instructions). You cannot change report parameters from this URL.

No progress bar or update will display, as these interfere with some scheduling clients.

Use Cicode via ReportMailer

You can use a utility called ReportMailer to email basic reports. This command line utility is located in the PLS_Include project. It can be called by Cicode. You can create a button on the graphics page and have it call the Cicode function or use a scheduled process to trigger an email.

Before you can use ReportMailer, you need to create or edit the file called `ReportMailer.ini` file that is in your project (not in PLS_Include). The `ReportMailer.ini` file must include the text listed in the following table:

Text Field	Required Setting	Description
LoginUsername	demo	Username for logging in to reporting system for emailing reports
LoginPassword	demo	User's password, will be encrypted on the first run
IsEncrypted	False	Flag that indicates if the password is encrypted. If you change the password, edit the field (replacing the unreadable encrypted entry, if one exists). Then change this value to False. The new password will be encrypted at the next startup cycle, and this field will be updated to True.
ScadaBinPath	C:\Program Files (x86)\Schneider Electric\Power SCADA Operation 9.0\Bin	The bin path of Power SCADA Operation
LogOnUrl	http://SCADASERVER/Reporting/LogOn.aspx	The URL of the logon page(this is an example; use your own server name)
ReportServerName	SCADASERVER	The name or IP address of the server running the reporting application
LogLevel	All	The level of logging you want in the report mailer application. This log is saved to a ReportMailerLog.txt file in the running project's directory. Possible settings are ALL, DEBUG, ERROR, WARN.

After this file is configured, run the `ReportMailer.exe` with the following syntax:

`ReportMailer.exe <ReportName> <ReportConfiguration> <EmailList> <ScadaProjectPath>`

where:

- <ReportName> = the name of the report you wish to view
- <ReportConfiguration> = the name of the saved configuration to use
- <EmailList> = the name of the email list you wish to use
- <ScadaProjectPath> = the full path to your SCADA project

This command line application may be called from Cicode using the following example:

```
FUNCTION
PLS_EmailReport()
ErrSet(1);
STRING FilePath = ParameterGet("CtEdit","User","") + "\PLS_Include\
ReportMailer.exe " + "MultiDeviceReport SampleConfiguration SampleList
" +
"^"C:\ProgramData\Schneider Electric\Power SCADA Operation\User\PLS_
Example^"";
Exec(FilePath);
END
```

NOTES:

- The SCADA project path must be enclosed in escaped quotes ("^").
- This is an asynchronous (non-blocking) call. While the EXEC() method will return immediately, it may take a few moments to run and email the report. See the web.config timeout value (see option 2 above) for more information.
- You can also call the ReportMailer application directly from a command line. In this case, you can add the term "blocking" to the command line (as a fifth parameter). This causes ReportMailer to act in a synchronous state (block the call) and to return any error messages to the console. Never use the "blocking" parameter by Cicode, as it could prevent EXEC() from returning in a timely fashion.

Scheduling basic reports

You can schedule the emailing of basic reports by executing the above Cicode as an action from a timed event. For more information, see **Configuring Events** in the Citect SCADA help file (...\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin\Help\Citect SCADA).

You can also use the Windows Task Scheduler to send these reports. Refer to Microsoft's documentation on Using the Task Scheduler (Microsoft Docs).

URL routing for basic reports

The basic reporting application uses ASP.NET extension-less URL routing. Depending on your operating system, you might need to complete additional steps to enable URL routing in your project.

Windows 2008 R2 and Windows 7

Microsoft has discovered an issue with extension-less URL routing in certain installations of Internet Information Services (IIS) 7.0 and IIS 7.5. To address this issue, Microsoft released a hot fix referenced by KB article 980368. This hot fix is available at http://support.microsoft.com/kb/980368.

This hot fix is included in Service Pack 1 for Windows 2008 R2 and Windows 7. To receive the hot fix, you should install Service Pack 1. This installation provides additional important updates to the operating system. To obtain Service Pack 1 for Windows 2008 R2 and Windows 7, go to either Windows Update or http://support.microsoft.com/kb/976932.

Set up IEC 61850 advanced control

The advanced control window provides two advanced controls (synchro check and interlock check) that you can use with IEC 61850 IEDs.

⚠ WARNING

INACCURATE DATA RESULTS

- Do not incorrectly configure the tag.
- Ensure that you understand the effects of using the "bypass" option so you do not shut down critical equipment.
- Do not rely solely on software messages and reports to determine if the system is functioning correctly or meeting all applicable standards and requirements.
- Consider the implications of unanticipated transmission delays or failures of communications links.

Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

Enable the advanced control

Before you can use the advanced control, you must add the appropriate variable STRING tag to be used when you send the command. For breaker control, the "operate" tag typically used is:

```
S33K_A_INC\CSWI1\Pos\ctVal
```

For this tag, you need to then add the corresponding STRING tag:

```
S33K_A_INC\CSWI1\Pos\ctVal\str
```

If you are using select before operate, you also need to add a STRING tag for it.

See "Perform IEC 61850 advanced control" on page 485 for information on using these advanced controls.

Create Real-Time Data Views

Create and view LiveView templates and views for real-time data tables. Some basic predefined templates are included with the software; you can create new templates or make copies of the predefined templates and edit the copies.

Before you view LiveView templates and views, you must set up data acquisition parameters. To do this, use the Application Configuration Utility. See "Set up data acquisition parameters" on page 152 for instructions.

NOTES:

- If you find that a predefined table does not include enough cells for the data you want to display, use the duplicate feature to make a copy of the predefined table. Then add the needed cells to the duplicate.

- If you install Matrikon Explorer on the same computer as Power SCADA Operation 9.0, the LiveView and reporting features will not launch. To prevent this, install Matrikon before you install Power SCADA Operation 9.0. If you install Matrikon after you install Power SCADA Operation, you need fix the issue in this way: Go to IIS > ISAPI Filters, and then reset the DLL that is already selected (click browse and re-select *v4.0.30319 aspnet_filter.dll*). Click OK.).

You can only view data in these templates if your system is online and you are connected to devices that provide data.

To set up LiveView real-time data tables in the Power SCADA Runtime:

1. Open the LiveView Viewer in your Internet browser:

   http://localhost/LiveViewViewer
2. Create a custom template or choose an existing template.
3. Select devices from which to show real-time data
4. Save the view, providing a name.

   Keep track of the names of your saved views. You will need to use them when you create menu items that display these views in the Power SCADA Runtime.

LiveView Viewer

Use this screen to view table templates, and to view or create table views, in the LiveView Viewer.

To open this screen, in the Power SCADA Runtime, click the menu links that have been set up when you created the graphics page (see "Create menu item for LiveView page" on page 318). In the PLS_Example project, there is a tab for LiveView. For information about an individual table, click a link from the Contents folder.

NOTE: If you plan to view a table using the "Rapid access labels (QR codes)" on page 499 feature, do not change its name after you print the QR code. If the name is changed, you must generate a new rapid access label.

Open LiveView from a URL

Before you can open LiveView from a URL, you must select a template and the desired devices, display the table, and save it as a View.

To open this view using a URL, use one of the following options:

- From the computer where LiveView is installed, enter http://localhost/LiveViewViewer
- From a remote client computer, enter http://10.10.10.10/LiveViewViewer (where 10.10.10.10 is the URL of the server where LiveView resides)

To automatically open a specific table when you launch LiveView Viewer, add the table name to the end of the address. For example, to open the basic readings summary view while on the local computer, you would enter: http://localhost/LiveViewViewer/Basic Readings Summary View

LiveView Viewer Display

The Live View Viewer displays with two tabs, Templates and Views.

Templates: A template includes all setup data (placeholders, formulas, thresholds, and formatting); but it does not have devices selected. The templates include those that are predefined (designated by the locked symbol:), as well as those that have been defined in the Setup window.

To view a template:

1. Select the template from the list.
2. Select the device(s) for which you want to display values. (Only devices that have at least one assigned topic from the topic placeholders in this template are available for selection.
3. Click Display Table to view the template in the right-hand pane.

To save a template as a view:

1. With a template displaying, click **Save** ( ) on the upper right of the Viewer page.
2. In the View Name window, edit the name, then click **OK**.

The new view is saved in Tables > Views on the server. The view will also display in the left-hand pane of the Views tab.

Views: A view is a template that is saved with its device selection(s). The views listed are saved on the server in Tables > Views. Views are available to all users, whether on the server or a client. They also display on the Views tab of the Live View Viewer.

To open a view:

Select a view and then click **Display**.

The view displays in the right-hand pane with updated data. You can delete a view (click Delete, to the right of the View tab). You can change a view by adding or deleting devices and then either overwriting the view or saving it as a new view.

Update List: This link forces the cached table and view lists to be refreshed, displaying any newly added tables and views.

Select Device(s) and Update Device List: This link forces any new devices (with at least one assigned topic) to display. In the Select Device(s) list, you can move devices higher or lower in the list that you see, so that they display in the order you prefer. To do this, right-click and highlight a device, then click one of these icons:

: Move to the top (double arrow) or move up one step (single arrow)

: Move to the bottom (double arrow) or move down one step (single arrow)

"Where's My Device?" on page 316: Click this link to explain why an expected device does not display in the table.

Template and View Features

The template (after you click Display) or view displays with devices and data. The following information is included:

Placeholders: All placeholders that were added during setup will display with the appropriate device name or tag value.

Thresholds: If any of the tag values are outside of the normal range established in the Thresholds that were added during setup, the font color will reflect the high or low status of that tag.

On the right-hand side of the top of the screen are these buttons:

Save (): Click to save a template as a "view." You are prompted to name the view (default: table name appended with "view"). The view is saved in Tables > Views on the server. The view includes the devices that were selected for the table.

Notes ( ): Click to display a description of the table that was added when the table was set up.

Print ( ): Click to print a copy of the table with its current values.

Last Update: This is the most recent date/time that the template or view values were updated.

Update Interval: Choose the interval of time that will pass between requests to update the data in the template or view. Options are:

- **Manual:** Updates only occur when you click Update Now.
  - 5 seconds
  - 10 seconds
  - 30 seconds
  - 1 minute
  - 10 minutes
- **Update Now**: Click to manually update values and refresh the template or view.

Where's My Device?

This help topic displays when you click "Where's My Device?" below the device list in LiveView Viewer.

Missing topics

Only devices that have topics available for the selected template will appear in the device selection list. If you do not see an expected device, it is missing because it does not include topics that are used in this template.

If this is a template that you have created, you can open the template in the "LiveView Placeholders" on page 320 screen of LiveView Template Editor to add the placeholder(s). If this is a predefined template, you cannot change it; you will need to make a duplicate template and then add the desired placeholder(s).

Clear cache and platform refresh

If the Schneider Electric CoreServiceHost has not been refreshed after devices or topics have been added, you should clear the cache and refresh the platform in order to access the new devices or topics.

See "Clear cache and refresh platform" on page 373 for instructions.

Set up LiveView

Use LiveView Template Editor to begin creating, duplicating, modifying, and deleting LiveView templates and views.

You can configure a LiveView template in LiveView Template Editor, and then display it on the server or on a web client. A "template" includes all of the setup data except devices.

You can create views of templates in LiveView Viewer. A *view* is a template that includes devices.

To open LiveView Template Editor:

From Start click Schneider Electric >Template Editor.

Only one user at a time can access LiveView Template Editor. When a user accesses LiveView Template Editor, a file called *TemplateEditor.lock* is saved on the Power SCADA Operation folder of the server (default location: Program Files > Schneider Electric > Power SCADA Operation > 9.0 > Applications > LiveView > Viewer). If necessary, an administrator can unlock the utility by deleting *TemplateEditor.lock* from the server.

Here you can see:

Notes icon (On the far right): Opens a free-form field to add any descriptive information about the template that will be useful. The information displays in a notes field, to the right of the template. Click **Done** to close the Notes field.

In the left-hand pane are the following:

New: (You are prompted to save if you are editing a template that is not saved.) Click to save the template you are editing, and then to add a new template. The "New Template" name displays in the list, a new template file is uploaded to the server in Table > Templates, and an empty template displays in the right-hand pane. All fields are set to their defaults.

Duplicate: Click to save a copy of the selected template. The current template name is used with "Copy" appended. Use this option to edit a predefined template.

NOTE: If you find that a predefined table does not include enough cells for the data you want to display, create a duplicate. Then add the needed cells to the duplicate.

Delete: Click to delete the current template (you cannot delete predefined templates). Confirm that you want to delete it. All views associated with the template will also be deleted.

Select Template: This list includes all of the templates that are set up. Predefined templates display a lock icon ( ) to the left of the name. These templates cannot be deleted or edited.

Template Name: Overwrite the current name, which updates the template here and in the list of templates. This will also update the views that are associated with this template.

Single Device (default) or **Multiple Device**: Click one of these options for the type of template you want.

View Area: Use this field to determine the area of the table that will be viewed in LiveView Viewer. When you set up a table, there may be information (such as formulas or notes) that you do not want to display in the final table in the Viewer. To select only the material that you want to view, do one of the following:

- In **View Area**, type the cell range that you want to view (for example, A1:D20).
- Select the cells that you want to include, then press **Use Selection**.

In either case, a border displays around the cells in the range you select.

Save the template. When you view it in LiveView Viewer, it will only include the cells you selected.

Save: This button is enabled when you make a change to a template that is edited. The template is saved as an .xlsx file; it is uploaded to the server in Table > Templates. The saved template appears in the View tab after you click Save. (You do not need to click Save when you create a new template or a duplicate; these files are automatically saved.)

To create a new template, see "Create a LiveView template" on page 318.

See also:

"LiveView Formulas" on page 321

"LiveView Placeholders" on page 320

"LiveView Thresholds" on page 322

"LiveView Formatting" on page 319

Create menu item for LiveView page

The following steps describe how to interact with the LiveView application via the runtime environment.

1. In Power SCADA Studio:click **System > Menu Configuration**.
2. Add the new menu item that you want for each of your LiveView tables.
3. In each of these menu items, in the Menu Command line, add the Cicode method that will display a LiveView tab. You can create your own custom method or use the default:

   ```
   PLS_LiveViewDsp(STRING sIPAddress, STRING sViewName = "", STRING
   sTitle = "")
   ```

   Example:

   ```
   PLS_LiveViewDsp(“10.10.10.10", "BasicReadingsSummary”,
   "ShowConfiguration/MyConfiguration", “Basic Readings Summary”);
   ```

   which opens a configured LiveView table view with the saved configuration name "MyConfiguration".

Create a LiveView template

To begin creating LiveView templates:

1. In Programs, click Schneider Electric Table Editor.

   The LiveView Template Editor screen displays.
2. Open LiveView Template Editor click **New**.

   An empty template displays with a "New Template" name.
3. In **Template Name**, enter the template name. You can use up to 100 characters; limited to A–Z, a–z, 0–9, spaces, underscores, hyphens, and parentheses.
4. In **Single Device/Multiple Devices**, keep the default single device or click **Multiple Devices**.

5. To continue setting up the template, click one of the following links:
   - To add data formulas to the real-time table, see "LiveView Formulas" on page 321.
   - To add data (device names and tag names) to the real-time table, see "LiveView Placeholders" on page 320
   - To add visual alerts (color changes) when the value of the tag associated with a cell becomes too high or too low, see "LiveView Thresholds" on page 322.
   - To add formatting to cells, such as font and font size, see "LiveView Formatting" on page 319.

NOTE: Table grid lines do not display in the LiveView Viewer, however, they do display in LiveView Setup.

LiveView Formatting

Formatting lets you format the appearance of the cell; such as font, font size, and color.

NOTE: Formatting changes become visible only after you click outside of the cell that you change.

To use cell formatting:

1. In LiveView Template Editor, click the **Formatting** sub-tab.

   A formatting toolbar displays on the screen. It allows you to set the appearance of the cells in the template.
2. To format a cell or range of cells, select the cell or cells. When you select a format, the active cells will be set to the specified format attribute. When a cell becomes active, the format selections on the toolbar will reflect the selections for that cell. When you select multiple cells, the format selections will reflect those of the first cell you select.
3. Format the cell appearance by choosing the following:
   a. Font and font size
   b. Bold, italics, or underline
   c. A font color (default is black), and for the background of the cells (default is white)
   d. Horizontal alignment: flush left, centered, or flush right.
   e. Vertical alignment: top, center, or bottom.
   f. If more than one cell is selected, **Merge Cells** is enabled. Check this box to merge the selected cells into one large cell.
   g. In the **Data Type** drop-down box, select the type of data that will be in the selected cell(s):
      - **Text** (default); the *Wrap Text* box displays; check this box if you want text to wrap and stay within the cell.
      - **Date**: In the *Format* field that displays, type the format you want to use (Excel formatting is supported):
        24-hour format: m/d/yy h:mm:ss
        AM/PM format: m/d/yy h:mm:ss AM/PM or m/d/yy hh:mm:ss AM/PM

- **Number**: In the *Decimal Places* field that displays, choose the number of decimal places you want; if desired, check the *Use 1000 Separator* box to insert the separator (for example, comma, depending on your regional settings).

4. You can resize the row height or column width by dragging row/column header. A tooltip displays the height or width as you resize it.
5. Alternatively, right click anywhere in the template to display a context menu that allows you to insert or remove columns or rows, or to type the column width and row height.
6. Save your changes.

See also:

- "LiveView Formulas" on page 321
- "LiveView Thresholds" on page 322
- "LiveView Placeholders" on page 320

LiveView Placeholders

Placeholders provide the data—device names and tag names—to a LiveView template. The placeholders are the identifiers that are added when setting up the template, but are replaced with the name of the selected device or the tag value when the template is viewed.

To use this feature:

1. In LiveView Template Editor, click the **Placeholder** sub-tab .
2. Place the cursor in a cell. Note that the Insert Location displays the cell number for the placeholder you are setting.
3. From the drop-down field in the top left corner of the page, choose one of the following:
   - **Tag Value:** Select the tag group, such as Alarm, Current, Energy. Beneath the tag group, select the specific tag you want. The list is filtered to include only the most common tags that belong to the group you selected. To view all the tags available in this tag group, check **Show Advanced**.
   - **Device Name:** The list of devices is filtered to include only devices for which this template's data is available. To display the device name in this cell of the template, select Device Name. You will choose the actual device during runtime.
4. **Insert Location:** This offers a second way of inserting the placeholder location. After choosing the device or tag, type the cell number for the placeholder cell.
5. **Insert:** Click to add the selected placeholder to the specified cell.
6. Continue adding placeholders as needed.

See also:

- "LiveView Formulas" on page 321
- "LiveView Thresholds" on page 322
- "LiveView Formatting" on page 319

LiveView Formulas

Formulas let you include data in a LiveView template. You can add formulas to:

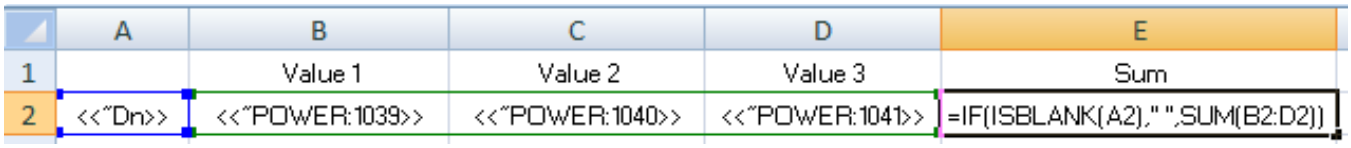
- Add, subtract, multiply, or divide the contents of two individual cells
- Add, multiply, or average the contents of a range of cells

To use formulas:

1. In LiveView Template Editor, click the **Formulas** sub-tab.
2. Choose one of the following fields:
   - **Cell:** Use this field to enable a formula for two individual cells. Then enter:
     - **Cell 1 Address:** Enter the cell address. The cell address displays in this field.
     - **Operator:** Choose the operator you want to use: +, –, *, or /.
     - **Cell 2 Address:** Enter the cell address. The cell address displays in this field.
   - **Cell Range:** Use this field to enable a formula for a range of cells. Then enter:
     - **Operation:** Choose average, product, or sum.
     - **Cell Range:** Enter the cell range (format C4:C20), or select the range of cells to include in the formula. The cell range displays in this field.
   - **Insert Location:** Enter the cell number.
   - **Insert:** Click this button to build the formula you have specified, and to add it to the cell you added to Insert Location.
3. Repeat the above procedure for the rest of the formulas you want to use for this Live View template.

NOTES:

- You must "Protect Current Sheet" for formulas to be maintained and visible in the LiveView Template Editor.
- If you want to use conditional formulas ("IF" formulas), you must first create them in Excel. To do this, you must access the template you want on the server (Program Files > Schneider Electric > Applications > LiveView > TemplateEditor >Templates Temp). Open the template in Excel and add the conditional formulas that you want. After you save the changes, the formulas will function correctly in Live View. You must copy the IF statement into every cell of the column that displays the result of the IF statement.
- In multiple device tables that rely on formulas to display information for each device, the results column will display zeroes when that row has no device in it. To avoid this, use a formula that will display no result if there is no device in that row. In the following example, when no device is in cell A2, no results will display (no zeroes) in cell E2.



	A	B	C	D	E
1		Value 1	Value 2	Value 3	Sum
2	<<"Dn>>	<<"POWER:1039>>	<<"POWER:1040>>	<<"POWER:1041>>	=IF(ISBLANK(A2)," ",SUM(B2:D2))

LiveView Thresholds

Thresholds let you display tag readings that fall outside of the normal range. You can apply it to an individual cell or a range of cells. You determine the tag or tags for which you want to display out-of-normal (threshold) readings. When the value of the tag in a cell (or any tag in a cell range) is below the minimum or above the maximum that you set, the tag value displays in the threshold cell.

You can set both minimum and maximum values for a cell or cell range. Use different colors to indicate the high and low readings.

To add a threshold:

1. In LiveView Setup, click the **Threshold** sub-tab.
2. Depending on the number of cells, do one of the following:
   - **Cell:** For a single cell: Select the cell for which you want the font color to change. The font color will change when the value for the tag in that cell goes above the specified Max Value (or below the Min Value) for the threshold.
   - **Cell Range:** For a range of cells, either select the range, or type the range in the format C4:C20.

   When setting up a multiple-device table, you should use a cell range to ensure that threshold font colors display for each device in the table.
3. In **Min Value**, type the low value for the "normal" range. If the tag value drops below this value, the cell font color will change as specified in step 4.
4. **Below Min Threshold Color:** Open the color palette and select the font color that you want to indicate the "low" status.
5. in **Max Value**, type the high value for the "normal" range. If the tag value goes above this value, the cell font color will change as specified in step 6.
6. **Above Max Threshold Color:** Open the color palette and select the font color that you want to indicate the "high" status.
7. **Insert Location:** Choose an empty cell, one that is not part of the table. This cell will be the location for the threshold definition that you are creating.

   The default cell for the threshold definition is the next available cell in the template. For example, if the tag in cell B7 has an unused cell to the right of it (C7), the threshold definition defaults to C7. Then, when the value in B7 exceeds the threshold defined in C7, the value in B7 displays in the font color you specified. To override the default cell location, change it in the Insert Location field.
8. Click **Insert** to create the thresholds.

The threshold definition is in the form: `<<Threshold;B2:B20;Min=100;Max=1000>>`

See also:

- "LiveView Formulas" on page 321
- "LiveView Placeholders" on page 320
- "LiveView Formatting" on page 319

Modify LiveView template

You can modify any template except one that is predefined. Predefined templates have a lock icon ( ) beside their names.

1. Open LiveView Template Editor.
2. In the Power SCADA Runtime, click the menu links that have been set up when you created the graphics page (see "Create menu item for LiveView page" on page 318). In the PLS_ Example project, there is a tab for LiveView.
3. Highlight the name of the template that you want to modify. The template displays.
4. You can change any field on the template. Click any of the sub-tabs (Placeholder, Formula, Threshold, or Formatting) to edit the related information. For help on the sub-tabs, see the "See Also" links below.
5. When you have finished making changes, click **Save**.

Continue working with other templates.

See also:

- "LiveView Placeholders" on page 320
- "LiveView Formulas" on page 321
- "LiveView Thresholds" on page 322
- "LiveView Formatting" on page 319

Duplicate LiveView template

You can duplicate an existing template, including predefined templates. The duplicated template will not be locked, allowing you to edit and save it as a different template.

1. Open LiveView Template Editor.
2. In Runtime mode, click the menu links that have been set up when you created the graphics page (see "Create menu item for LiveView page" on page 318). In the PLS_Example project, there is a tab for LiveView.
3. Highlight the name of the template that you want to duplicate. The template displays.
4. Click **Duplicate** (on the top of the left-hand pane).

   The duplicate template is added to the list. It has the same name of its original template, appended with "Copy."
5. Change the name of the duplicated template to differentiate it from its original.
6. Make the desired changes and then click **Save** to save them.

LiveView delete

You can delete any template except one that is predefined.

1. Open LiveView Template Editor.

2. In Power SCADA Runtime, click the menu links that have been set up when you created the graphics page (see "Create menu item for LiveView page" on page 318). In the PLS_Example project, there is a tab for LiveView.
3. Highlight the name of the template that you want to delete. The template displays.
4. Click **Delete** (on the top of the left-hand pane).
5. You are prompted to verify the deletion.
6. Click **Yes** to delete the template, or click **No** to cancel the deletion.
7. Continue working with other templates.

Enable Windows Authentication for LiveView

You can use Windows Authentication for logging in to LiveView. If you want to use Windows Authentication, you must follow standard IIS authentication methods.

NOTE: These steps are specific to Windows 7; they may be different for other operating systems. For further assistance, view Microsoft's documentation on this topic at: http://technet.microsoft.com/en-us/library/cc754628(v=ws.10).aspx

To enable Windows Authentication for LiveView:

1. Turn on Windows Authentication:
   a. From the Control Panel, click Programs and Features > Turn Windows features on or off.
   b. Check Windows Authentication.
2. Enable Windows Authentication in IIS:
   a. From the Control Panel > Administrative Tools, choose Internet Information Services (IIS).
   b. Select the root node from the tree on the left (or the LiveViewViewer node, if this server hosts multiple sites).
   c. From the right pane, in the IIS section, click Authentication.
   d. Enable Windows Authentication.
3. Modify web.config to specify Windows Authentication:
   a. In Windows Explorer, navigate to ...\Power SCADA Operation\v9.0\Applications\LiveView\Viewer
   b. Open `web.config`.
   c. Change the line:

      ```
      <authentication mode="Forms">
      ```

      to

      ```
      <authentication mode="Windows">
      ```
4. Add roles to `web.config` to allow access to the LiveVIew application. For example, to allow the role (group) Administrators, add the following to the `web.config` file:

```
<authorization>
<allow roles="Administrators"/>
<deny users="?" />
</authorization>
```

Modifying the web.config file is an advanced topic that is covered on the Microsoft Web site:

http://www.iis.net/configreference/system.webserver/security/authentication/windowsauthentication

Additional information is available in the following Microsoft knowledge base article:

http://support.microsoft.com, and search on kb/815179.

Compile the Project and Launch the Power SCADA Runtime

After you install the software and create the project (along with clusters, network addresses, and servers, perform your first system compile. You will also do this periodically during system setup.

It is always a good idea to "pack" before you compile. From the **Projects** tab of the Power SCADA Studio, click **Pack**. Then, from the left side of the page, click **Compile**  . Correct any errors and note any warnings.

To run the Computer Setup Wizard:

1. In Power SCADA Studio: Click **Projects > Home**, then click **Setup Wizard**.
2. Choose **Custom Setup** and **Multi-Process** mode.
3. Click **Networked** (instead of Stand alone.)
4. Enter a "Server Password". You do not need to remember this password.
5. Choose **Kernel on Menu** which will help with future troubleshooting.

To launch the Power SCADA Operation runtime:

In Power SCADA Studio: Click **Run the active project**  .

If you are running Power SCADA as a Service, navigate to the Power SCADA Operation bin directory, and launch the Service Display client shortcut.

Notifications

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Do not rely solely on Notifications Settings for alarm notifications where human or equipment safety relies on successfully delivered notifications.
- Do not use Notifications Settings for critical control or protection applications where human or equipment safety relies on the operation of the control circuit.
- Consider the implications of unanticipated transmission delays or failures of communications links.

Failure to follow these instructions can result in death or serious injury.

NOTE: Other parts of the overall communication system, such as email servers and cellular phone systems, could fail and result in notifications not being delivered. If notifications are not delivered to recipients, conditions that cause alarming may persist and result in safety critical issues.

Overview

Notifications alert specific people in your facility about critical power incidents no matter where they are. Notifications deliver timely alerts of power system events to the mobile phone, email or pager of designated users and helps them quickly identify system abnormalities and take appropriate action.

Notifications provide:

- View-based alarm grouping
- Basic and custom alarm filtering
- Flexible notification schedules
- SMS and email notification relay
- Primary and Standby Alarm Server synchronization
- Maintenance mode

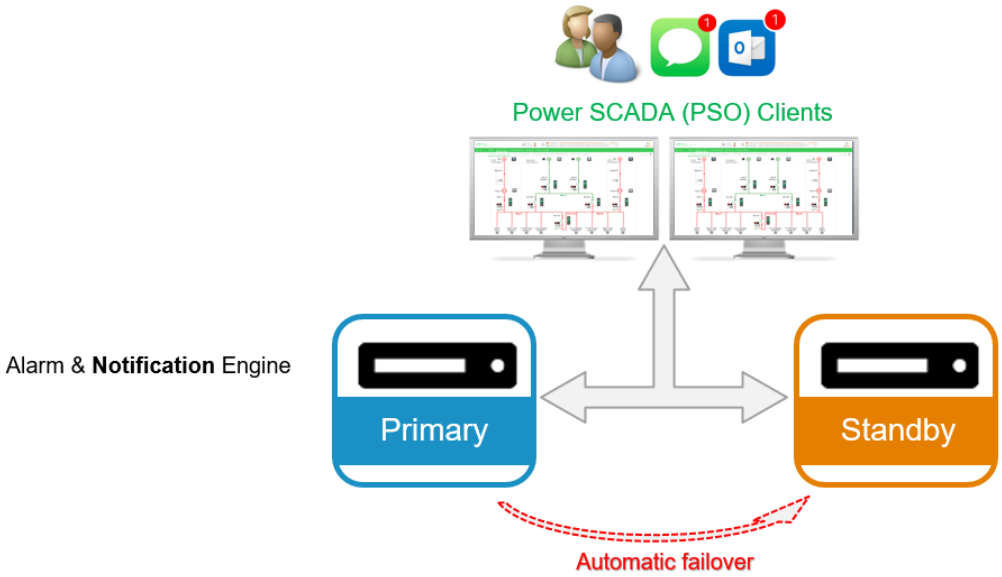
Notifications Settings accepts alarms for the Power SCADA Operation Alarm Server. Notifications Settings alerts specified recipients based on the configured notification.

Subsequent topics explain how to configure and maintain your system notifications.

Notifications Settings Architecture

Notifications are invoked by the Power SCADA Operation Alarm Server.

In redundant systems, configuration settings are automatically synchronized between the primary and standby alarm servers. Only the active server sends the notification.



Prerequisites

Before you can use Notifications Settings, verify the following:

- You have a Power SCADA Server license as well as an Event Notification Module license.
- The device alarms are configured
- The Alarm Server process is running
- (On redundant Power SCADA systems) The standby Alarm Server is running
- Users have the correct privilege level to open Notifications Settings
- For Notifications Settings reports: A program that can open and view CSV files
- The Power SCADA project must be compiled and running.

NOTE: You must enable 64-bit processes to run on the alarm servers . To do this: In Power SCADA Studio, click **Topology > Edit > Alarm Servers**. For each Alarm Server you want to include (primary, or primary and standby), in the **Extended Memory** column, enter `TRUE`.

Before migrating notifications from Event Notification Module (ENM), see "Migrating notifications" on page 327.

Licensing

In Power SCADA Operation 9.0 the Notifications Settings service runs by default, however, sending out notifications requires a Power SCADA Server license as well as an Event Notification Module license. Without these licenses you can still open Notifications Settings to create, test, and save notification configurations. However, the notifications will not be sent.

NOTE: To verify that Notifications Settings is licensed, click **Settings and Diagnostics**. The server license status is indicated on the **General** tab.

Migrating notifications

You can migrate notifications from Event Notification Module (ENM).

Prerequisites

Before migrating your notifications from ENM, ensure the following:

- The ENM database on SQL Server is running and accessible (you can connect to it)
- Your system is using the latest version of ENM (8.3.x).

  NOTE: If you do not have ENM version 8.3.x, you will need to update it before you can migrate your existing system's notifications.

Migrate notifications from ENM

1. In Notifications Settings, click **Settings and Diagnostics**, and then click **Migration**.
2. Click **Migrate from ENM**.
3. Connect to the ENM database using one of the following methods:
   - Enter the ENM SQL instance and database information.
   - Enable **Integrated Security** and then enter your user name and password.

4. Click **Test Connection** to verify that you entered the correct database information.
5. Click **Start**.

   Depending on the number of alarm notifications in ENM, the migration process may take several minutes to complete.

   NOTE: After the ENM alarm notifications are migrated, they are not committed to Notifications Settings until you click **Save** (step 7).
6. (Optional) Create notifications reports and then compare the report outputs to your ENM system to determine whether all of your alarm notifications were successfully migrated.
7. Click **Save** to commit the migrated alarm notifications.

After the ENM migration completes successfully, consider decommissioning ENM.

NOTICE
POTENTIAL COMPROMISE OF SYSTEM CONFIDENTIALITY • Use cyber security best practices to securely uninstall or decommission Event Notification Module (ENM). **Failure to follow these instructions can result in unauthorized or unintended access to sensitive or secure customer data.**

Malicious attackers can potentially gain access to and exploit old or unsupported versions of the software and databases. After migrating alarm notifications from ENM, restrict access to the old ENM system and uninstall the software.

Software decommissioning cyber security and auditing policies vary from site to site. Work with the facility IT System Administrator to ensure that the software decommissioning adheres to the site-specific cyber security and auditing policies.

Configuring notifications

Before your system can send out notifications, you must configure the email server and the modem COM port to send SMS messages.

Configure the Email Server

To send notifications using email, you must configure the email server.

To configure the email server:

1. "Open Notifications Settings" on page 333
2. Click **Settings and Diagnostics**, and then click **Email Setup**.
3. Enter the email server settings.

   Refer to the following table for a description of the email server values:

Email Server Setting	Description
SMTP Server	The server name or IP address of the provider.
From Address	Appears in the “From” field of the sent email.
User Name	Login for the SMTP Server, if required.
Password	Password for the SMTP Server, if required.
Enable SSL	Indicates whether the email is sent using Secure Sockets.
Service Port	The port number on the SMTP host. The default value is 25.
Timeout	The duration (in seconds) to wait before not sending an email.
Retries	The number of unsuccessful send attempts are made before the email is not sent.
Backoff	The delay (in seconds) between retries.

Configure SMS Text Notification

Short Message Service (SMS) sends a notification as a text message when an alarm occurs in a configured notification, or when you click the SMS Notifications **Test** button.

Prerequisites

Before you can configure SMS text notification, you need:

- A modem that accepts a standard SIM card and connects to the computer via USB cable (the connection is a serial connection). Compatible modems include: MultiTech MTD-H5, or MultiTech MTC-H5-B03.

  Moxa OnCell G3111 is not supported.
- A SIM card from a carrier that allows you to send automated messages and large numbers of text messages at one time.

  NOTE: Certain carriers restrict how you can use their services.
- The modem COM port. To determine the modem COM port:
  a. Open Windows Device Manager.
  b. Expand **Ports (COM & LPT)**.

     The port is listed beside the modem in brackets.
  c. Take note of the COM port value. You will need to enter this value in Notifications Settings.

To configure SMS text notifications:

1. "Open Notifications Settings" on page 333.
2. Click **Settings and Diagnostics** and then click **SMS Setup**.
3. In COM Port, enter the modem COM port.
4. (Optional) Set the other SMS values.

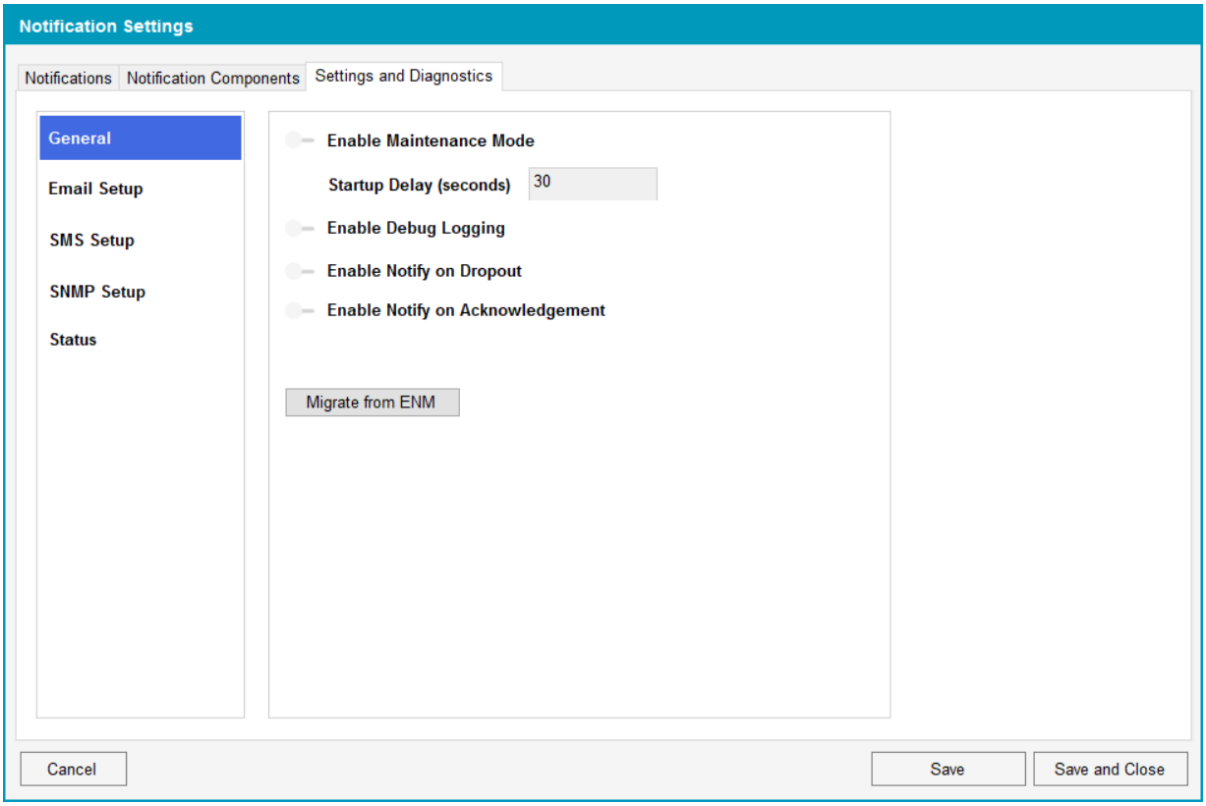
   Refer to the following table for a description of the SMS values:

SMS Setting	Description

COM Port	The modem COM port. The default value is COM1.
Timeout	The duration (in seconds) to wait before not sending an SMS.
Retries	The number of unsuccessful send attempts are made before the SMS is not sent.
Backoff	The delay (in seconds) between retries.
Max SMS Length *	The maximum number of message characters. **NOTE**: Mobile carriers impose limits on the length of text messages that—if exceeded—could possibly result in messages not being delivered. Determine your mobile carrier's limit and enter the value here.

Notifications Settings

There are a number of global settings you can apply to notifications. Unlike alarm rules that apply to a specific notification, global settings control how all system notifications behave.



You can define the following settings:

- **Enable Maintenance Mode** – Disables notifications. See "Using Maintenance Mode" on page 331 for more information.
- **Startup Delay (seconds)** – Disable nuisance start up notifications for a defined period of time.
- **Enable Debug Logging** – Enables logging. See Notifications FAQs for more information on logging.
- **Enable Notify on Dropout** – Sends a message when the alarm is back to normal.

- **Enable Notify on Acknowledgment** – Sends a message when the alarm has been acknowledged.

After you change settings, click **Save**. For redundant systems: In Save Configuration, select the servers to which you want to apply the settings.

Using Maintenance Mode

Maintenance mode lets you configure and troubleshoot notifications without notification messages being sent. You will not receive notifications from Power SCADA while the Alarm Server remains in maintenance mode.

NOTE: No heartbeat alarms are sent when Maintenance Mode is on.

When you put Notifications Settings in maintenance mode, Power SCADA sends a message indicating that the Alarm Server is in maintenance mode. Power SCADA sends another message when Notifications Settings resumes. You can optionally disable these messages (see step 4 for details.).

1. Click **Settings and Diagnostics**.
2. Click **Maintenance Mode** to enable it and then click **Save**.
3. (For redundant systems) In the Save Configuration window, select the servers that you want to put in maintenance mode.
4. (Optional) In the Save Configuration window, uncheck **Send Configuration Announcements**.

   Typically, you would only uncheck this setting when you are commissioning a live system and you do not want maintenance mode alerts to go out.
5. When you have completed your system updates, click Click **Maintenance Mode** to disable it and then click **Save**.

Creating Notifications

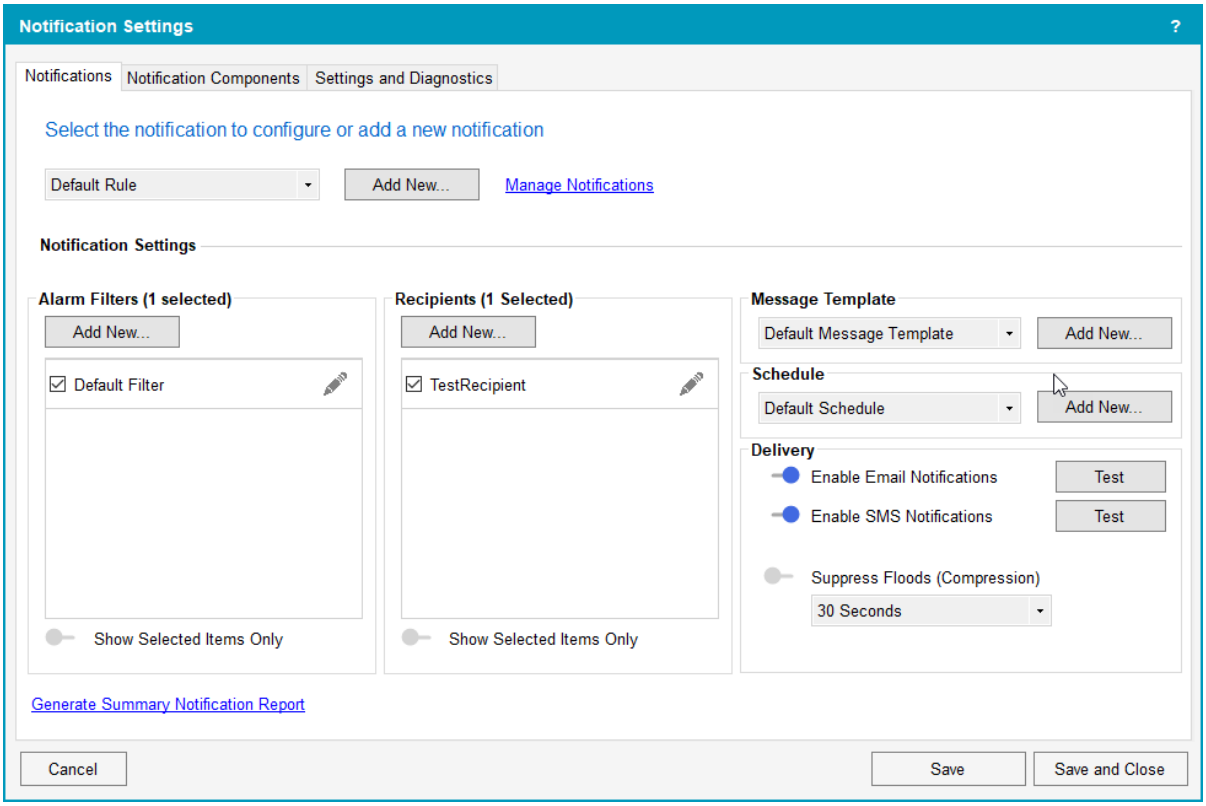
⚠ WARNING

INACCURATE DATA RESULTS

- Do not base your maintenance or service actions solely on messages and information displayed by the software.
- Do not rely solely on software messages and reports to determine if the system is functioning correctly or meeting all applicable standards and requirements.

Failure to follow these instructions can result in death or serious injury.

All of a notification's components are displayed on the **Notifications** pane, letting you quickly see the components that comprise the notification. For example:



Notification Components

A notification consists of the following notification components:

Component	Description
Alarm Filters	What alarms trigger the notification.
Recipients	Who will receive the notification.
Schedules	When the notification will be sent.
Delivery	How the notification message will be delivered (email, SMS)

Managing Notification Components

Design your notifications as much as possible before you create them. A notification can be very complex (consisting of multiple alarm filters, with many recipients and schedules). Understanding how to use notification components—especially how alarm filters work—is key to creating system notifications.

Subsequent topics provide details on how to use Notifications Settings to notify people when a system alarm requires their attention.

Create a notification workflow

Create your system notifications either by editing and duplicating the default notification, or by adding a new ones.

Creating a notification involves the following tasks:

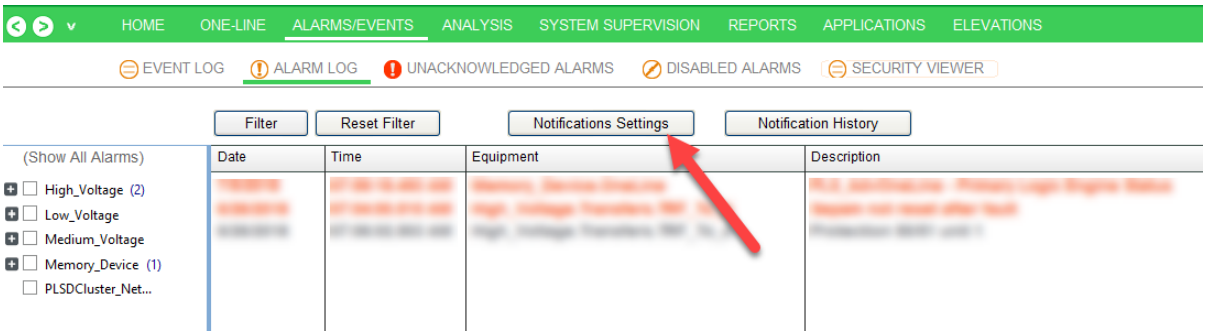
1. Add a new notification or duplicate an existing notification
2. Filter the alarms to be included in the notification.
3. Add recipients to the notification.
4. Define the schedule when recipients can receive the notification.
5. Set the notification relay.
6. Test the notification.

TIP: If the components of a new notification vary only slightly from those of an existing notification, duplicate an existing notification and then edit the copied notification components.

Subsequent topics provide detailed description on how to accomplish these tasks.

Open Notifications Settings

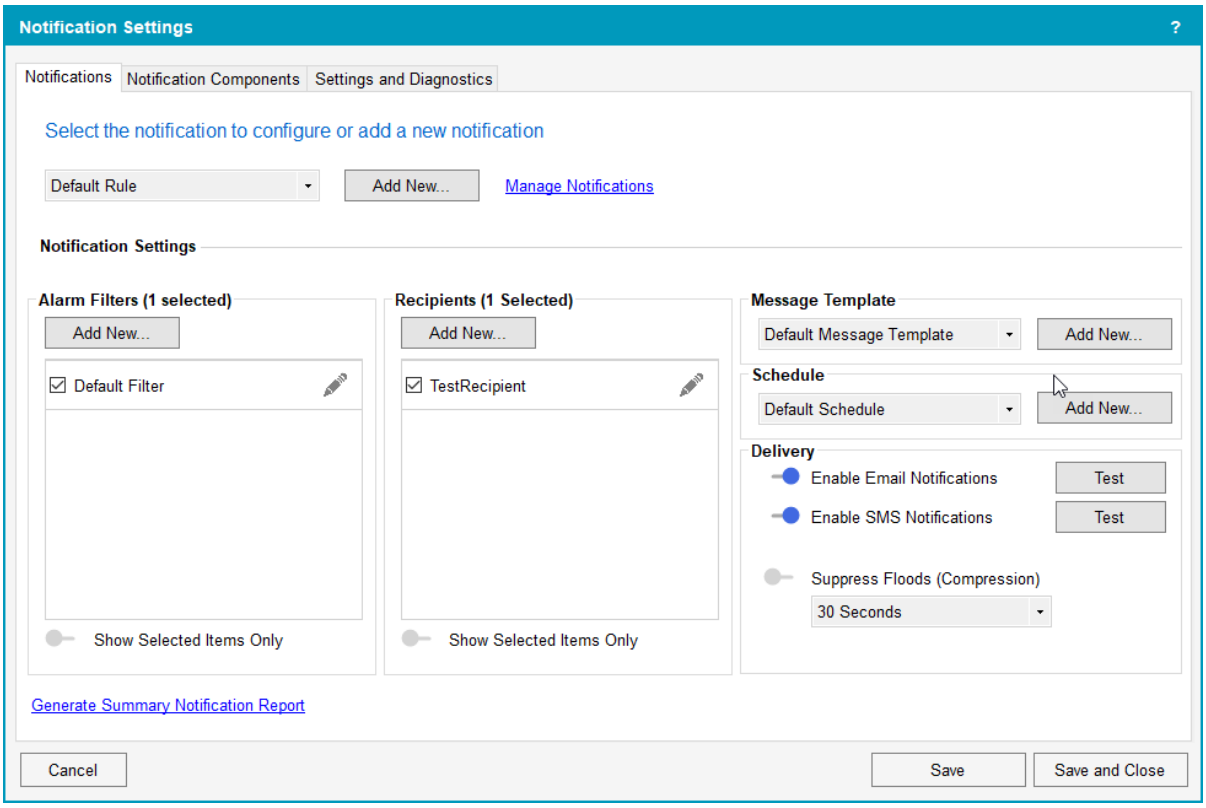
Open Notifications Settings from the Power SCADA Runtime.



NOTE: Notifications Settings can be customized to open anywhere in the Power SCADA Runtime.

In the Power SCADA Operation runtime, click **Alarm Log > Notifications Settings**.

Notifications Settings Settings appears:



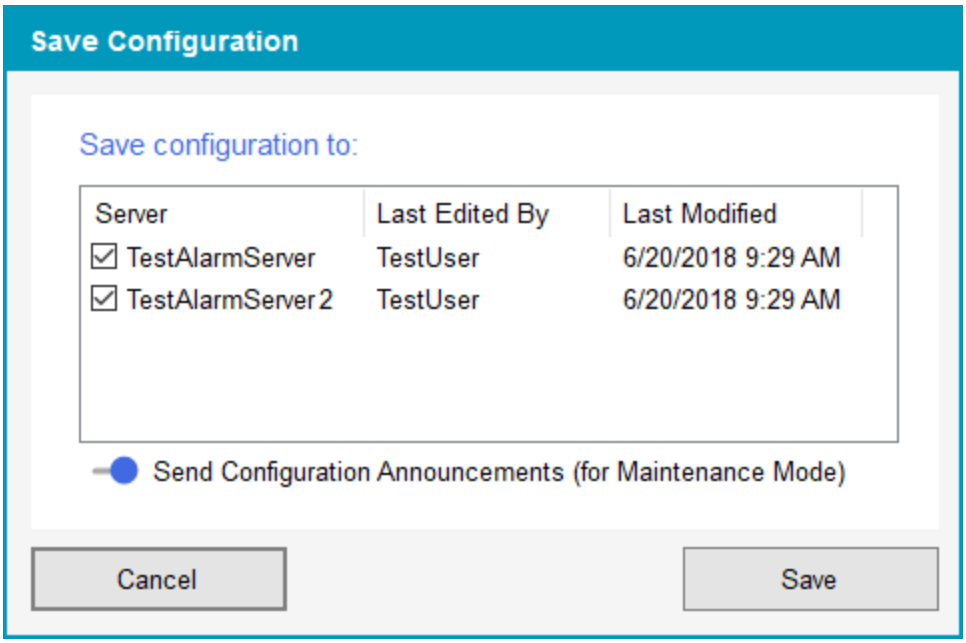
Assuming notifications from Event Notification Module (ENM) were not migrated, when you first open Notifications Settings, a notification is included by default. This default notification includes an alarm filter that includes all alarms in the system, a fictitious recipient, a message template, and a schedule.

Create your system notifications either by editing and duplicating the default notification, or adding a new ones.

Notifications in a redundant system

Whenever you change a notification or a notification component and then click **Save** or **Save and Close**, you will be prompted to save your changes to an Alarm Server.

For example:



Click the server or servers to which you want to apply the changes and then click **Save**.

Create a notification

A notification is a set of rules that determine when someone should be notified about an alarm.

To create a notification:

1. In the **Notifications** pane, click **Add New**.
2. Enter a notification name then click **OK**.

   The newly- added notification appears in Notifications Settings and a default alarm filter called Default Rule is added to the notification.
3. Define the notification components by completing the following tasks:
   a. Create alarm filters.
   b. Add recipients.
   c. Add schedules.
   d. Test the notification.
   e. Save the notification.

Subsequent topics discuss how to define notification components.

About Alarm Filters

A notification can consist of one or more alarm filters. An *alarm filter* is a set of alarm tags that trigger a notification. You create alarm filters by adding rules, lists, and exclusions that—taken together—define the filter.

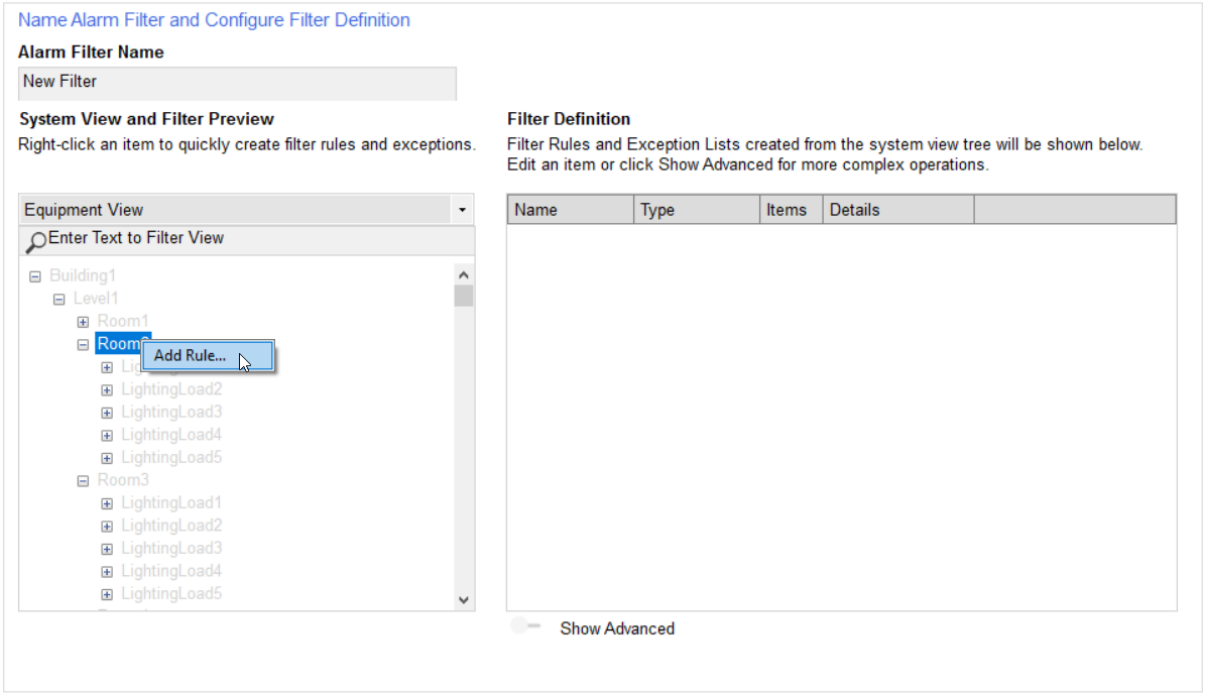
Rules

A *rule* adds all the tags to the filter definition. You can apply a rule to a system node or a tag.

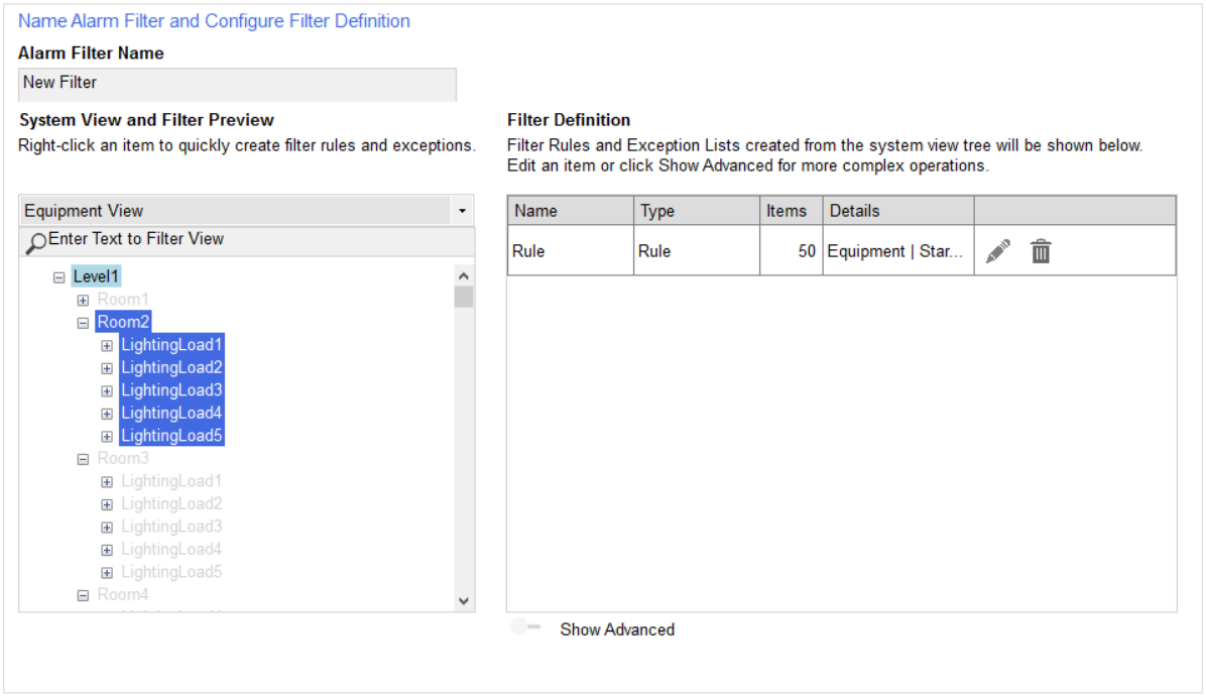
Rules and nodes

When you add a rule to a system node, all the tags belonging to that node and all the tags belonging to any child nodes are added to the filter definition.

For example, when you add a rule for a room that contains 5 lighting loads (with 10 tags each), all of the tags in the room nodes are added to the rule:

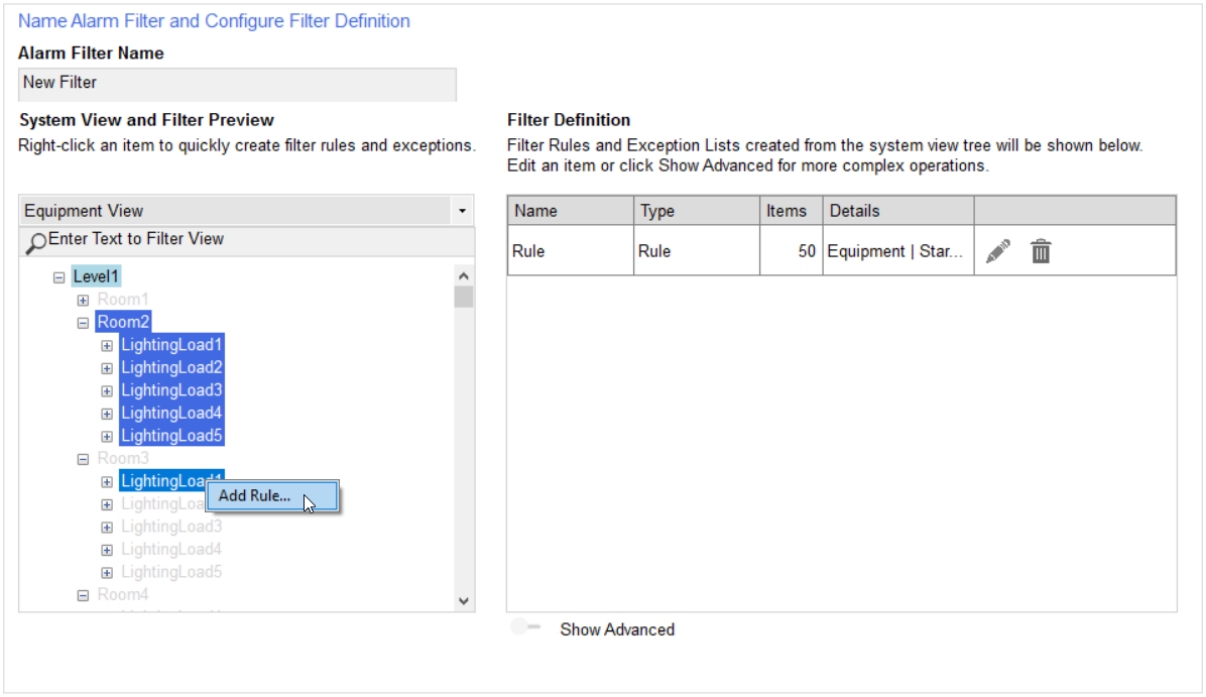


Result: All the tags in the node and child nodes are added to the filter definition.

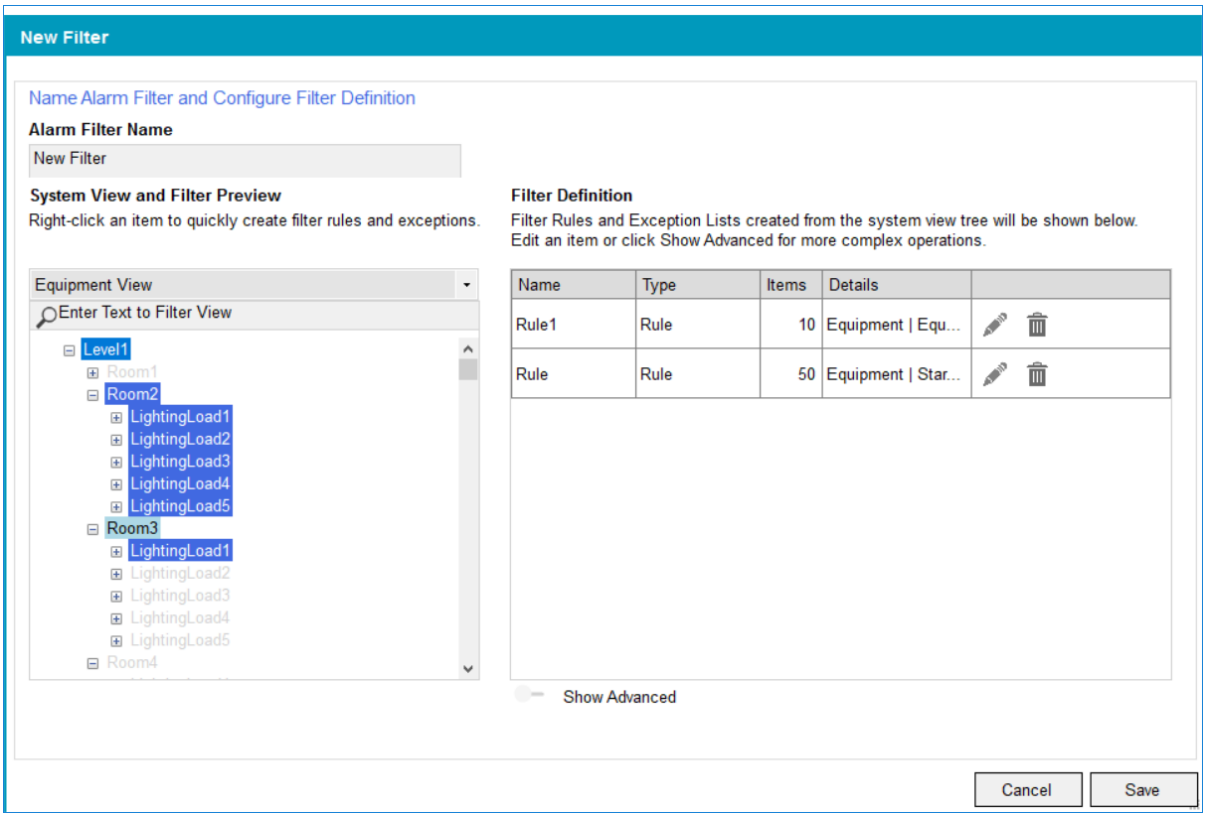


TIP: Notice the shading in the **Filter Preview**: Room2 and all its child nodes are highlighted in blue because all of their tags are part of the filter definition. Level1 is highlighted in light blue to indicate that some of its child node tags have been added to the filter definition.

You can also add more tags to the filter definition. In the following example, the 10 tags from Room3 > LightingLoad1 are added as a rule to the filter definition:



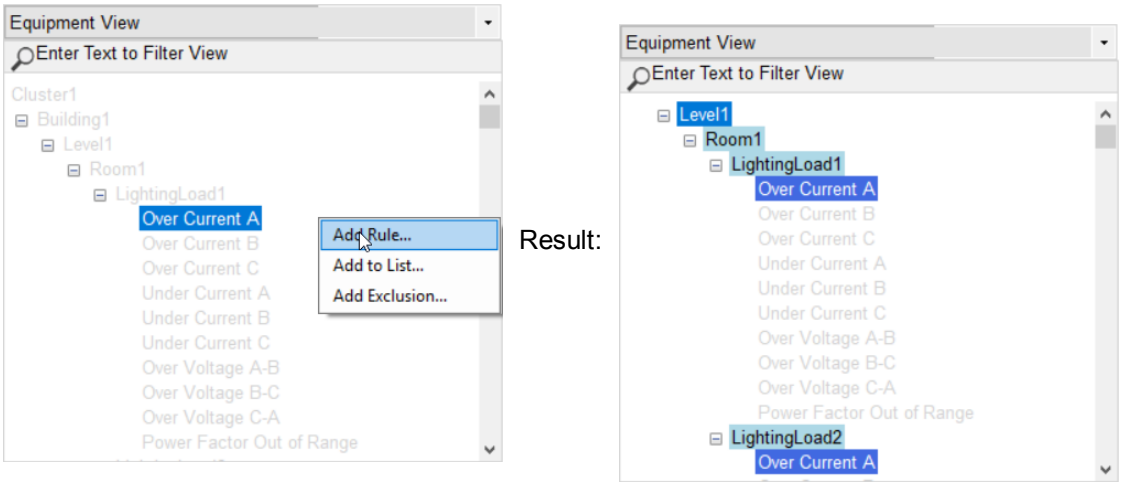
Result: All the tags in LightingLoad1 are added to the filter definition.



TIP: Notice the shading in the **Filter Preview**: Room2 and all its child nodes are highlighted in blue because all of their tags are part of the filter definition. Room3 is highlighted in light blue to indicate that some of its child node tags have been added to the filter definition.

Rules and tags

You can also add a rule to an individual tag. When you do this, all tags of that type are added to the filter definition. For example:

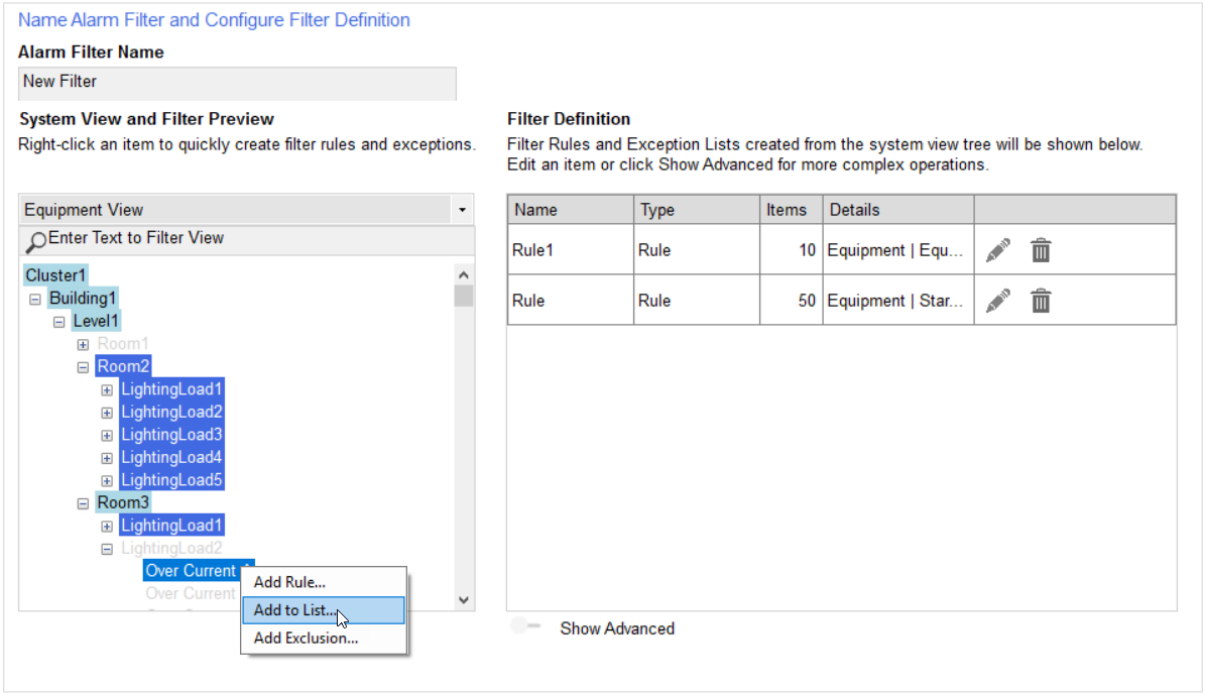


Lists

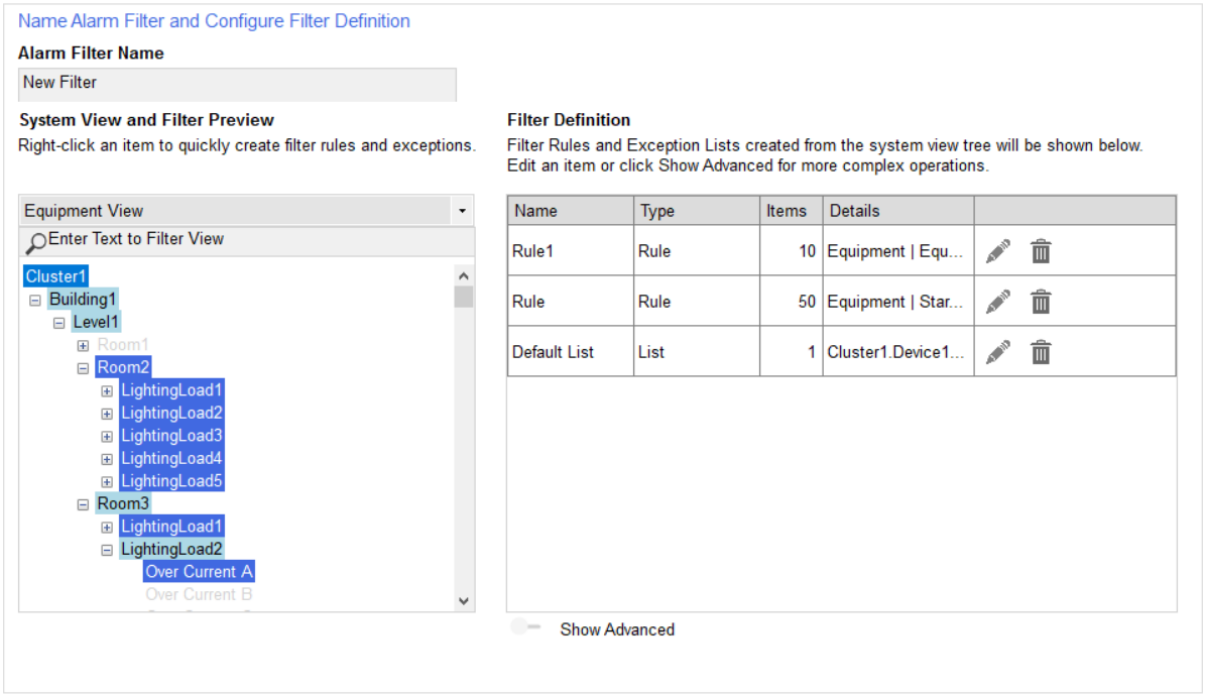
Use *list* to add specific tags one at a time to a filter definition.

NOTE: Use lists very carefully. Unlike rules, when you add a list to an alarm definition, if the tag name changes the notification will not automatically update. Instead, you must edit the alarm filter to include the re-named tag. If not, your system will not send out a notification if the old tag name triggers an alarm.

In the following example, a tag to the filter definition:



Result: The tag is added to the filter definition.



Exclusions

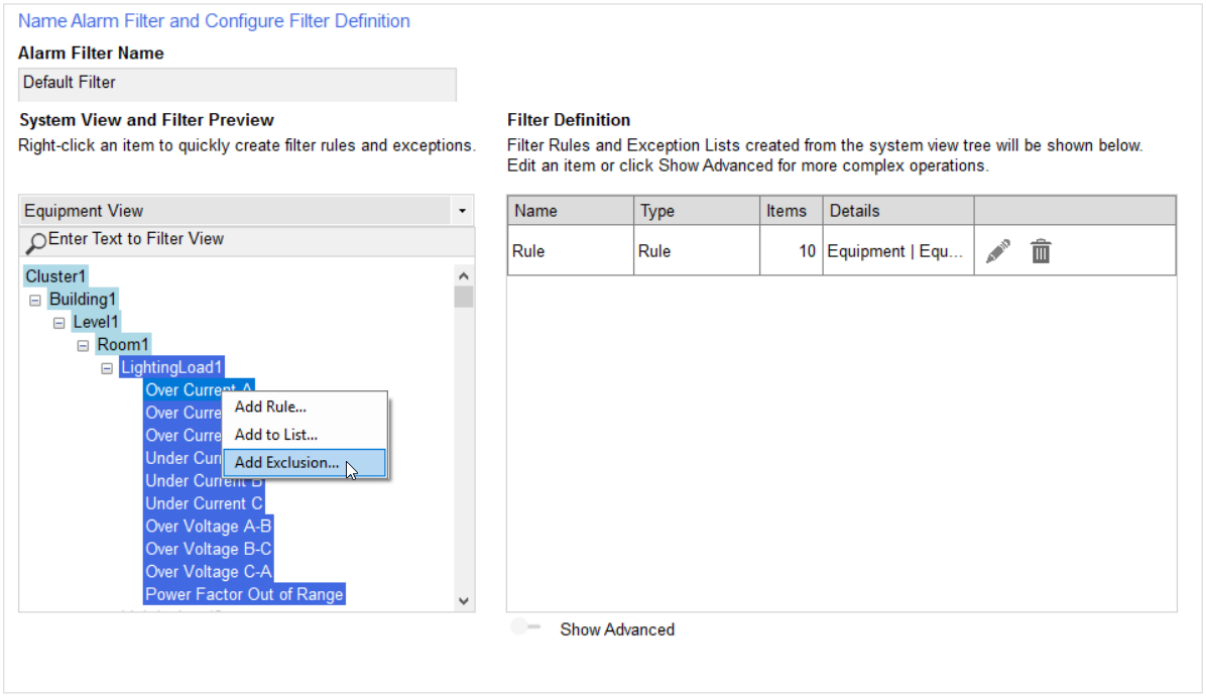
Use *exclusion* to exclude specific tags one at a time to a filter definition.

NOTE: Use exclusions very carefully. Unlike rules, when you add an exclusion to an alarm definition, if the tag name changes the notification will not automatically update. Instead, you must edit the alarm filter to include the re-named tag. If not, your system will not send out a notification if

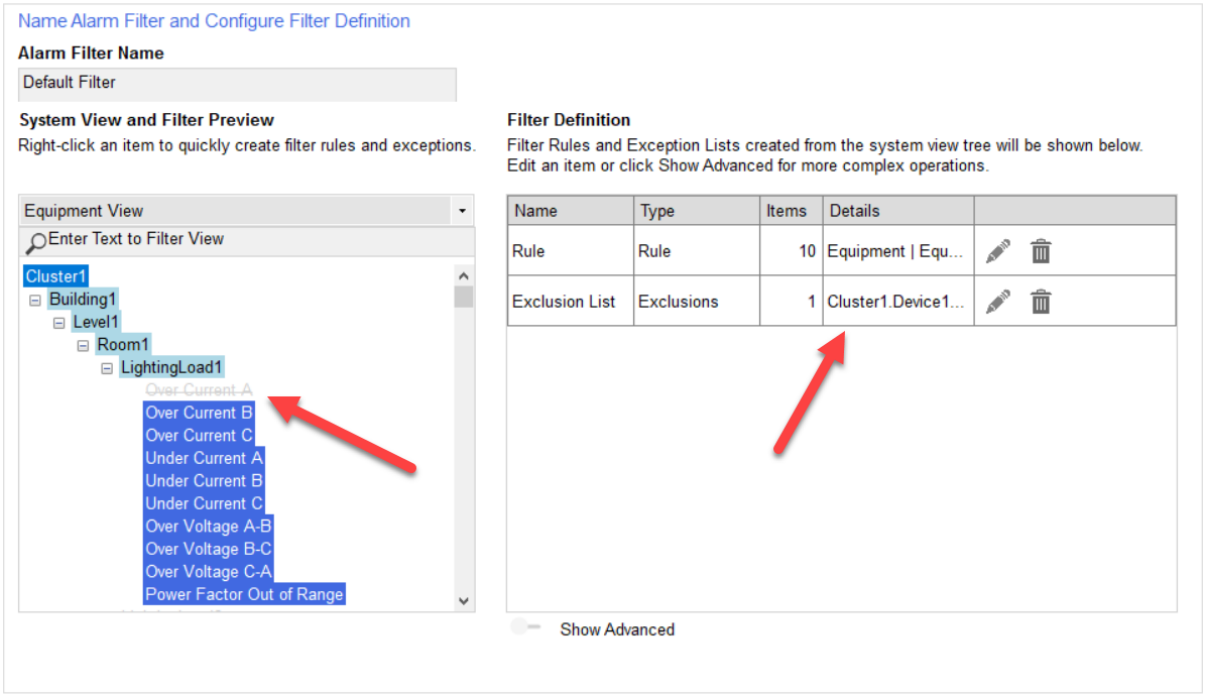
the old tag name triggers an alarm.

If an alarm filter contains an exclusion that is met, the notification will not be sent. Consider creating one alarm filter that includes all exclusion lists.

In the following example, a tag is removed from the filer definition.



Result: The tag is removed from the rule.



TIP: Notice that the tag is no longer highlighted; instead it appears with strikethrough text in the preview list. Also, the excluded tag appears in the Filter Definition.

Create basic alarm filters in the New Filter or Edit Filter window. See "Create Basic Alarm Filters" on page 341 for more information. Create advanced alarm filters using the dedicated rule, list and exclusion filter windows. See "Show advanced alarm filter settings" on page 343 for more information.

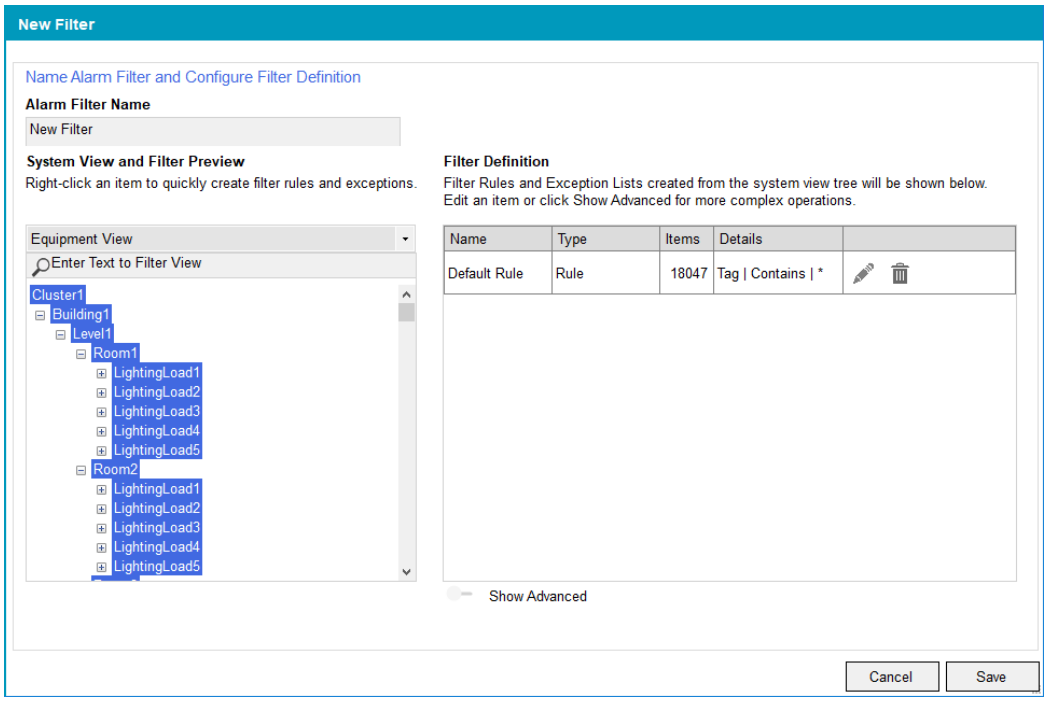
Create Basic Alarm Filters

An *alarm filter* is a set of criteria that filters the alarms to include or exclude in a notification. An alarm filter is comprised of one or more alarm rules, lists, and exclusions.

NOTE: Before creating alarm filters, you should have a good understand of alarm filter rules, lists, and exclusions. See "Rules and nodes" on page 336 for details. Also note the following:
– If an alarm filter contains an exclusion that is met, the notification will not be sent. Therefore, use exclusions with care.
– Thoroughly test your alarm notifications before deploying them on a live system.

To create a basic alarm filter:

1. Open the New Filter window using one of the following methods:
   - In the **Alarms Filters** section of the **Notifications Settings** pane, click **Add New**.
   - In the **Notification Components** pane, click **Alarm Filters** and then click **Add New**.



   By default, the new alarm filter has a default rule that includes all alarm tags. You can build the filter definition by editing the Default Rule or deleting it and then adding new filters.

2. In **Alarm Filter Name**, enter a unique alarm filter name.

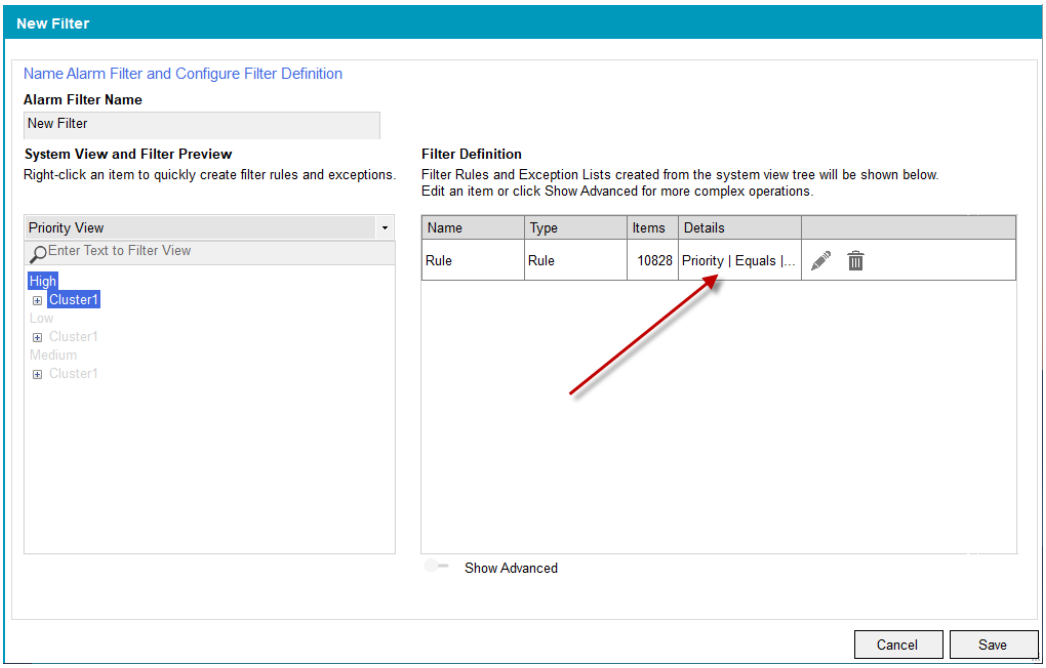
3. Under **System View and Filter Preview**:
   a. Select the system view that sorts the alarms for your needs.

      For example, if you want to include and exclude equipment, use the Equipment View. If you want to create an alarm filter for high priority alarms only, select Priority View. For more information on system views, see .

   b. Navigate to the level of alarm you want to use by expanding or collapsing the alarm nodes.

   c. Right-click the node you want to filter on and then click **Add Rule**.

      The alarm rule is added to the alarm filter. The Alarm Filter section displays the rule name, type, items and details. For example:



NOTE: The steps for adding a list and exclusion is the same as that for rules. However, you can only add a list or an exclusion to tags.

4. (Optional) Repeat step 3 to add more alarm rules to the alarm filter definition.
5. When you are finished adding alarm rules, click **Save**.

For detailed information on creating advanced alarm filters, see "Show advanced alarm filter settings" on page 343.

Create Advanced Alarm Filters

You can use **Notifications Settings** to create advanced alarm filters.

NOTE: When using advanced criteria, the multiple criteria are logically AND'd together, meaning that all criteria have to be satisfied for an alarm to ultimately be selected into the rule.

An advanced alarm filter consists of custom criteria you define to customize the alarm filter definition. You can filter alarms using the same objects that are available in basic filters. However, you can also define alarm filters using the search terms **contains**, **equals**, and **starts with** to further fine tune the alarm filter definition.

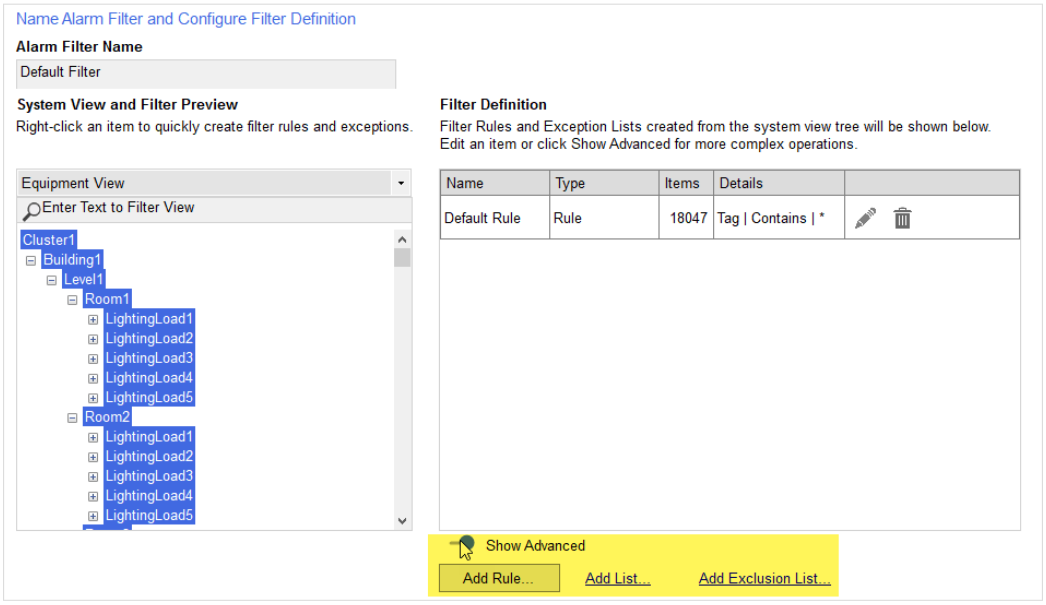
NOTE: Before creating advanced alarm filters, you should have a good understand of alarm filter rules, lists, and exclusions. See "Rules and nodes" on page 336 for details. Also note the following:
– If an alarm filter contains an exclusion that is met, the notification will not be sent. Therefore, use exclusions with care.
– Thoroughly test your alarm notifications before deploying them on a live system.

Show advanced alarm filter settings

To view the advanced alarm filter settings:

1. Open the New Filter window using one of the following methods:
   - In the **Alarms Filters** section of the **Notifications Settings** pane, click **Add New**.
   - In the **Notification Components** pane, click **Alarm Filters** and then click **Add New**.
2. At the bottom of the Filter Definition pane, click **Show Advanced**.

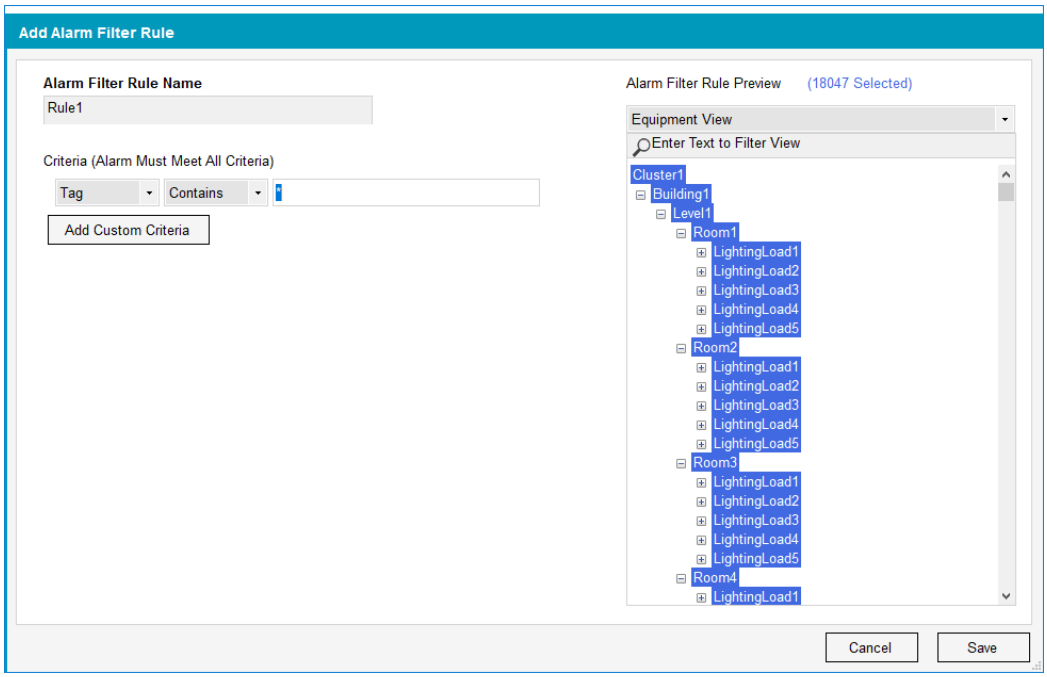
   The **Add Rule**, **Add List**, and **Add Exclusion List** items appear.



Add a custom rule

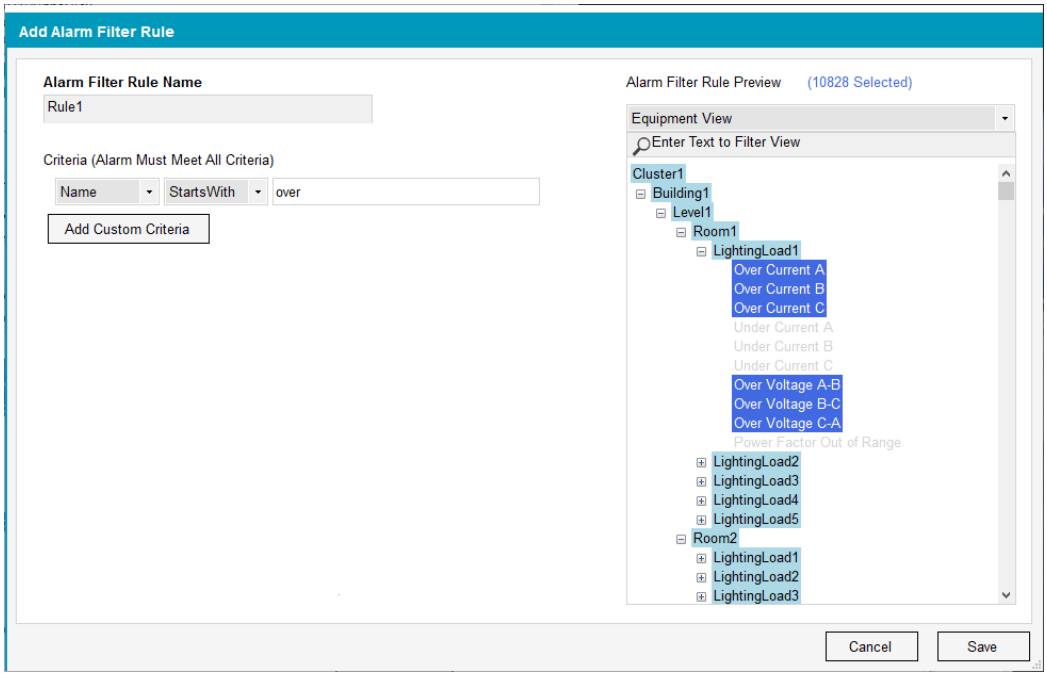
1. Click **Add Rule**.

   The Add Alarm Filter Rule window appears.



2. Enter an alarm filter rule name.
3. From the first drop down, select an object type. For example, Name.
4. From the second drop down, select a search condition. For example: StartsWith.
5. Enter the text you want to include. For example: over

NOTE: You can only also use * (wildcard) alone; it cannot be used with other text.



All the tag names that begin with 'over' are included in the custom filter:

6. (Optional) Click **Add Custom Criteria** to add another rule. You can add up to 10 criteria per rule.

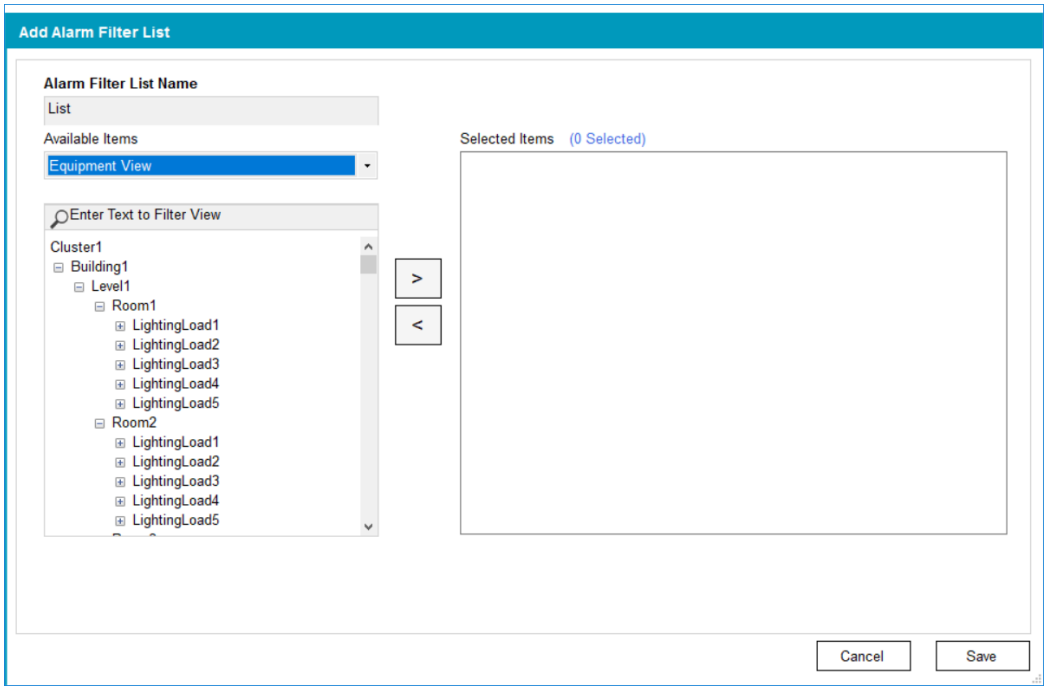
7. When you are finished adding custom criteria, click **Save**.

Add a custom list or exclusion list

NOTE: The procedure for adding lists and exclusion lists is the same. This following procedure adds an alarm filter list.

1. Click **Add List**.

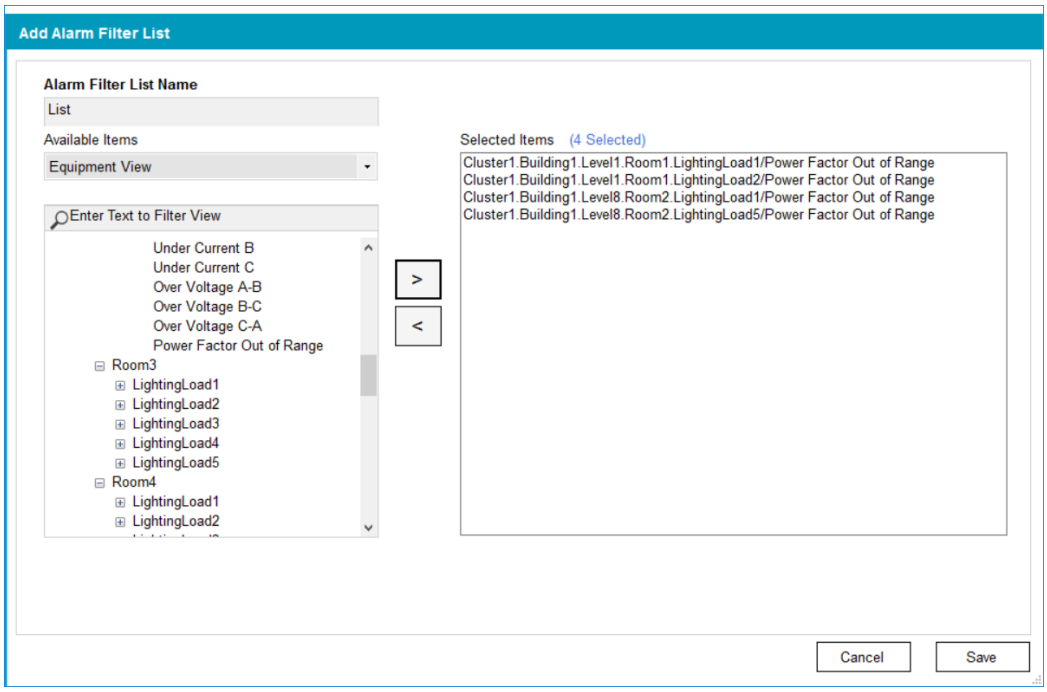
   The Add Alarm Filter List window appears.



2. Enter an alarm filter list name.
3. From the **Available Items** drop down, select the view you want to use.

   NOTE: You can only add tags to lists and exclusion lists.

4. Navigate to and then select the tag you want to add to the list, and then click the add arrow [>] .
5. (Optional) Remove tags form the list by clicking the remove arrow [<].
6. Repeat step 4 to add additional tags.



7. When you are finished adding tags, click **Save**.

Adding alarm filters to a notification

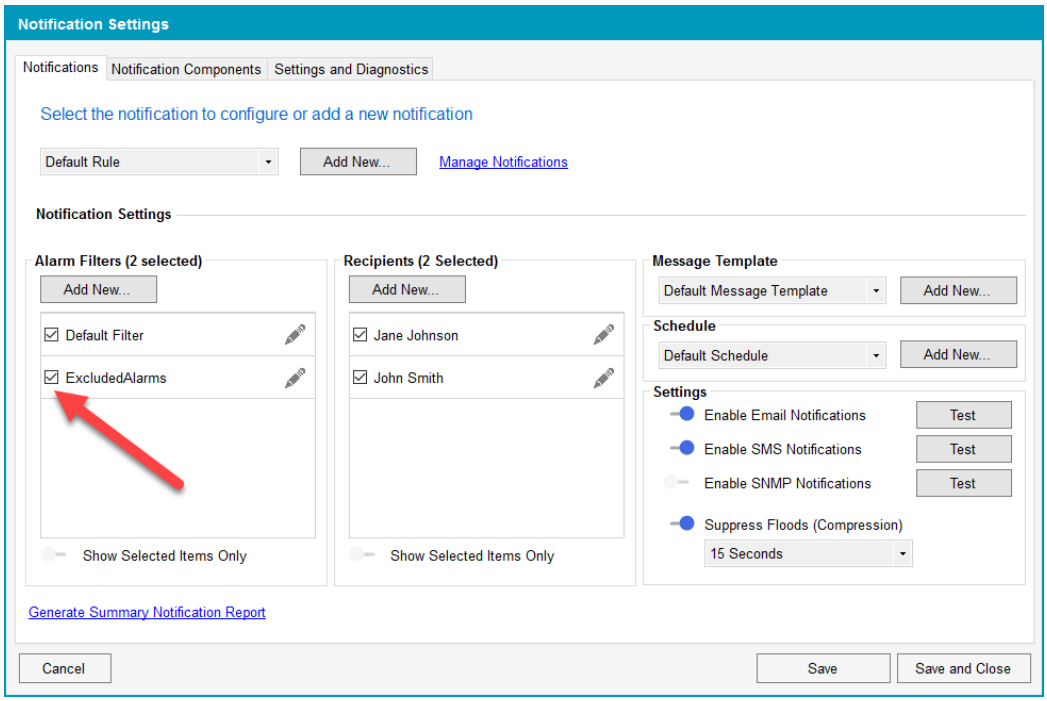
After you create an alarm filter, you need to add it to the notification.

To add an alarm filter to a notification:

1. Click the **Notifications** tab.
2. In the **Alarm Filers** section, click the alarm filters you want to include in the notification.

   TIP: You can uncheck any alarm filters you want to temporarily or permanently exclude from the notification. Doing so lets you update the notification without having to disable all system notifications using Maintenance Mode.

   For example:



3. Click **Save** or **Save and Close**.
4. (For redundant systems) In Save Configuration, select the servers to which you want to apply the settings, and then click **Save**.

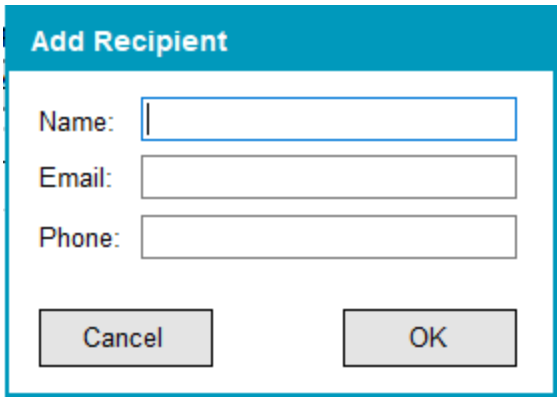
Add Recipients to a notification

A *recipient* is the person will receive the notification. To be notified of an alarm, at least 1 recipient must be added to a notification.

For more information on recipients, see "Manage Recipients" on page 348.

To add a recipient to a notification:

1. In the **Recipients** section of the **Notifications** pane, click the person you want to add as a recipient.
2. (Optional) If a recipient is not listed in the **Recipients** section:
   a. Click **Add New** and then add the recipient to the system.



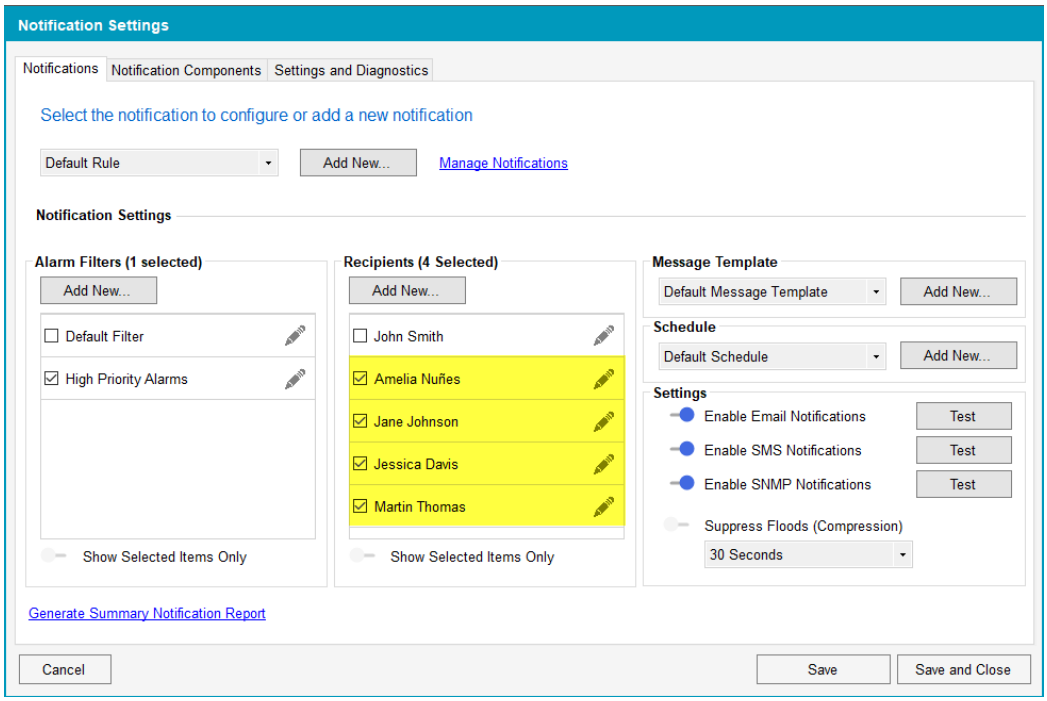
   b. Enter the recipient details.

> **NOTE**: For **Phone**, enter numbers only. Do not enter parentheses or hyphens.

c. Click **OK**.

3. In the **Notifications** tab, click the recipient you just added to include them in the notification.

   Note that in the following example, John Smith will not be notified when a high priority alarm occurs:



4. Click **Save**

5. (For redundant systems) In Save Configuration, select the servers to which you want to apply the settings, and then click **Save**.

Manage Recipients

From the **Notification Components** tab, you can add, edit, and delete recipients.

Add a Recipient

1. Click the **Notification Components** tab and then click **Recipients**.
2. Click **Add New**
3. Enter the recipient details.

   NOTE: For **Phone**, enter numbers only. Do not enter parentheses or hyphens.

4. Click **OK**.

   The recipient appears in the recipient list and can be assigned to a notification in the **Recipients** section of the **Notification** pane.

Edit a Recipient

1. From the recipient list, click **Edit** .
2. Edit the recipient details and then click **Save**.

Delete a Recipient

1. From the recipient list, click **Delete** and then confirm the deletion.

Set Schedules

A *schedule* is the defined time period when a notification is sent. For a notification to be received, a notification must include at least one schedule.

For information on schedules, see "Manage Schedules" on page 349.

Manage Schedules

Add, edit, and delete schedules in the **Notification Components** pane.

NOTE: For **Phone**, enter numbers only. Do not enter parentheses or hyphens.

Add a Schedule

1. Click the **Notification Components** tab and then click **Recipients**.
2. Click **Add New**.
3. Enter the recipient details.

   NOTE: For **Phone**, enter numbers only. Do not enter parentheses or hyphens.
4. Click **OK**.
5. Click **Save**

   The schedule appears in the schedule list and can be assigned to a notification in the **Schedule** drop down list of the **Notification** pane.

Edit a Schedule

1. From the schedule list, click (**Edit**).
2. Edit the schedule details and then click **OK**.

Delete a Schedule

1. From the schedule list, click (**Delete**) and then confirm the deletion.

NOTE: You cannot delete the Default Schedule.

About Message Templates

A *message template* is the message the recipient will receive that includes information about the notification. A notification must have an associated message template.

Notifications Settings includes three default email and SMS templates that you can associate with a notification:

- **Single Notification** – The message that is sent with a single notification.
- **Flood Start** – The message that is sent at the beginning of a flood of alarms. Typically, this message includes information that subsequent notifications containing more alarms will arrive.
- **Flood End** – The message that is sent at the end of a flood period. Typically, this message includes how many alarms occurred during the flood suppression period.

NOTE: Email and SMS message size and frequency are governed by carriers. If you are not sure about carrier limitations or restrictions, do not create message templates that include a lot of information.

The default templates were designed to include basic alarm information. You can use the default templates, edit the default templates, or create your own template messages.

Add a Message Template

TIP: Review the default message templates; they provide good direction on what type of information you should include in your messages.

To add a message template:

1. Click **Notifications Components**.
2. Click **Templates**.
3. Click **Add New** to create a new message template, or click  to edit the default message template.
4. Click **Email** or **SMS** to select a relay method for the message template.
5. Click the message type you want to create: **Single Notification**, **Flood Start**, or **Flood End**.
6. In the text entry fields, enter the information you want to include in the message:
   a. Type any custom information you want to include.
   b. Right-click anywhere in the text entry fields and then click **Insert > *system value*** to add system values.
7. Review the **Preview** section to see an example of your message.
8. Click **Save**.
9. (Optional) Repeat these steps for other message templates you want to create.

Manage Message Templates

Rename a Message Template

1. In the **Notifications Components** pane, click **Templates**.
2. For the message template that you want to rename, click **Edit**  .
3. In the Edit Message Template window, edit the message and then click **Save**.

Delete a Message Template

1. In the **Notifications Components** pane, click **Templates**.
2. For the message template that you want to delete, click **Delete** .
3. Click **Yes** to confirm that you want to delete the message template.

Enable and Test Delivery

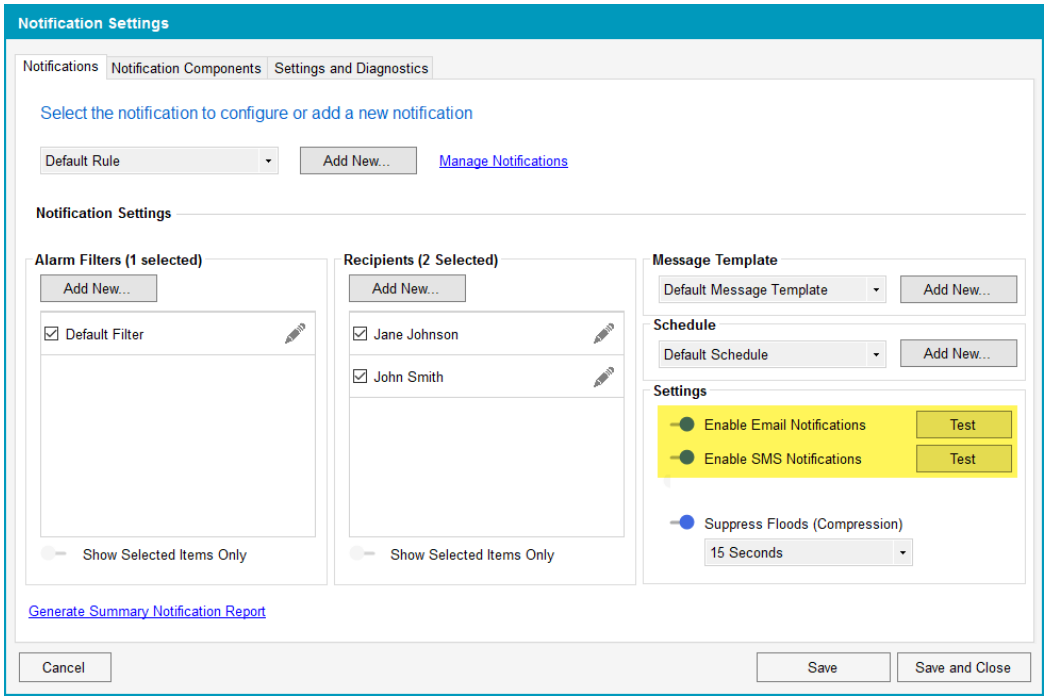
After you have configured all the notification components, choose the delivery methods that Notifications Settings will use to notify people if an alarm occurs.

Prerequisites

- Email and SMS setup is complete
- Email and SMS templates are defined
- The notification has at least 1 alarm filter
- The notification has at least 1 recipient

To enable notification delivery:

1. In the **Delivery** section of the **Notifications** pane, click the delivery methods you want to use to notify people.
2. For each delivery method you enable, click **Test** to make sure the it works as expected.



3. Click **Save**.

Manage notifications

Edit notifications as your facility or system evolves. For example, add or remove recipients as staff change, edit schedules if shifts change, create notifications and alarm rules when tags are added or renamed, or when there is a Power SCADA Server change.

You can put Notifications Settings into Maintenance Mode. Maintenance mode lets you configure and troubleshoot notifications without notification messages being sent. See "Using Maintenance Mode" on page 331 for more information.

After you edit a notification, save your changes.

TIP: If your notification includes a lot of alarm filters and recipients, click **Show Selected Items Only** to view only the included notification components.

Rename a Notification

1. In the **Notifications** pane, click **Manage Notifications**.
2. In Manage Notifications, click **Edit Name**.
3. Edit the name then click **OK**.

Duplicate a Notification

You can quickly create a new notification by duplicating and renaming an existing one, and then modifying it to meet your needs.

1. In the **Notifications** pane, click **Manage Notifications**.
2. In Manage Notifications, click **Duplicate**.

   The newly-duplicated notification is added to the list of notifications.
3. Rename the notification and then click **OK**.
4. From the Notification drop down list, select the notification you duplicated and renamed and then edit it to meet your needs.

Delete a Notification

1. In the **Notifications** pane, click **Manage Notifications**.
2. In Manage Notifications, click  .
3. Click **Yes** to confirm the deletion.

Suppress Floods

Suppressing floods compresses all the notifications that occur during a defined time period. When you suppress floods, Notifications Settings encapsulates how many times the alarm occurred over the suppression time period into a single message.

Example:

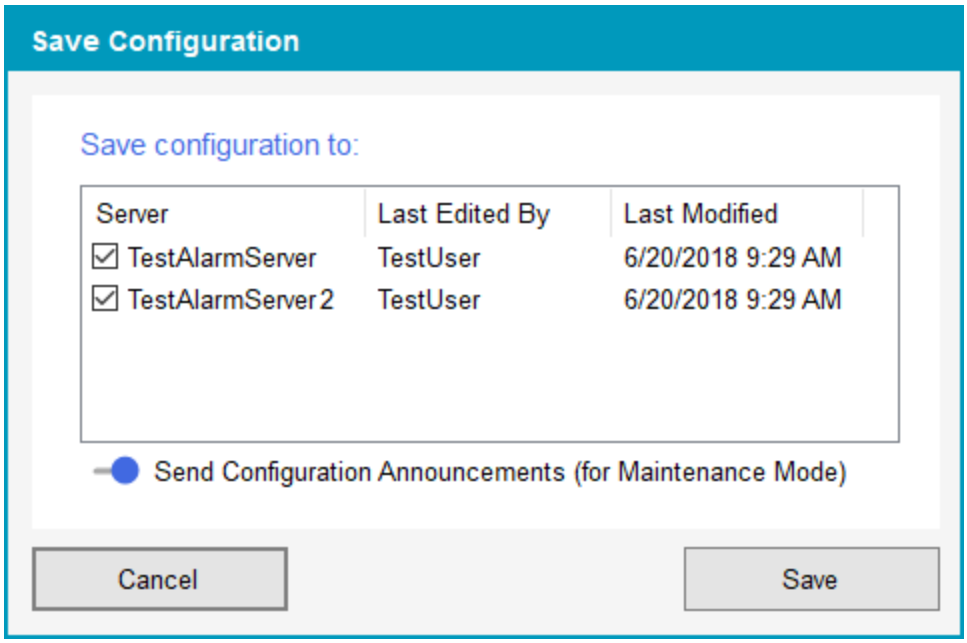
You enable suppress floods and set the time period to 30 seconds. If 500 alarms occur during that time period, Notifications sends out 2 messages:

- The first message notifies you of the alarm.
- The second message notifies you that the alarm occurred 499 times over the 30 second suppression period.

To suppress floods:

1. In the Notifications Settings pane, select the notification that you want to suppress.
2. Enable **Suppress Floods** and then select a time duration.
3. Click **Save** or **Save and Close**.
4. (On redundant systems) Select the servers to which you want to apply the suppression and then click **Save**.

   For example:



Create Summary notification reports

Summary notification reports can help you determine how your system alarms are configured, troubleshoot your notifications, and validate that your notifications migrated successfully from Event Notification Module (ENM).

You can generate the following reports:

- Alarms to Recipient Report – One record for every alarm / recipient pair.
- Alarms to Recipients Report – One record for every alarm / multi-recipient pair
- Alarm to Rule Report – One record for every alarm / rule pair.
- Alarm to Rules Report – One record for every alarm.
- Alarms with No Rule Report – One record for every alarm that is not included in a rule.
- Excluded Alarms Report – One record for every alarm that is excluded.
- Rule Configuration Report – A summary of all configured notifications on the server.

For detailed information on the information contained in each report, see "Notification reports" on page 354.

NOTE: With the exception of the Rule Configuration Report (which is a TXT file), you need a program that can open and view CSV files to view and open reports.

To create a notifications report:

1. On the **Notifications** pane, click **Generate Summary Notification Report**.
2. From the reports list, select the reports that you want to create and then click **OK**.

   The reports you selected are created in the logs folder: C:\ProgramData\Schneider Electric\Power SCADA Operation\v9.0\Logs

   The Notifications Settings report file name include the cluster name, a timestamp, and the report name.

Troubleshooting notifications

This section contains information on how to troubleshoot notifications by using reports and logs.

Notification reports

Notifications Settings includes reports that you can run to see how your system alarms are configured. Use notification reports to help manage and troubleshoot your system notifications, as well as to validate that your notifications migrated successfully from ENM.

The following table lists the information contained in each report:

Report	Notification Information
Alarms to Recipient	Cluster, Equipment, Alarm, Tag, Recipient, Email, SMS, Schedule, Rule, Priority
Alarms to Recipients	Cluster, Equipment, Alarm, Tag, Priority, Recipients
Alarm to Rule	Cluster, Equipment, Alarm, Tag, Rule
Alarm to Rules	Cluster, Equipment, Alarm, Tag, Rules
Alarms with No Rule	Cluster, Equipment, Alarm, Tag
Excluded Alarms	Cluster, Equipment, Alarm, Tag, Rule, Filter
Rule Configuration	For each rule in the system: Rule Name, Email, SMS and Flood Suppression enabled or not, Alarm Filters, Recipients, Message Template, Schedule

Notifications Settings FAQs

How does Notifications Settings logging work during failover?

Notifications Settings logs informational messages (such as start-up messages, activity updates, and warnings) to the log file.

The size of `NotificationLog_<Cluster>_<Server>.txt` is limited to approximately 1000 Kilobytes (K). When the size is exceeded, Notifications Settings messages are logged to new,

empty NotificationLog.txt file, and the existing NotificationLog.txt file is renamed to `NotificationLog_Backup.txt`. If a `NotificationLog_<Cluster>_<Server>_Backup.txt` file already exists, it is replaced by the new one.

If the Notifications Settings log file is not available, (the file is set to read-only, or the file permissions change) the Notification Service continues to run, however, it will not log messages.

Service-related informational logging will also go the Citect Alarm Server kernel window.

SOEEventAdd function alarms

Citect hardware alarms and user events that are created from the SOEEventAdd function will not be notified upon.

Why am I getting duplicate notifications?

If the alarm servers are unable to communicate with each other, they will each assume the Active (or main) state. In the unlikely event that both alarm servers can communicate with the SMTP server, they will both send out notifications.

Assign and control user privileges

You need to give users appropriate levels of access, depending on the work they will do. For safety reasons, only advanced users should be given access to such features as controls and resets.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Do not use the software or devices for critical control or protection applications where human or equipment safety relies on the operation of the control action.
- Do not use the software to control time-critical functions.
- Do not use the software to control remote equipment without proper access control and status feedback.

Failure to follow these instructions can result in death or serious injury.

Because Power SCADA Operation lets you set user permissions on runtime graphical objects, thoroughly test the deployed project to ensure that permissions are applied as intended.

⚠ WARNING

POTENTIAL COMPROMISE OF SYSTEM AVAILABILITY, INTEGRITY, AND CONFIDENTIALITY

- Use cybersecurity best practices when configuring user access.

Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

Cybersecurity policies that govern user accounts and access – such as least privilege and separation of duties – vary from site to site. Work with the facility IT System Administrator to ensure that user access adheres to the site-specific cyber security policies.

For cybersecurity purposes, use Windows Authentication when you create user accounts.

Use Windows Integrated Users

You can incorporate Power SCADA Operation users and security options with the standard Windows security system. Using the integrated Windows security feature, the Windows user can log on to Power SCADA Operation runtime with runtime privileges and areas configured within the project. For a Windows user to be able to log on to runtime, it must be linked to a Power SCADA Operation "role," which is defined in the project with associated privileges.

To link a Windows user to a Power SCADA Operation role, add the "role" that specifies the Windows security group of which the Windows user is a member.

The pre-existing AutoLogin capability is extended to include the client, when the user is a Windows user, having an associated Power SCADA Operation role.

To invoke this functionality for a Windows user, you need to set the `[Client]AutoLoginMode` parameter in the `Citect.ini` file.

Instead of using auto-login when the system starts up, users can also log in to Power SCADA Operation using any Windows user credential that is a member of the linked group.

When the name of a Power SCADA Operation user also has the same name as a Windows user, the Power SCADA Operation user takes priority at runtime. However, if a valid Power SCADA Operation user login fails for some reason, the Windows user credentials will not be checked and an alert will be generated to advise that the login was not effective.

For more information, see Windows Security Usage Scenarios in the Citect SCADA help file (C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin).

Integrate with the Schneider Electric Security Access Module

If the Schneider Electric Security Access Module (SAM) is a part of the customer's solution, any domain/users created in the SAM can be used in Power SCADA Operation in the same manner as described above in Use Windows Integrated Users. To do this, add a role to the Power SCADA Operation project, and use the name of the SAM security group in the role's "Windows Group" field.

Default User Access Settings (Privileges)

The following table describes the access rights. These privileges are included in the `PLSSecurity.ci` file (in the PLS_Include project). Any changes you make to this file will be overwritten when PLS_Include is updated (every new release).

NOTE: Document every change you make, so that you can update `PLSSecurity.ci` when PLS_Include is upgraded.

Access Right			Roles / Global Privileges				
Description	Access Level Label	Access Level Value	Operator 1	Operator 2	Engineer	Administrator	Kernel
			2	4	6	7	8
Circuit breaker, switch control	PL_Sec_CBControl	1		X	X	X	X
IED configuration	PL_Sec_IEDConfig	2			X	X	X
Circuit breaker tagging	PL_Sec_Tagging	3			X	X	X
Alarms acknowledgment	PL_Sec_AlmAck	4	X	X	X	X	X
Alarm deletion	PL_Sec_AlmDelete	5			X	X	X
Alarm configuration	PL_Sec_AlmConfig	6			X	X	X
Add/remove log-in users	PL_Sec_UserConfig	7				X	X
Reset alarms in device	PL_Sec_AlmReset	8			X		X
Shutdown runtime	PL_Sec_Shutdown	9			X	X	X
View waveforms	PL_Sec_ViewWaveform	10	X	X	X	X	X

NOTE: Privileges 1, 3, and 5 are currently not used. You can use them as you wish.

One way to limit access to the design time/configuration environment is to remove the user rights from certain Power SCADA Operation files. From the server, use Windows security to remove user accounts from individual features. For example, access to the following EXE files should be restricted to users who have design time/configuration privileges:

- `CtDraw32.exe` – Graphics builder
- `CtEdit32.exe` – Project Editor
- `CtExplor.exe` – Project Explorer
- `ProfileWizard.exe` – I/O Device Manager
- `ColorSwap.exe` – Color Swap Tool
- `ProfileEditor.exe` – Profile Editor

These files need to have the same level of security, as they are interrelated.

Additionally, to prevent users from accessing and changing the code responsible for enforcing user security in the Power SCADA Runtime, you need to lock down the user rights for the Cicode files (.ci extension) in the PLS_Include project.

TIP: Another way to lock users out from changing an existing project is to implement read-only projects.

Change access rights

You can edit the default access rights to each of the eight levels, thus changing the privileges that are available at each level. This is done in the `PLSSecurity.ci` file (in the PLS_Include project).

To change access rights:

1. Open `PLSSecurity.ci`.
2. Locate the AccessRights section.
3. For each right that you want to add (for example, to add access privileges for working with switches), add a new CASE.
4. Save and close PLSSecurity.ci.

NOTE: PLSSecurity.ci is overwritten any time that the PLS_Include project is overwritten. This happens with every release of the product, including service packs.

To ensure that you do not lose changes that you enter:

– Note the changes that you make. Keep a copy of these changes.
– Re-merge the changes every time that the PLS_Include project is overwritten.

Add users

This section discusses how to set up user IDs and passwords for the project.

⚠ WARNING

POTENTIAL COMPROMISE OF SYSTEM AVAILABILITY, INTEGRITY, AND CONFIDENTIALITY

- Use cyber security best practices.

Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

For safety reasons, only advanced users should be given access to such features as controls and resets. User access rights (privileges) are defined in **Security > Roles**, located in the Power SCADA Studio.

You can use single sign-on (SSO) to associate passwords for different products (such as Power SCADA Studio with Power SCADA Operation and Advanced Reporting and Dashboards Module). SSO allows the project user, when logged in to the Power SCADA Runtime, to access external applications, such as dashboards. For more information see "Configure Single Sign-On (SSO)" on page 388

Add and modify user accounts

You must add at least one user to any project before you can run and view it. Each user must have a role and a user account.

NOTE: We recommend that you use Windows Authentication when you create user accounts.

Terms you need to understand are:

- privilege: The level of access that is applied to a system element. A user account has individual privileges, that the user can then control.
- role: Contains a defined set of privileges that are assigned to users.

The Users screen controls the user access levels for each project.

To assign user access:

1. In Power SCADA Studio: Click **Projects** and then choose the project for which you want to assign user access.
2. Click **Security** **> Roles**.
3. For the first user, assign a user role, Windows group name (optional), and global privileges. If you need additional information, click Help from that screen. For global privileges, see the table in "Default User Access Settings (Privileges)" on page 357 for the level of each type of access right.
4. After you add the first role, click **Add**.
5. Click **Security** **> Users**.
6. Assign a user account. If you need more information, click Help.
7. After you add the first user, click **Add**.

To add additional users:

1. With a user displayed, click **Add**. This creates a copy of the user.
2. Enter the new user information in place of the old information and then click Replace to over-write the information.

Note that the record count increases by one with each addition. To view the users one at a time, scroll through the list.

NOTE: If your system includes Advanced Reporting and Dashboards Module, you can use single sign-on (SSO) to associate a Citect user with a Power SCADA Operation username/password or a Power Monitoring Expert username/password. See "Use Single Sign-On" on page 387 for details.

Cybersecurity

This section includes information on how to help reduce the threat of a cyber attack.

⚠ WARNING

POTENTIAL COMPROMISE OF SYSTEM AVAILABILITY, INTEGRITY, AND CONFIDENTIALITY

- Use cybersecurity best practices to help prevent unauthorized access to the software.

Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

Work with facility IT System Administrators to ensure that the system adheres to the site-specific cybersecurity policies.

Two-Factor Authentication (One-Time Password)

NOTE: For cybersecurity purposes, it is strongly recommended that you configure two-factor authentication in your projects; especially in deployments with control functionality.

Power SCADA Operation uses a one-time password (OTP) to accomplish two-factor authentication. OTP is implemented in Power SCADA Operation using a USB key device called a YubiKey. The YubiKey is designed to fit on a key ring or attached to a badge. It must be plugged into the client machine when the user authenticates.

NOTE: You can export one-time password settings to other servers. See "Export and import One-Time Password settings" on page 461 for details.

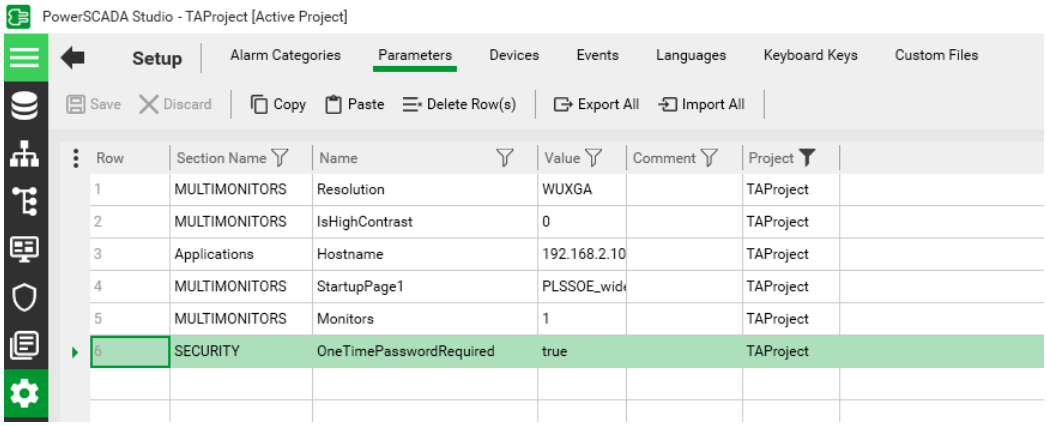
Add the Citect parameter

You need to add the parameter that allows Power SCADA Operation to communicate with the YubiKey. You can do this before or after you configure the YubiKey.

NOTE: Before you add the parameter, make sure the correct project is active.

To add the parameter:

1. From Power SCADA Studio, click **Setup** > **Parameters**.
2. Enter the following:
   - Section Name: Security
   - Name: OneTimePasswordRequired
   - Value: true



3. Compile the project.

Set Allow RPC to TRUE for all YubiKey-user roles

To use YubiKey in Power SCADA Operation, you must set Allow RPC to TRUE for all roles that include users with assigned YubiKeys. The default for Power SCADA Operation 9.0 is FALSE.

To change Allow RPC to TRUE:

1. In Power SCADA Studio, click **Security**  **> Roles**.
2. For each YubiKey-user role, change **Allow RPC** to **TRUE**.

YubiKey configuration

You can autoconfigure a YubiKey or program it manually.

In most cases, you can autoconfigure the YubiKey, thus avoiding the lengthier process of programming it. Autoconfiguration may not work with all YubiKey models; however, all OTP-compliant keys can be manually programmed.

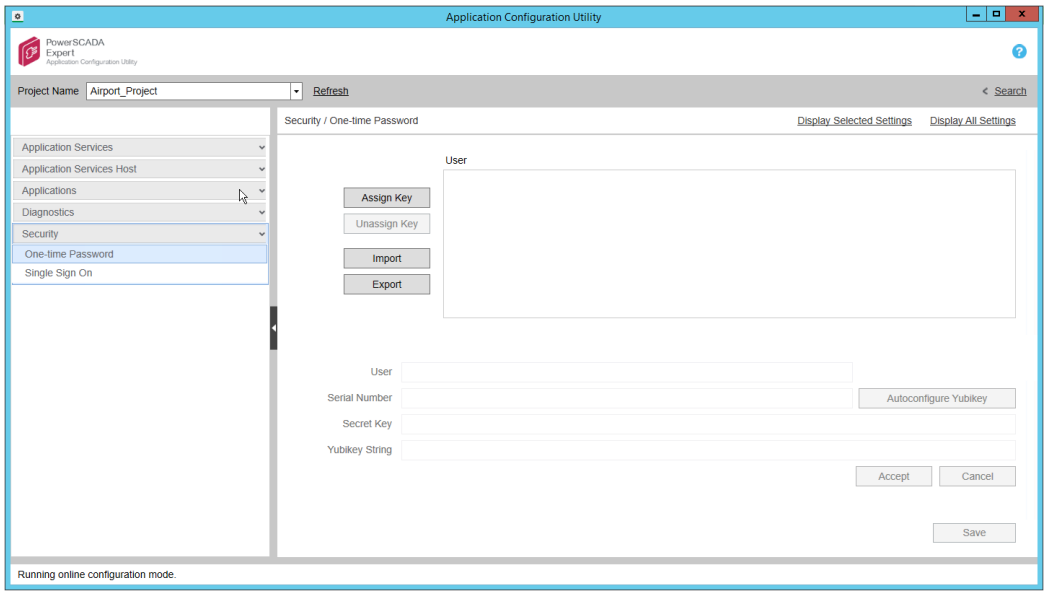
NOTES:

- Autoconfigure requires that you have a USB port available on your computer.
- If you do not have a USB port available on the server – because it is in a virtual machine or you do not have physical access– program the key on a remote machine (see "Manually configure the YubiKey" on page 363, below), and then transfer the configuration to the server (see "Two-Factor Authentication (One-Time Password)" on page 361, below).
- You can only have one YubiKey inserted at a time.
- If autoconfigure will not work, you must manually program the YubiKey. See "Manually configure the YubiKey" on page 363 for instructions.

Auto-configuring the YubiKey

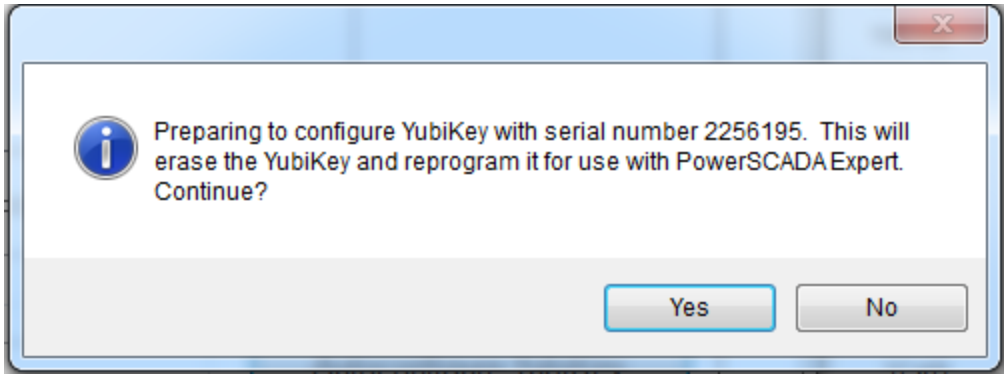
To auto-configure the YubiKey:

1. Insert the YubiKey into the USB port of the computer.
2. In the Application Configuration Utility, click **Security > One-Time Password**.



3. Click **Assign Key**.

   The grayed-out fields are enabled.
4. In the **User** field, type the Power SCADA Operation username (or user name from Active Directory) to which you want to assign the YubiKey.
5. Click **Autoconfigure YubiKey**. The following message appears:



   This message tells you that all settings on the key will be erased, including any key assignments.
6. To continue, click **Yes**. The key will receive a new secret key.
7. Click **Accept**.

Manually configure the YubiKey

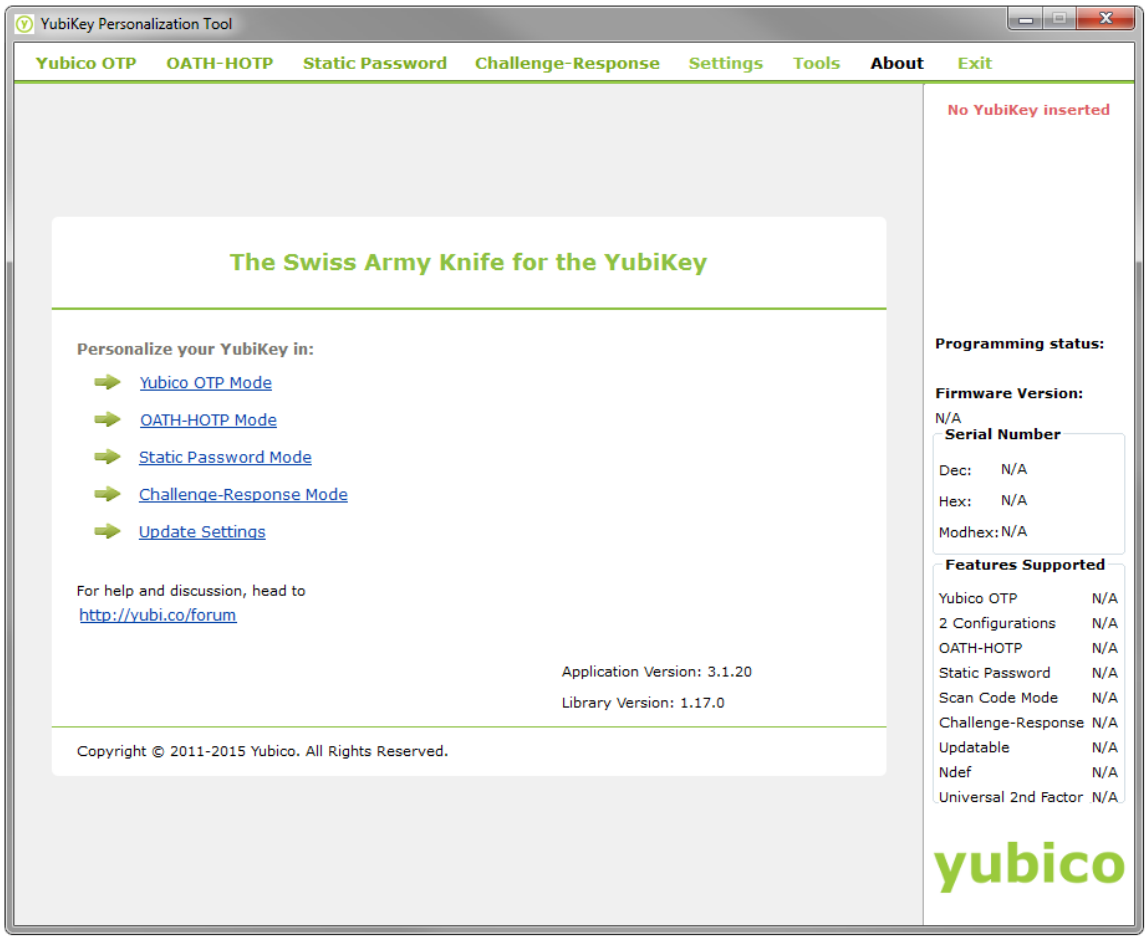
If you cannot auto-configure the YubiKey, program and configure it manually.

After you obtain the YubiKey from a third-party vendor, (such as Amazon), download the YubiKey Personalization Tool from the Yubico web site: www.yubico.com; click Products > Services & Software > Personalization Tools > Download YubiKey Configuration Tools.

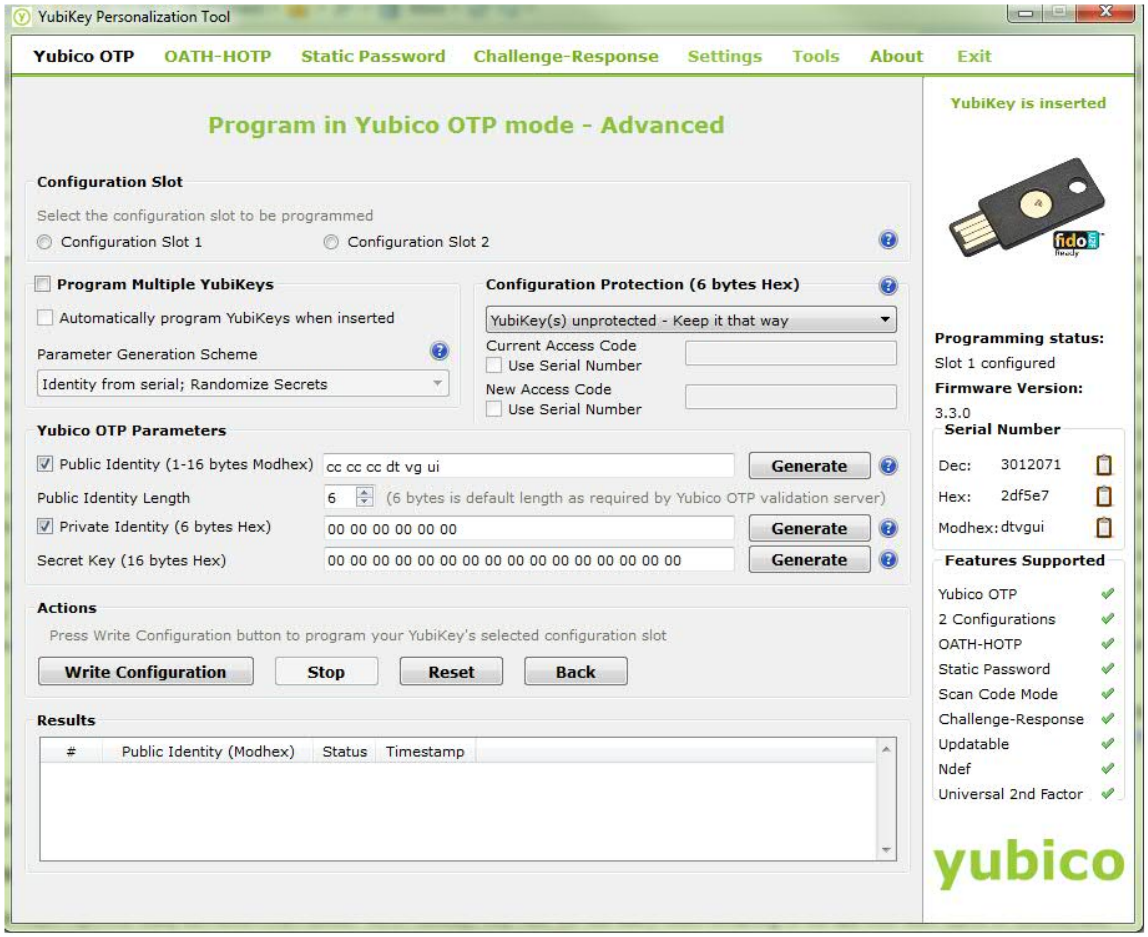
NOTE: This procedure outlines how to configure a single slot. If you want to use both of the key's configuration slots, download the YubiKey documentation, located under the Support tab of the Yubico website.

To manually configure the key:

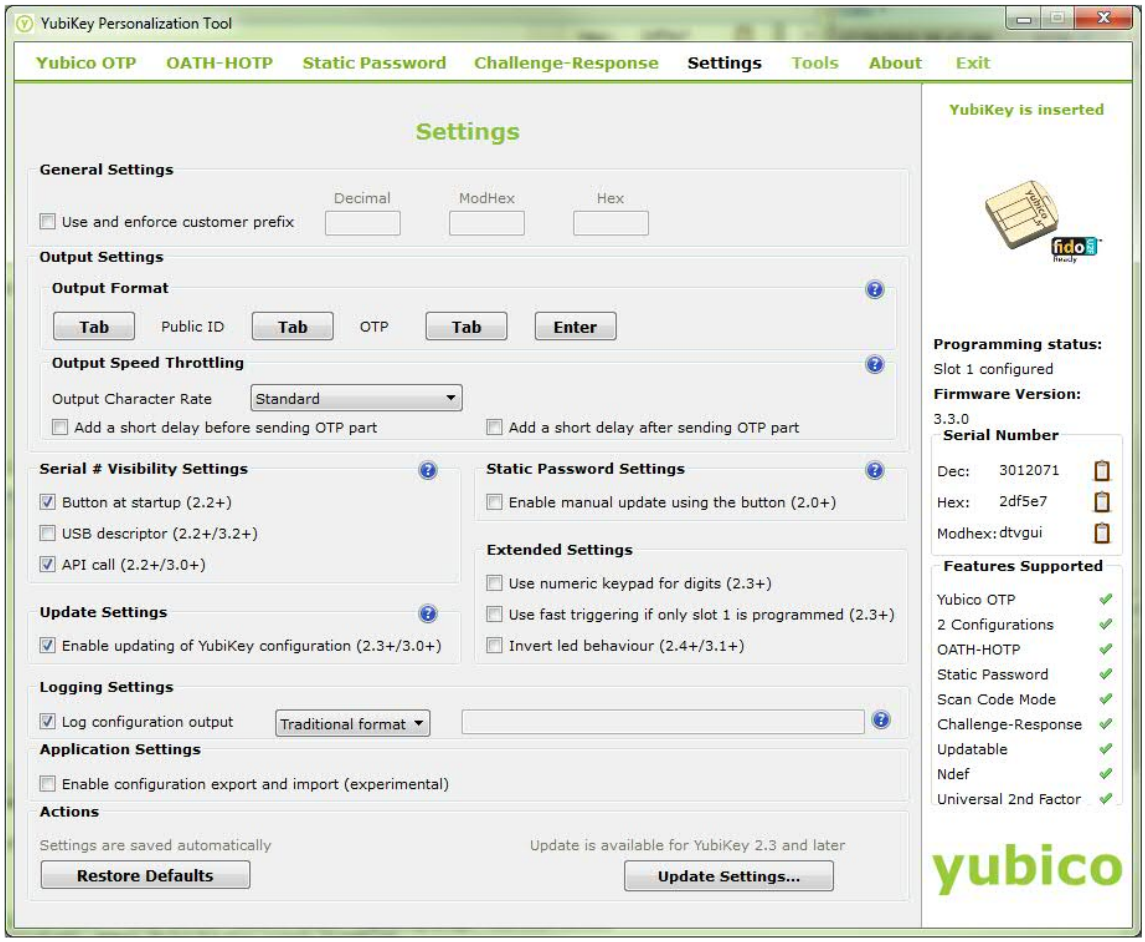
1. Launch the YubiKey Personalization Tool. The following screen appears:



2. Insert the YubiKey into a USB port of your computer. Click the Yubico OTP Mode link. At the next screen, click **Advanced**. The following screen appears:



3. In the **Configuration Slot** section, select the slot you want to configure.
4. In the **Yubico OTP Parameters** section:
   a. Click **Public Identity**, and then click **Generate**.
   b. Do not edit the default **Public Identity Length**.
   c. Click **Private Identity** and then click **Generate**.
   d. Beside **Secret Key**, click **Generate**.
   e. Make note of the secret key that displays, including all characters and spaces. You will need it when you add the key to the Application Configuration Tool.
5. In the **Actions** section, click **Write Configuration**.
6. Click the **Settings** tab. This following screen appears:



7. Enter the following information:
   a. Under **Output Settings**, click **Enter** to enable it; when enabled the button turns blue. Do not enable any of the **Tab** buttons.

      This causes a return and an "OK" to automatically occur when you press the Yubikey as part of login in Power SCADA Operation.
   b. Ignore the remaining settings. Click **Update Settings** at the bottom right of the screen.

      The key is programmed.
8. Next, configure the key on the Power SCADA Operation computer:
   a. In the Application Configuration Utility, click **Security > One-Time Password**.
   b. Click **Assign Key**.
   c. The fields on the lower half of the screen are enabled.
   d. For **User**, type the user name that you are adding. This should be a Power SCADA Studio user.
   e. For **Serial Number**, type the number that is printed on the underside of the key.
   f. For **Secret Key**, enter the Secret Key from the YubiKey Personalization Tool (created above). Enter the secret key exactly as it was created, including all spaces. After you enter it, the key will be encrypted and will display as bullets (••••) in the future.

g. Press the button on the top of the YubiKey.

h. **YubiKey String:** This field is populated when you press the button in step 6.

i. Click **Accept**.

9. Repeat step 8 for any additional keys.

NOTE: Repeat steps 1 to 8 on each server computer in a redundant or distributed system.

Log in with a programmed YubiKey and One-Time Password

After the key is programmed and associated with a user in Power SCADA Operation, and you have enabled YubiKey usage, the user will use the key to log in to the system.

To log in:

1. Insert the programmed YubiKey into a USB port of the Power SCADA Operation server.
2. Launch Power SCADA Operation Runtime, or access runtime via a remote web client.
3. Run the project you want to view.
4. In the upper right corner of the Startup screen, click **Login**.
5. Enter your name and password and then click **OK**. The One-time Password screen appears.
6. Press the button on the YubiKey.

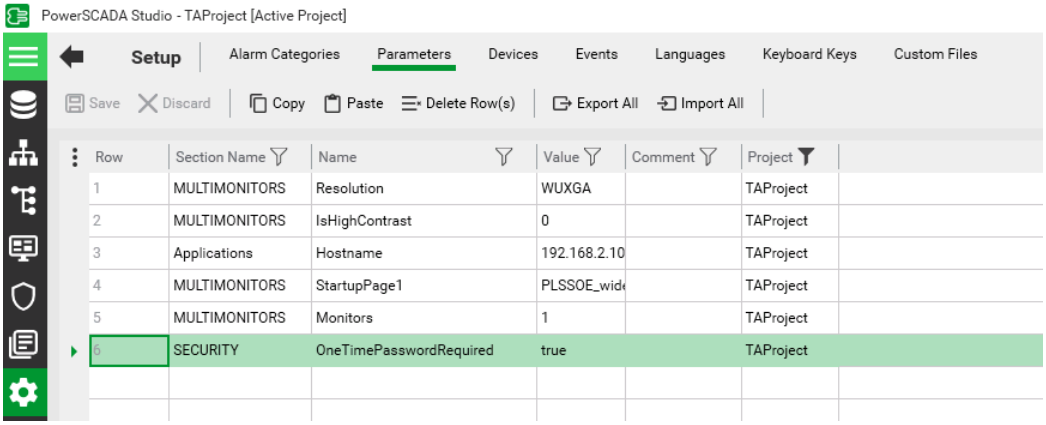
   The one-time password is generated. The key and software communicate behind the scenes to verify the uniqueness of the one-time password and to click OK.

   You can start using runtime screens.

Disabling YubiKeys

To disable a YubiKey:

1. In Power SCADA Studio, click **Setup**  **> Parameters**, locate the parameter for the YubiKey.
2. Change the **Value** from true to false, and then compile the project.



McAfee White Listing

McAfee Application Control (Application Control) is a dynamic whitelisting program that is used to prevent block unauthorized applications from running on your systems. The installation files for the software are included on the Power SCADA Operation disk, but you must purchase the license separately.

Install Application Control on the Power SCADA Operation primary and secondary servers, as well as the Advanced Reports and Dashboards server. For detailed information about installing Application Control on each server, see the McAfee Installation Guide located on the Power SCADA Operation installation disk (McAfee Embedded Control > Documents > Installation-Guide-v6.2.0).

NOTE: Allow the install to add a desktop shortcut; you need it for all interactions with Application Control. Also, before you run Application Control, make sure that you have installed all other software that you want on the computer.

To begin using Application Control, right-click the desktop icon and select the Run As Administrator option.

First, you need to create and solidify the whitelist. To do this:

1. Invoke the *sadmin* command line as an administrator and type the command `sadmin solidify`.

   This process can take some time to complete. When it is complete, you see a line telling you total files scanned and the number that are "solidified."
2. Verify the whitelist with the command `sadmin status`.

   Verify that the whitelist status of drives or volumes is *solidified*.
3. When this is complete, you need to enable the enforcement of the whitelist: type the command `sadmin enable`.
4. Add updaters: Updaters are components for which you provide permission to update the system. Any program or script that will be able to update the system must be configured as an updater. To add an updater, enter on the command line:

   `sadmin updaters add <xxx>`

   where xxx is the name of the component

   For a complete discussion of updaters, see "Using Updaters" in the McAfee Product Guide (on the Power SCADA Operation installation disk, see McAfee Embedded Control > Documents > Product-Guide-v6.2.0)

When running in Enabled mode, Application Control protection can prevent a legitimate application from executing (if the required rules are not defined). Application Control tracks all such failed attempts made by authorized executable to modify protected files or run other executable files.

You can review information for failed attempts to identify updater rules to allow legitimate applications to run successfully. To do this:

1. Enter the command `sadmin dia`
2. To add the suggested updaters to the authorized list, use the command `sadmin diag fix`.

When you deploy Application Control to protect a system, it scans the system and creates a whitelist of all executable binaries and scripts present on the system. The whitelist also includes hidden files and folders.

The whitelist lists all authorized files and determines trusted or known files. In Enabled mode, only files that are present in the whitelist can execute. All files in the whitelist are protected; you cannot change or delete them. An executable binary or script that is not in the whitelist is said to be "unauthorized," and it is prevented from running.

You can also use Application Control to write protect files, directories, drives or registry entries. Additionally, you can use it to Read Protect Files, Directories, or Drives. For more information about these applications, see the Product Guide.

Tofino Firewall

The ConneXium Tofino Firewall is an industrial firewall designed for use in industrial control system networks. The firewall offers deep packet inspection of Modbus TCP, allowing restriction at the Modbus command level as defined by the network designer. It is highly configurable using software called ConneXium Tofino Configurator (included with Tofino purchase). The software allows a user to define entire networks, referred to as projects, which can have multiple Tofino firewalls protecting a myriad of devices (referred to as Assets) at different points in the network.

The setup steps are:

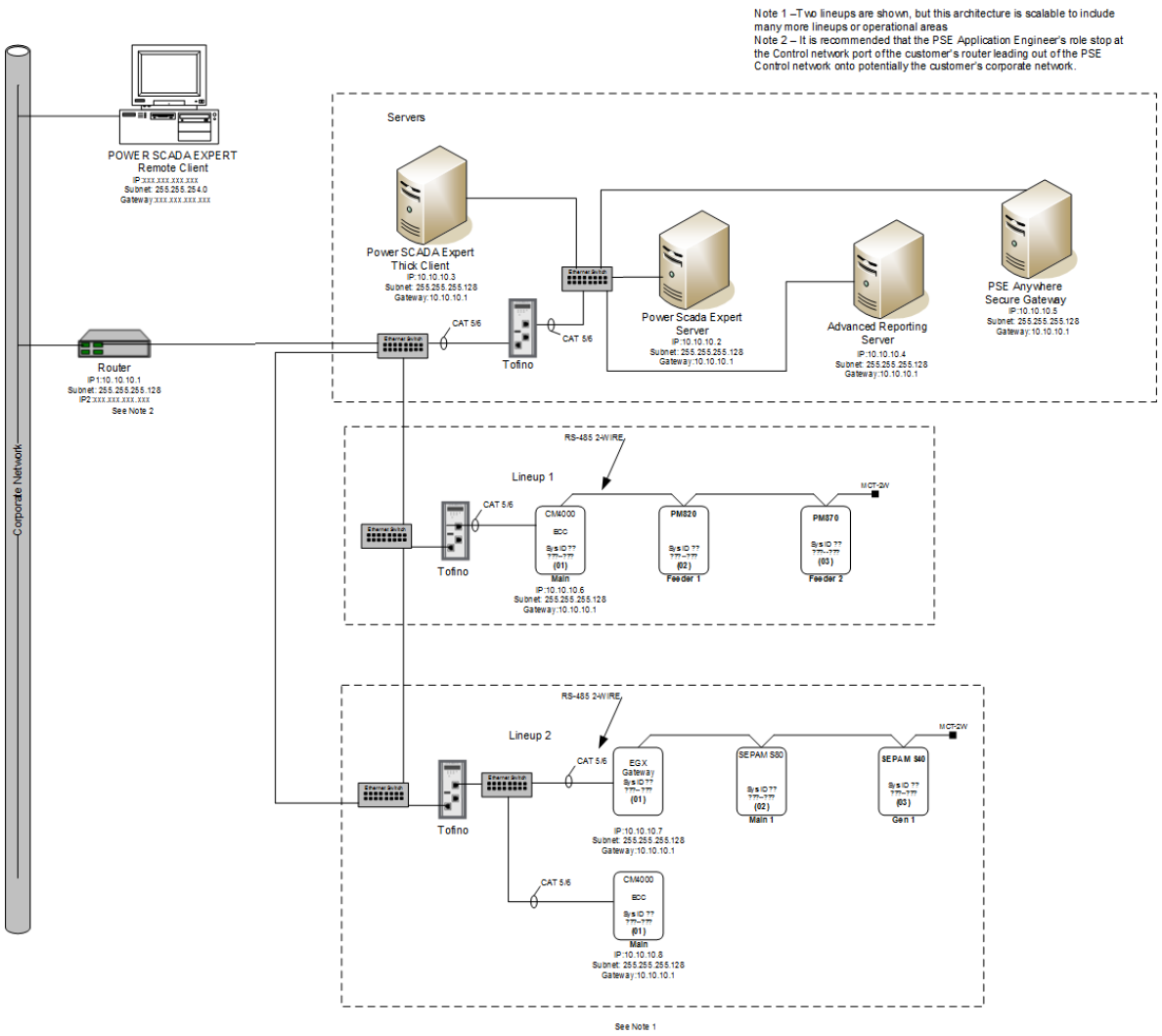
1. Install the Tofino Configurator and create a project.
2. Add all the Tofino devices to your network.
3. Add all the other devices on the network.

You configure the rules for the network that define the traffic that can pass through which firewall.

We recommend that you begin with the firewalls in test mode so you can see what would have been blocked and then adjust accordingly. The firewall configurations should be then loaded onto a USB flash drive that is used to upload the configuration to each firewall.

Detailed information about the setup and configuration of the Tofino architecture is provided in the ConneXium TCSEFEA User Manual V1. You should not use this firewall as an “edge” device, bridging the Control Network with public networks.

The following is an example architecture that can serve as reference for how one of the networks might be constructed. It is a small network that can be scaled out to fit a much larger system.


Note 1 –Two lineups are shown, but this architecture is scalable to include many more lineups or operational areas
Note 2 – It is recommended that the PSE Application Engineer's role stop at the Control network port of the customer's router leading out of the PSE Control network onto potentially the customer's corporate network.
Corporate Network
POWER SCADA EXPERT Remote Client
IP xxx.xxx.xxx.xxx
Subnet: 255.255.254.0
Gateway xxx.xxx.xxx.xxx
Router
IP1:10.10.10.1
Subnet: 255.255.255.128
IP2:xxx.xxx.xxx.xxx
See Note 2
Servers
Power SCADA Expert Thick Client
IP:10.10.10.3
Subnet: 255.255.255.128
Gateway:10.10.10.1
CAT 5/6
Tofino
Power Scada Expert Server
IP:10.10.10.2
Subnet: 255.255.255.128
Gateway:10.10.10.1
Advanced Reporting Server
IP:10.10.10.4
Subnet: 255.255.255.128
Gateway:10.10.10.1
PSE Anywhere Secure Gateway
IP:10.10.10.5
Subnet: 255.255.255.128
Gateway:10.10.10.1
RS-485 2-WIRE
Lineup 1
MCT-2W
CM4000
EDC
Sys ID ??
??h–??h
(01)
Main
IP:10.10.10.6
Subnet: 255.255.255.128
Gateway:10.10.10.1
PM820
(02)
Feeder 1
PM870
(03)
Feeder 2
Lineup 2
EGX Gateway
IP:10.10.10.7
Subnet: 255.255.255.128
Gateway:10.10.10.1
SEPAM S80
(02)
Main 1
SEPAM S40
(03)
Gen 1
See Note 1

Customize default behaviors

In this section, you will find these topics:

Customize a project using Cicode

Cicode is a programming language designed for use in this product to monitor and control plant equipment. It is a structured language similar to Visual Basic or 'C'. You need no previous programming experience to use it. However, it is assumed that you will have received Cicode training before you attempt to use Cicode.

Using Cicode, you can access all real-time data (variables) in the project: variable tags, alarms, trends, reports, and so on. You can also use Cicode to interface with the computer's operating system and communication ports.

The following Cicode modules have been written specifically for use in PLS_Include:

For information about other parameters, see the **Cicode Programming Reference** help file in the Citect SCADA help file (...\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin\Help\Citect SCADA).

For information about driver-specific INI parameters that you can configure, see "Citect INI Parameters" on page 577.

PLSProviderEngine.ci Module

Use this module when you want to invoke a provider to produce results that can be displayed or acted on in a custom table or report that you create. Providers invoked by this method must be written so that they take a single string as input and return a single string as output.

Module construction

The following string functions are included in this module:

CallProvider

This function invokes a provider (whose GUID-based identifier must appear in the sProvider argument) with a single string as input (the sArgs argument). The input string can consist of anything that is meaningful to the provider that you invoke.

The provider then returns a string-based token.

Construction of CallProvider:

```
STRING FUNCTION CallProvider(STRING sProvider, STRING sArgs)
        INT hHandle;
        STRING sResult;
        ErrSet(1);
        sProvider = "^"" + sProvider + "^"";
        sArgs = "^"" + sArgs + "^"";
        hHandle = DLLOpen("ProviderGatewayUnmanaged.dll", "MakeRequest",
"CCC");
        sResult = DLLCall(hHandle, sProvider + "," + sArgs);
        DLLClose(hHandle);
        IF IsError() THEN RETURN "ERROR"; END
        RETURN sResult;
END
--------------
```

GetProviderStatus

This function reports the status of a provider invocation by showing the percentage of its completeness. A provider has completed its work when the status reaches 100 percent,

To retrieve status with this function, pass in a token (obtained previously by calling CallProvider) and examine the number contained in the function's return string (from 0 to 100).

Construction of GetProvider Access:

```
------------
STRING FUNCTION GetProviderStatus(STRING sToken)
       INT hHandle;
        INT iPercent;
        ErrSet(1);
                sToken = "^"" + sToken + "^"";
        hHandle = DLLOpen("ProviderGatewayUnmanaged.dll", "GetPercent",
"JC");
        iPercent = DLLCall(hHandle, sToken);
        DLLClose(hHandle);
        IF IsError() THEN RETURN "ERROR"; END
        RETURN iPercent;
END
-------------
```

GetProviderResult

This function retrieves the result from a provider. Pass a unique token (obtained previously by calling CallProvider) to this function. It returns the provider result as a string. Note that you should only call this function after you verify that the provider work is 100 percent complete.

Construction of GetProviderResult:

```
------------
STRING FUNCTION GetProviderResult(STRING sToken)
        INT hHandle;
```

```
        STRING sResult;
        ErrSet(1);
        sToken = "^"" + sToken + "^"";
        hHandle = DLLOpen("ProviderGatewayUnmanaged.dll", "GetResult", "CC");
        sResult = DLLCall(hHandle, sToken);
        DLLClose(hHandle);
        IF IsError() THEN RETURN "ERROR"; END
        RETURN sResult;
END
--------------
```

Clear cache and refresh platform

When you add, delete, or update a device or topic, you need to shut down and then restart the Power SCADA Runtime. At that time, we recommend that you also clear the cache and then refresh the platform. This ensures that data is .

Clearing the cache removes stale data. Refresh updates the Schneider Electric CoreServiceHost list of devices and topics, making it available to App Mods.

Clearing and refreshing uses the PLSProviders.ci module. See "PLSProviderEngine.ci Module" on page 371 for instructions on creating the statements needed.

PLS_ClearCache

In the Schneider Electric CoreServiceHost, when you call a provider and it returns its result, it caches that result for a given amount of time (which varies by provider). If someone calls that provider again, the system will return the cached result.

If someone adds a device during this time, and then restarts run mode, the device is not available for features like LiveView or basic reporting. Thus, if someone tries to view a table or run a basic report, using the new device, it will not display. The next call that is made to the cache will refresh it.

NOTE: You can create a graphics page that includes a button that calls the cache or refresh.

To clear the cache, call the `PLS_ClearCache` function by doing one of the following:

- If the Schneider Electric CoreServiceHost is on the machine from which you are invoking the function, you can call it with no input parameters:

  ```
  PLS_ClearCache();
  ```

  This can be done during startup or by using a button handler.

- If the Schneider Electric CoreServiceHost is on a different machine, you must supply parameters to identify where the Application Services core resides. For example, if the customer's Schneider Electric CoreServiceHost resides on an I/O Server named "IOServer1" on "Cluster1", to call PLS_ClearCache, enter:

  ```
  PLS_ClearCache("IOServer", "IOServer1", "Cluster1");
  ```

  NOTE: This cannot be done at startup; you must do it after the startup routine is run. For example, you can use a button handler.

PLS_PlatformRefresh

After you clear the cache, run the platform refresh to update the Schneider Electric CoreServiceHost, causing it to refresh its list of devices and topics.

To run the refresh, call the `PLS_PlatformRefresh` function by doing one of the following:

- If the Schneider Electric CoreServiceHost is on the machine from which you are invoking the function, you can call it with no input parameters:

  ```
  PLS_PlatformRefresh();
  ```

- If the Schneider Electric CoreServiceHost is on a different machine, you must supply parameters to identify where the Application Services core resides. For example, if the customer's Schneider Electric CoreServiceHost resides on an I/O Server named "IOServer1" on "Cluster1", to call PLS_PlatformRefresh, enter:

  ```
  PLS_PlatformRefresh("IOServer", "IOServer1", "Cluster1");
  ```

Localizing Power SCADA Operation

You can localize the following Power SCADA Operation components:

- Power SCADA Runtime
  - PLS_Include Library Contents (including Notifications Settings)
  - Default Starter Project
- Power SCADA Applications
  - Basic Reports
  - Diagnostics
  - LiveView

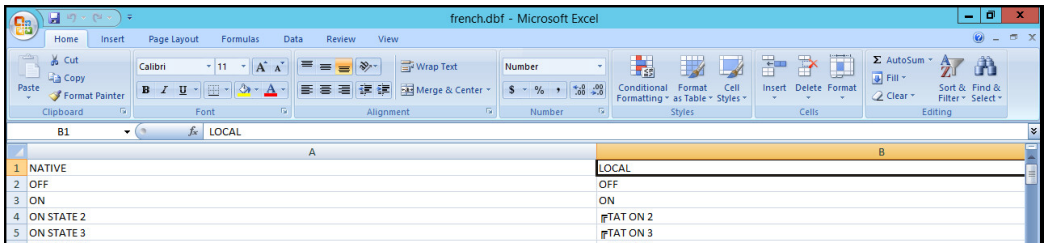
You must create all custom project content in the local language.

Localizing Power SCADA Runtime

You can localize the runtime HMI by creating a localized .dbf file, and setting it to be your project language source file in Power SCADA Studio.

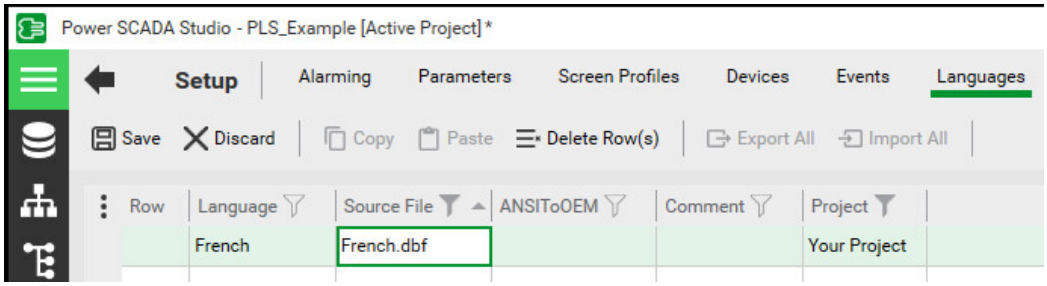
To localize the Power SCADA Runtime:

1. Navigate to C:\ProgramData\Schneider Electric\Power SCADA Operation\v9.0\User\Include.
2. Using Apache OpenOffice™, or Microsoft® Excel with the .dbf extension, open English.dbf.
3. In Column B (LOCAL), enter translations for the Column A (NATIVE) runtime strings.

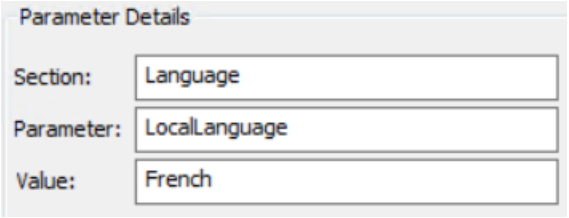


4. Click **Save As**, enter [Localized Language].dbf as the File Name.

5. In Power SCADA Studio, click **Setup**  **> Languages**.
6. Enter the appropriate names in the Language, Source File, and Project fields, and then click **Save**.



7. In **Projects**  , click the **Setup Wizard** drop-down arrow and then click **Setup Editor**.
8. In Parameter Details, enter Languages in the Section field.
9. From the **Language Parameters** list in the right pane, select **[Language]LocalLanguage**.
10. In the **Value** field, enter the localized language.



11. In the **Comment** field, enter a custom comment, or click **Generate** to use the default message.
12. Click **Add**.

Localizing Power SCADA Applications

You can localize PowerSCADA applications by creating localized .resx files for each application your project requires:

Application	Folder Path	.resx File Name

Common Data Model (CDM) files: Alarm Proxy Basic Reports * LiveView **	C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\Applications\AppServices\bin\Resources	CDMMetadataNameResources.resx CDMMetadataValueResources.resx CDMTopicDescriptiveNameResources.res CDMUnitResources.resx
Basic Reports	C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\Applications\AppServices\bin\Resources	ReportDefinitionResources.en-US.resx Reporting.RapidAccess.resx Reporting.StandardReports.resx Reporting.Utilities.en-US.resx * CDM files also required
Diagnostics	C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin\DiagnosticsLanguages	DiagnosticsResources_en-US.resx
LiveView	C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\Applications\LiveView\Viewer\App_GlobalResources	LiveViewViewer.resx ** CDM files also required

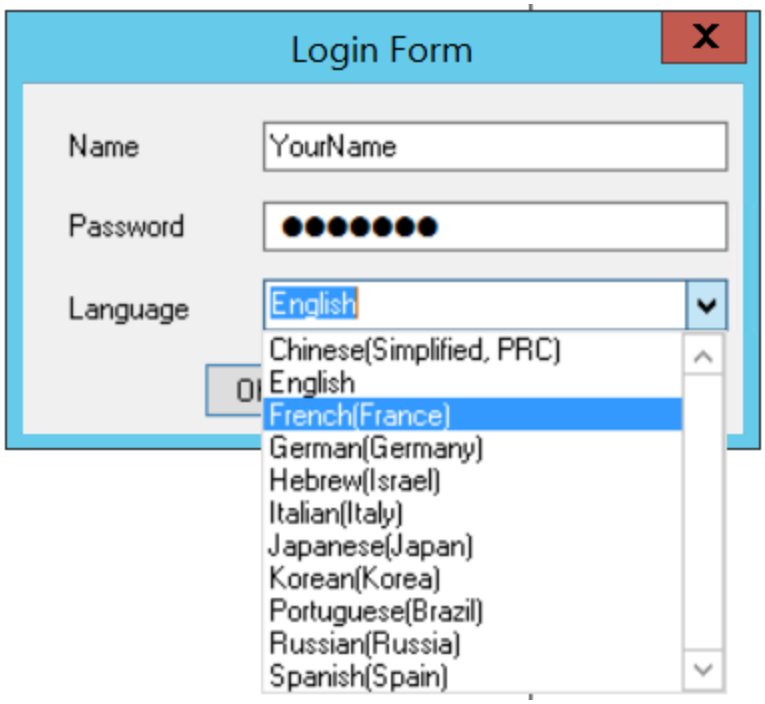
To localize a Power SCADA application:

1. Navigate to the specified application folder(s) and create a copy of each .resx file associated with the application.
2. Open a copy .resx file in Visual Studio and replace the terms in the left column with the translated terms.
3. Click Save As, and replace en-EN with the appropriate new Language tag found in the Language table.
4. Repeat Steps 2 to 3 for all the .resx file copies you created for the application.
5. Repeat Steps 1 to 4 for all required project applications.

   NOTE: You only need to complete Steps 1 to 4 for all the Common Data Model (CDM) files once and it will apply to all the applications that reference the CDM files.

6. Launch Power SCADA Operation runtime, from the Login Form Language drop-down list,

select the localized language and then click **OK**.



Translating device information

There are several description or comment fields throughout Power SCADA Operation that you use to create copy for translation purposes. If you type a comment in the following format:

`@(XXX)` where XXX = the copy that will be translated

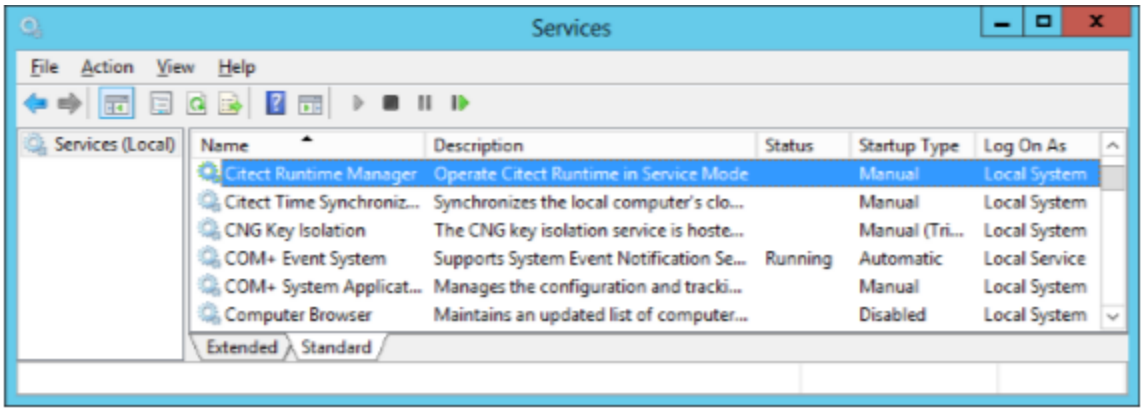
The text you enter in the Comment field is added to the default language file, named `English.DBF`. After the project is compiled, this file is located in ...\Documents and Settings\All Users\Application Data\Schneider Electric\Power SCADA Operation\9.0\User\<*your project*>. `English.dbf` contains terms that will be translated from English.

To create another language file for translation, set the `Citect.ini` parameter `[Language]LocalLanguage` to the specified language, then re-compile. So, for example, if you set this parameter to French, a `French.dbf` file is created in the project folder when you compile. You can then enter the translated text in the LOCAL field of the file. Repeat this same step for each additional language file you want in this project.

At runtime, the user can choose the DBF file that will be used in the display.

Running Power SCADA Operation as a Windows Service

When you install Power SCADA Operation, a Windows service – called Citect Runtime Manager – is created:



By default, the service Status is Stopped, the Startup Type is set to Manual, and Log On As is set to the Local System account.

Running the Power SCADA Operation Windows service automatically provides the following benefits:

- Protects applications that provide runtime and historical data to clients and allows data to be preserved across user log in sessions.
- The application can be started automatically at system power on, minimizing downtime in the event of a system reboot or unexpected issue.
- Security benefits, as well as efficiency improvements, are gained when users do not have to log in to the operating system. Access to the server can be restricted and locked down to suit specific security requirements.

Configuring the Power SCADA Operation service

Before you can set the Citect Runtime Manager Service to run automatically, you might need to configure dependencies with another service.

In the following example, since the Citect Runtime Manager has a dependency on the FlexNet Licensing Service to acquire valid licenses, the FlexNet Licensing Service must start first.

You can set service dependencies by modifying the Service Control Manager settings.

To configure the required dependency:

1. Launch a command prompt with Administrator privilege.
2. To create a dependency that starts the FlexNet License Service before the Citect Runtime Manager Service, enter the following command:

```
sc config "Citect Runtime Manager" depend= "FlexNet Licensing
Service"
```

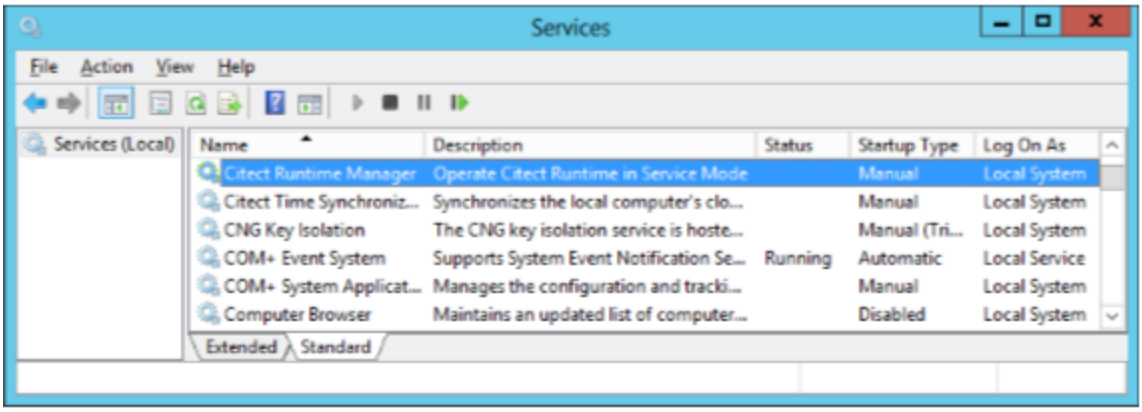
NOTE: You must insert a space character after `depend=`. You must also enclose service names in quotes, which includes spaces in their names.

Windows Service Operation

With the Citect Runtime Manager Service now configured, note the following:

The service is run as Local System account on Session 0.

When an application is run in Session 0, it is not possible to raise this session to the active desktop to interact with it. It will remain hidden.

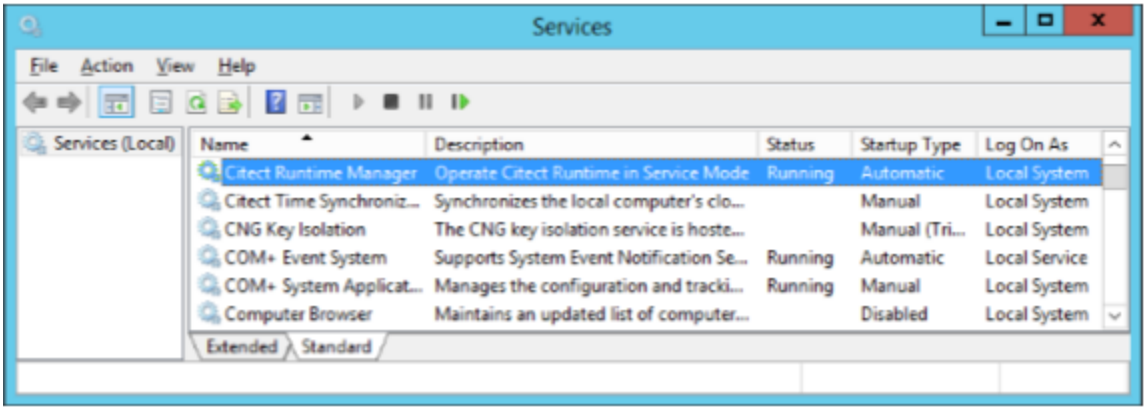


To run Power SCADA Operation as a Windows service:

1. Set the Citect Runtime Manager service Startup Type to Automatic.
2. Reboot the machine to allow Power SCADA Operation to run as a Windows service.

Alternatively:

1. Right-click the Citect Runtime Manager service, and then click Start Service to run Power SCADA Operation without rebooting the machine.



You can now log in and log off without disrupting the system.

Launch Power SCADA Operation from a Remote Client

After you configure Power SCADA Operation to run as a service, end users can use one of two shortcuts to launch the runtime screens from a remote client:

Service Display Client (Control) – Gives users the access provided in the Control Client license (PSA1020xx).

Service Display Client (View only) – Gives users the access level provided in the View-only Client license (PSA1030xx).

These shortcuts are located in the Power SCADA Operation \bin folder (default: `C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin`).

NOTE: You must have the appropriate license for the type of client the user will launch.

To launch Power SCADA Operation from the remote client:

The end user double-clicks the client they will use.

Power SCADA Operation locates the license that was purchased for that client and displays the log in page.

At the Power SCADA Operation log in page, the user logs in with their normal user credentials.

TIP: To make it easier for the end user to find the shortcut, copy the shortcuts to the desktop.

System Startup and Validation Checks

To test and validate the project:

1. Test two-factor authentication. (For more information see "Log in With a Programmed YubiKey and One-Time Password" on page 467.
2. Test the Web Client: Open the Web Client and verify that links are working properly.
3. Test the Advanced One-Line.
4. Test single sign-on to Dashboards, Advanced Reports, and Web Diagrams.
5. "Verify that I/O Devices are Communicating" on page 381

Log in With a Programmed YubiKey and One-Time Password

Use this procedure to log in to Power SCADA Operation sing a YubiKey.

Prerequisites

The YubiKey is programmed and associated with a user in Power SCADA Operation, and the YubiKey is enabled.

To log into the system using YubiKey:

1. Insert the programmed YubiKey into a USB port of the Power SCADA Operation server.
2. Launch Power SCADA Operation Runtime, or access runtime using a remote Web Client.
3. Run the project you want to view.
4. In the upper right corner of the Startup screen, click **Login**.
5. In the Power SCADA Studio login screen, enter your name and password and then click **OK**.

   The One-time Password screen appears.
6. Press the button on the YubiKey.

The one-time password is generated. The key and software communicate behind the scenes to verify the uniqueness of the one-time password and to click OK.

You can start using Power SCADA Runtime.

Verify that I/O Devices are Communicating

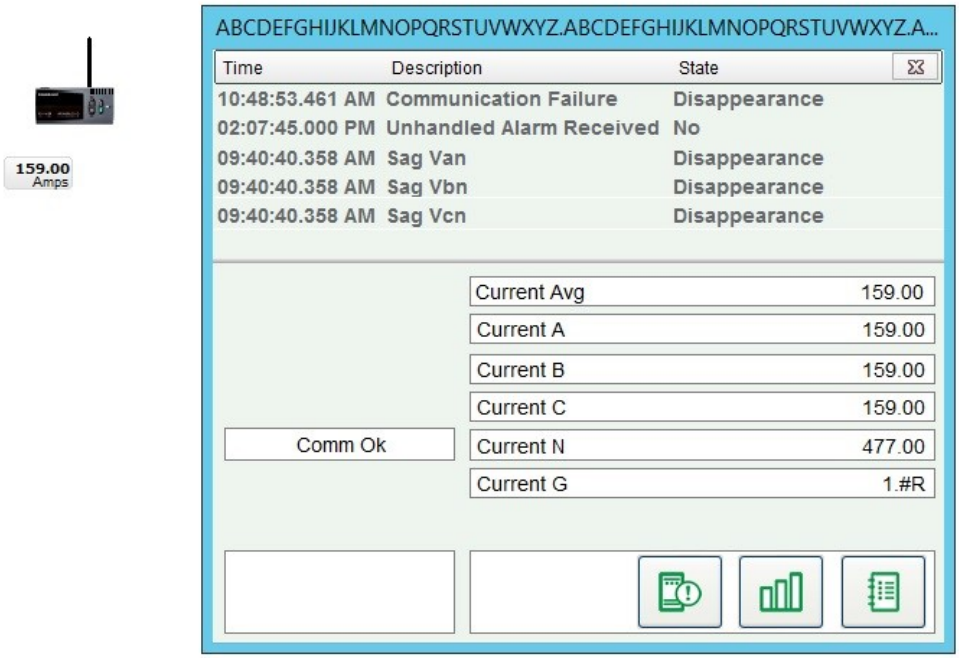
After the system is configured and communicating in runtime mode, verify that all devices are communicating correctly. All devices that are not communicating will trigger "Communication Failure" alarms, which can be seen in the active alarm log screen. For more information on how to add this screen to the project, see "Add Alarm Pages" on page 303. On the Menu Configuration page, use PLSDspShowAlarm(0) as the menu item Menu Command.

Use one of the following methods to test communication.

View the graphics pages

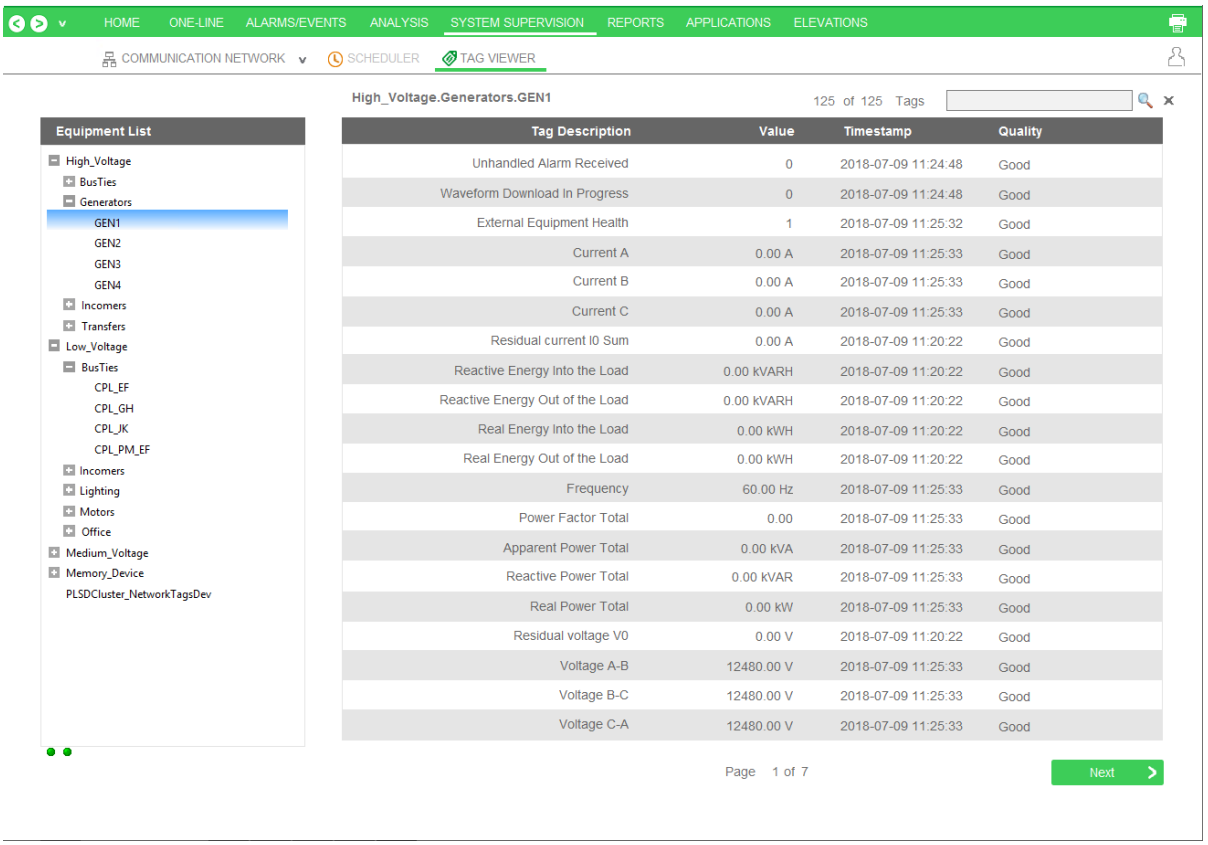
1. Create a graphics page containing an appropriate genie selected from the pls_meter library, found in the PLS_Include project.
2. Assign the selected genie to the specific device needed to verify communications.
3. Save the page and compile the project.
4. In the Power SCADA Runtime, double-click the genie to open the genie pop-up. Verify that the updated readings displayed by the genie match the actual values on the meter itself. If the readings match, you have verified the device is communicating.

   The following image shows a genie and its related genie pop-up:



Use the Tag Viewer to learn the status of all project tags

During runtime, open one of the pages that displays real time tag values. The example below is PLSTagView. Compare the values displayed on the Tag Viewer page to actual values displayed on the meter itself. If the compared values match, then you have verified communications with that device.



Use the One-Line Configuration Utility to verify that devices are connected and animations are working

The electrical system must be in a non-critical state so that the breakers being used will not cause any adverse effects (such as putting a person's safety at risk or affecting a process). Breaker genies should be able to remotely operate the breaker.

⚠ DANGER

EQUIPMENT ELECTRIC SHOCK, EXPLOSION, OR ARC FLASH

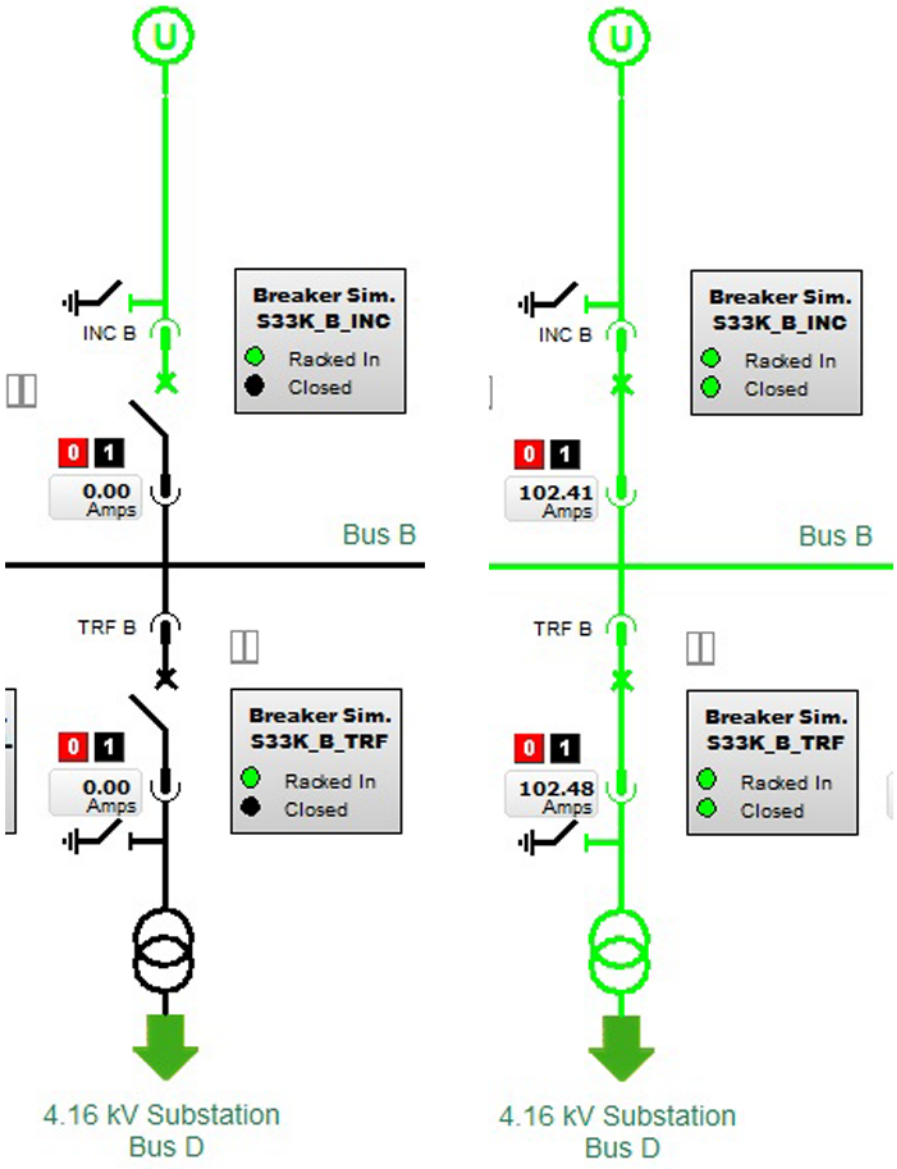
- Do not rely solely on the display of the graphic on the one-line.
- Use this procedure only during development, and not on a live deployed system.
- Before energizing or de-energizing any equipment from this software, verify that all personnel are a safe distance from all energized equipment.
- Before testing, verify that the proper lock out/tag out procedure is followed, to ensure that the equipment is in an electrically safe condition.
- Ensure that all safety regulations and procedures have been followed before you work on the equipment.

Failure to follow these instructions will result in death or serious injury.

In the Graphics Builder, create a one-line diagram with breaker genies that use the breakers you want to verify. Use the proper logic and passwords to configure the one-line on the diagram. After the diagram is successfully created, open the graphic page in runtime mode.

The breaker genie status indicator should mirror the current breaker state. Also, the busbar color should accurately reflect the electrical state of the conductors connected to the breaker.

The following illustrates the appearance of the one-line drawing with breakers first open and then closed. Note the color change, from black to green (energized), and the position and current changes on the breakers.



Communications Losses

When you bring your system on line, if you find that Power SCADA Operation has lost communications with a device, verify the following:

- That the physical connection is correct and secure.
- The IP address.
- The Modbus address.
- statusRegister, statusRegistersCount, and statusRegisterType

Use Diagnostics

The I/O Device Settings page provides a quick view of the I/O device INI settings for all protocols, clusters, servers, ports, and devices. Use this information as the first step to troubleshooting device and communication issues in your system.

For information on how to use Diagnostics, see "Diagnostics Overview" on page 502.

Distributed systems

Use the information in the tables below to find the content you are looking for.

Section	Description
"Setting up more than two I/O Servers per cluster" on page 386	How to add multiple I/O Servers per cluster
"Set up the Advanced Reporting and Dashboards Server" on page 390	How to configure the Advanced Reporting and Dashboards Module
"Configure the Power SCADA Anywhere Server" on page 430	Information on how to configure Power SCADA Anywhere.
"EcoStruxure Web Services setup" on page 432	Information on how to configure EcoStruxure Web Services.
"Time synchronization" on page 434	Considerations for synchronizing time across a distributed system.
"Time zone settings" on page 434	Information on how distributed time zones are handled in Power SCADA Operation.
"OFS system time stamping" on page 435	How to configure OPC Factory Server (OFS) time stamping in Power SCADA Operation.
"Configure Power SCADA Operation as an OPC-DA Server" on page 457	How to configure an OPC-DA Server.
"Configure Power SCADA Operation as an OPC-DA Client" on page 458	How to configure an OPC-DA Client.

Setting up more than two I/O Servers per cluster

If you need to add more than two I/O servers to a cluster, you need to define a redundant I/O device called *NetworkTagsDev* for each of the servers. If you do not do this, you can lose device status information during runtime.

If the cluster includes only one or two I/O Servers, the I/O devices are automatically added when you add the cluster during I/O Device Manager configuration (see Citect SCADA Help for details). If a system has more than two I/O Servers in a cluster, you must manually add the *NetworkTagsDev* I/O device for the remaining servers (after the first pair).

To create the board, port, and *NetworkTagsDev* I/O device, ensure the following:

- All redundant *NetworkTagsDev I/O* devices have the same number
- The Startup Mode field is set to Standby; do this for all standby *NetworkTagsDev* I/O devices, including the one created by the I/O Device Manager
- The Equipment field is set to <Cluster>_NetworkTagsDev

The field values for the forms in each of the I/O servers should be:

Boards Form
Board Name: <any unique name> (suggestion: BOARDy_SVRz)
Board Type: DISKXML
Address: 0
Leave everything else blank.

Ports Form
Port Name: <any unique name> (example: Px_BOARDx_PRJz)
Port Number: <any unique number within the I/O server> (suggestion: x)
Board Name: <use the board name defined above>
Leave everything else blank.

I/O Devices Form
Name: NetworkTagsDev
Number: <same number as the one defined in the corresponding device>
Address: NetworkTagsDev
Protocol: DISKXML
Port Name: <use the port name defined above>
Startup Mode: Standby
Equipment: <Cluster> _NetworkTagsDev
Leave everything else blank.

NOTES:

- Startup Mode is only visible when in extended form mode (press F2 to toggle between simple form mode and extended form mode, while in the I/O device form).
- The Equipment field is hidden by default. To change it to visible, open units.dbf (in the project folder) in Excel.
- If the system has one or two I/O servers per cluster, the startup mode of the standby *NetworkTagsDev* I/O device could be set to StandbyWrite in the I/O Device Manager. If the system has more than two I/O servers per cluster, the startup mode of all standby *NetworkTagsDev* I/O devices must be set to Standby.
- One side effect of this is that, when the system switches to a redundant I/O server, affected devices will momentarily lose communication as the system transitions to the redundant server.
- If the primary and redundant alarms servers are synchronizing, data will be slow to display in the Alarm Log and Events Log.

Use Single Sign-On

With single sign-on (SSO), you associate a Citect user with a Power SCADA Operation username/password or a Power Monitoring Expert username/password. This allows the Citect user to access external applications, such as Dashboards, using an SSO user password from Power Monitoring Expert.

To set up single sign-on, see:

- "Add Single Sign-On Settings to Citect.ini" on page 388
- "Configure Single Sign-On (SSO)" on page 388

Add Single Sign-On Settings to Citect.ini

Open the Citect.ini file (typically in C:\ProgramData\Schneider Electric\Power SCADA Operation\v9.0\Config). In this file, you will add the following SSO values (if they are not already there):

~~[SSO] (deprecated in version 8.1, now use [Applications])~~
~~RemoteCallHandlerServer= (deprecated in version 8.1, do not use)~~
~~RemoteCallHandlerCluster= (deprecated in version 8.1, do not use)~~
~~SupportsVisitorDashboard= (deprecated in version 8.2, do not use)~~

```
[Applications]
Hostname=
WebReachServer=
Area=
PrivLevel=
UseHTTPS=
PSEHostname=
```

Complete each parameter with the value specified below. Then save the modified citect.ini file:

- `Hostname` – The name or IP address of the computer that hosts Advanced Reports and Dashboards (Power Monitoring Expert).
- `WebReachServer` – Default value: empty string. This parameter specifies the host name or IP address of the WebReach server machine. In most cases this is the same as the `Hostname` above. Required for integration with WebReach to display Diagrams in the runtime graphic pages.
- `Area` – Allows the use of the “area” field associated with Power SCADA project users. It can be configured on a per application level including: Power SCADA Operation reporting, Reporting (PME), WebReach, and Dashboards, and provides the ability to limit the use of SSO operations to specific areas.
- `PrivLevel` – Allows the use of the “privilege level” field associated with Power SCADA project users. It can be configured on a per-application level including: Power SCADA Operation reporting, Reporting (PME), WebReach, Dashboards, and provides the ability to limit use of SSO operations to specific privileges.
- `UseHTTPS` – Default value: `TRUE`. Required in Power Monitoring Expert 9.0
- `PSEHostname` – If you want to use Power SCADA Operation basic reports, use this parameter. This parameter specifies the IP address for the Power SCADA server.

Configure Single Sign-On (SSO)

Use single sign-on (SSO) to associate a Power SCADA project user (a Citect user) with either a Power SCADA Operation or Power Monitoring Expert (PME) username/password. When the user is logged in to the Power SCADA Runtime and accesses an external application—such as Dashboards—the SSO user password is used to authenticate with the external application.

When you use SSO, we recommend that you maintain the components on the same computer or on a secure network. If higher security is needed, use Transport Layer Security.

⚠ WARNING

POTENTIAL COMPROMISE OF SYSTEM AVAILABILITY, INTEGRITY, AND CONFIDENTIALITY

- Store system keys, AES encryption files, or other files containing passwords to a secure site.

Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

Cybersecurity policies that govern how sensitive system files are securely stored vary from site to site. Work with the facility IT System Administrator to ensure that such files are properly secured.

To configure SSO:

1. Open the Application Configuration Utility:
   - From Programs click Power SCADA Operation > Application Config Utility.

     Or
   - In Power SCADA Studio: Click **Projects > Home**, click **Power Applications > Application Config Utility**.
2. Click the **Security** tab.
3. From the **Application** drop-down list, choose the application (such as Dashboards, Basic Reporting, Advanced Reporting, Diagrams, LiveView) to which you want to map a Power SCADA Operation user.
4. In **Timeout**, enter the time after which the system will stop trying to find a match. If no match is found, SSO for this user will not take place.
5. Click **Guest User**, then click **Edit** to launch the Edit User dialog.
6. In the Edit User dialog, type the SSO user and password that match the username and password of the Power Monitoring Expert (PME) or Power SCADA user to which the Guest User is mapped.

   NOTE: Guest User allows the Power SCADA Runtime Operator to access the integrated applications in PME or Power SCADA Operation (basic reports), however, the Operator will be acting as a Guest User and will have fewer feature privileges.

   For example, you could create a guest user that only has access to dashboards, and link a PME user to this account. The Power SCADA Operator could then access dashboards without logging into the Power SCADA Runtime.
7. In the **Users** area, manage users access to the applications. Use this area to add users who need to have a Power SCADA project user account.

- **Citect User:** The project username for the user logging in to the Power SCADA Runtime.
- **SSO User/SSO Password:** The established credentials for this user, either from Power SCADA Operation or Power Monitoring Expert.

SSO Calls from a Web Client

Power SCADA Operation automatically detects calls that are made from a Web client. The calls are sent to an I/O Server. For this to work properly, the user needs Remote Procedure Call (RPC) privileges for web client access.

To enable SSO calls from a Web client:

1. In Power SCADA Studio: Click the **Security > Roles**.
2. For the desired Power SCADA role or Windows Group, change **Allow RPC** to **TRUE**.

NOTE: See Integration Parameters for information on other INI parameters that must be set for Single-Sign On with integrated applications.

Configure SSO for Active Directory Users

SSO allows the use of Windows Active Directory users. Follow the instructions above to create a Guest User. When the Power SCADA Runtime Operator uses the system and logs into the Power SCADA Runtime interface with a Windows user, the operator will be treated as a Guest User and will be able to access integrated Advanced Reports and Dashboards through SSO.

See also:

- "Add Single Sign-On Settings to Citect.ini" on page 388

Set up the Advanced Reporting and Dashboards Server

NOTICE
INOPERABLE SYSTEM Ensure that you have received Power SCADA training and understand the importance of the Power SCADA Operation productivity tools and workflows. **Failure to follow these instructions can result in overly complex projects, cost overruns, rework, and countless hours of support troubleshooting.**

NOTE: Power SCADA Operation is build on Citect Studio and includes productivity tools that are designed and optimized to create the tags you need to configure power-based SCADA projects. If you have prior experience using Citect Studio, do not rely exclusively on Citect tools to build a power SCADA project.

NOTE: Review the topics that comprise this section if your Power SCADA system includes the Advanced Reporting and Dashboards Module.

The installation medium for Advanced Reporting and Dashboards is located on the same DVD or .ISO as the Power SCADA Operation installation, in the Advanced Reporting Module folder.

On the server that you will use for the Advanced Reporting and Dashboards Module, install software in the following order:

1. Microsoft SQL Server
2. Advanced Reporting and Dashboards Module: Use the Power SCADA Operation with Advanced Reporting and Dashboards installation medium and installation guide.

At this stage of the commissioning procedure, many I/O devices have been added to the Power SCADA Operation system. However, to obtain Power Quality reports, Branch Circuit Monitoring reports, and hierarchy functionality, these types of devices must be added again on the Advanced Reporting and Dashboards server. To do this, use the Management Console application to add and configure these devices.

For more information on using the Management Console, see the *Power Monitoring Expert 9.0 – System Guide*.

NOTE: You can use single sign-on (SSO) to associate a Power SCADA project user (a Citect user) with a Power Monitoring Expert (PME) username/password. See "Configure Single Sign-On (SSO)" on page 388 for more information.

ETL for Power SCADA Operation

For Power SCADA Operation with Advanced Reporting and Dashboards, the ETL Administration Tool extracts data from Power SCADA Operation and loads it into Power Monitoring Expert. Once loaded into the Power Monitoring Expert database, the data can be used in Reports and Dashboards.

⚠ WARNING

INACCURATE DATA RESULTS

- Do not incorrectly configure the software or the devices.
- Do not base your maintenance or service actions solely on messages and information displayed by the software.
- Do not rely solely on software messages and reports to determine if the system is functioning correctly or meeting all applicable standards and requirements.
- Consider the implications of unanticipated transmission delays or failures of communications links.

Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

Before using the ETL Administration Tool

Before using the ETL Administration Tool (PSO to PME), ensure the following:

- The Power Monitoring Expert system is installed and configured.
- Power SCADA Operation is installed and configured.
- All devices have been added and configured on both systems; see **Important note about device synchronization**, below.

- The ETL (PSO to PME) is properly installed on the Power Monitoring Expert server.
- The ETL has remote access to the PSO Server. See "Allowing ETL remote access to the PSO Server" on page 392 for details.

Important note about device synchronization

When a PSO device is included in a PSO to PME ETL job and the job is run, that device (and its data) is added to PME as a source. Because these PME sources are not visible in the PME Management Console, ensuring device synchronization between the integrated systems can present challenges.

For example, if a PSO device included in an active PSO to PME ETL job is deleted or renamed, update the PSO to PME ETL job to include the change. Furthermore, since the historical source (and its data) does not change in PME, you might also have to update the source and its data in the database.

For this reason, it is strongly recommended that before you create a PSO to PME ETL job, make sure that your sources are named correctly.

See "Limitations" on page 419 for more details on managing sources.

Allowing ETL remote access to the PSO Server

The PSO Server must allow the ETL to access it remotely from the PME Server.

To allow remote access to the PSO Server:

1. In Windows Explorer, navigate to ...\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\Applications\AppServices\bin.
2. Open `Services.xml`.
3. Search the file for `<EndpointName>Data/RequestHandler</EndpointName>` and change only this hosted service's `AllowRemoteAccess` value to `true`:

   ```
   <AllowRemoteAccess>true</AllowRemoteAccess>
   ```

4. Save the file.

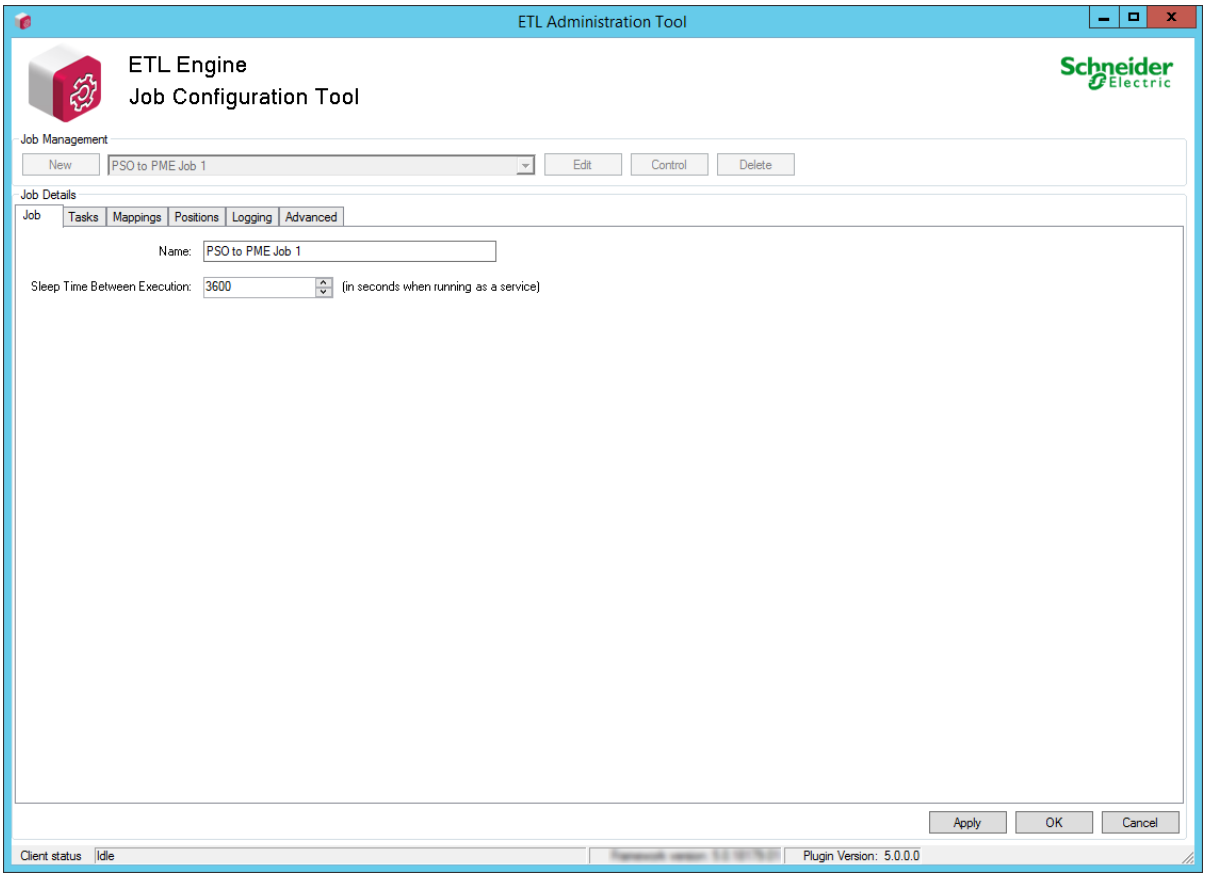
Opening the ETL Administration Tool

NOTE: On Windows Operating Systems with restricted permissions, the ETL tool might not initialize and load its plugins on start up. This is due to limited write permission on the ETL install folder (for example: `C:\Program Files\`). The workaround is to install ETL to a custom folder with write permission.

To open the ETL Administration Tool:

1. Double-click the ETL desktop shortcut. Depending on your operation system, you can also open the ETL Administration Tool from the **Start** menu or by typing the name of the ETL.

The ETL Administration Tool opens:



Upgrading a PSO to PME ETL job

You can upgrade an ETL job that was created in a previous release of the ETL tool when the ETL job includes a PSO Extract Task.

Upgrading an ETL job is useful when an existing PSE 8.2 ETL job exists, and the underlying PSE 8.2 system has been upgrade to PSO 9.0. Upgrading to Power SCADA Operation 9.0 made the 8.2 ETL job obsolete.

To upgrade a PSO ETL job:

1. On the **Advanced** tab, click **Upgrade**.

   The Upgrade Mapping Items window appears.

2. Click **Upgrade Source and Quantity Mapping Items**.

   The ETL tool starts the mapping upgrade process.

   The mapping item upgrade routine attempts to update as many device and topic IDs in the job's internal state (for example: position counters and device-topic representations within the job).

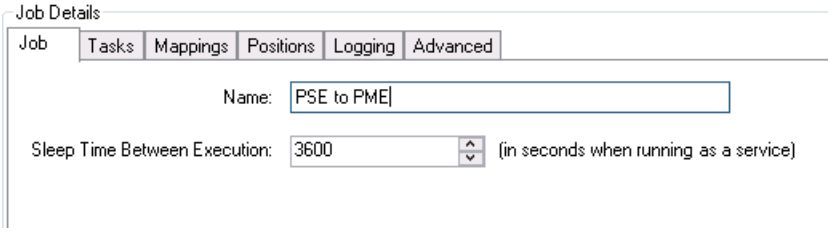
   When the mapping upgrade process is complete, detailed results are displayed. This information is also save to a new XML file in the MappingResults folder (under the ETL root). This XML file is useful for tracking and troubleshooting device and topic mappings before and after the upgrade operation ran.

NOTE: Running the upgrade routine is technically optional, since a new job could be created after upgrading from PSE 8.2 to PSO 9.0. The **Load Sources** button on the mapping screen could be run again. The downside to this would be the existing position counters would be lost. Therefore the new job would not necessarily pick up where the old job left off.

Creating a PSO to PME ETL job

To create a PSO to PME ETL job:

1. In the ETL Administration Tool, click **New**.
2. Enter the name of the job in the **Name** field.



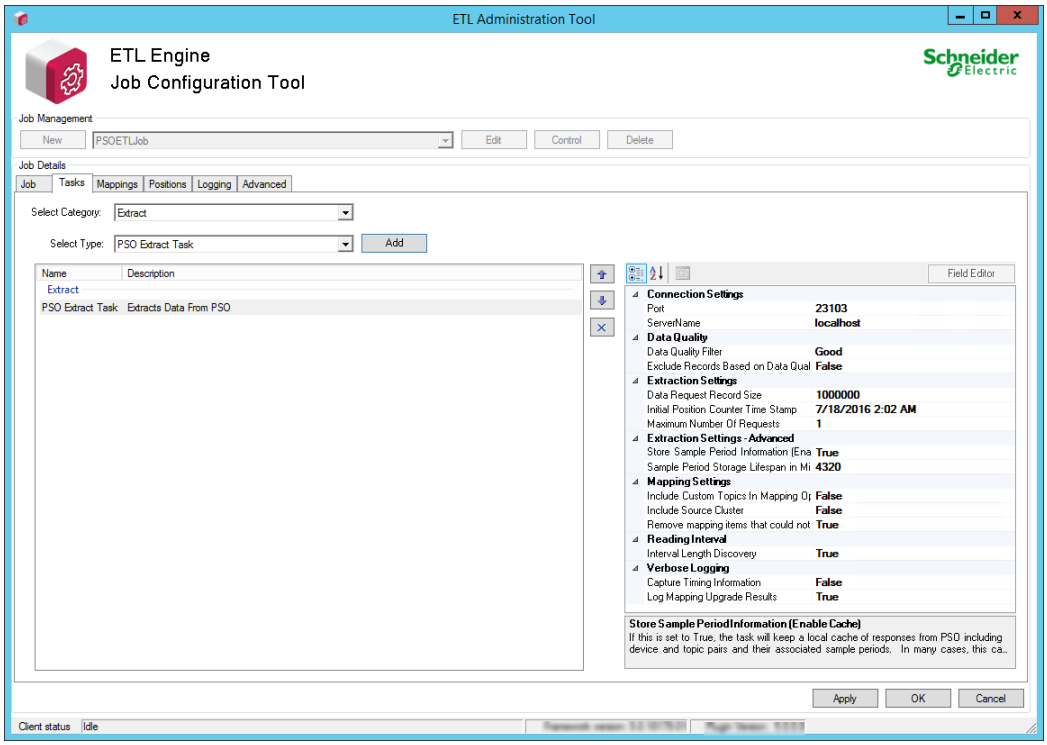
NOTE: The ETL job name has to be unique. Make sure your ETL job name does not conflict with any other ETL jobs on your system. This is particularly important to consider when registering ETL jobs to run as Windows services.

3. For testing purposes, use the default **Sleep Time Between Execution** interval of 3600 seconds.

NOTE: After you confirm that the ETL job runs successfully, the initial data transfer has occurred, and the ETL job is ready to be scheduled to run as a service, you can set the **Sleep Time Between Execution** to `900` seconds; PSO uses a 15 minute interval to collect trend data.

4. Click the **Tasks** tab.
5. From **Select Category** select **Extract**.
6. From **Select Type** select **PSO Extract Task**.
7. Click **Add**.

   The extract task name and description appear under the Extract heading:

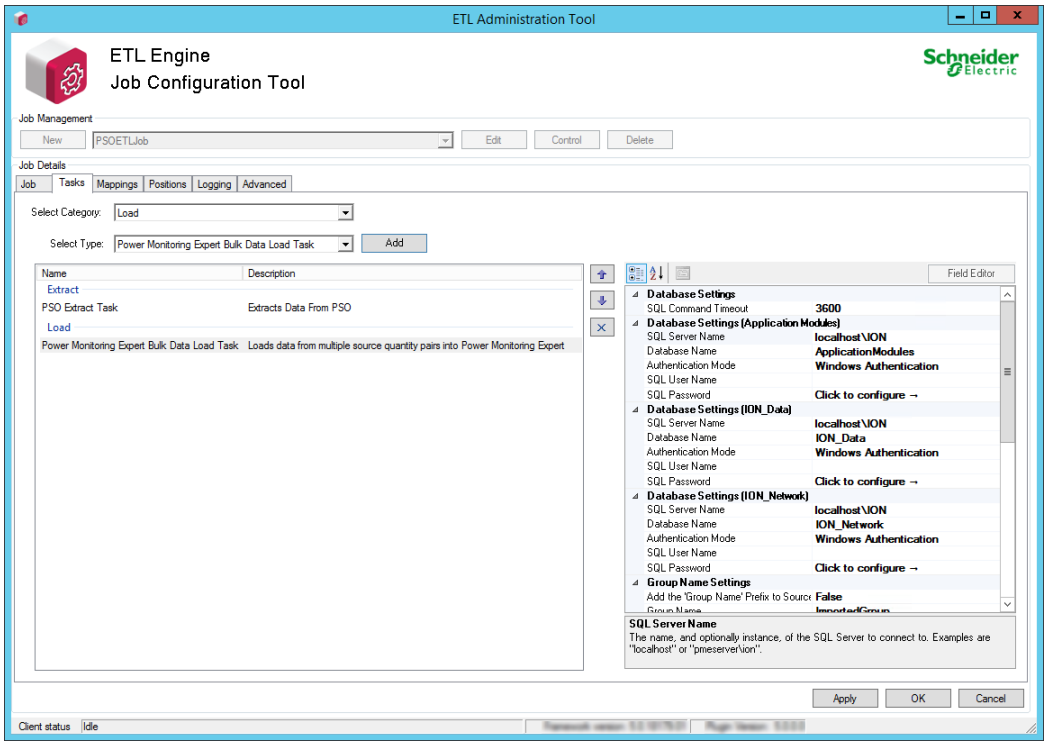


8. In the Field Editor pane, configure the extract task. See "Configuring the PSO to PME extract task" on page 396 for details.
9. (Optional) Add and configure a transform task. See "Configuring the PSO to PME transform task" on page 402 for details.

   NOTE: For most PSO to PME ETL jobs, the transform task is not needed.

10. From **Select Category** select **Load**.
11. From **Select Type** select **Power Monitoring Expert Bulk Data Load Task**.
12. Click **Add**.

    The load task name and description appear under the Load heading:



13. In the Field Editor pane, configure the load task settings. See "Configuring the PSO to PME load task" on page 403 for details.

    NOTE: If you plan to use the Energy Cost Report or Load Profile Report, review the **Enable Recorder and Channel Creation** setting in the "Configuring the PSO to PME load task" on page 403 table.

14. Click **Apply** to save without exiting the job, or click **OK** to save and exit the job.

After the ETL tasks are configured, map the extracted data sources to the target data store. See "Configuring PSO to PME mappings" on page 406 for details.

Configuring the PSO to PME extract task

Configure the **PSO Extract Task** after you add it to the ETL job. Click the extract task to display the configurable settings. Click a setting name to change its value. Some settings are configured by entering a value, while others are configured by selecting an option from a drop down list.

After you complete the extract task configuration, click **Apply** to save the ETL job without exiting the job, or click **OK** to save and exit the ETL job.

Setting Name	Description	Setting Parameters / Recommended Values
Connection Settings		
Port	Thee port used for communicating with PSO.	Default: 23103; leave as is.
Server Name	The name or IP address of the PSO server.	Default: localhost

Setting Name	Description	Setting Parameters / Recommended Values
Data Quality		
Data Quality Filter	When the 'Exclude Records Based on Data Quality' setting is set to True, only data records with this Data Quality value will be extracted. Other records will be ignored.	Values are: Good, Bad, NotApplicable, Disabled Default: **Good**
Exclude Records Based on Data Quality	If this setting is **True**, the quality property of each data record extracted from PSO will be examined and only records with the desired quality (as indicated by the 'Data Quality Filter' setting) will be included in the extracted data set. If this setting is **False**, no extracted records will be excluded by the task based on their quality property.	Default: **False**
Extraction Settings (see Note below)		
Data Request Record Size	The maximum number of records in each data request, sent to SCADA. See "Grouping" on page 399 for more information on how to use this setting.	Default: **1,000,000** Min: 100,000 Max: 3,000,000 **NOTE:** 1 device, 1 topic, 2 years, at 15-minute interval is about 70,000 records.
Initial Position Counter Time Stamp	The starting time stamp for extracting data.	Default: back-dated 2 years from the load task creation.
Maximum Number Of Requests	The maximum number of requests per job run. See "Grouping" on page 399 for more information on how to use this setting.	Default: **1** Min: 1 Max: 100 **NOTE:** In many cases this setting should be increased. Example: When processing two years' worth of data per device-topic pair, or when running the job for the first time.
Extraction Settings - Advanced		
Store Sample Period Information (Enable Cache)	When True, the task keeps a local cache of responses from PSO including device and topic pairs and their associated sample periods. When False, the device and sample period is requested every time the job runs.	Default: **True** In many cases, setting this to True improves performance.

Setting Name	Description	Setting Parameters / Recommended Values
Sample Period Storage Lifespan in Minutes	When Store Sample Period Information is set to True, this setting determines the lifespan of the sample cache. Each time the sample period information is retrieved from PSO, a timestamp is captured. Each time the cache is used, this timestamp is checked against the lifespan to determine if the sample period cache needs to be refreshed. Each time the cache is refreshed, a new sample period information request is sent to PSO. If new devices were added to the PSO system and they are not showing up in the ETL mapping grid after you click **Load Sources**, try disabling this cache (or temporarily setting the lifespan to 1 minute) and then mapping try again. Once you have the devices you need, the cache settings can be set back to the values shown above.	Default: **4320** (3 days)
Mapping Settings		
Include Custom Topics in Mapping Operations	Include (true) or exclude custom topics from the lookup operation. **NOTE:** If the PSO system includes any custom topics, set Include Custom Topics In Mapping Operations to True. If False, all custom topics will be ignored.	Default: **False**
Include Source Cluster	Determines whether the cluster name is included in the source name.	Default: **False** Set to True if you want to include the cluster name in PME device names (required if the same device is used in more than one cluster).
Remove mapping items that could not be updated	When True, all mapping items that could not be upgraded are removed from the job. This takes effect only when running a job upgrade operation ("Upgrading a PSO to PME ETL job" on page 393	Default: **True**
Reading interval		
Interval Length Discovery	Have the extract task determine the reading interval for each pair based on each pair's data.	Default: **True**
Verbose Logging		
Capture Timing information	When True, additional timing information is logged to the trace log file during job execution.	Default: **False**
Log mapping Upgrade Results	When True, information about mapping items that were upgraded is logged.	Default: **True**

Note:

Data Request Record Size: Maximum Number of Requests and Threading must be balanced for system performance and consumption.

Power SCADA Core Services Memory: Total data records requested at any given time from Power SCADA must be kept under 10,000,000. This number should be below 3,000,000. If too many requests are sent, Power SCADA Core Services may run out of memory and need to be restarted.

The total records requested at any given time can be calculated as Data Request Record Size x Number of Threads.

ETL Memory: The total data records requested per job run is dependent on the available RAM on the local machine. You should keep this number below 50,000,000, but this is only limited by the local machine RAM.

The 'total data records requested per job run' can be calculated as Data Request Size x Maximum Number of Requests.

Requests per job are dependent on the available RAM on the local machine. You should keep this number below 50,000,000, but this is only limited by the local machine RAM. Requests per job can be calculated as Data Request Size x Maximum Number of Requests.

Example:

Data Request Record Size	Maximum Number Of Requests	Threading	Total requests at any given time	Requests per job
100,000	100	25	2,500,000	10,000,000
500,000	100	10	5,000,000	50,000,000
1,000,000	**1**	**25**	**1,000,000**	**1,000,000**
1,000,000	50	3	3,000,000	50,000,000
3,000,000	1	25	3,000,000	3,000,000

Grouping

The PSO to PME ETL includes a new grouping feature that breaks the device-topic pairs that the ETL job processes into groups. Grouping processes a subset of all device-topic pairs (or tags) concurrently each time the job runs, thereby increasing the concurrent action within the job and improving performance.

Grouping is enabled by selecting **Process item groups across multiple job runs** (on the **Advanced** tab.)

How grouping groups and processes device-topic pairs is determined by the following settings:

Advanced tab:

- Max Data Request Per Group
- Max Groups Per Job Run

PSO Extract Task settings:

- Data Request Record Size
- Maximum Number of Requests

For information on how to use the grouping settings, see "PSO to PME ETL job performance" on page 400 .

PSO to PME ETL job performance

NOTE: The following settings do not represent a recommendation for production environments due to the numerous variables involved when approximating them.

They simply show the details of an in-house test system that was used to show the effect of these settings in a test environment.

They may be used as starting points for the application engineer when determining how to configure jobs in the field.

The application engineer should determine appropriate settings for each job based on observations of job execution time and other factors.

Testing Environment and Setup

Power SCADA Operation – Server 2012 with 4GB of RAM, 2 Processors 3.46 GHz

- Added CM4000 meters with 70 trend tags logging 15 minutes intervals.
- For the 35K trend tags: 500 CM4000 meters
- For the 105K trend tags: 1500 CM4000 meters.

All the CM4000s were in memory mode. Outside of just logging the trend tags, the SCADA project was not doing anything else.

Power Monitoring Expert – Server 2012 with 4GB of RAM, 2 Processors 3.46 GHz

Test execution

In these tests, the number of requests was set to cover 1 day worth of data. For 35,000 tags, that equals 3,360,000 rows of data. For 105,000 tags, that equals 10,080,000 rows of data.

The ETL task for 35,000 tags was configured to make 35 requests with each request containing up to 100,000 records.

The ETL task for 105,000 tags was configured to make 30 requests with each requests containing up to 1 million records.

Due to the 4GB of RAM available on the virtual machines, the maximum number of records inserted into SQL had to be set to 10,000 records. If the number of records were higher, SQL insertion performance could be affected and the ETL task would stop and write a message to its log files.

For these tests, the ETL job was run again right after it finished. For the 35,000 tag test, it ran 2 to 3 times and for the 105,000 tag test, it ran 6 to 7 times.

Using a value of 1 hour for the 'sleep time between executions' job setting, it would take 2 to 3 hours to catch up for the 35K tag scenario, and 6 to 7 hours to catch up in the 105,000 tags scenario.

NOTE: The grouping tests that were conducted were not done under load. 2.5 GB of RAM was dedicated to SQL Server. If other tasks were occurring on the server, then it is very likely the ETL job would take longer to execute. Since systems vary so much, use the settings listed here as a starting point, not as a recommendation. Application engineers should calibrate PSO to PME ETL performance on each system based on observations such as job execution time and other factors.

Test 1 Grouping Settings – 35,000 tags (recorded every 15 minutes for 3,360,00 records per day):

Setting	Value
PSO Extract Task > Data Request Record Size	100,000
PSO Extract Task > Maximum Number of Requests	35
PME Load Task > Enable Limit on Records per Insert	True
PME Load Task > Maximum records per insert	10,000
Advanced > Grouping Options > Max Data Request Per Group	7
Advanced > Grouping Options > Max Groups Per Job Run	5

Test 2 Grouping Settings – 105,000 tags (recorded every 15 minutes for 10,080,000 records per day):

Setting	Value
PSO Extract Task > Data Request Record Size	1,000,000
PSO Extract Task > Maximum Number of Requests	30
PME Load Task > Enable Limit on Records per Insert	True
PME Load Task > Maximum records per insert	10,000
Advanced > Grouping Options > Max Data Request Per Group	6
Advanced > Grouping Options > Max Groups Per Job Run	5

Recommendations

In general, running ETL jobs on servers with more RAM can have a positive effect on performance.

The time between execution can be set accordingly. If you want to run the ETL tasks more frequently, the time in between the job execution can be set lower. However, this can lead to the ETL task running and requesting data when no new data is available in Power SCADA Operation.

Background information

Internally, the ETL job processes all available data for each device-topic pair before moving on to the next pair. This operation is based on the position counter for each device-topic pair, and the max number of records per request.

Example: A device-topic pair records data every 15 minutes, each pair would log approximately 70,000 records every 2 years:

4 records/hour * 24 hours/day * 365 days/year * 2 years = 70,080 records

When running the job for the first time every device-topic pair will be starting from the default position of 2 years ago relative to job creation time. This can also be changed via the 'Initial Position Counter Time Stamp' task setting.

Assuming the job is configured as follows: 1 million records per request, and only 1 request, then 1 million records will fit 14 device-topic pairs each time the job runs.

(1,000,000 / 70,080 = 14.27)

If you increase the number of requests allowed per job to 3,000,000 then 42 device-topic pairs (each having 2 years worth of data) could be extracted each time the job runs. Once the job progresses forward (closer to the current time), then the expected number of records per device-topic pair gets smaller, and thus the number of pairs that fit into 1,000,000 records increases.

So the first few times the job is run, it is advantageous to configure it to run more often than it will once it catches up to the current time. For example, when running the job as a service, set 'sleep times between executions' to be 30 seconds or lower -- if appropriate for this PSO installation. Once the job progresses closer to the current time for all device-topic pairs, then the 'sleep time between executions' could be set back to 900 seconds (15 minutes).

You can tell how far along the job is by checking the Positions tab when editing a job. Timestamps for each pair are listed there.

The appropriate choice for 'sleep time between executions' during the initial runs will depend on how much data is in the system and other variables. If there is less than 2 years of data available in the system, then it will help to adjust the 'Initial Position Counter Time Stamp' task setting forward in time. This will mean that more device-topic pairs would fit into the allotted 1,000,000 records per request.

Configuring the PSO to PME transform task

NOTE: For most PSO to PME ETL jobs, the transform task is not needed.

Configure the **Intervalize Data Transform Task** after you add it to the ETL job. Click the transform task to display the configurable settings. Click a setting name to change its value. Some settings are configured by entering a value, while others are configured by selecting an option from a drop down list.

After you complete the extract task configuration, click **Apply** to save the ETL job without exiting the job, or click **OK** to save and exit the ETL job.

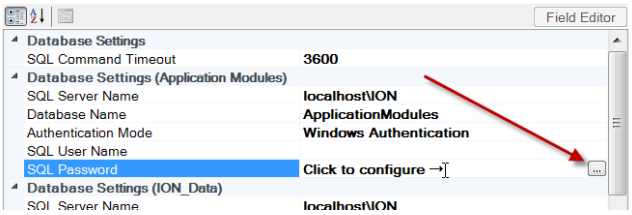
Setting Name	Description	Setting Parameters / Recommended Values
Transform		
Intervalization Method	The method for converting the values from an irregular interval to a regular interval.	LastKnownValue
Intervalize to present time	When set to True, the data is intervalized up to the current system time. If set to false, the data is intervalized up to the most recent data point.	**False**
Target Reading Interval	Data is intervalized to a reading interval specified in this field.	FifteenMinutes

Configuring the PSO to PME load task

Configure **Power Monitoring Expert Bulk Data Load Task** after you add it to the ETL job. Click the load task to display its configurable settings. Click a setting name to change its value. Some settings are configured by entering a value, while others are configured by selecting an option from a drop down list.

After you complete the extract task configuration, click **Apply** to save the ETL job without exiting the job, or click **OK** to save and exit the ETL job.

Setting Name	Description	Setting Parameters / Recommended Values
Database Settings		
SQL Command Timeout	The wait time (in seconds) before stopping the attempt to execute a SQL command and generating an error.	Default: **3600**
Database Settings (Application Modules / ION_Data / ION_Network)		
SQL Server Name	The name and optional instance of the SQL Server to connect to.	Default: **localhost\ION**
Database Name	The name of the target database.	**Database > Default values:** Application Modules > ApplicationModules ION_Data > ION_Data ION_Network > ION_Network
Authentication Mode	Authentication mode to connect to the database.	Windows Authentication (default) SQL Server Authentication
SQL User Name	The SQL Server Authentication Mode user name.	

Setting Name	Description	Setting Parameters / Recommended Values
SQL Password	The SQL Server Authentication Mode password	Click the field to display the Browse button. Click the button to enter your password. 
Group Name Settings		
Add the 'group Name' prefix to Sources if Needed	When set to 'True', the task adds a group name prefix to all sources that do not already have one. When set to 'False', a group name prefix will not be added.	Default: **False**
Group Name	The name provided in this setting is used as the Group Name prefix setting described above.	If the previous setting is 'False', this setting does not need to be filled in.
Load Options		
Disable in-memory table constraints	When True, constraints are disabled when building up an in-memory table prior to inserting.	Default: **True** **NOTE:** In some cases when True, this can improve performance.
Enable Limit on records per insert	When True, the Maximum record per insert setting is applied.	Default: **False**
Maximum records per insert	The maximum number of records passed to any one PME stored procedure call. The load task can break inbound data into batches and invoke the stored procedure for each batch.	Default: **10000** **NOTE:** The value is used only when Enable Limit on records per insert is True.
Mapping Options - Source and Quantity End Names		
Populate Button - Automatically Set Quantity 'End Names' to 'Start Names'	When True, all quantity End Names will be filled in and given the same value as the Start Name column. When False, all quantity End Names will be left blank.	False

Setting Name	Description	Setting Parameters / Recommended Values
Populate Button - Automatically Set Source 'End Names' to 'Start Names'	When True, all source End Names will be filled in and given the same value as the Start Name column. When False, all quantity End Names will be left blank.	False
Null Values		
Allow Null Values	When set to 'False' the task ignores any null values. When set to 'True', the null values are inserted into the database.	Set to 'False'.
Recorders and Channels		
Enable Recorder and Channel Creation	When set to 'False', the task does not create recorders and channels while inserting data.	The default setting is 'False' to prevent Log Inserter from creating unwanted downstream devices in the database. If the setting is 'True' and you add a device to PME with the same name as a pre-existing ETL source, Log Inserter will create unwanted downstream devices. **NOTE:** Some reports – such as Energy Cost Report and Load Profile Report – use Recorder and Channel information when retrieving data from PME. If loading data into PME for the purpose of viewing it in one of these reports, set this to 'True'.
Set the IsCurrentConfiguration Flag to False for New Channels	Indicates whether new channels are marked as non-current (True), or current (False.)	Default: **True**
Source And Quantity Creation Settings		
Enable Quantity Creation	When set to 'False' the setting disables creating quantities if they are not already in the database.	Set to 'False'.
Enable Source Creation in ION_Data	When set to 'True', the setting enables the creation of sources that are not already in the ION_Data database.	Set to 'True'.
Enable Source Creation in ION_Network	When set to 'True', the setting enables the creation of sources that are not already in the ION_Network database.	Set to 'True'.
Source Namespace Settings		

Setting Name	Description	Setting Parameters / Recommended Values
Source Namespace Override	Namespace given to all sources that do not have a namespace or that are created during the Load Task.	IONEnterprise
Source Type Settings		
Override Source Type	When set to 'True', enables the use of the Source Type Override value when creating sources.	Set to 'True'.
Source Type Override	The source type to use when creating sources.	presumed downstream device.
Verbose Logging		
Capture Timing Information	When True, additional timing information is logged in the trace log file.	Default: **False**

After the ETL tasks are configured, map the extracted data sources to the target data store. See "Configuring PSO to PME mappings" on page 406 for details.

Configuring PSO to PME mappings

⚠ WARNING

INACCURATE DATA RESULTS

- Do not incorrectly configure the software or the devices.
- Do not base your maintenance or service actions solely on messages and information displayed by the software.
- Do not rely solely on software messages and reports to determine if the system is functioning correctly or meeting all applicable standards and requirements.
- Consider the implications of unanticipated transmission delays or failures of communications links.

Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

Use the **Mappings** tab to map PSO devices and topics to PME sources and quantities. Depending on the size and the design of your system, loading sources may take some time to scan both systems.

To map PSO devices and topics to PME sources and quantities:

1. In a PSO to PME ETL job that has the extract, transform and load tasks configured, click the **Mappings** tab.

2. Click **Load Sources**.

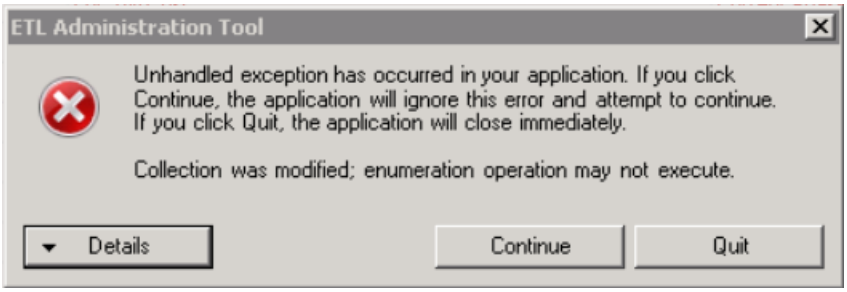
   NOTE: You might need to restart the Schneider Electric CoreServiceHost service and reset Internet Information Services (IIS) on the Power SCADA Server if the device that you add to Power SCADA Operation does not appear in the **Mappings** pane after you click **Load Sources**. Performing a restart could affect all other web applications and Power SCADA components running on the server.

   Please review the state of the system before performing a service restart or IIS reset.

   After you click **Load Sources**, the Client status details appear at the lower left of the dialog and display the number of tags loaded and folders searched.

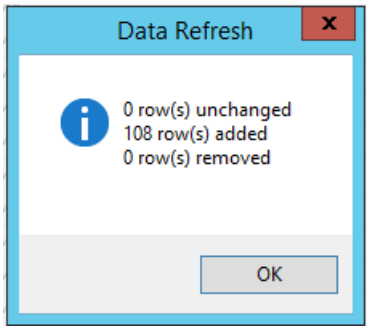
   NOTE: If you have a large system with a lot of devices, wait until **Load Sources** re-enables prior to using the mappings grid.

   If you get the following error:



   Click **Quit**, restart the ETL tool, and then click **Load Sources** again. Wait until the **Load Sources** button re-enables before using the tool.

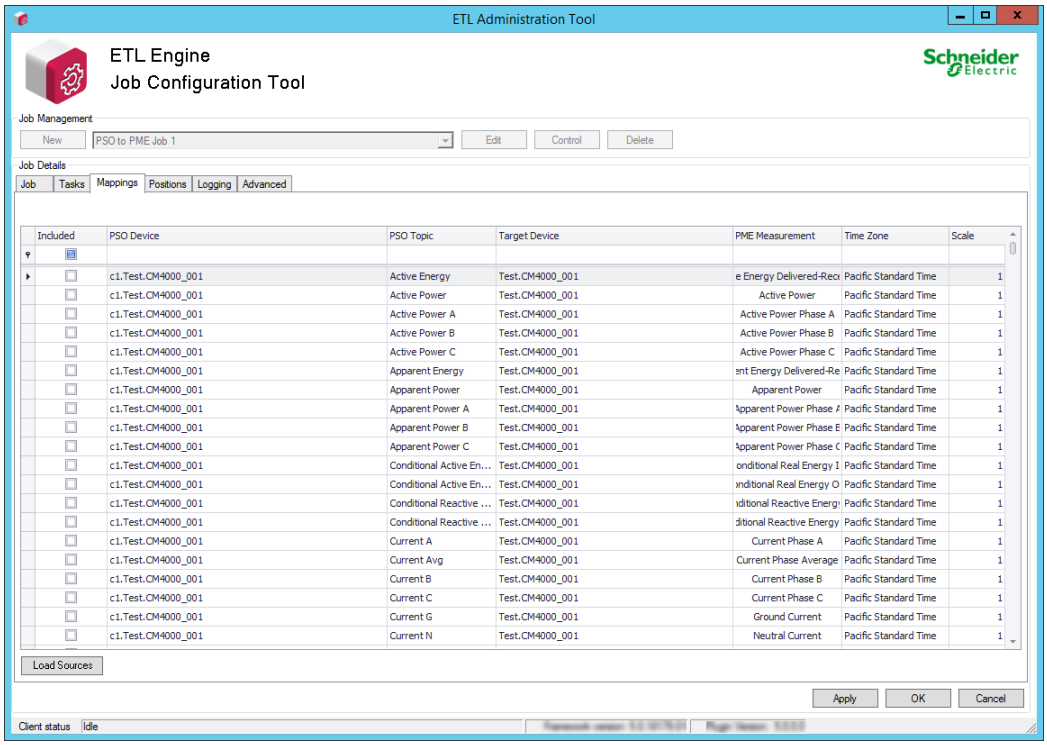
   When the devices are loaded the Data Refresh dialog appears:



   This dialog displays the number of devices loaded, and the number of rows added and removed.

3. Click **OK** to close the Data Refresh dialog.

   If everything is set up correctly, the ETL polls the PSO Server to retrieve a list of available PSO device-topic tags, polls the PME Server to retrieve a list of sources and measurements, and then displays them as suggested PME source-quantity pairs. For example:



4. Review the PSO to PME mappings.
5. (Optional) Edit the default mappings if they do not meet your needs. See"Editing the PME source" on page 409 for details.
6. For each PSO device that needs to be available in dashboards or reports, click **Included** to mark the rows that will be included in the ETL processing.

   TIP: You can select multiple source-quantity pair rows that you want to include in the PSO to PME ETL job, right-click and then click **Include Selected Mappings**. See"Highlighting rows" on page 411 for details on how to use **Mappings**.

7. (Optional) Set the **Time Zone** and **Scale** values.

   NOTE: Time zone and scale are standard ETL values. Typically you will not need to edit these values.

8. After you map all the PSO device-topic pairs to PME source-quantity pairs that you want to include in the ETL job, click **Apply** to save the job.

You can continue to configure the PSO to PME ETL job by setting the logs and adding position counters; see "Position counters" on page 421. Or you can run the ETL job; see "Grant database permissions for the ETL job to run as a service" on page 415

Editing PSO to PME mappings

Load Sources automatically pairs PSO devices and topics to PME sources and quantities. You can edit the default pairings by changing the PME source and the PME quantity.

Editing the PME source

You can edit the PME source associated with a PSO device-topic pair by selecting a different PME source, or by creating a new one.

To edit the PME source:

1. In the **Mappings** grid, click the cell of the PME source you want to edit.
2. Assign a new or different new PME source to the PSO device-topic pair:

   To assign a new PME source:

   a. Type the name of the new PME source. Click **Create New**.

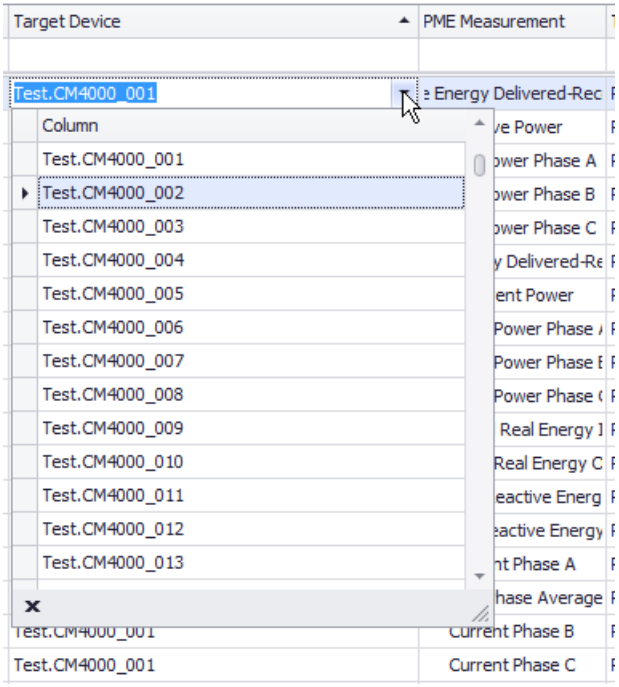
   > **NOTE**: The **PME Source** name has to match the Power Monitoring Expert device naming convention of *Group.DeviceName* with no special characters, such as: \ | + = : ; < > ? or , .

   If you do not follow this device naming convention:

   - In Web Reports you will have to find your ETL'd devices in the "other" group.
   - In Dashboards, the devices will be grouped under "Devices".

   To assign a different PME source:

   a. Select the **PME Source** from a drop-down menu of existing devices.



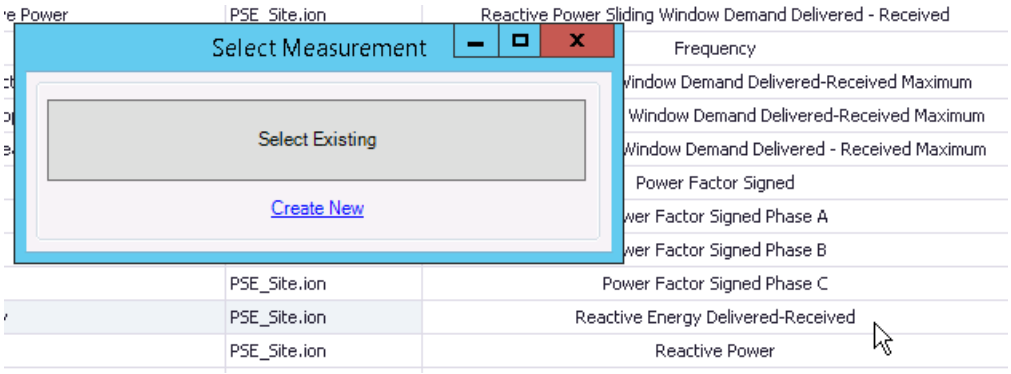
Editing the PME quantity

You can edit the PME quantity associated with a PSO device-topic pair by selecting a different PME quantity, or by creating a new one.

To assign a non-default PME quantity to a PSO device-topic pair:

1. In the Mappings grid, click the PME Quantity cell that you want to rename.

   The Select Measurement dialog appears:

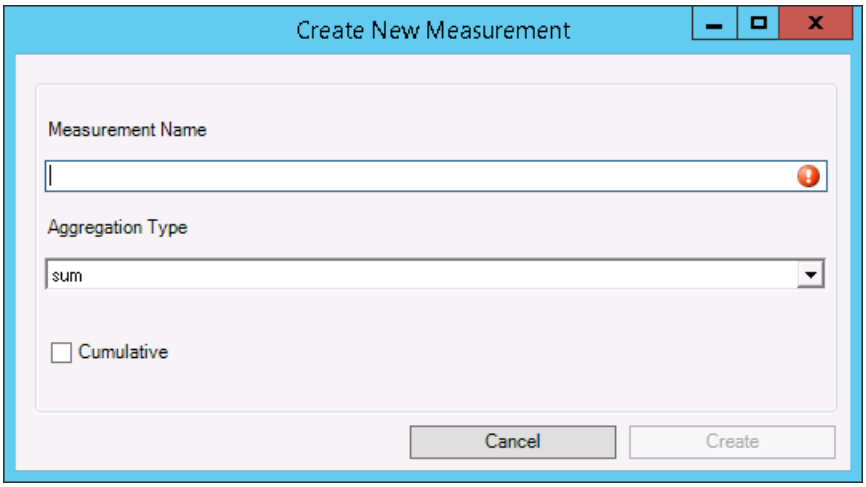


2. Assign a different or new PME quantity to the PSO device-topic pair :

   To assign a new PME quantity:

   a. Click **Create New**.

      The Create New Measurement dialog appears:

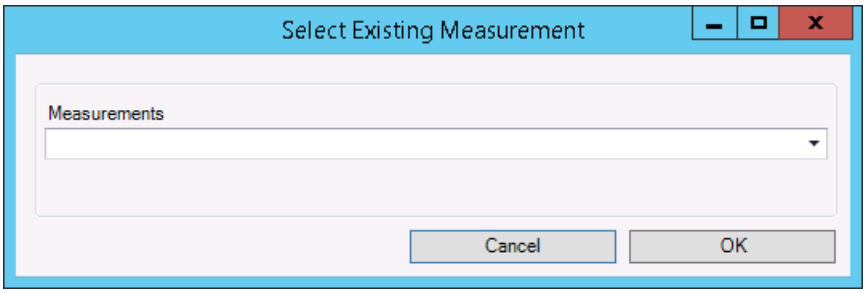


   b. Enter the new measurement name, set its values, and then click **Create**.

   To assign a different PME quantity:

   a. Click **Select Existing**.

      The Select Existing Measurement dialog appears:



   b. From the **Measurements** drop down, select an existing PME quantity, and then click **OK**.

Continue mapping the ETL job.

Tips for working with mappings

Loading sources can return thousands of rows. To help you manage a large result set, the ETL Administration Tool includes several features to help you search, filter, and update loaded sources.

Highlighting rows

Highlighting a source row lets you work with that source. When you highlight a row you can copy, include or exclude the row from the ETL job, or perform a batch edit on the row.

To highlight a row:

1. Click the row.

To highlight successive rows:

1. Click the row.
2. Press **Shift** and click another row.

To highlight non-successive rows:

1. Press **Ctrl** and click the desired rows.

To highlight all rows:

1. Press **Ctrl + A**.

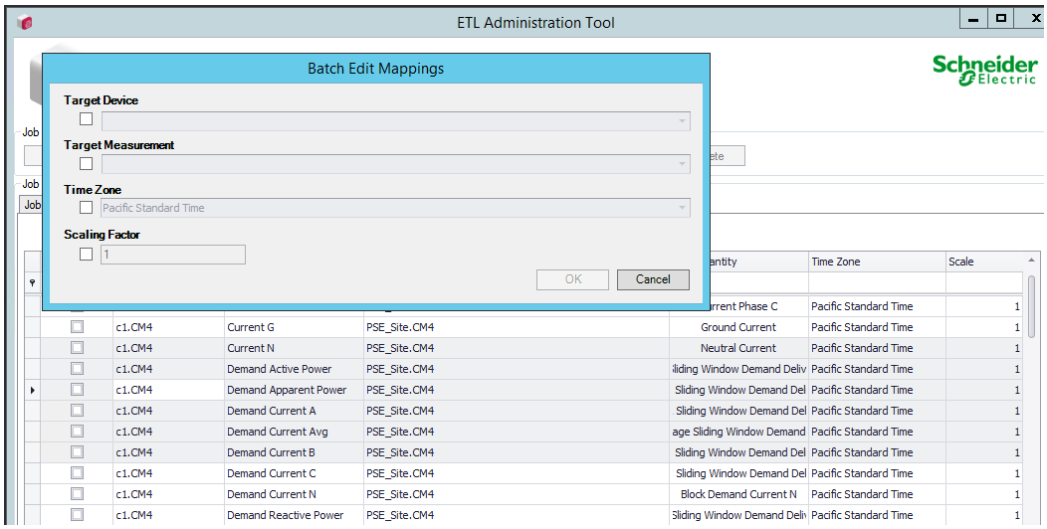
Batch Edits

A batch edit lets you update all highlighted rows at once.

To perform a batch edit:

1. In the **Mappings** pane highlight the rows you want to edit.
2. Right-click and click **Batch Edit**.

   The Batch Edit Mappings dialog appears.



3. Complete all applicable fields in the dialog as needed.

   NOTE: You have to complete the **Target Device** and **Target Measurement** fields before you can select Included for the row.

4. While the rows are still highlighted, right-click and click **Include Selected Mapping(s)**. The **Included** check box is checked for the selected rows and these devices are included in the job.

5. Click **OK**, and then click **Apply** to save the changes to the job. The Batch Edit values appear for the selected rows.

Sorting contents by column

To sort contents by column:

1. Right-click a column heading and from the sort menu choose to sort column contents by ascending or descending order.

Searching by column

To search by column:

1. Click in the Auto Filter Row (search field below a column heading.)
2. Begin typing characters. Column contents appear based on the search criteria you enter. Note that characters are not case sensitive.

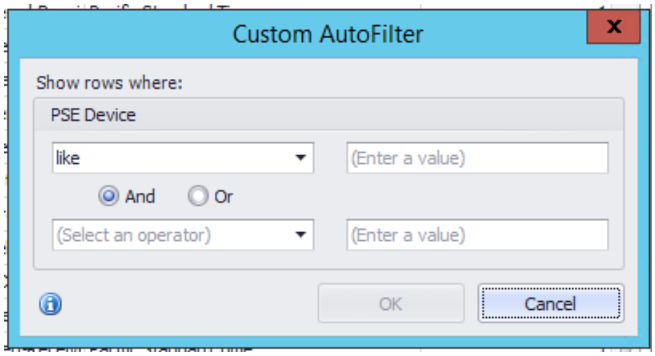
PSO Device	PSO Topic
	Demand
c1.Test.CM4000_001	Demand Active Power
c1.Test.CM4000_001	Demand Apparent P...
c1.Test.CM4000_001	Demand Current A
c1.Test.CM4000_001	Demand Current Avg
c1.Test.CM4000_001	Demand Current B
c1.Test.CM4000_001	Demand Current C
c1.Test.CM4000_001	Demand Current N
c1.Test.CM4000_001	Demand Reactive Po...
c1.Test.CM4000_001	Peak Demand Active...
c1.Test.CM4000_001	Peak Demand Appar...
c1.Test.CM4000_001	Peak Demand Curre...
c1.Test.CM4000_001	Peak Demand Curre...
c1.Test.CM4000_001	Peak Demand Curre...
c1.Test.CM4000_001	Peak Demand Curre...
c1.Test.CM4000_001	Peak Demand Curre...
c1.Test.CM4000_001	Peak Demand Reacti...
c1.Test.CM4000_002	Demand Active Power

Filtering content by column

To filter the contents by column:

1. Click the filter symbol to the right of the column heading, and then choose (Custom), (Blanks), (Non blanks), Checked, Unchecked, or a specific device.
2. If you choose (Custom), you can define a unique filter, based on your input, in the Custom

AutoFilter dialog. Complete the fields in the dialog and then click **OK**.



Filtering content using the Filter Editor

To filter the contents using the Filter Editor:

1. Right-click the column header you want to filter and them click **Filter Editor**.

   You must complete the Target Device and Target Measurement fields before you can select Included for the row.

2. Click an operator or enter a filter value.
3. Click **Apply**.

   The sources are filtered based on the filtering criteria you enter.

4. Click **OK** to return to the **Mappings** tab.

Copying and pasting devices

You can select and copy one or more devices PSO and paste that data into a document, such as a text editor or a spreadsheet.

To copy and paste devices into a document:

1. In the **Mappings** tab select one or more device rows.
2. Press **CTRL+C** or right-click and click **Copy**.
3. Open your document and place the cursor where you want to paste.
4. Press **CTRL+V** or right-click and click **Paste**.

   The device data appears in the document.

Testing your ETL job

When you complete configuring the extract and load tasks and the mapping of source and quantity pairs, you can test your ETL job by running it once using the ETL Administration Tool.

TIP: Add the HTML File Load Task to an ETL job to help validate and troubleshoot the extract task portion of the ETL job.

1. Select your job name from the list in the **Job Management** field and click **Control**.

   The **Job Control** tab opens.

2. Click **Run Once** to test your job.

The Job Execution Complete dialog opens and a high-level message indicates whether or not your job succeeded.

If the job is not successful:

a. Click **Open Log** to open the folder containing the log files.

b. Open the error.log file and scroll to the last set of **Job Logger Started** and **Job Logger Finished** entries at the bottom of the file. The error details are contained within these two entries for the latest job run.

For example, one of the most common errors is that the connection string for the ION_ data database is incorrectly specified for the extract task.

3. Click **OK** to close the dialog.
4. Click **OK** to close the **Job Control** tab.

Running an ETL job

You can run an ETL job by:

- Running the job as a Windows service. This is the default method.
- Running the job as a batch file using Windows Task Scheduler.
- Running the job from the command line.

This section describes how to schedule an existing ETL job to run in an unattended and repeated fashion, or by running the ETL job from the command line.

Running the ETL job as a Windows Service

This is the default method and is appropriate for most installations. The ETL Administration Tool provides a built-in way to create a Windows service from the ETL job. The ETL job runs and then waits for a configurable duration before it runs again. You can define the amount of time between each run.

Advantages:

- The ETL Administration Tool simplifies setting up the service.
- The ETL service appears in the Windows Services console.

  This is desirable in cases where the administrator is already managing other services for related systems.

Disadvantages:

- Very few scheduling features are available. The only configurable option in terms of scheduling is the sleep time between executions.
- The service does not perform a true periodic execution of the job.

  Each single run of the job takes a variable amount of time depending on many factors, such as how much data it needs to process, or how much activity is taking place on the server during

the job run. The sleep time is fixed. This means that for each run the start time for the job drifts. This may be undesirable in situations where you want the job to start at a specific time each day.

Running the ETL job as a service may not be optimal when you have many different ETL jobs. The service remains in memory even when the underlying job is sleeping.

Grant database permissions for the ETL job to run as a service

By default, when an ETL job is run as a service it runs under the NT AUTHORITY\SYSTEM Windows user account.

With SQL Server 2012 and later, the NT AUTHORITY user does not have database permissions. If an ETL job is run using the NT AUTHORITY user, the ETL job cannot connect to the Power Monitoring Expert database and the job is not successful.

For the ETL job to succeed, you must first grant database permissions to this user.

To grant database permissions to the NT AUTHORITY user, log in to SQL Server Management Studio as an administrator and run the following script:

```
USE [ION_Data]
GO
CREATE USER [NT AUTHORITY\SYSTEM] FOR LOGIN [NT AUTHORITY\SYSTEM]
GO
EXEC sp_addrolemember N'db_owner', N’NT AUTHORITY\SYSTEM'
GO
USE [ION_Network]
GO
CREATE USER [NT AUTHORITY\SYSTEM] FOR LOGIN [NT AUTHORITY\SYSTEM]
GO
EXEC sp_addrolemember N'db_owner', N'NT AUTHORITY\SYSTEM'
GO
```

NOTE: If security concerns limit you from using the default NT AUTHORITY user, create a dedicated Windows user to run the ETL job as a service:

1. Create a Windows user. Note that if the ETL is installed to its default location, C:\Program Files\..., the Windows user must have Administrator access.
2. Set the ETL job to run as a service under the new Windows user.
3. Log in to SQL Server Management Studio as an administrator and run the above script, substituting NT AUTHORITY\SYSTEM with the new Windows user.

Running the ETL job as a batch file using Windows Task Scheduler

Create a batch file and use Windows Task Scheduler to schedule when the ETL job runs. The batch file contains the command line entry to run the job.

Advantage:

- The scheduled task performs a true periodic execution of the job. Windows Task Scheduler allows you to schedule the job to start at precise times.

Disadvantages:

- It is more difficult to set up than the services option because you must create and test the batch file before scheduling it. There is currently no built-in feature to create a batch file automatically for the job.
- You must have a fully configured ETL job that runs successfully. Follow these steps if you want to run the ETL job using the Windows Task Scheduler.

To create the batch file:

1. Use your favorite text editor and create a command line batch file (.bat) that executes the ETL job once (using the `-SingleRun` option).
2. To determine what to put in your batch file:
   - Try running your ETL job from the command line. Open a command prompt, and change directories to your ETL Engine's bin folder.
   - Optional: View the list of available ETL Engine commands by entering the following:

     ```
     ETLEngine.exe -?
     ```

   - Run your ETL job once using the following as an example, and substitute your ETL job's name:

     ```
     ETLEngine.exe -SingleRun -job enterjobnamehere
     ```

     NOTE: Your job name is listed on the Job tab in the ETL Administration Tool. If your job name contains spaces, enclose the job name in double quotes on the command line.
3. After you determine the correct command line arguments to use, create a batch file containing the full command.

Schedule that batch file for repeated execution using Windows Task Scheduler. Refer to the Windows Task Scheduler documentation for details.

Running the ETL job using the command line

The syntax for running an ETL job from a command line is:

```
ETLEngine.exe [OPTION] -Job JobName
```

Where `OPTION` can be one of the following values:

-?, -help	Prints a help message and exits.
-SingleRun	Performs one single run of processing and exits.
-Service	Registers a specific job as a Windows service.
-UnregService	Unregisters the service associated with a specific job.
-WaitSingleRun	Useful for debugging only.

Manage ETL jobs

You can set up logging to help manage ETL jobs. You can also switch between ETL jobs, change the order of ETL tasks, and remove ETL tasks from an ETL job.

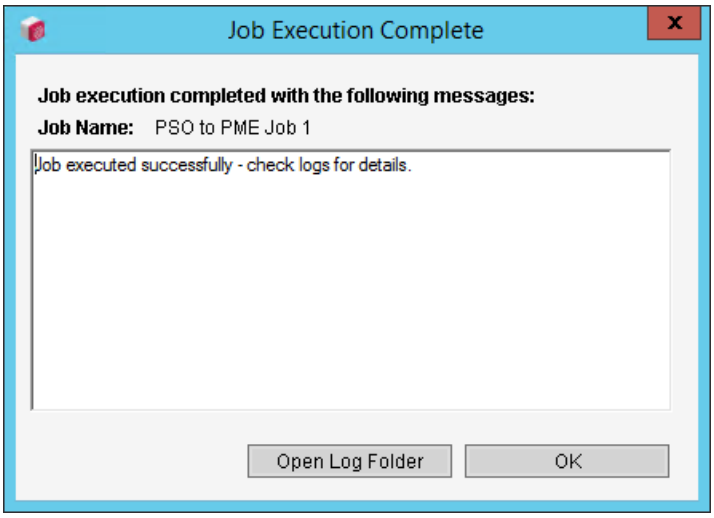
Enabling ETL logging

Logging lets you enable the various logs where ETL writes the information regarding the status of your ETL job. These logs can assist in tracking down the cause of an unsuccessful ETL job.

To enable the ETL logs:

1. Open the ETL Administration Tool.
2. In the **Job Management** list click the applicable ETL job and then click **Edit**.
3. Click the **Logging** tab. The Logging panel appears.
4. For Trace Log, Error Log, and Customer Log, click **Enabled** as required.
5. (Optional) Provide the location for the log file in the Log File field, or leave at the default location.
6. (Optional) Set the **Maximum Log File Size** and **Maximum Log Files** for each log, or leave at the default settings.
7. (Optional) Select the **Enabled** check box for Email Notifications and complete the fields for: **To Email Address**, **From Email Address**, and **SMTP Server Address**.
8. Click **OK** when finished to exit the job.

After you run an ETL job, the Job Execution Complete dialog appears. You can click **Open Log Folder** to review the log files. For example:



Confirming the ETL job

If the ETL Administration Tool returns a **Job execution failed** message, click **Open Log Folder** to open the error log. Locate the timestamp that corresponds to your job and review the log. Based on this information, make the appropriate changes to the job and run the job again.

Cloning an ETL job

When creating a new job in ETL, you can clone an existing ETL job.

To clone an ETL job:

1. In the **Job Management** list click the applicable ETL job and then click **Edit**.
2. In the **Job** panel, change the name to define the new ETL job.
3. Click the **Task** tab and then edit the new ETL job as necessary.
4. Click **Apply** or **OK**.

The ETL job saves with the new name. Sources and quantities are carried over from the original ETL job. It is recommended that you clear the mappings from the cloned ETL job.

Renaming an ETL job

1. In the **Job Management** list click an existing ETL job.
2. Click **Edit**.
3. In the **Job** panel, change the name to define the new ETL job.
4. Click **OK**.
5. (Optional) In the **Job Management** list, click the original ETL job and then click **Delete**.

Removing a task from an ETL job

1. In the **Job Management** list click the applicable ETL job and then click **Edit**.
2. Click the **Tasks** tab.
3. Highlight the task that you want to remove from the left pane.
4. Click **Delete**  .
5. Click **OK** to save and exit the job.

Switching between ETL jobs

1. Click **OK** at the bottom right to save and exit the current job.
2. In the **Job Management** list select an ETL job and click **Edit**.

Synchronizing devices

Power SCADA Operation with Advanced Reporting and Dashboardsrequires a PSO to PME ETL job to transfer device data logging from PSO to the Power Monitoring Expert database. If devices change in PSO, the ETL job does not automatically recognize the change and the two systems are no longer synchronized.

The following scenarios describe what happens to the integrated systems when you make changes to PSO sources (after the initial PSO to PME ETL job is configured and running):

Scenario: Adding devices in PSO

When you add a device in PSO, the source and its data are not automatically available in PME.

You must edit the ETL job to map the new device to PME and then run the ETL job. See"Prerequisites" on page 420 for details.

Scenario: Editing sources in PSO

When you edit a device name or change the device's measurement logging in PSO:

- The old source name continues to exist in PME
- If you edit the ETL job to include the new source name and then run the job, a new historical source (with the edited name) is created in PME and the source's logged data is associated with the new source name. However, the data that was logged before the source name change will continue to be associated with the old source name.

You must edit the ETL job to map the edited source name or measurement to PME and then run the ETL job. See"Prerequisites" on page 420 for details. You might also need to update the database to associate the historical source data with the edited source.

Scenario: Deleting a source in PSO

When you delete a device in PSO:

- The old source name and its historical data continue to exist in Power Monitoring Expert
- If you edit the ETL job to include the new source name and then run the job, a new historical source (with the edited name) is created in PME and the source's logged data is associated with the new source name. However, the data that was logged before the source name change will continue to be associated with the old source name.

You must edit the ETL job to remove deleted source from the ETL job and then run the ETL job. See"Prerequisites" on page 420 for details.

You might also need to update the database to associate the historical source data with the deleted source.

Scenario: Upgrading a source in PSO

When you upgrade a source in PSO, the data transfer for the source continues seamlessly as long as the Trend Tag Name and I/O Device Name remain the same. Even if the Communication Protocol or I/O Device Address changes the Variable Tag and the Trend Tag will remain unchanged.

Limitations

The following scenarios are not supported by the PSO to PME ETL:

- Moving a device from PSO to PME
- Viewing historical data from ETL sources in Vista or Diagrams in PME.

The following scenarios require that you to contact technical support:

- Renaming an ETL source in PME.
- Deleting an ETL source in PME

Verifying PSO sources in PME

Before you can update an ETL job to synchronize PSO and PME devices, it is recommended that you obtain a list of the device names that are already in the system. Doing so will prevent device naming conflicts and will also help you to edit the ETL job.

You cannot see PME source names that were created by ETL in PME Management Console; you must run a SQL query to return this information.

NOTE: You can also look for PSO devices and their associated PME sources by creating and generating a tabular report in PME or by creating a dashboard that uses a trend from the PSO source. See *Power Monitoring Expert Help* for more information.

To match PSO devices to PME sources:

1. In Microsoft SQL Server Management Studio, click **New Query**.
2. To return all sources in alphabetical order, enter and execute the following query:

   ```
   SELECT * FROM ION_Data.dbo.vSource
   ```

   To sort the sources beginning with the most recently added device, enter and execute the following query:

   ```
   SELECT * FROM ION_Data.dbo.vSource ORDER BY SourceID DESC
   ```

   Alternatively, you can click **Use list of sources (allows aliasing)** and click **Recommended Pairs**. Choosing this option returns the sources and quantities available at the time you clicked **Recommended Pairs**. To discover additional sources and quantities, you must click **Recommended Pairs** again.

   TIP: Copy the entire query result and paste it into Microsoft Excel to more easily sort and filter the devices.

3. (Optional) Use this information to"Prerequisites" on page 420.

Editing a PSO to PME ETL job

Edit a PSO to PME ETL when you:

- Add a new device in PSO
- Edit an existing device name in PSO
- Change device logging in a PSO device that is mapped to PME.

NOTE: Sources cannot be deleted from Power Monitoring Expert Management Console. To delete PSO devices from PME you must contact technical support.

Prerequisites

In order to edit an ETL job, you must know:

- The name of the PSO device you want to map. (If you are adding a new PSO device to the ETL job.)

- The name of the mapped PME source. (If you are editing a PSO device name or measurement in the ETL job.)

To edit a PSO to PME ETL job:

1. Open ETL (PSO to PME).
2. (Optional) If the ETL job that you want to edit is registered to run as a service, select the job you want to stop and click **Control** and then **Stop** to stop the service. Then click **OK** to close the Job Control page.
3. From the list of jobs, select the job that you want to edit and then click **Edit**.
4. In the **Mappings** pane, click **Load Sources**.

   The ETL tool displays how many new records were added. The newly named PSO source device and measurement appears in the grid in the Source Tag column, along with a suggested Target Device.
5. Filter the list of devices to locate the PSO Source Tag row containing the PSO source you want to map.

   See "Highlighting rows" on page 411 for details on how to filter loaded sources.
6. Review the Target Device value in the same row.

   The Target Device value is the PME source under which the PSO data will be loaded.
7. If the Target Device name and measurement matches the expected name in PME, click **Include Selected Mappings**. If the Target Device or Target Measurement does not match the expected value, edit the Target Device field and then click **Include Selected Mappings**

   NOTE: If you want to log data for the PSO device continuously under the same PME device as it did prior to the PSO tag renaming, map the new PSO tag to the same PME target device. This may be useful when viewing historical data in PME over the time span of the PSO tag renaming.
8. Click **Apply**.

Resetting and resending data

You can use the ETL Administration Tool to restore lost Power Monitoring Expert (PME) data from Power SCADA Operation (PSO).

Position counters

Position counters keep track of the data that is extracted from the source system and then loaded into PME. Each PSO Source tag specified in ETL has a position counter associated with it. The position counter represents a timestamp of the most recent data point loaded for each source tag. When an ETL job is run, only data after this timestamp value is extracted from the source system's Trend log.

After you run an ETL job you can check the position counters to verify that the job ran as expected. If the position counter value for a given source-quantity pair continues to increase after each job run, then you can be confident that the job picked up new data for that pair on the most recent run.

If you need to re-extract previously extracted data, or if you want to load data after a specific date, you can manually update the position counter and then run the ETL job.

Prerequisites

- The name of the mapped PSO device. (If you are editing a PSO device name or measurement in the ETL job.)

To reset or resend data for mapped Trend logs:

1. Open PSO to PME ETL.
2. From the list of jobs, select the job that you want to edit and then click **Edit**.

   NOTE: To restore lost data, you can either edit or clone the existing PSO to PME ETL job.

   TIP: Editing the original PSO to PME ETL job might be easier to manage when working with only a few source tags. You can also save a copy of your job and work with that.
3. Click the **Positions** tab.
4. For each device whose data you want to recover, enter a specific value in **Initial Value** to set all position counters. Use the same format as shown in the existing records/rows on the positions tab, or in the "Initial Value" text box.
5. Click **Initialize**. Mapped Trend logs appear with associated timestamp data for each.

   You should see a row for each pair selected in the **Mappings** tab. The Key is a long string that represents the pair.

   Now, the next time you run ETL, only data after the given timestamp is loaded.
6. Run the ETL job.

   The Target Device value is the PME source under which the PSO data will be loaded.
7. (Optional) Verify the data transfer. See "Prerequisites" on page 422 for details.

Verifying PSO data transfer to PME

After a PSO to PME ETL job runs successfully, you can check the database to verify that the data transfer occurred for a PSO source device.

Prerequisites

- You need to know the SourceID of the PME source in question. You can find this information in the ION_Data database. See"Verifying PSO sources in PME" on page 420 for details on how to obtain this value.

To verify PSO data transfer to PME:

1. In Microsoft SQL Server Management Studio, click **New Query**.
2. Select a device whose data you want to verify by obtaining its SourceID.

3. Enter and execute the following query:

   ```
   SELECT * FROM ION_Data.dbo.DataLog2 WHERE SourceID =
   DeviceSourceID
   ```

   The query returns all data for all quantities under that SourceID.

Add the WebReach Server Parameter

To add PME server properties to the Citect.ini file:

1. Open the Computer Setup Editor: In Power SCADA Studio, click **Projects > Setup Wizard** drop down, and then click **Setup Editor**.
2. Add a new Section named “Applications” and a parameter named “WebReachServer” with a value of either a server_name or the IP address of the PME server.
3. Save and then compile the project.

Get the Advanced Reports Report ID

1. In SQL Server Management Studio, select the ION_Network database.
2. Create and run the following query:

   ```
   SELECT TOP 1000
   [ReportID], [DisplayName], [SubFolder], [Name]
   FROM [ION_Network].[dbo].[RPT_Report]
   ```

This SQL script displays the names and IDs of all the reports that have been configured and saved.

NOTE: It is possible to have two reports with the same name, but the [SubFolder] designation will make them unique.

Get the device name and test the WebReach Diagrams URL

To display the diagram, determine the device name using SQL and test the URL in a browser.

To determine the device name:

1. In SQL Server Management Studio, select the ION_Network database.
2. Create and run the following query:

   ```
   SELECT Name FROM dbo.device
   ```

3. Find the device name that you want.

To test the diagram display:

1. Open a browser window and enter the following URL:

   ```
   http://<servername>/Ion/default.aspx?dgm=OPEN_TEMPLATE_
   DIAGRAM&node=<devicename>
   ```

2. Replace <*devicename*> with the name you found in the previous step, and <*servername*> with the name of the Power Monitoring Expert server.

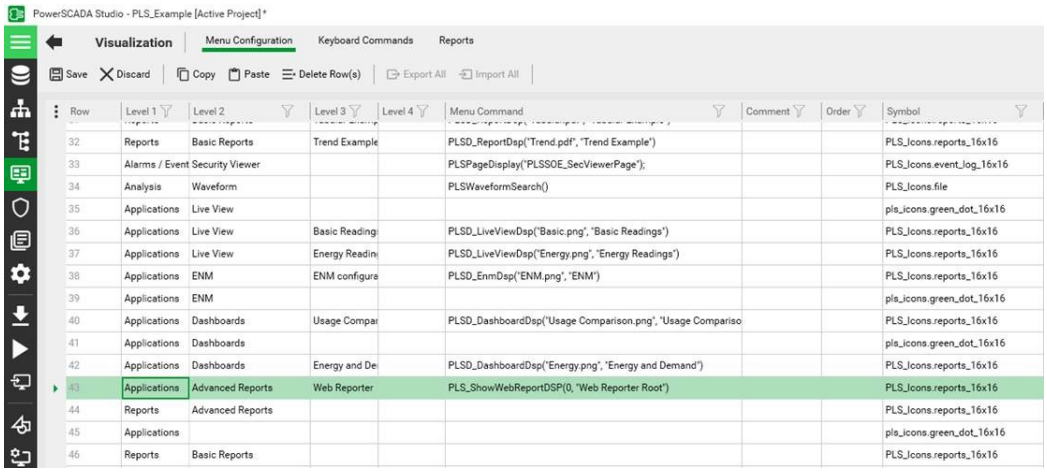
   For example, a real URL would look like:

```
http://10.168.94.77/Ion/default.aspx?dgm=OPEN_TEMPLATE_
DIAGRAM&node=TVD.7650
```

The device diagram should display in the browser window, and you should be able to navigate around the diagram, per normal WebReach function.

Add the Advanced Reports Root Page Menu Item

1. From the Power SCADA Studio, click **System > Menu Configuration**.



2. Enter the call to the `ShowWebReportDsp` function (found in the `PLS_Applications.ci` file), with 0 entered for the ReportID and the page title.
3. If you have multiple reports configured, and want to display a different report for different devices, repeat this procedure for each button, with the correct ReportID.
4. Save, compile, and run the project to test the functionality.

NOTE: Carefully consider how and where you display the web report root. Power SCADA Operation has native reports, and the customer should see as consistent interface an as possible. When you modify the menu, you can maintain the experience of a single HMI if you remove certain native links (in the PLS_Example project) and if you are selective about where the root is displayed.

About the PLS_ShowWebReportDsp Cicode: In this step, you call the `PLS_ShowWebReportDsp` function from a menu configuration. This function is part of the Cicode in the `PLS_Applications.ci` file, which is packaged with this document. The code is shown below for reference.

```
FUNCTION PLS_ShowWebReportDsp(INT iReportID, STRING sTitle = "")
IF ("" = sTitle) THEN sTitle = "Reporting"; END
STRING sUrl = _PLS_Apps_BuildWebReporterUrl(iReportID);
IF ("" <> sUrl ) THEN
PLS_WebDsp(sUrl, sTitle, "PLS_ShowWebReportDsp",
IntToStr(iReportID) + ",^"" + sTitle + "^"");
END
END
```

Important things to note about this code:

- `iReportID` is the unique identification number of the desired report, determined in the step below.
- `sTitle` is the title of the page.
- The function builds a URL based on the provided Host in the Citect.ini.
- It will also dynamically create the object with PLS_WebDsp so there is no need for an AN object name reference.

NOTE: After you are on the Web Reporter page, you stay logged in until you close the browser or refresh the page.

Add Advanced Reports page menu items

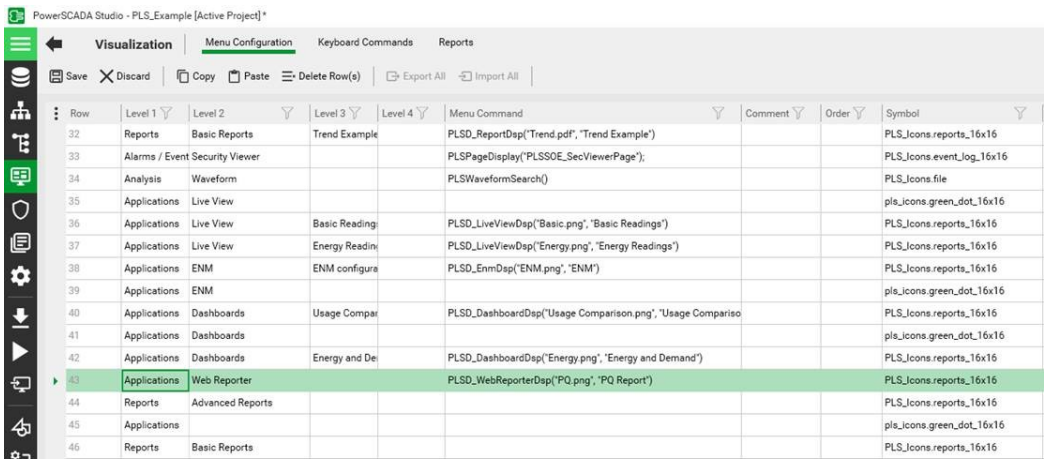
You can add menu items that navigate directly to a saved Advanced Report, such as a report for Energy Analysis over the last two months.

NOTE: Carefully consider how and where you display the web report root. Power SCADA Operation has native reports, and the customer should see as consistent interface an as possible. When you modify the menu, you can maintain the experience of a single HMI if you remove certain native links (in the PLS_Example project) and if you are selective about where the root is displayed.

NOTE: After you are on the Web Reporter page, you stay logged in until you close the browser or refresh the page.

To add specific Advanced Reports page menu items:

1. In Power SCADA Studio, click **Visualization**  **> Menu Configuration**.



   To determine the ReportID that you enter see "Get the Advanced Reports Report ID" on page 423. You can repeat this procedure to add menu items for each of the saved reports that you want to display from the Power SCADA Operation navigation menus.
2. Enter the call to the ShowWebReportDsp function (found in the PLS_Applications.ci file), with 0 entered for the ReportID and the page title.
3. If you have multiple reports configured, and want to display a different report for different

devices, repeat this procedure for each button, with the correct ReportID.

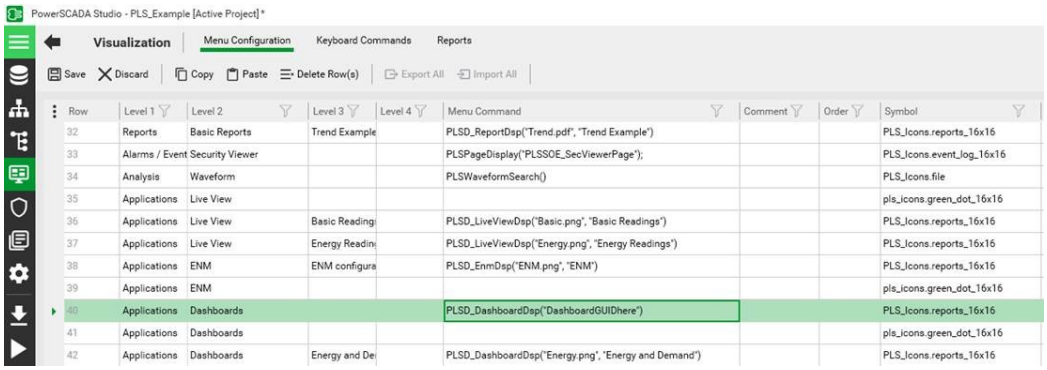
4. Save, compile, and run the project to test the functionality.

Add the Dashboards Page Menu Item

To add a menu item to launch a specific dashboard:

1. In Power SCADA Studio, click **Visualization** > **Menu Configuration**.
2. Open a browser and navigate to the dashboard you want to add to the menu item. The specific dashboard GUID is in the URL: https://localhost/web/#Dashboards/lib/**DashboardGUIDhere**
3. Enter the call to the PLS_ShowDashboardDsp function (found in the PLS_Applications.ci file) and the page title.

   The following image illustrates the settings for "with optional dashboard GUID," which loads a specific dashboard:



4. If you want to display multiple dashboards, repeat these steps for each menu item, using the correct dashboard GUID.
5. Save and compile. Then run the project to test functionality.

To add a menu item to launch the Dashboards home page:

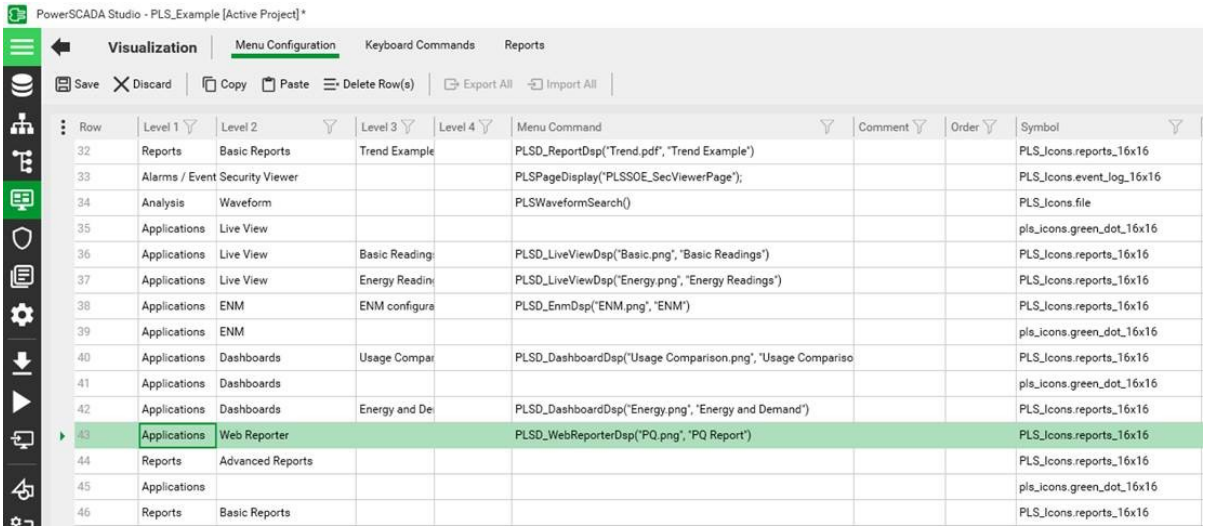
1. In Power SCADA Studio, click **Visualization** > **Menu Configuration**.
2. Enter the call to the PLS_ShowDashboardDsp function (found in the PLS_Applications.ci file) with a custom page name and an empty dashboard ID: `PLS_ShowDashboardDsp("", "CustomPageName")`, or with no custom parameters to use the default page name: `PLS_ShowDashboardDsp()`.

Finish Advanced Reports Page Menu Items

Revisit each project menu configuration item previously created for displaying Advanced Reports pages. Do not update the menu item created for the Advanced Reports Root Page.

For each item, update the menu command with the respective Report ID. For more information see "Get the Advanced Reports Report ID" on page 423.

For example:



Row	Level 1	Level 2	Level 3	Level 4	Menu Command	Comment	Order	Symbol
32	Reports	Basic Reports	Trend Example		PLSD_ReportDsp("Trend.pdf", "Trend Example")			PLS_Icons.reports_16x16
33	Alarms / Event	Security Viewer			PLSPageDisplay("PLSSOE_SecViewerPage");			PLS_Icons.event_log_16x16
34	Analysis	Waveform			PLSWaveformSearch()			PLS_Icons.file
35	Applications	Live View						pls_icons.green_dot_16x16
36	Applications	Live View	Basic Reading:		PLSD_LiveViewDsp("Basic.png", "Basic Readings")			PLS_Icons.reports_16x16
37	Applications	Live View	Energy Readin		PLSD_LiveViewDsp("Energy.png", "Energy Readings")			PLS_Icons.reports_16x16
38	Applications	ENM	ENM configura		PLSD_EnmDsp("ENM.png", "ENM")			PLS_Icons.reports_16x16
39	Applications	ENM						pls_icons.green_dot_16x16
40	Applications	Dashboards	Usage Compar		PLSD_DashboardDsp("Usage Comparison.png", "Usage Compariso			PLS_Icons.reports_16x16
41	Applications	Dashboards						pls_icons.green_dot_16x16
42	Applications	Dashboards	Energy and De		PLSD_DashboardDsp("Energy.png", "Energy and Demand")			PLS_Icons.reports_16x16
43	Applications	Web Reporter			PLSD_WebReporterDsp("PQ.png", "PQ Report")			PLS_Icons.reports_16x16
44	Reports	Advanced Reports						PLS_Icons.reports_16x16
45	Applications							pls_icons.green_dot_16x16
46	Reports	Basic Reports						PLS_Icons.reports_16x16

Add a Menu Item to Launch a Web Diagram

Use this procedure to access a WebDiagram by invoking Cicode from your project menu. Alternately, the following procedure describes how to add a WebDiagram view in your genie equipment popup:

"Add Web Diagrams to Equipment Popups" on page 428

To add a page to the project that will display a given WebDiagram

1. Create a new menu configuration item that calls the PLS_WebReachDsp Cicode explained below.
2. Enter the call to the PLS_WebReachDsp function (found in the PLS_Applications.ci file), with the slideshow (if desired), and the page title.

About the PLS_WebReachDsp Cicode

In the following step, you will call the WebReachDsp function from a button. This function is part of the Cicode in the PLS_Include.ci file, which is packaged with this document. The code is shown here for reference:

```
FUNCTION PLS_WebReachDsp(STRING sDeviceName, STRING sTitle = "")
STRING sPage = PLS_GetWebReachURL(sDeviceName);
IF ("" = sPage) THEN RETURN; END

IF ("" = sTitle) THEN sTitle = sDeviceName; END
PLS_WebDsp(sPage, sTitle);
END
```

There are some important things to note about this code:

- `sDeviceName` is the name of the device, determined in the step above.
- `sTitle` is the title of the page

If the diagram does not display, try the following troubleshooting steps:

Enter the URL of the diagram directly into a browser window; verify that it launches. The URL is: http://[servername]/ION/default.aspx?dgm=OPEN_TEMPLATE_DIAGRAM&node=[device name]

If this does not work, verify that the WebReachServer is correct in your Citect.ini, and the diagram appears correctly in WebReach.

The steps above should resolve most issues. One last option is to test by putting the web browser in a window on the calling page.

Finish WebDiagram Page Menu Items

Revisit each project menu configuration item previously created for displaying WebDiagram pages.

For each item update the menu command with the respective DeviceName. For more information on how to determine the device name, see "Get the device name and test the WebReach Diagrams URL" on page 423.

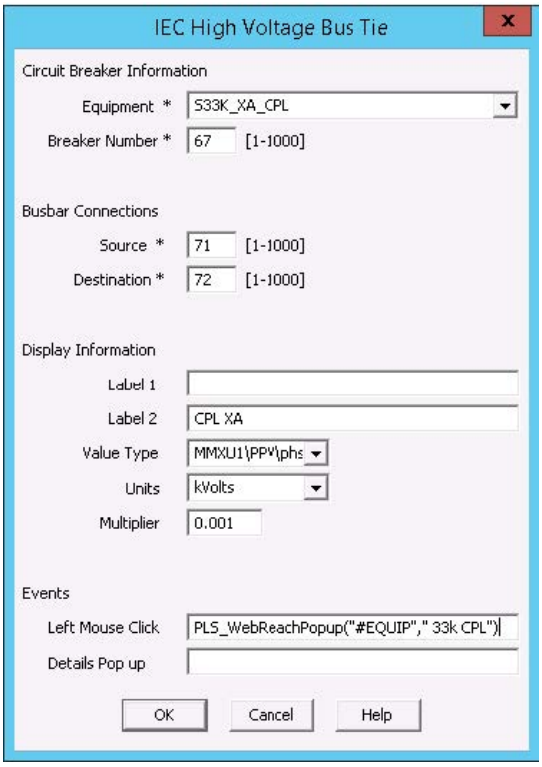
Add Web Diagrams to Equipment Popups

NOTE: This method only works when Power Monitoring Expert device names are identical to Power SCADA Operation equipment names.

To launch the diagram from a meter genie equipment page:

1. Open the Power SCADA Operation Graphics Builder and navigate to the page on which you want to insert the meter genie.
2. Click **Edit > Paste Genie**.
3. Under Library, click pls_meter and select the desired meter genie.
4. Near the bottom of the page, locate the **Events** fields.
5. In the **Details Pop Up** field, enter the PLS_WebReachPopup Cicode method.

Your Genie Properties dialog should resemble the the following:



NOTE: Unlike the other two button types (from a menu or popup page), you do not specify the sDevice name. Instead, you pass #EQUIP. This value is a property of the genie. This only works when the Power SCADA Operation equipment name is the same as the Power Monitoring Expert group.devicename.

The result is an equipment popup that contains a button that looks like this:

To test the WebReach URL:

1. Verify that the diagram launches, by entering the URL of the diagram in a browser.

   The URL is: `http://<servername>/ION/default.aspx?dgm=OPEN_TEMPLATE_DIAGRAM&node=<devicename>`

If this does not work, verify that the WebReachServer is correct in your citect.ini, and the diagram appears correctly in WebReach.

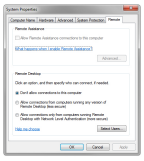
Configure the Power SCADA Anywhere Server

NOTICE
INOPERABLE SYSTEM Ensure that you have received Power SCADA training and understand the importance of the Power SCADA Operation productivity tools and workflows. **Failure to follow these instructions can result in overly complex projects, cost overruns, rework, and countless hours of support troubleshooting.**

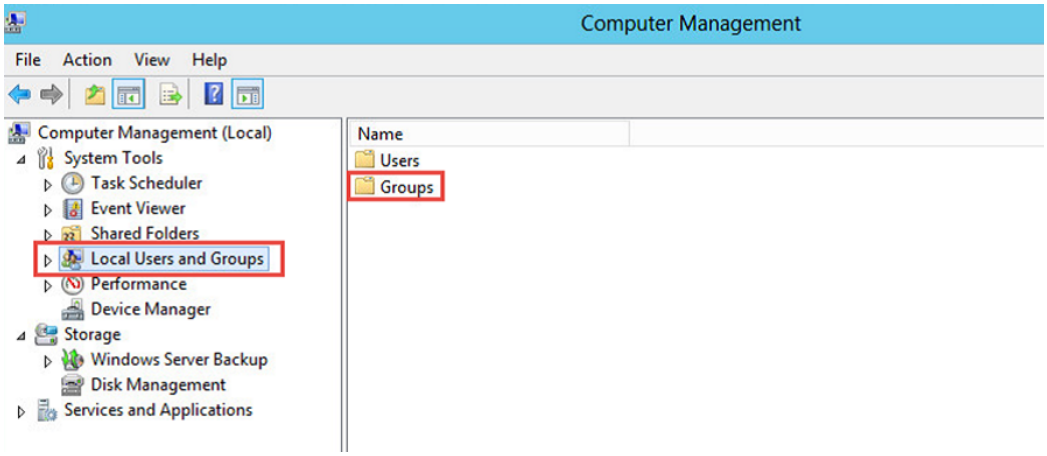
NOTE: Power SCADA Operation is build on Citect Studio and includes productivity tools that are designed and optimized to create the tags you need to configure power-based SCADA projects. If you have prior experience using Citect Studio, do not rely exclusively on Citect tools to build a power SCADA project.

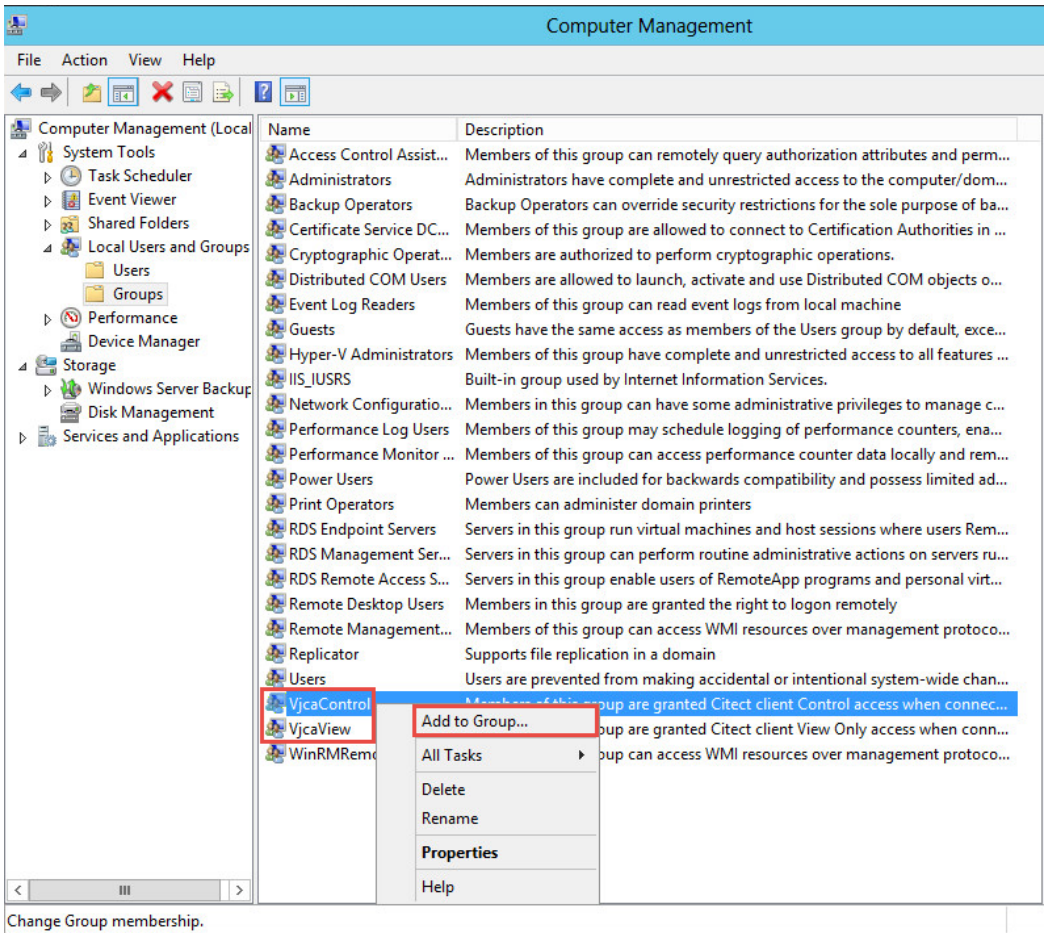
To configure the Power SCADA (Citect) Anywhere Server:

1. Configure Remote Desktop settings to allow remote access:
   a. From the Control Panel, open the System Properties window and click the **Remote** tab:

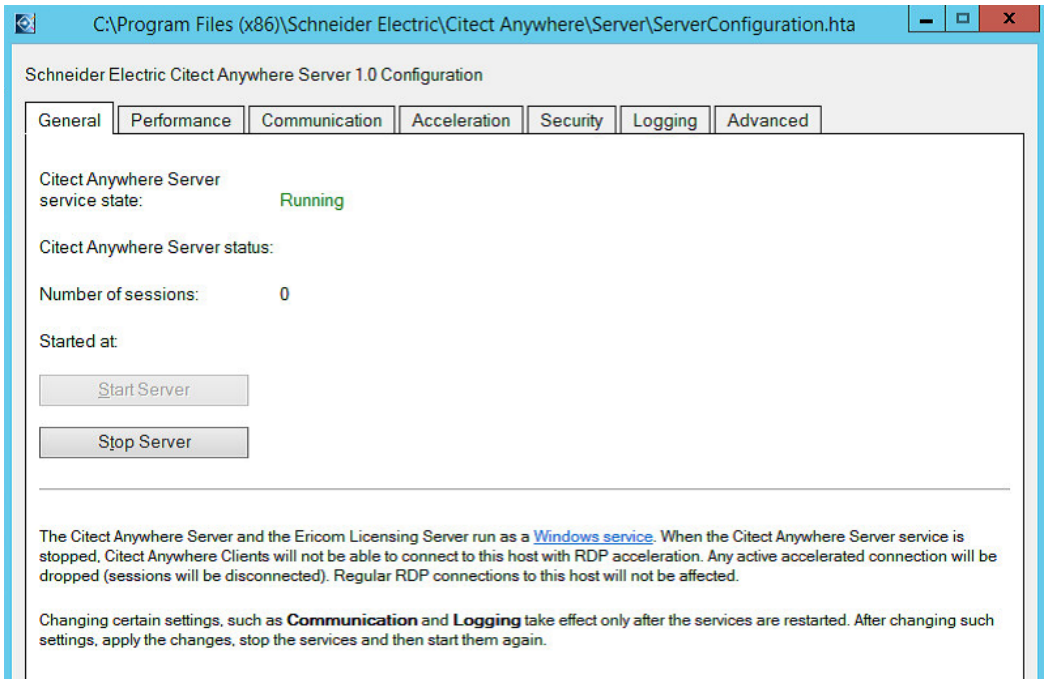


   b. Click **Allow Remote Assistance connections to this computer**.
   c. Click **Allow connections from computers running any version of Remote Desktop (less secure)**.
   d. Click **Select Users** to begin adding user accounts to the Remote Desktop Users group.
2. Access to the client type is granted through two special Windows user groups created by the installer on the computer where the Citect Anywhere Server is installed. You must add users to the VJCAControl and VJCAView groups manually using Administrative Tools > Computer Management:





3. Ensure that the Citect Anywhere service is started. To confirm this, use the Server-Configuration for Citect Anywhere:



If the server is stopped, click **Start Server**.

Connect to Power SCADA Anywhere

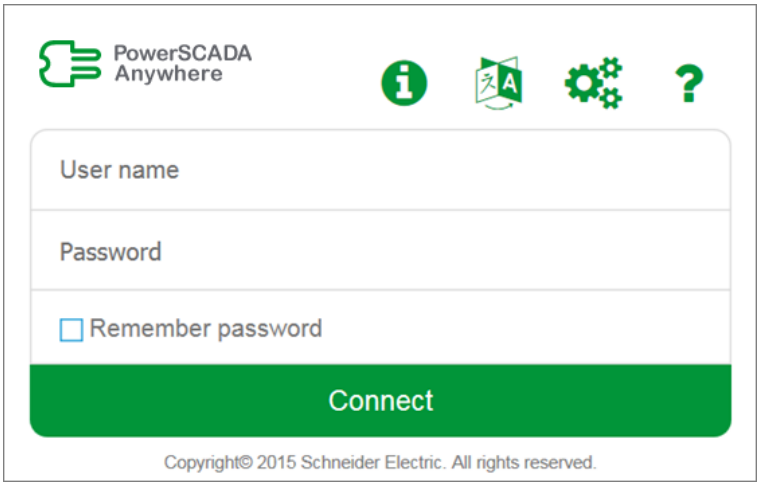
The following browsers are supported by Power SCADA Anywhere:

- Internet Explorer 10 and 11
- Microsoft Edge
- Google Chrome 33
- Safari 8 on Apple iOS

Connect to a Power SCADA Anywhere Server by navigating to the following web address in a supported browser:

http://<*VJCA Server Node Name or IP address*>:8080/

The logon screen appears.



Log in with Windows user credentials from the Citect Anywhere server. The user needs to belong to the VjcaView or VjcaControl group on the Citect Anywhere server.

EcoStruxure Web Services setup

NOTICE
INOPERABLE SYSTEM Ensure that you have received Power SCADA training and understand the importance of the Power SCADA Operation productivity tools and workflows. **Failure to follow these instructions can result in overly complex projects, cost overruns, rework, and countless hours of support troubleshooting.**

NOTE: Power SCADA Operation is build on Citect Studio and includes productivity tools that are designed and optimized to create the tags you need to configure power-based SCADA projects. If you have prior experience using Citect Studio, do not rely exclusively on Citect tools to build a power SCADA project.

This feature configures the Power SCADA Operation EcoStruxure Web Services (EWS) server. See "EcoStruxure Web Services (EWS)" on page 87 for a description of this server.

Do not confuse this information with the EWS server that was released with PowerSCADA Expert 7.40. That implementation is specific to the Citect core. It was developed only for real-time tag data acquisition. The implementation being released with this product also acquires historical data and alarms.

The fields are:

- **Alarm Acknowledgment Wait Period:** The amount of time allowed for Power SCADA Operation to process an alarm acknowledgment request. Choose a value that allows the system enough time to allow acknowledgments to be processed, while not so long as to delay processing.
- **Initial Alarm Request Length:** The number of days worth of alarm to request from Power SCADA Operation.
- **Max Request Size:** The number of alarms returned with one request. The default (1000 alarms) should be sufficient to maintain alarm data integrity (ensuring that all alarms are returned in each call), while also maintaining system performance.
- **Alarm Settle Time:** The number of seconds "grace period" to allow the Citect Alarm Server to finish inserting alarms that are in process at the time of the poll. If you set this too low, you could miss alarms. If you set it too high, it may take longer for alarms to come into EWS.
- **EWS/Citect User Association:** Use this block to manage user names and passwords. This provides EWS Digest Authentication for the user, permitting them to view data. However, for the user to be able to acknowledge alarms, the username/password must match a user-name/password added to the Power SCADA Operation project. When this user acknowledges an alarm through EWS, Citect verifies the credentials of the user and acknowledges the alarm under this user's identity.

To add a user:

1. Click **Add User**.
2. At the Add User screen, type an established Power SCADA Studio username and password.
3. Click **Test Citect Credentials** to verify the name and password.

When you enter a valid username and password, a message displays telling you they are valid.

Avoiding EWS Provider Timeouts

When attempting to retrieve large amounts of data from the EWS server, the provider call might timeout, resulting in an error. To correct this, you can temporarily increase the timeout period, which will allow the target application to receive the data. To do this, modify the key named *ProviderTimeoutInMinutes*, found in the EWS virtual directory, under Web.config.

Default location:

C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\Applications\EWS\Web.config

After the data is processed, edit this key to its original setting.

Time synchronization

Current time can be sent to the corresponding device by means of Set Time command or (in case of Sepam) by writing directly to the corresponding registers within the device. In addition to the manual procedure, this process can be scheduled to occur periodically (using Power SCADA Operation events).

Non-manual time synchronization causes the Set Time command to be sent automatically, based on a device state or event originating from within the device.

Automatic time synchronization applies only to Micrologic and PM devices and takes place based on the following rules:

- For Micrologic devices, the value of the top-most bit of the register 679 is examined (for both the Circuit Breaker Manager and the Chassis Manager). If the bit is equal to 1, it means that the device is out of sync and needs to be synchronized.
- For PM devices, an alarm 50700 (“Unary Power Up / Reset”) indicates that the device needs to be synchronized. In addition, bit 6 of register 3055 of the device is examined. If this bit is equal to 1, the device has a real-time clock; so automatic time synchronization should never take place.

Time zone settings

To interact with devices located in different time zones, the system converts any alarm/waveform timestamp as well as the actual time sent within the Set Time command from / to the local time zone. The Windows time zones database is used to take daylight saving time into account. Thus, time zone names must be taken directly from this database (case-insensitive), otherwise the system will default to the I/O Server’s local time zone. The Windows time zone database is in the Windows registry in HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones\. Examples of time zone names are:

- AUS Central Standard Time
- China Standard Time
- Pacific Standard Time (Mexico)

Device time zones can be specified on two levels:

1. Use section [ProtocolName.ClusterName.PortName.IODeviceName] to specify the time zone for a particular device.

   For example:

   [PLOGIC870.Cluster1.PM870_Port.PM870_Device1]
   Time zone = Singapore Standard Time

2. Use general section [POWERLOGICCORE] to specify the time zone for all devices.

   For example:

   [POWERLOGICCORE]
   Time zone = Mountain Standard Time

The device-specific time zone specification takes precedence. In other words, if both examples are present in the `Citect.ini` file, the PM870_Device1 would be located in “Singapore Standard Time” time zone, and all the other I/O devices in the project would be located in “Mountain Standard Time” time zone.

If there is no time zone specification, or if it does not match the time zone from Windows database, the device would be in the same time zone as the machine where the I/O Server is running; thus, no time conversion will occur.

If only the first of the above examples is present within the Citect.ini file, the PM870_Device1 would be located in “Singapore Standard Time,” and all the other devices would use the current local time zone.

OFS system time stamping

Power SCADA Operation provides the System Time Stamping method for the electrical distribution monitoring and control system.

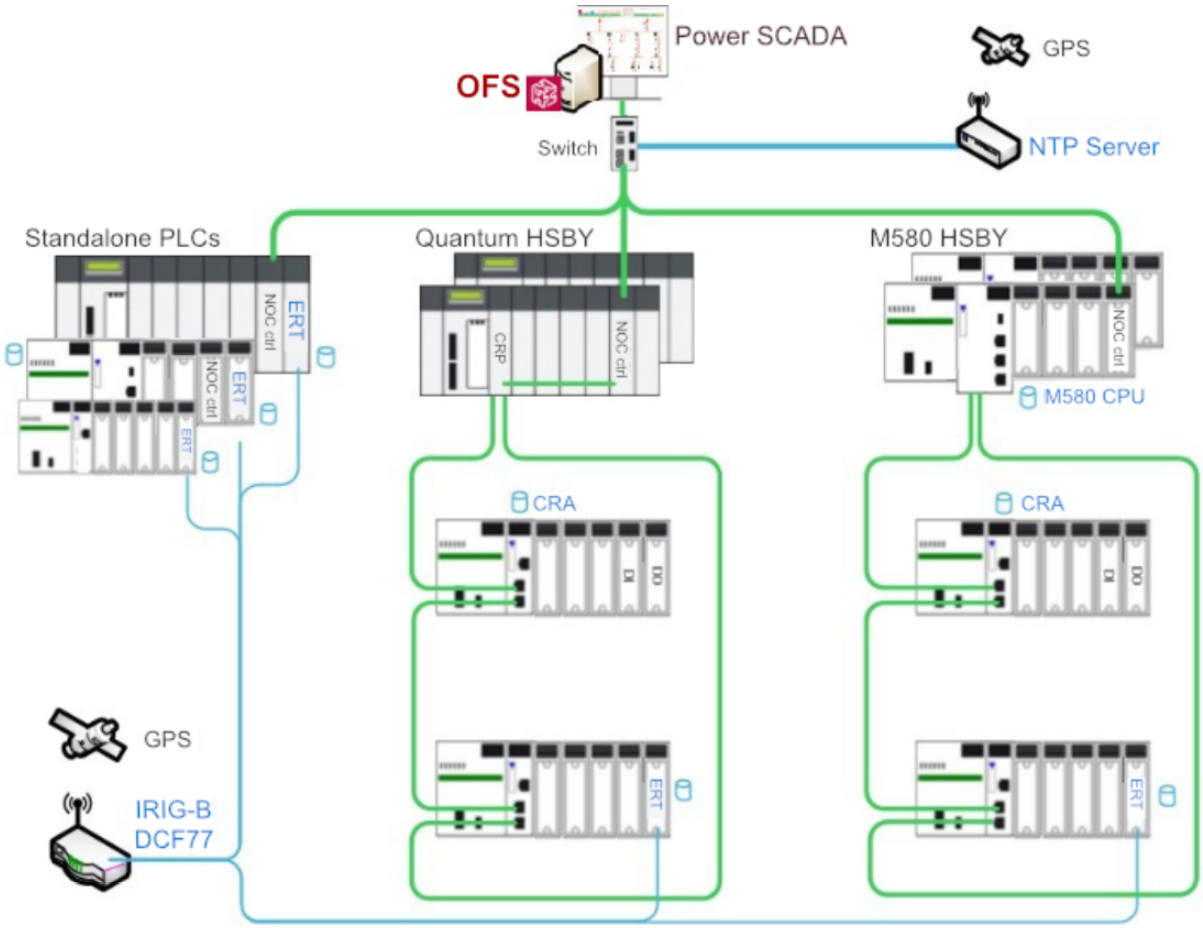
System Time Stamping helps the user analyze the source of abnormal behaviors in an automation system.

The benefits of the system time stamping mode are:

- No PAC programming required: All the time stamped events are managed and transferred automatically by OFS
- Direct communication between the time stamping modules and the client: The available communication bandwidth in the PAC is preserved
- Advanced diagnostic functions:
  - Signaling of uncertain SOE (sequence during which some events may be lost) to the client
  - Time quality information is associated with each time stamped event
- No loss of events in normal operating conditions:
  - An event buffer stores the events in each event source module. The event buffer behavior is configurable
  - Both rising and falling edge transitions can be stored for both discrete I/O and PAC internal variables
- Works with both a redundant hot-standby PAC and redundant SCADA

The current limitations of the system time stamping are:

- A communication path between OFS and the time stamping sources is required, so, routing is necessary in multi-layer architectures.
- 2 OPC servers (running for HMI and SCADA) cannot simultaneously access the same time stamping source. A reservation mechanism is implemented.
- No detection of transition edges; the event detection is processed only on both edges.



The following table describes the main features and differences between these two methods.

Process	System Time Stamping
1. Synchronize the time clock	ERT module is synchronized by IRIG-B/DCF77 link and x80CRA & M580 CPU are synchronized by the NTP server
2. Time stamping of events generation	I/O events are stamped by x80 ERT modules & CRA Internal variable values are stamped by the M580 CPU
3. Manage the time stamped events in PAC buffer	Events are managed and transferred to Power SCADA automatically by OFS
4. Transfer time stamped events from PAC to SCADA	Events are managed and transferred to Power SCADA automatically by OFS

System time stamping

System time stamping is an important feature of Power SCADA Operation. It helps the user analyze the source of abnormal behaviors in an automation system.

The benefits of the system time stamping mode are:

- No PAC programming required: All the time stamped events are managed and transferred automatically by OFS
- Direct communication between the time stamping modules and the client: The available communication bandwidth in the PAC is preserved

- Advanced diagnostic functions:
  - Signaling of uncertain SOE (sequence during which some events may be lost) to the client
  - Time quality information is associated with each time stamped event
- No loss of events in normal operating conditions:
  - An event buffer stores the events in each event source module. The event buffer behavior is configurable
  - Both rising and falling edge transitions can be stored for both discrete I/O and PAC internal variables
- Works with both a redundant hot-standby PAC and redundant SCADA

The current limitations of the system time stamping are:

- A communication path between OFS and the time stamping sources is required, so, routing is necessary in multi-layer architectures.
- 2 OPC servers (running for HMI and SCADA) cannot simultaneously access the same time stamping source. A reservation mechanism is implemented.
- No detection of transition edges; the event detection is processed only on both edges.

Competencies

Before configuring OFS system time stamping in Power SCADA Operation, you should have experience with the following Schneider Electric products:

Software:

- Unity Pro
- OFS configuration tool
- Power SCADA Operation
- Citect SCADA

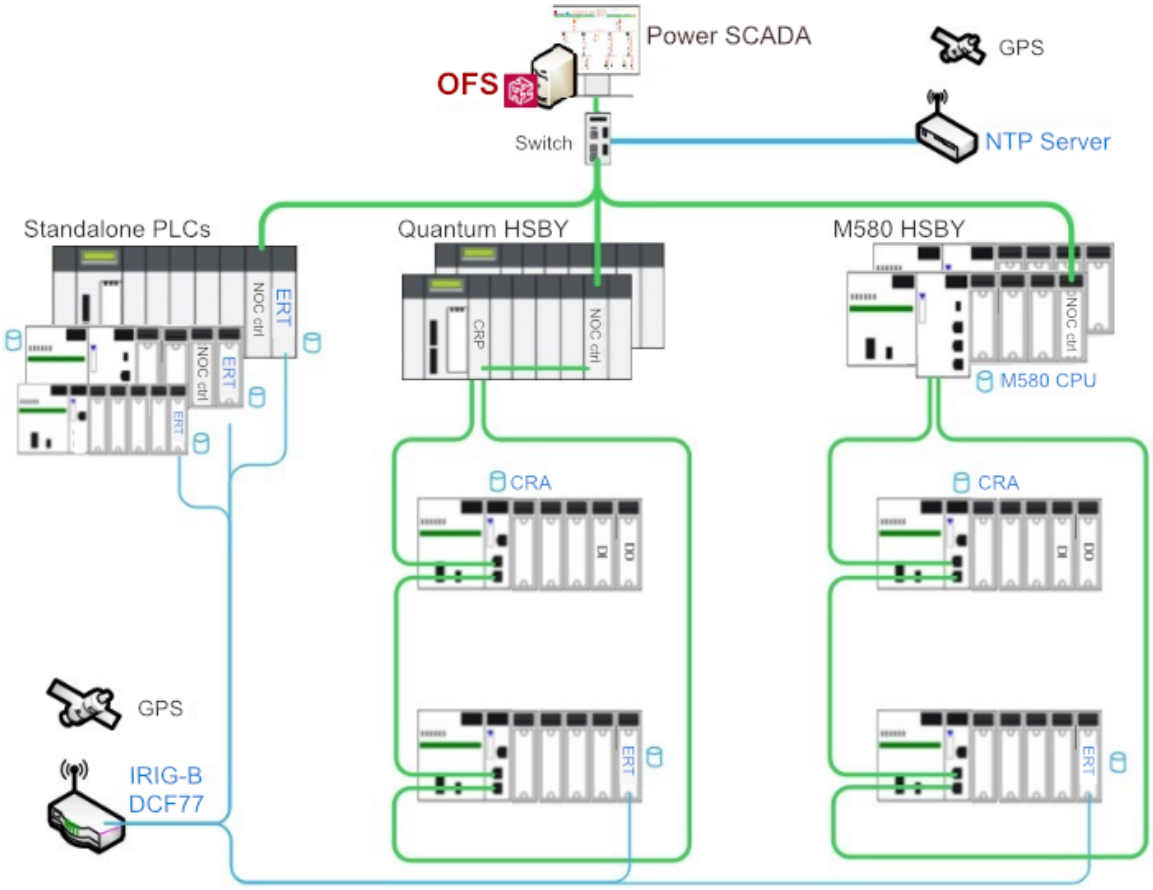
Hardware:

- Programmable Automation Controller (PAC) and Remote Input / Output (I/O) – Quantum, M340, and M580
- Ethernet module with routing capabilities
- ERT modules: M340/eX80 BMX ERT 1604 T

Selection

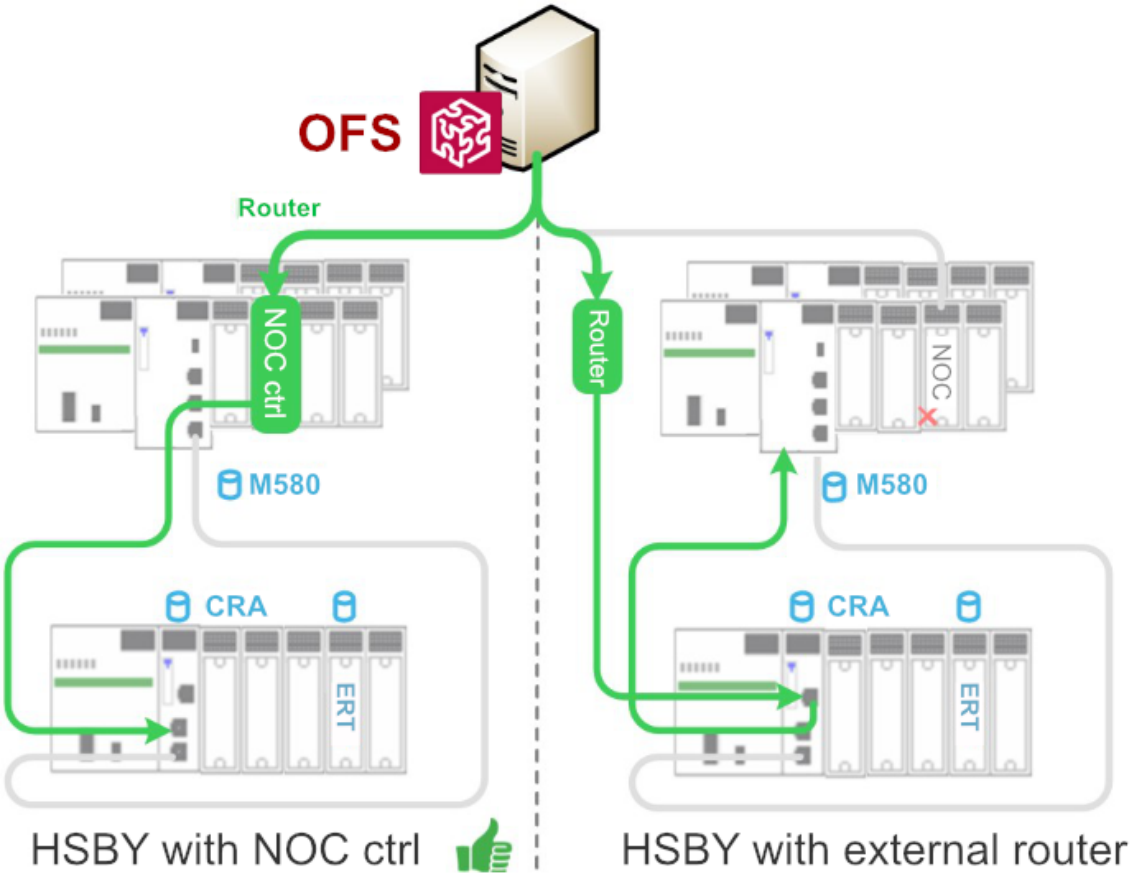
This chapter discusses how to select the architecture for the system time stamping application. We also introduce the method to synchronize the time clock between the multiple time sources and the time stamping modules, and list the time resolution with the different time stamping solutions.

Architecture selection

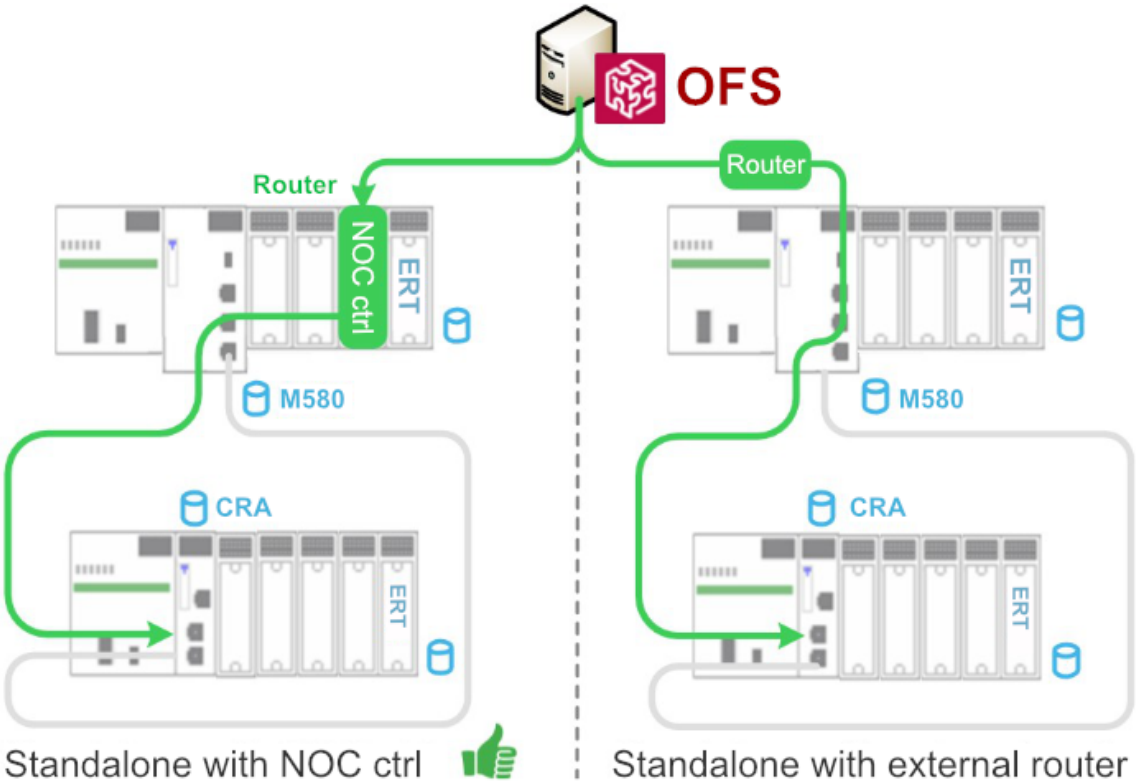


There are three types of modules which are supported by the system time stamping solution, including the M340/eX80ERT, eX80CRA, and M580 CPU. In the system time stamping architecture, OFS is used to automatically transfer the events from the time stamping module to the SCADA. As the time stamping module and OFS are on separate subnets, it is necessary to select a router to link these two subnets.

- In the standalone architecture, we can either select the NOC control module or a third-party router connected to the CPU service port/NOC module which is linked to RIO network in order to set up the connection between OFS and the time stamping module.

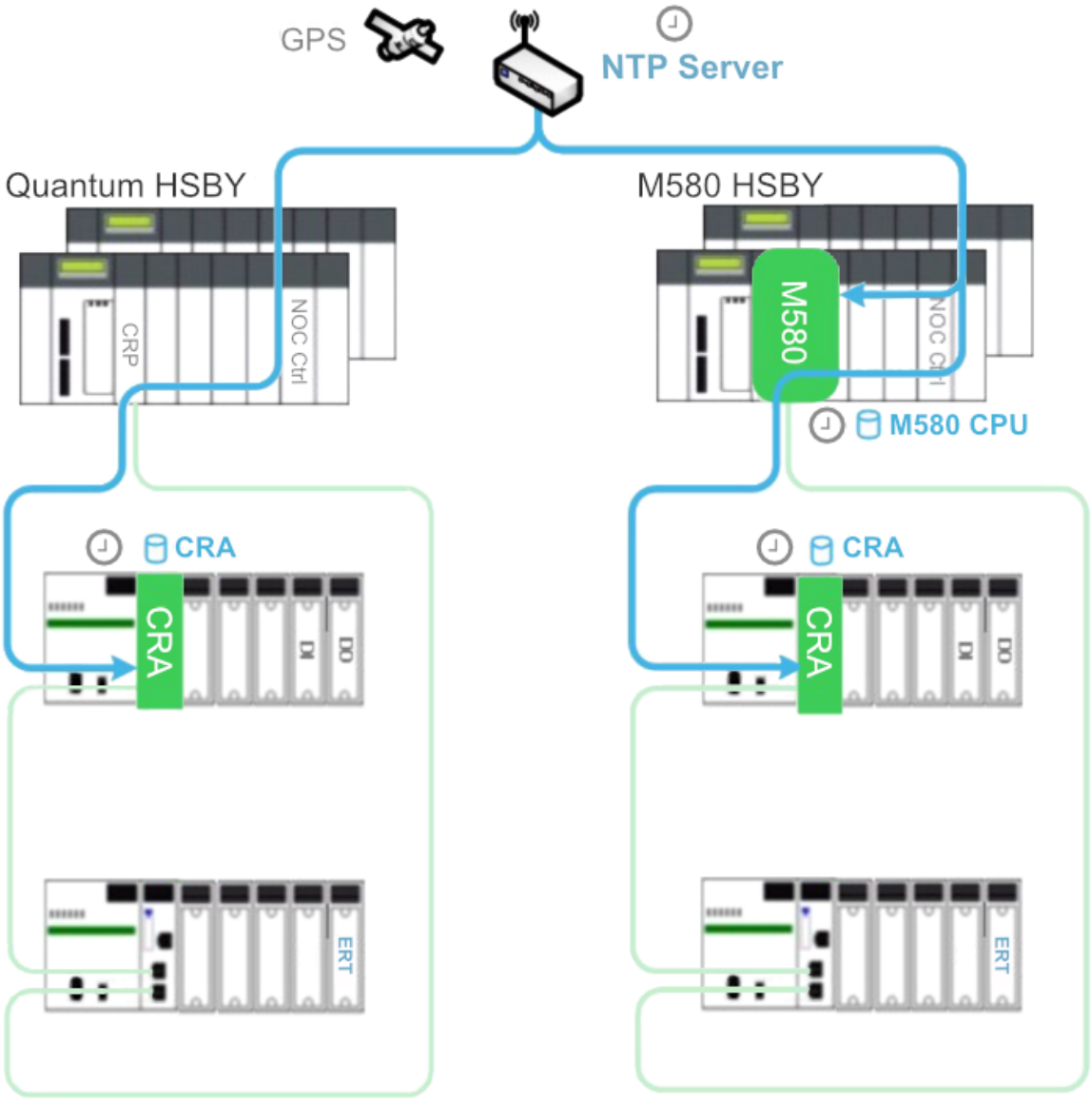


- In the HSBY architecture, we can either select the NOC control module as a router, or select a third-party router directly connected to the RIO network to set up the connection between OFS and the time stamping module.

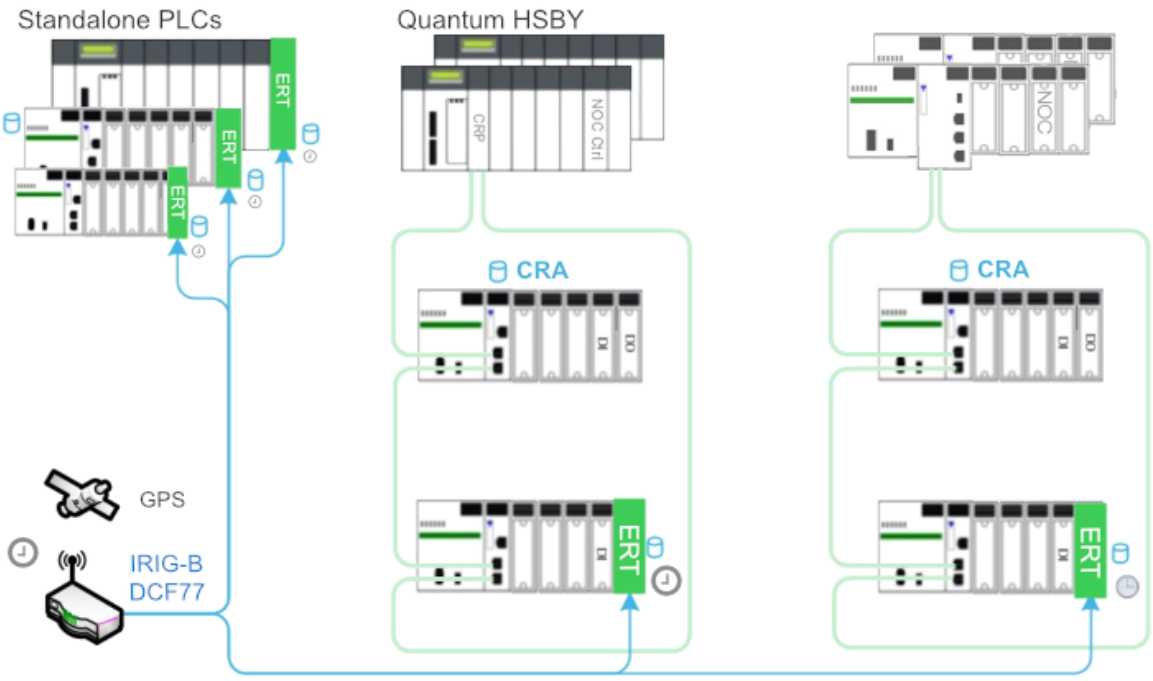


Time synchronization

- The external NTP server provides the time clock for the CPUs and CRAs. We have to configure the NTP server's IP address and polling period for each NTP client. In the M580 architecture, the M580 CPU can act as an NTP server to synchronize its CRA module's time clock.

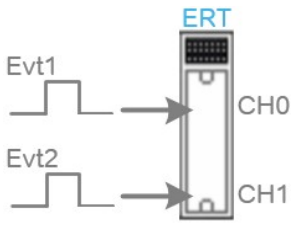
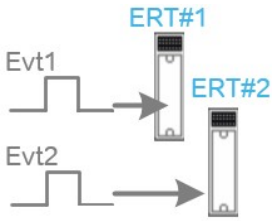
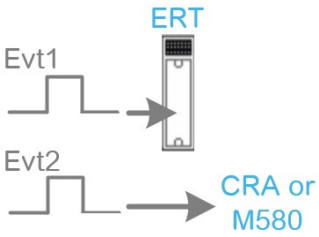
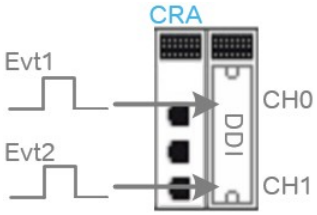
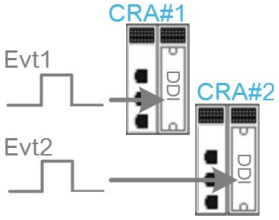
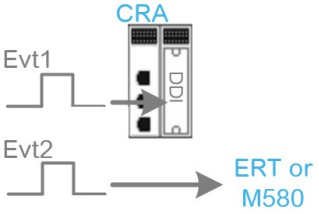
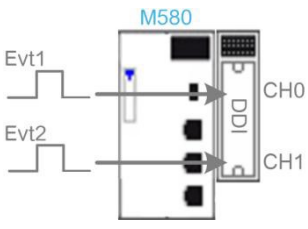
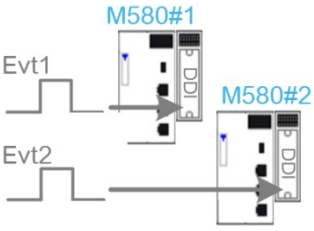
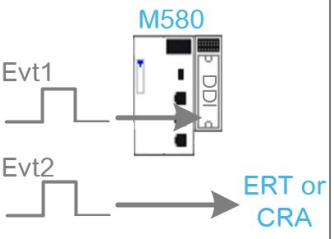


- The IRIG-B 004/5/6/7 or DCF77 signals generated by the GPS receiver are used to synchronize the ERT module's time clock.


Standalone PLCs
Quantum HSBY
ERT
ERT
ERT
CRP
NOC Ctrl
NOC
CRA
CRA
ERT
ERT
GPS
IRIG-B
DCF77

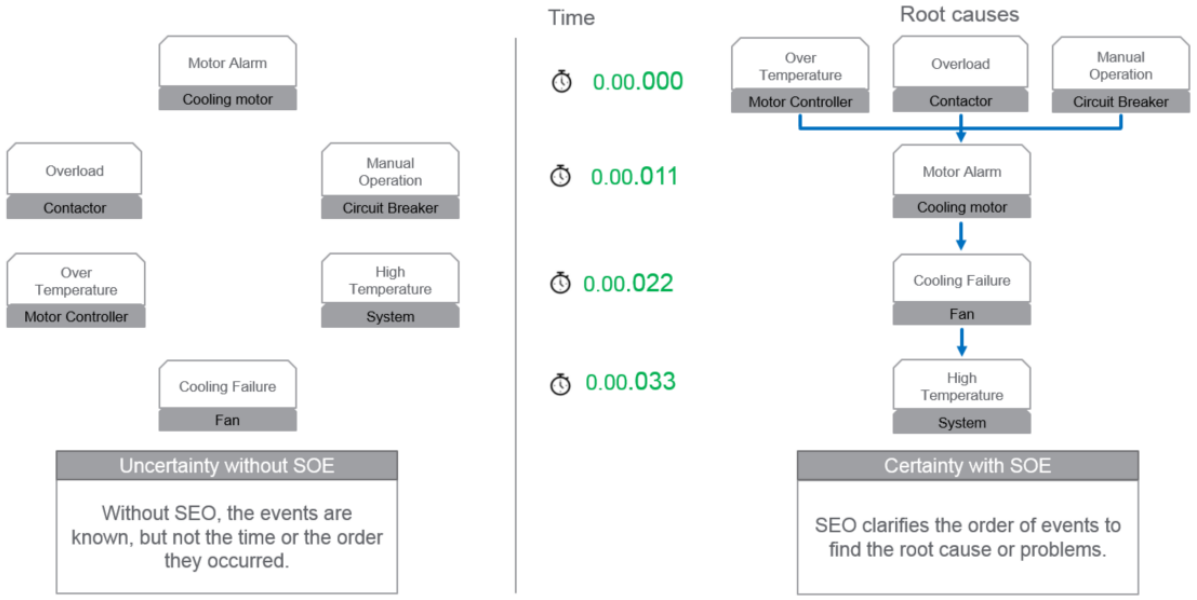
Event resolution

The resolution time is an important parameter for the time stamping application as it impacts the precision of the sequence of events. Below is the list of the resolution times depending on where the events are detected.

TS source module	Events recorded by one module	Events recorded by two modules of the same type	Events recorded by two modules of different types
M340/x80 ERT			
	Min 1ms resolution	Min 2ms with IRIG-B 004/5/6/7 Min 4ms with DCF77	Depends on CRA or M580 scan time
(e)X80 CRA			
	CRA scan time, average 3ms	Average 10ms resolution	Depends on CRA or M580 scan time
M580 CPU			
	CPU MAST task scan time	Depends on large M580 scan time	Depends on CRA or M580 scan time

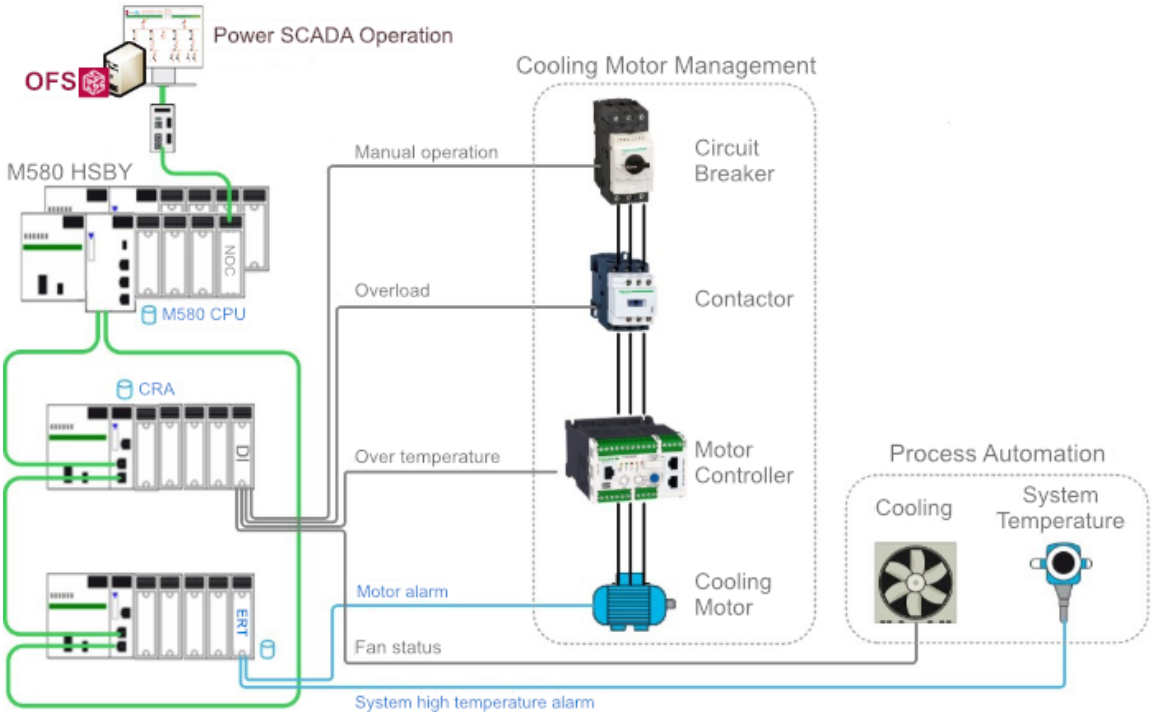
Design

The SOE function is the primary user of the time stamping application. This chapter uses the example of a cooling system for the temperature process control to show how to design an SOE function. In the example application, the SOE function will help us to easily find the root cause of the problem according to the sequence of events.



SOE architecture design

This guide uses the M580 HSBY architecture as an example to design an SOE function.

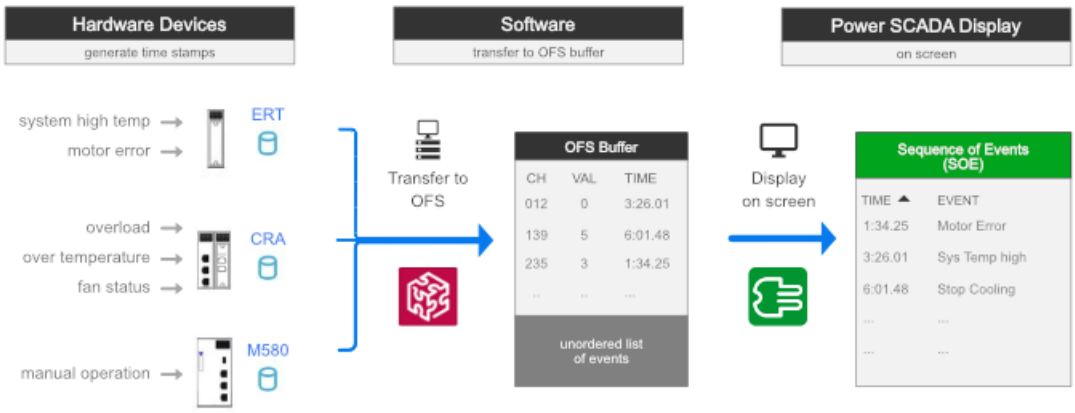


In the above diagram, a cooling control system includes a circuit breaker, a contactor, a motor controller, a motor, and a fan. The fan is used to cool down the system temperature when the temperature is higher than the pre-set value. For the process automation monitoring, some device statuses and process values need to be acquired by the PAC. Meanwhile, these statuses need to be time stamped by the PAC for building an SOE service. The first step to designing the SOE function is to define which time stamping module will be used to monitor the status of the devices, and the process for generating the time stamping events. The table below shows which time stamping module is associated with which event.

Event level	Event name	Source devices	TS module
Process events	High temperature alarm	System temperature instrument	M340/eX80 ERT module
Device events	Motor alarm	Motor	
	Overload	Contactor	eX80 CRA with RIO module
	Fan status	System cooling fan	
	Over temperature	Motor controller	
	Manual operation	Circuit breaker	M580 CPU with RIO module

Data flow design

The following image shows the flow of the time stamped data from the devices to the SCADA using the system time stamping solution:



1. Events are detected and time stamped by the time stamping module
2. Manage the time stamping events using OFS
3. Transfer these events to SCADA using OFS, and display them on the SCADA pages

Configuration

This chapter introduces how to configure the PAC, the time stamping module, OFS, and Power SCADA in order to implement the SOE application using the system time stamping solution.

PAC configuration

The PAC system configuration is the same for these three platforms.

Unity Pro

1. In the tree pane, expand **Project Settings > General > PLC embedded data** and then under Property Label, click **Data dictionary**.

   This allows any client (SCADA using OFS) to animate or modify all symbolized variables of the application embedded in the PLC's memory.

2. In the tree pane, expand **Project Settings > General > Time** and then set **Time Stamping Mode** to **System**:

Max events stored is used for adjusting the buffer size of the time stamping by the M580 CPU. The value is between 0 and 4000.

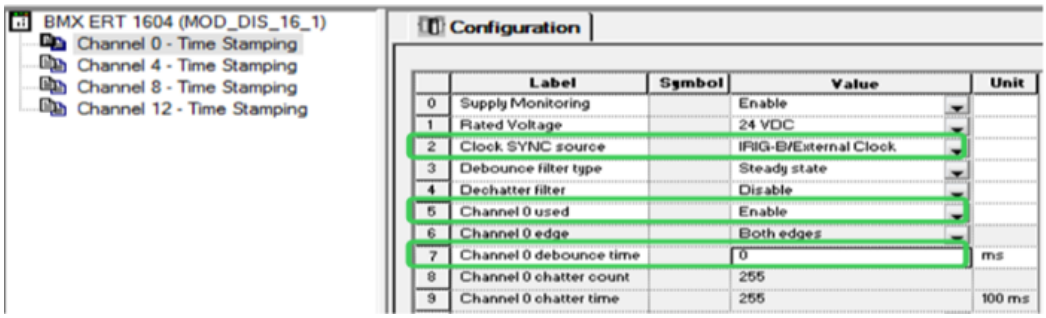
NOTE: Its minimum value = 4 * number of events configured (including SOE_UNCERTAIN). If this configured value is too small, Unity Pro will show a build error and indicate the minimum events number in the message window.

BMX ERT

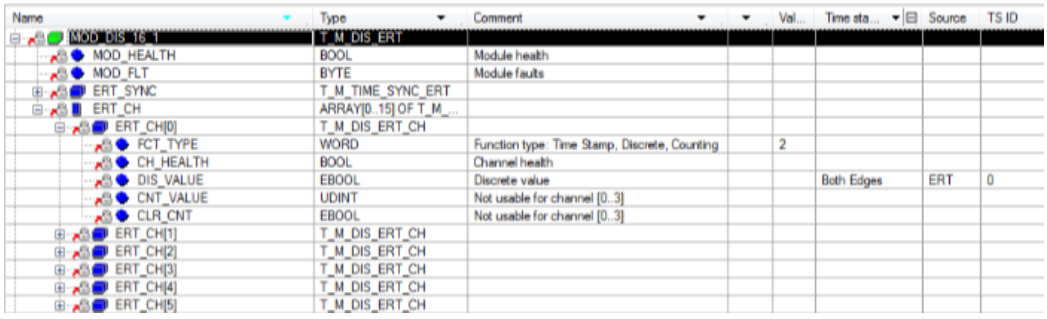
The BMX ERT module is installed in the M580/M340 backplane or x80 drop using the device DDT mapping methodology:

1. Double-click on the BMX ERT 1604 T module to enter the Configuration window and then configure the following:
   - Define the 'Clock SYNC source' for the ERT module.
   - Enable or disable each of the 16 discrete channels in the field, 'Channel x used,' according to the application.
   - Set the 'debounce time' of the enabled channel to 0ms, if you need to meet the requirement of a 1ms event resolution.

   For example:

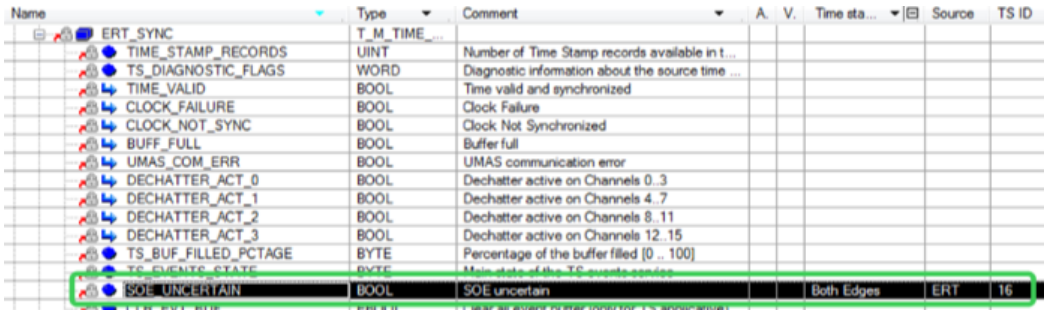


2. Open the module's 'Device DDT' tab and then click **Goto detail**. All the elements within this Device DDT are shown in the Data Editor.



3. The parameter, SOE_UNCERTAIN, is activated by default, and is time stamped by both

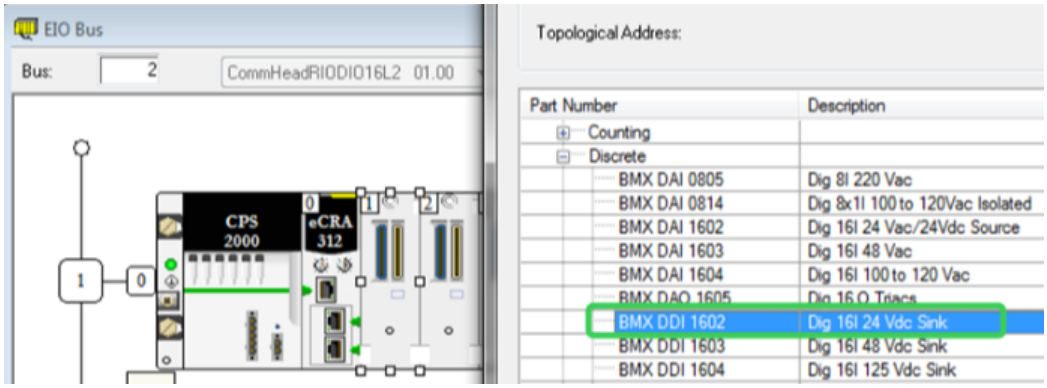
edges.



x80 CRA

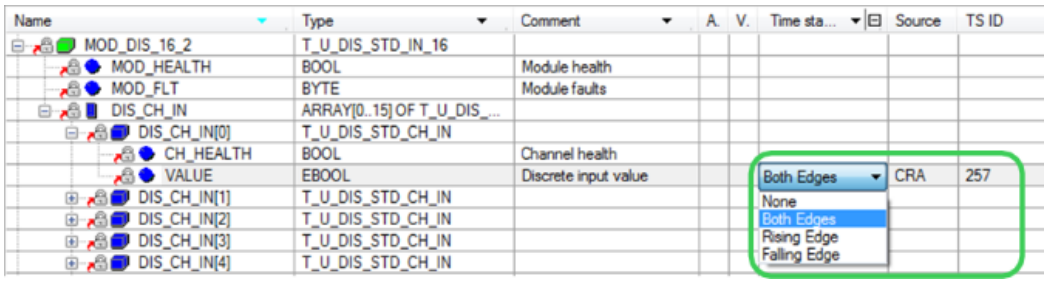
The x80 CRA module can be installed in the x80 remote I/O drops.

1. The x80 CRA can time stamp the discrete I/O events detected by modules inserted in the remote I/O drop. Add a discrete I/O module in the x80 drop by double-clicking on an empty slot. Select a BMX DDI 1602. For example:



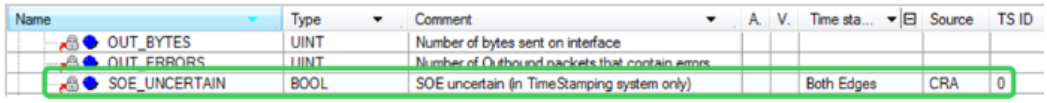
2. Open the properties page of the discrete I/O module. Select the **Device DDT** tab, and click **Goto details** to open the Data Editor window. The 'Name' of the 'Implicit device DDT' can be modified as the application requires.

3. Expand the elements under the implicit device DDT name of the BMX discrete I/O module. Expand the elements under 'DIS_CH_IN' of the input module, or 'DIS_CH_OUT' of the output module. Expand the elements under the required time stamping channel, and enable the channel by selecting the proper event in the 'Time stamping' cell.

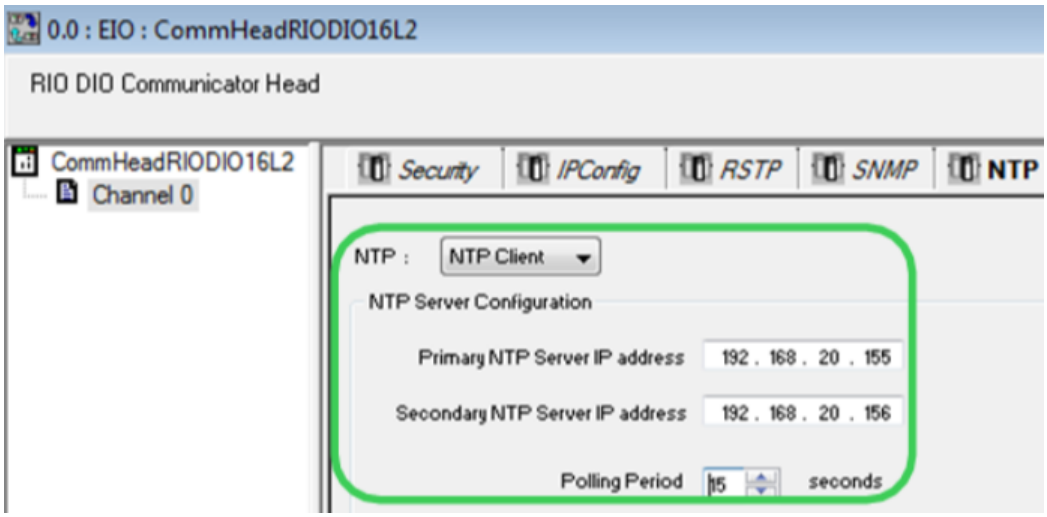


NOTE: For the M580, this attribute can be 'None,' 'Both Edges,' 'Rising Edge,' or 'Falling Edge.' For Quantum, however, the only options are 'None' or 'Both Edges.'

4. The parameter – SOE_UNCERTAIN – is already listed in the CRA drop's device DDT, and the 'Time stamping' attribute has automatically been set to 'Both Edges' and assigned a TS ID.



5. Open the Quantum CRP or the M580 communication configuration window. Enable the NTP service to provide the time synchronization service for x80 CRAs. Configure the primary or secondary server's IP and polling period. For example:

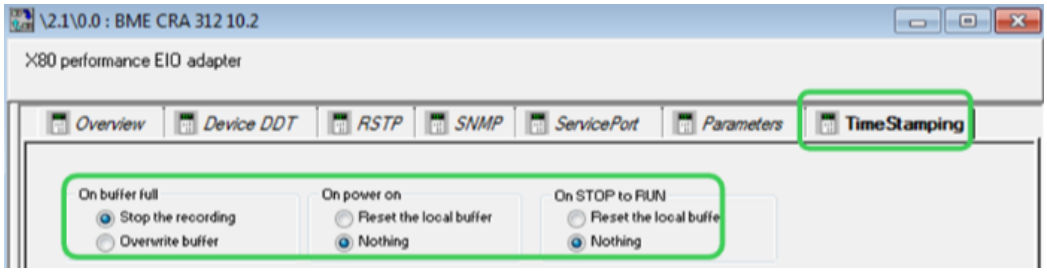


NOTE: It is recommended that the polling period be set to lower than 20s in order to get a time stamp resolution of 10ms between two events on different CRA modules.

For the M580, configure the CPU as either the NTP server or client. Both can provide time synchronization for the x80 CRAs.

6. In the M580 platform, the x80 CRA's buffer behavior settings can be adjusted in the 'Time Stamping' tag of its configuration window.
   - **On buffer full**: Stop the recording or overwrite the oldest value when the event buffer is full.
   - **On power on**: Erase the local buffer or do nothing when detecting a CPU powering on.

- **On stop to run**: Erase the local buffer or do nothing when detecting a PLC transitioning from stop to run.

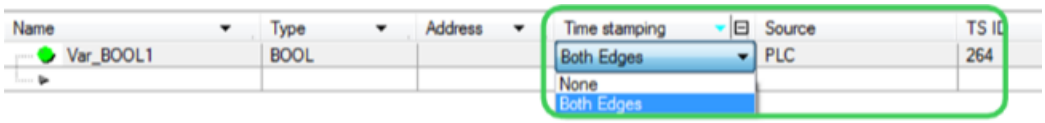


NOTE: While installed in Quantum remote I/O drops, the CRA's time stamping buffer behaviors are set to the default value (as per the figure above) and cannot be modified.

M580 CPU

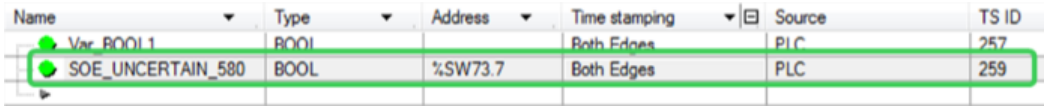
This section presents the configuration steps of the time stamping by internal variable changes in the M580 program.

1. In the 'Data Editor,' select a BOOL type internal variable which can trigger a time stamping event; then select the trigger condition. Unity Pro will generate a TS ID.

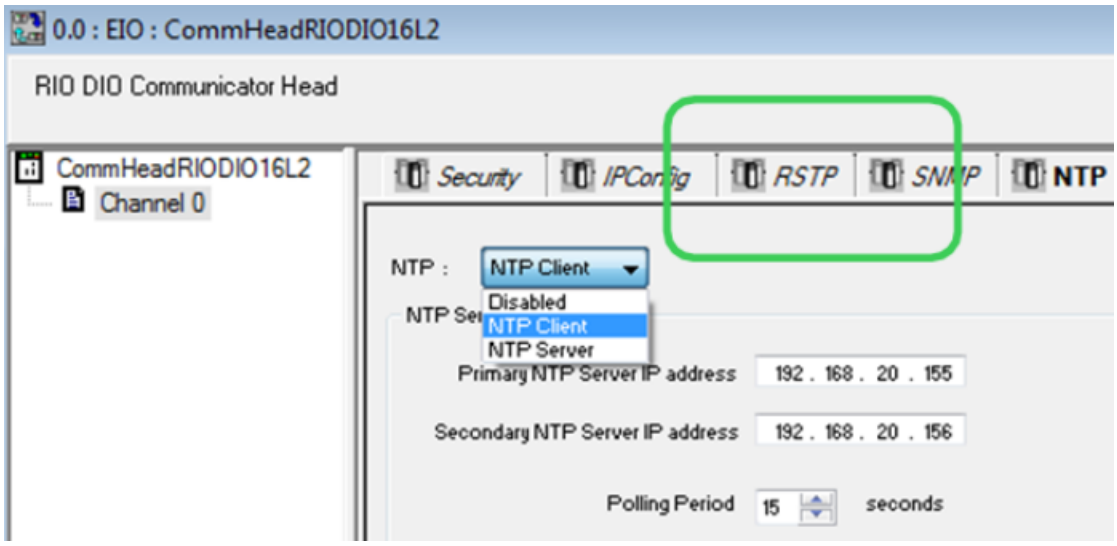


NOTE: The internal buffer of the M580 CPU's time stamped events will behave as follows: The CPU stops recording new events when the buffer is full.

2. Manually create the SOE_UNCERTAIN variable for the M580 CPU, and locate this BOOL at %SW73.7. Enable its time stamping selection.



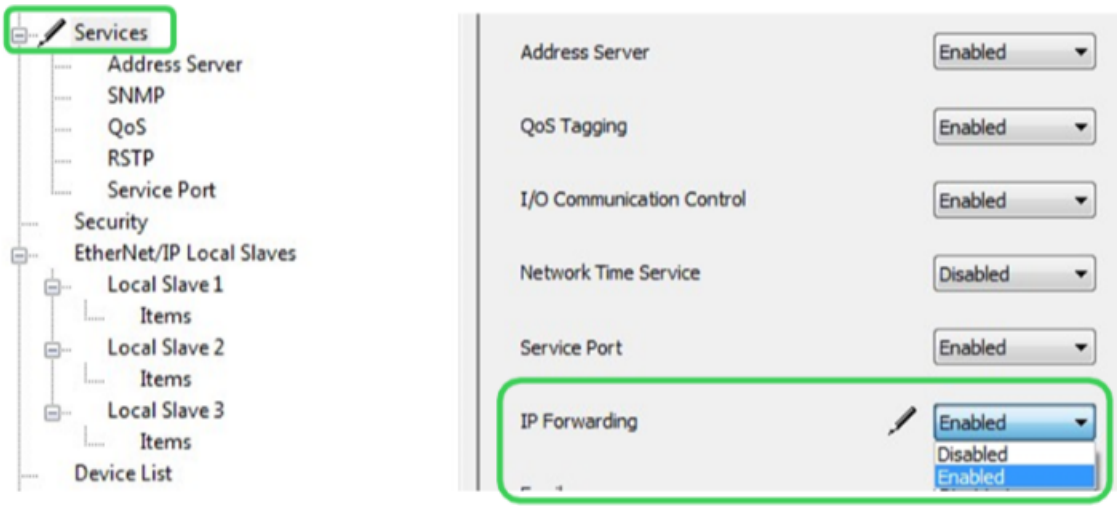
3. In the M580, two kinds of time synchronization methods are allowed:
   - External time source: The CPU is set as an NTP client and synchronizes its internal clock with an Ethernet NTP server, usually located on the control network.
   - Internal time source: The CPU is set as an NTP server. Using its internal clock, the M580 CPU provides the time synchronization service for the other connected devices.



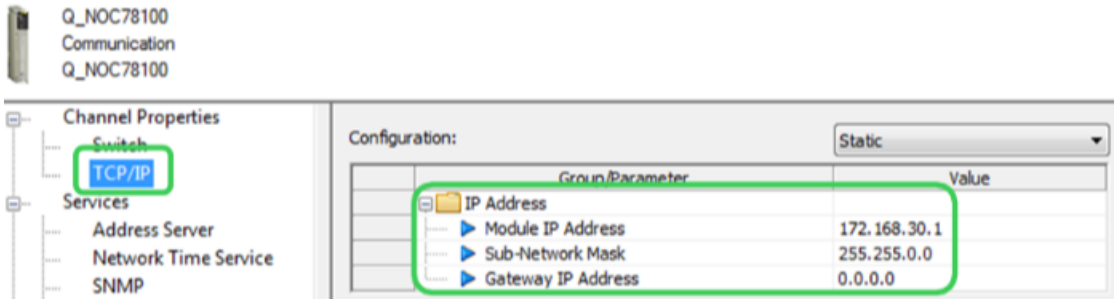
Quantum 140 NOC 78100

The Quantum Ethernet control module, 140 NOC 781 00, acts as the router between the x80 ERT or x80 CRA module installed in the device network and OFS installed in the control network.

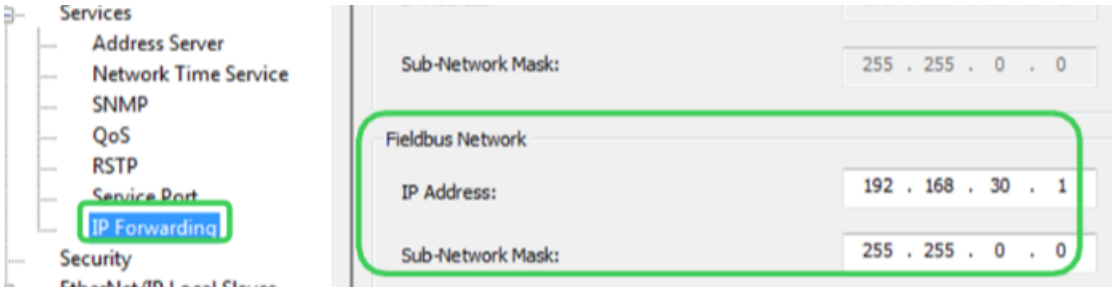
1. In the Unity 'DTM Browser,' enable the 'IP Forwarding' service.



2. Configure its IP address for the control network port (Eth port 3&4) on the 'TCP/IP' page.



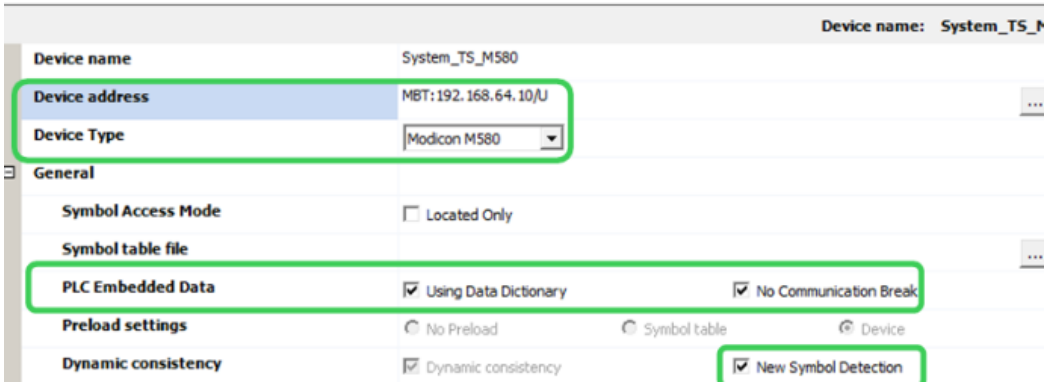
Configure its IP address for the device network port (Eth port 2) on the 'IP Forwarding' page.



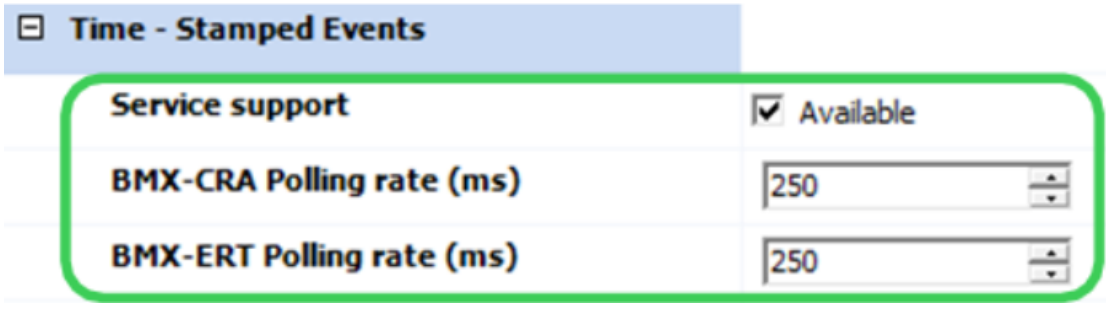
OFS configuration

Open the OFS configuration tool and create a new device alias.

1. Open the 'Device overview' page. Configure the protocol and address to communicate with the CPU:
   - From **Device Type**, select the PLC used.
   - Enable **Using Data Dictionary**, **No Communication Break**, and **New Symbol Detection**.



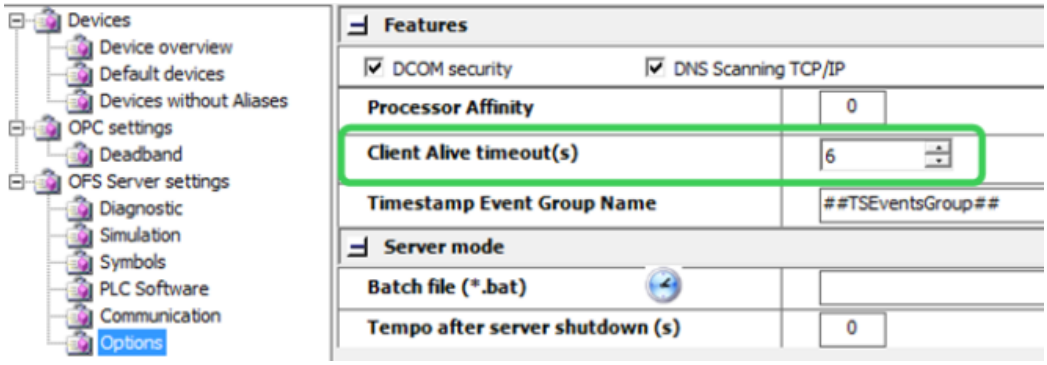
2. Check 'Available' under 'Time-Stamped Events,' and regulate the 'BMX-CRA Polling rate' and 'BMX-ERT Polling rate' to meet the system's requirements.



NOTE: Before setting the polling rates in OFS, the capability should be checked in advance.

If the 'Polling rate' is set to 0, then no event buffer read is performed. This can be used to temporarily disable the event sources.

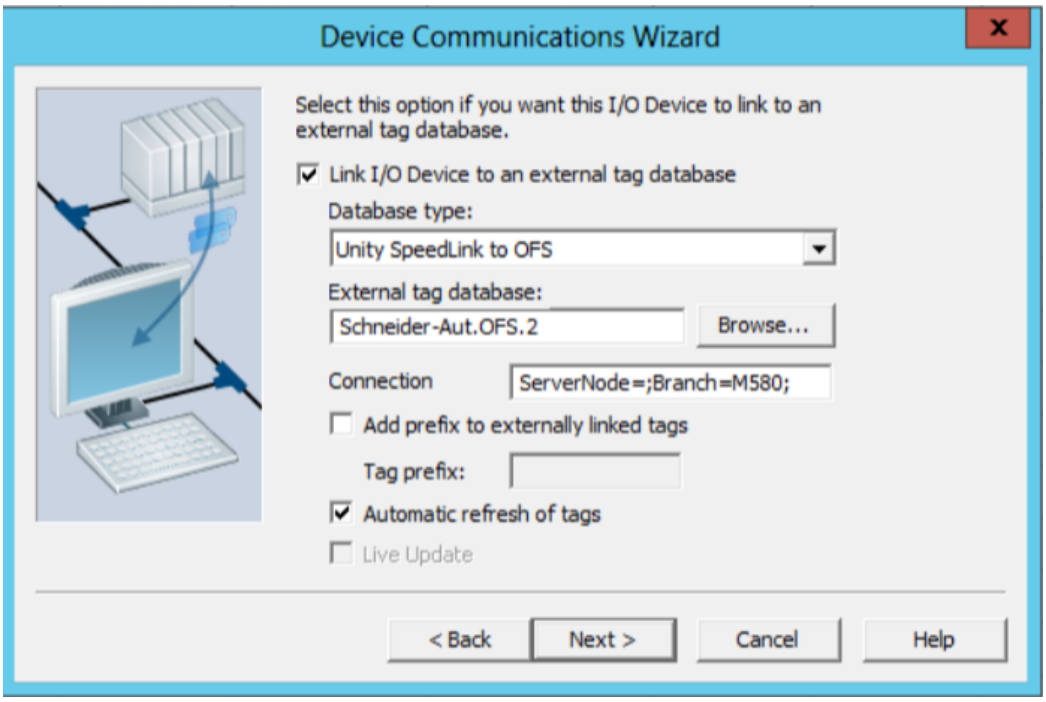
3. Set the 'Client Alive timeout' value which allows OFS to detect whether the OFS client (SCADA system) is responding or not.



Power SCADA configuration

The time stamped variable tags need to be configured in Power SCADA Studio to represent the corresponding time stamped variables in the PAC.

1. In Power SCADA Studio, create a new I/O device for the system time stamping. In **Topology > I/O Devices**, click **Express Wizard** and then configure the settings according to the device's requirement:
   a. Select the Power SCADA project that you want to create the device in.
   b. Click **Use an existing I/O Server**, select the existing server, and then click **Next**.
   c. Click **Create a new I/O Device**, enter an alias for the device, and then click **Next**.
   d. Click **External I/O Device**, and then click **Next**.
   e. Select the communication method, then click **Next**.
   f. In **Address**, enter the I/O device alias name. This value must be identical to the alias name you created in step c. Click **Next**.
   g. Link the device to an external tag database. Click **Link I/O Device to an external tag database**, browse to the database, and then enter the connection information. For example:



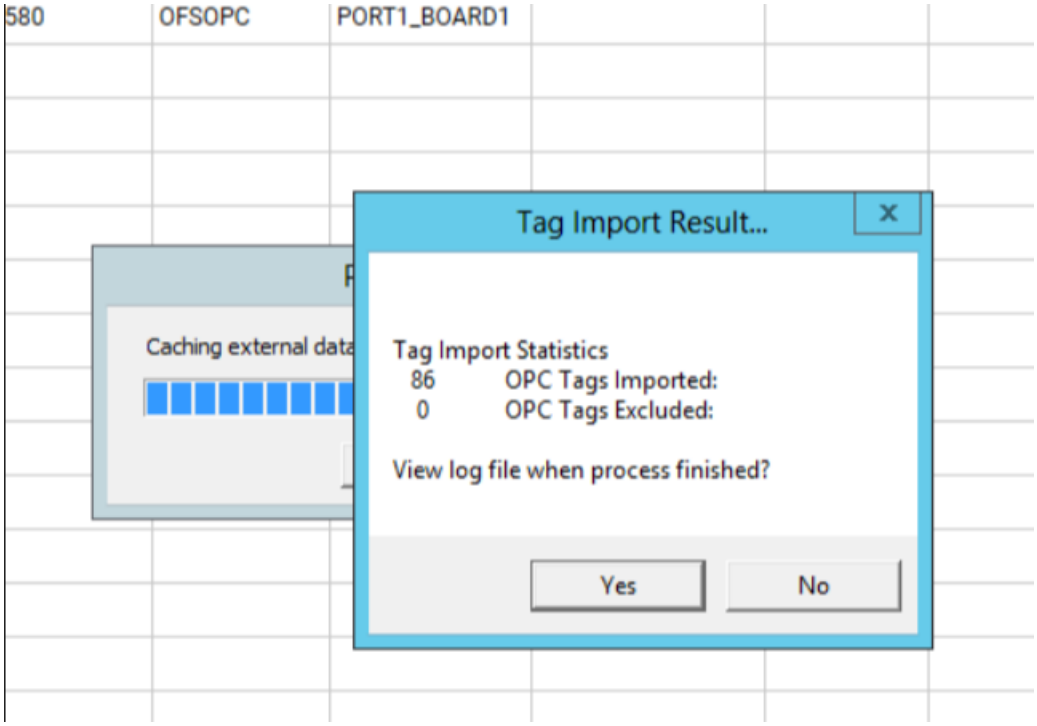
h. Click **Next**.

i. Review the summary. Click **Finish** to save the I/O device, or **Back** to change its settings.

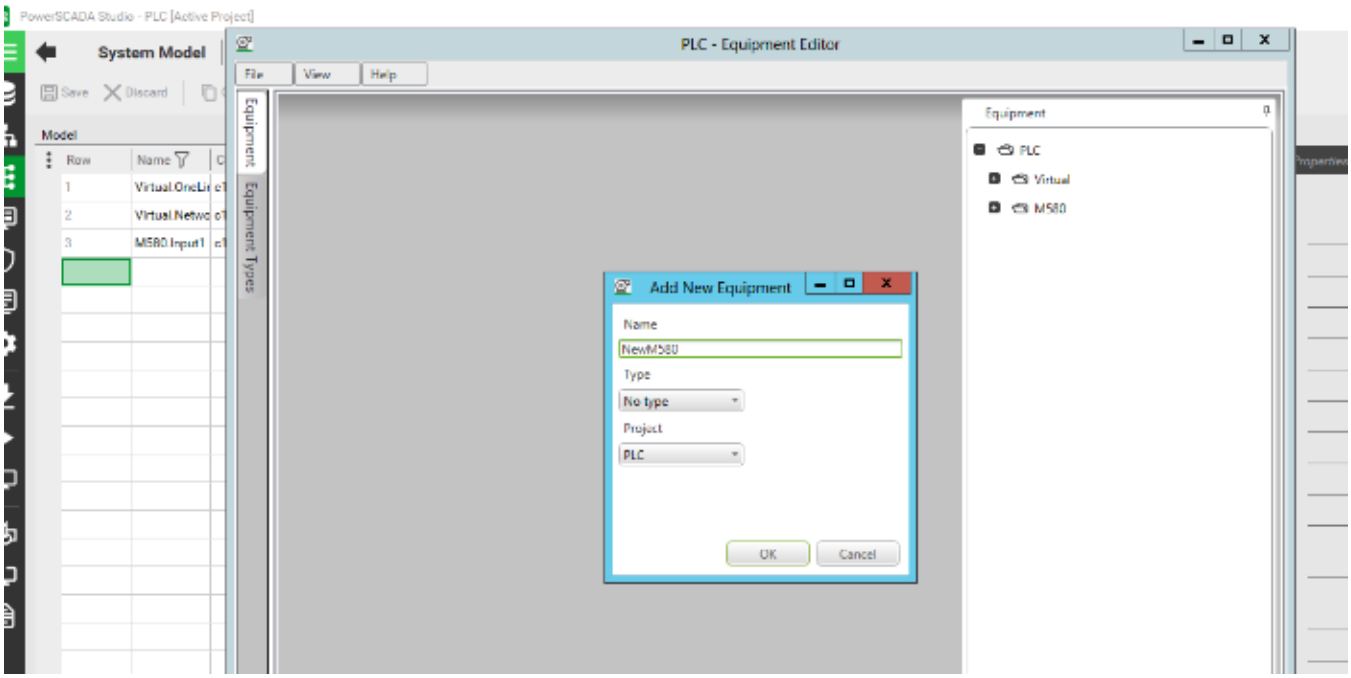
2. Import the device tags:

a. In **Topology > I/O Devices**, click **Import Tags**.

b. Select the OFS I/O device, verify that the source information is correct, and then click **Import**. and then The PAC tags are then automatically updated through OFS.

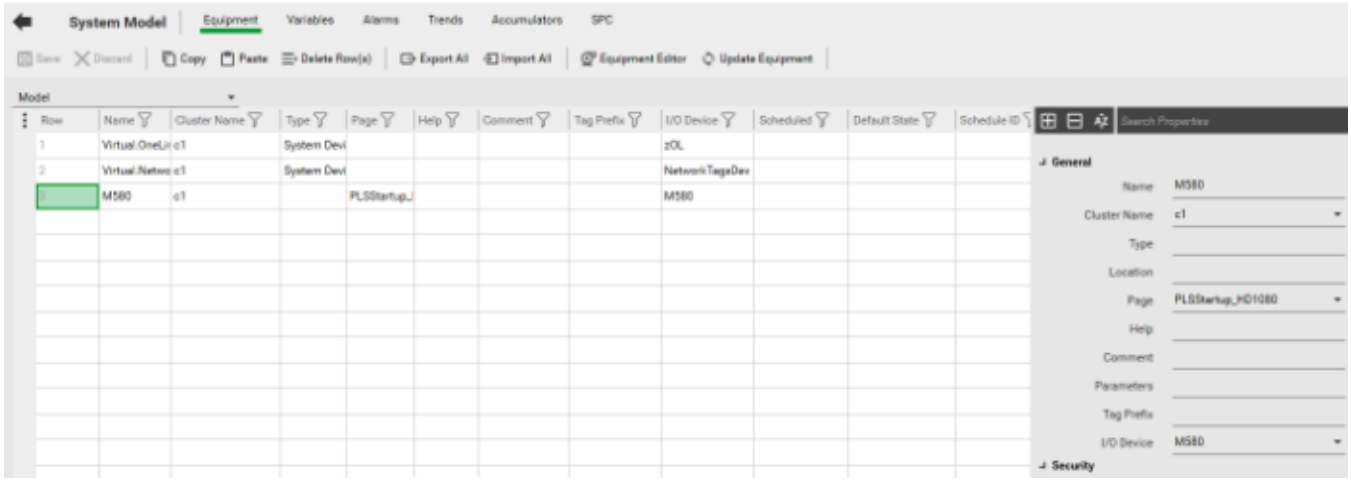


3. Create the equipment:
   a. In **System Model > Equipment**, click **Equipment Editor**.
   b. Add the **Equipment** and **Equipment Types** for the time stamped sources. For example:



4. Add a piece of equipment to associate with the I/O device:
   a. Click **System Model > Equipment**.

   All of the device's variables in this system will be linked to this equipment. For example:



5. (Optional) Manually configure the alarm category. Alternatively, if you want to use an existing alarm category (_PLSALM_EVENT, _PLSALM_HIGH, _PLSALM_MEDIUM, or _PLSALM_LOW), skip this step.

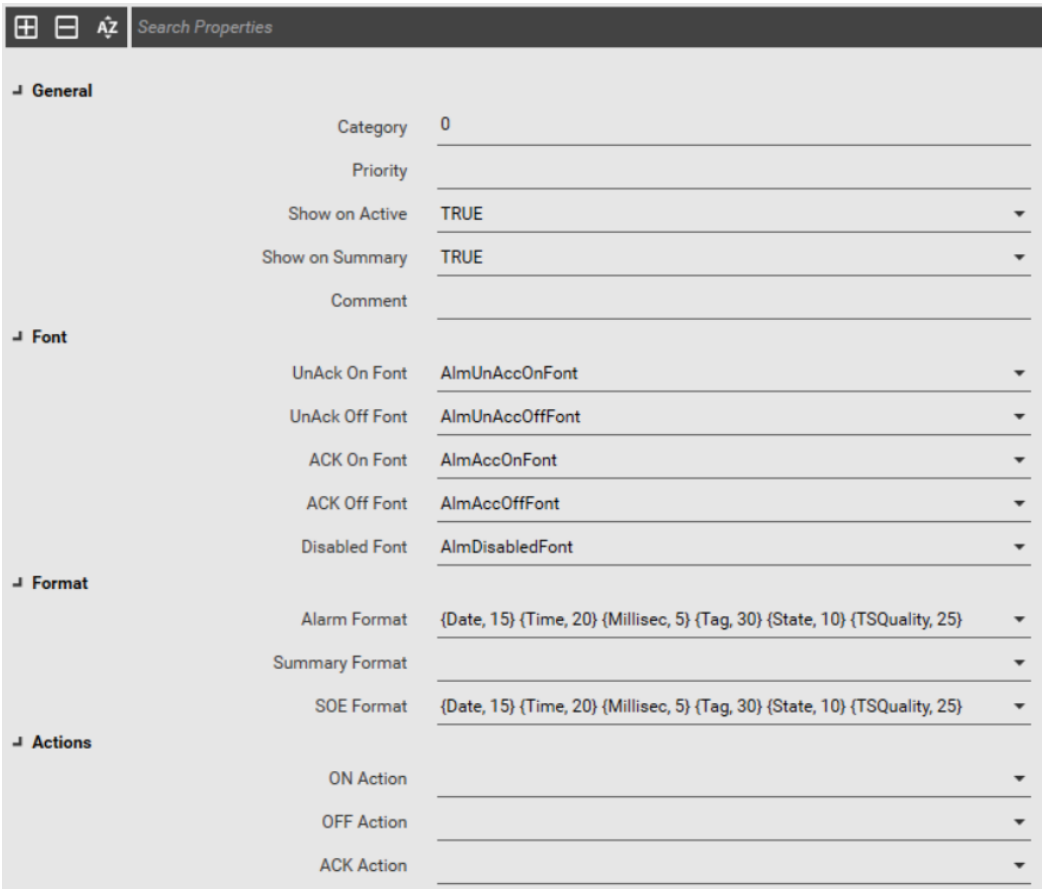
   To manually configure an alarm category:
   a. Click **Setup > Alarm Categories**.
   b. In the grid, enter the **Category** number.

c. Select whether the alarm category is **Show on Active** and/or **Show on Summary**.

d. Select the corresponding formats for the different alarm statuses.

e. In **Alarm Format**, enter the information to be displayed on the Active Alarm page, and, in **SOE Format**, the information to be displayed on the SOE history page.

For example:



6. Create the system time stamping alarms:

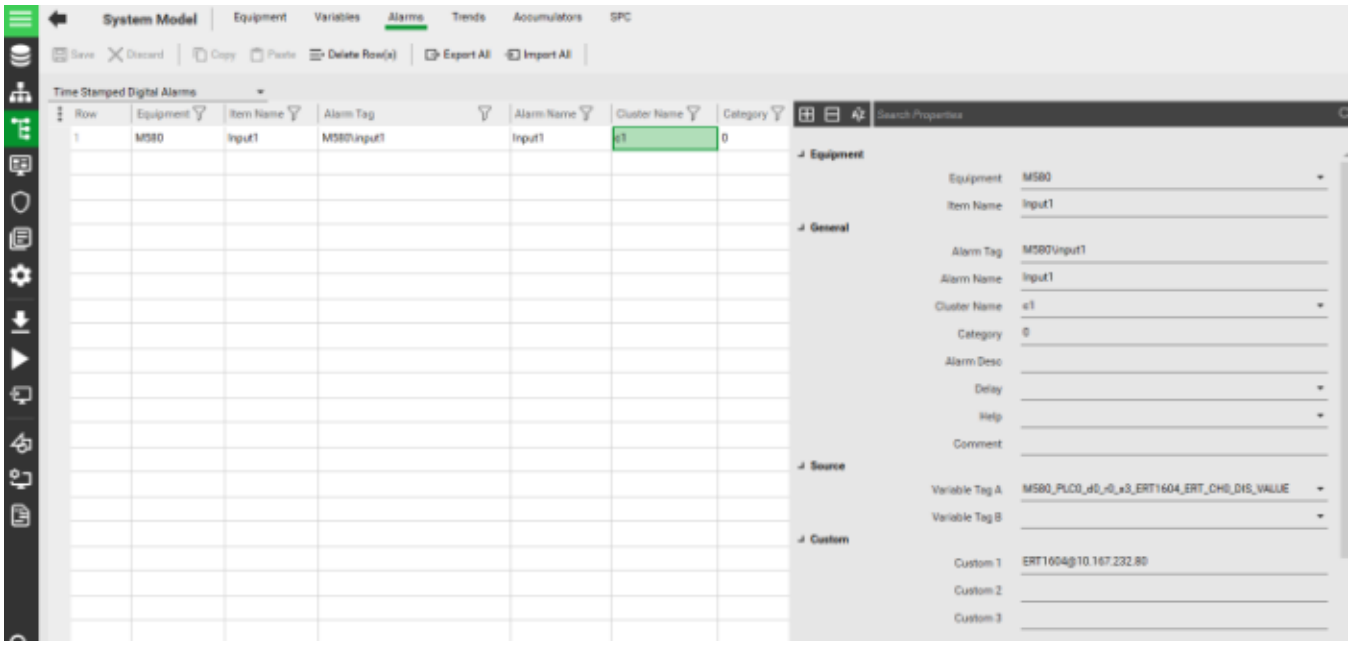
a. Click **System Model > Alarms**.

b. Select the corresponding **Equipment** for the alarm.

c. Enter the alarm's information, and select the time stamping variables to configure the **Variable Tag**.

d. In the alarm **Category**, enter the alarm category you created in step 5, or select an existing alarm category (_PLSALM_EVENT, _PLSALM_HIGH, _PLSALM_MEDIUM, or _PLSALM_LOW).

For example:



⚠ WARNING

LOSS OF ALARMS

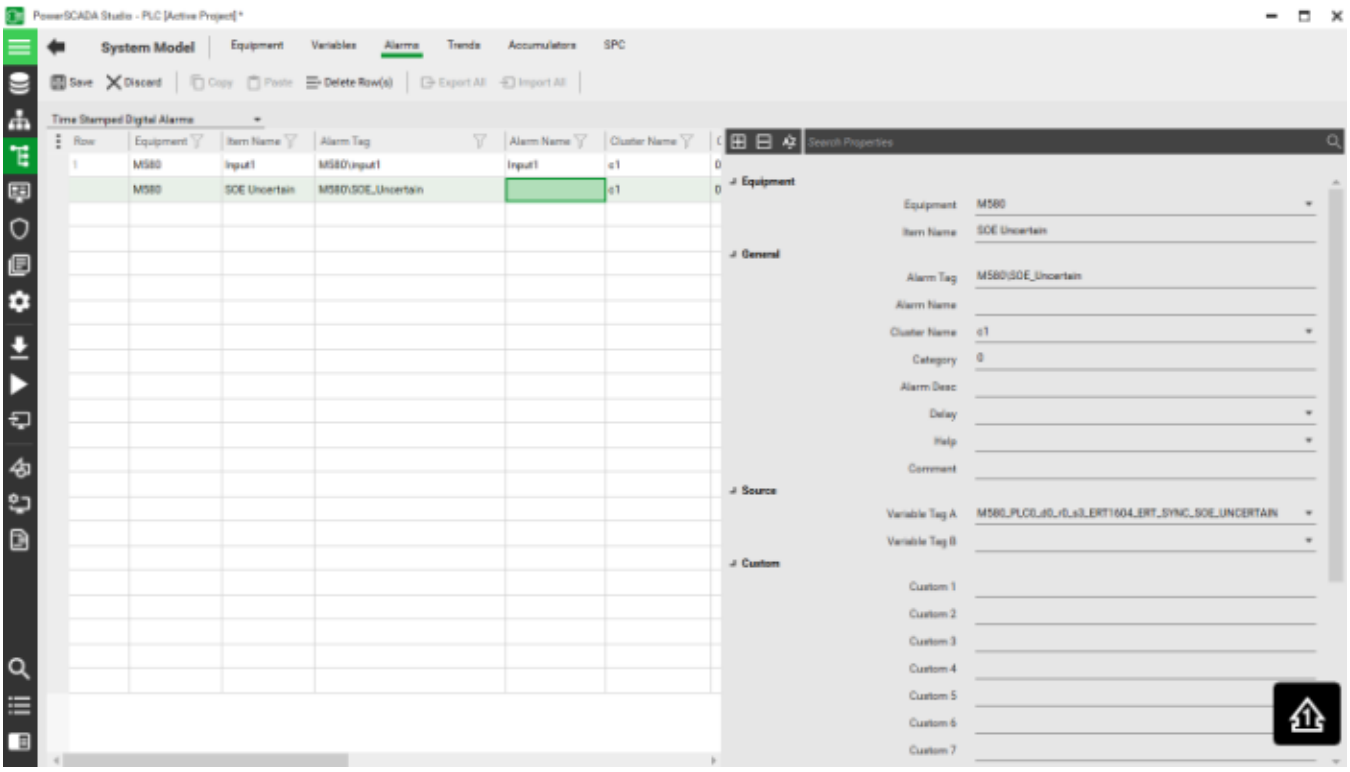
- To be able to detect that the event buffer is full, configure a tag and an alarm tag associated with the SOE_Uncertain parameter in UnityPro.
- Respond quickly to a buffer full alarm if it appears, as this will avoid a situation where the buffer becomes inoperable.

Failure to follow these instructions can result in death, serious injury, or equipment damage.

NOTE: When the source event buffer in the PLC is full, any new events will not be stored. In this case the value of SOE_Uncertain variable becomes TRUE. When the buffer becomes available again, the PLC will provide the values of all time stamped event variables. As these values are timestamped with a current time, the time quality of these values will be set to Invalid. The SOE_Uncertain is a variable in a time stamped event source whose value becomes TRUE when there is no space in the event buffer.

In other words, from the moment the SOE_Uncertain variable becomes TRUE to the moment it goes FALSE, all events occurring within that time period will have an invalid time quality. Do not rely on the time quality of events occurring within the time period where SOE_Uncertain is TRUE (event buffer full).

7. Add the ‘SOE_UNCERTAIN’ parameter to the ‘Time Stamped Digital Alarms’ to help check the status of the time stamping sources:



Implementation

This chapter presents the detailed steps for the engineering implementation.

PAC implementation

To implement PAC:

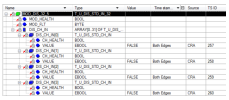
1. In Unity's data editor, select the ERT device DDT to enable two ERT channels for the demo SOE application, as follows:

   1) ERT_CH[0] → High temperature alarm; 2) ERT_CH[1] → Motor alarm

Name	Type	Value	Time stam...	Source	TS ID
MOD_DIS_16_5	T_M_DIS_ERT				
MOD_HEALTH	BOOL				
MOD_FLT	BYTE				
ERT_SYNC	T_M_TIME_SYNC_ERT				
ERT_CH	ARRAY[0..15] OF T_M_DIS...				
ERT_CH[0]	T_M_DIS_ERT_CH				
FCT_TYPE	WORD	16#0002			
CH_HEALTH	BOOL				
DIS_VALUE	EBOOL	FALSE	Both Edges	ERT	0
CNT_VALUE	UDINT				
CLR_CNT	EBOOL				
ERT_CH[1]	T_M_DIS_ERT_CH				
FCT_TYPE	WORD	16#0002			
CH_HEALTH	BOOL				
DIS_VALUE	EBOOL	FALSE	Both Edges	ERT	1
CNT_VALUE	UDINT				
CLR_CNT	EBOOL				

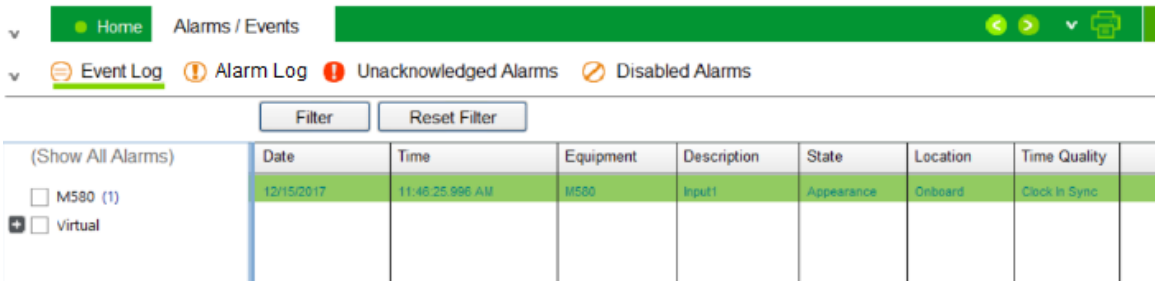
2. In Unity's data editor, select the DDI device DDT to enable four DDI channels for the demo SOE application, as follows:

1) DIS_CH_IN[0] → Overload; 2) DIS_CH_IN[1] → Fan status; 3) DIS_CH_IN[2] → Over temperature; 4) DIS_CH_IN[3] → Manual operation



Operation

You can view the SOE history in the Power SCADA Operation event log:



Configure Power SCADA Operation as an OPC-DA Server

Before you begin configuring OPC communications with Power SCADA Operation, refer to these help file locations:

- In the DriverReferenceHelp.chm help file (located in the Power SCADA Operation Bin folder), see the OPC Driver section.
- In the citectscada.chm help file (also in the Bin folder), see Using OPC Server DA.

You can configure Power SCADA Operation to act as an OPC-DA server. In this mode, it will supply data to an OPC client, such as Matrikon OPC Explorer (a free download available at Matrikon.com).

NOTE: We used Matrikon in our tests and validation, but you may have one of the many other OPC products. The information in this document is specific to Matrikon products. Thus, the screens you see in your OPC client software may not be the same as the instructions below.

To select device profiles, create tags, and begin using the Matrikon tool:

1. From the Profile Editor, select the device profiles to be used for the project that will be used when Power SCADA Operation becomes an OPC-DA server.
2. Use the I/O Device Manager (Start > Programs > Schneider Electric > IO Device Manager) to add the device. This will create the variable tags you need for the project.
3. To configure the OPC-DA server: In Power SCADA Studio, click Topology > Edit, then choose OPC DA Servers.
4. Complete the fields for the server.
5. Compile and run the project.

6. Launch the Matrikon OPC Explorer.

   The Matrikon OPC Explorer screen displays. On the left side of the screen, a list of available OPC servers displays.
7. Highlight the server you want. The Connect button to the right of the list is enabled.
8. Click Connect.

   NOTE: If you are connecting to an OPC Server on a remote networked computer, and it does not display in the list, you must manually add the server. From the top toolbar, click Server > Add/Connect Server. This displays the form used to enter the host and server. Choose the server on that form and click OK to connect.
9. After you have connected to the server, click Add Tags to display a new pop-up box, which lists the available tags in the project that is running:
10. To add a single tag to the group, hover over the tag name and right click. Select Add to Tag List. To add all items to the tag list, right click and select Add All Items to Tag List.

    Selected tags appear in the Tags to be added column on the right:
11. After you select all the tags you want, close the form: click File > Update and return.
12. You return to the main setup page, where the tag values are displayed.

Configure Power SCADA Operation as an OPC-DA Client

Before you begin configuring OPC communications with Power SCADA Operation, refer to the online help files in these locations:

- In the DriverReferenceHelp.chm help file (located in the Power SCADA Operation Bin folder), see the OPC Driver section.
- In the citectscada.chm help file (also in the Bin folder), see Using OPC Server DA.

You can configure Power SCADA Operation to act as an OPC-DA client. In this mode, it will draw data from an OPC server, such as the one Matrikon OPC Explorer uses.

NOTE: We used Matrikon in our tests and validation, but you may have one of the many other OPC products. The information in this document is specific to Matrikon products. Thus, the screens you see in your OPC client software may not be the same as the instructions below.

To create OPC tags in Power SCADA Operation:

1. Launch Matrikon Explorer to see tags that are available. Select the OPC Server to which you want to connect.

   For this example, we are using Matrikon.OPC.Simulation.1
2. Connect to the Server Matrikon.OPC.Simulation.1 on the remote computer.
3. Click Add Tags to display the Tag Entry tab:
4. Right click the Random folder (under Available Items…), and select Add All Items.
5. Select File > Update and return.

Matrikon Explorer displays a list of tags that it is regularly updating, similar to the list illustrated in this screen. To change the update rate (shown in the lower right-hand corner), right-click the group folder and choose properties.

6. Create a project: from the Power SCADA Studio Projects window, add the project.
7. Change to the Topology window. Click Edit, then add the following items: Choose from the drop down link each of the items:
   - Cluster
   - Network Addresses
   - I/O Servers
8. Add a board: on the Topology window, select Components & Mapping. Then click the drop down link, and choose Boards. Add the information for the board.

   NOTE: Type the IP address of the remote OPC Server in the Special Opt field. The address field is used to specify the update interval in milliseconds. Type zero (0) here to use the default value.
9. Create a port: from the Topology window, Components & Mapping, click the drop down link, and choose Ports. Add the port information.
10. Create an I/O device that references the OPC Server name: from the Topology window, choose I/O Devices. Be sure to use OPC for the Protocol.
11. Create the variable tags: from the System Model tab, choose Variables.
    a. Add a tag name.
    b. Use the OPC I/O device you created earlier.
    c. The address is the tag name given by the OPC server.

    One example in this case is Random.Int1, as shown in Matrikon Explorer display earlier.
12. Compile and run the project.
13. You can display the newly created Power SCADA Operation OPC tag values on a graphics page.

Performance Note: Using the setup described above with the default refresh rate (0), test results show that approximately 50,000 tags can be updated in less than one second . This was on a computer with an Intel Pentium dual-core processor running at 2.8 GHZ and 2 GB of RAM.

Redundant systems

NOTE: This section assumes that Power SCADA project and Primary Server are configured.

To configure redundant systems you must copy and export project files from the Primary Server to the Secondary Server.

Function	Task

"Configure the Power SCADA Primary Server" on page 460	Lists the procedures to copy and export that will be subsequently imported on the Secondary Server.
"Configure the Power SCADA Secondary Server" on page 462	Lists the procedures to import the Primary Server files onto the Secondary Server.

.

Configure the Power SCADA Primary Server

NOTE: This section assumes that Power SCADA project and Primary Server are configured.

Complete the following configuration tasks on the Primary Server.

- "Back up the Power SCADA Studio project" on page 460
- "Back up Application Configuration Utility settings" on page 460
- "Export One Line Engine Encryption" on page 461
- "Export and import One-Time Password settings" on page 461

The files you back up and export on the Primary Server will subsequently be copied or imported into the Secondary Server.

Back up the Power SCADA Studio project

Back up your Power SCADA Operation project. You will subsequently restore the project on the Secondary Server. (To back up the Profile Editor, use the Export feature on the **Projects** tab.)

To back up a Power SCADA Studio project file:

1. In Power SCADA Studio: Click **Projects**, and then click **Backup**.
2. In the Backup Project window, select the project you want to back up.
3. Browse to the location where you want to store the backup file.
4. In the **Options** box, click **Save configuration files**. This saves the citect.ini file. Also, click **Save sub-directories** and **Use Compression**.
5. Click **OK**.
6. Backup the citect.ini file from the Primary Server for later use in merging settings into the Secondary Server's citect.ini file.

The backup CTZ (Citect ZIP) file is written to the location that you chose during backup. You can open it with WinZip.

Back up Application Configuration Utility settings

Browse to the Power SCADA Operation installation directory, AppServices\bin directory (typically found in: C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\Applications\AppServices\bin).

Copy the `Configuration.xml` file.

Paste this file to the same location on the secondary Power SCADA Operation server.

Export One Line Engine Encryption

> **⚠ WARNING**
>
> **POTENTIAL COMPROMISE OF SYSTEM AVAILABILITY, INTEGRITY, AND CONFIDENTIALITY**
>
> - Store system keys, AES encryption files, or other files containing passwords to a secure site.
>
> **Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.**

Cybersecurity policies that govern how sensitive system files are securely stored vary from site to site. Work with the facility IT System Administrator to ensure that such files are properly secured.

To back up the one line engine:

1. Open the Application Configuration Utility:
   a. In Power SCADA Studio. click **Projects**.
   b. Form the **Power Applications** drop down, click **Application Config Utility**.
2. Expand **Applications** and then click **One Line Engine**.
3. Click **Redundancy**.
4. Click **Export Key**, navigate to the location where you want to export the encryption file. and select the AES file, and then click **Save**.

Save the AES file to a secure location, such as a secure network drive or a USB flash drive. Also, back up the `AdvOneLine.ini.txt` file. For redundant systems, copy these files to the Power SCADA Operation secondary server after accessing the AES file from that server during the restore process.

Export and import One-Time Password settings

You can copy and use one-time password settings on multiple server computers.

NOTE: When you import password settings into another server, you will overwrite any password settings that already exist there. You are not simply adding the new password settings to the existing ones.

To copy and use one-time password settings on multiple server computers:

1. In the Application Configuration Utility: click the **Security** drop down and then click **One-time Password**.
2. Click **Export**. A file named `ExportedOTPConfiguration.xml` is generated. You can rename it if you wish. Save it where you can access it from other servers, or copy it to a portable drive.

3. From a server to which you want to import the password settings, click **Import**. You are prompted for a location.
4. Browse to the location where you placed the XML file. Click **Open** and accept the XML file.

Configure the Power SCADA Secondary Server

NOTE: This section assumes that the Power SCADA project and the Primary Server are configured,

Complete the following tasks to configure the Power SCADA Secondary Server:

- "Restore the Power SCADA Studio project" on page 462
- "Import the One-Time Password" on page 463
- "Import the Advanced One-Line AES Encryption File" on page 463
- In the Application Configuration Utility:
  - Re-enter SSO passwords in . For more information , see "Configure Single Sign-On (SSO)" on page 388.
  - Re-enter the Citect Data Platform password. For more information, see "Set up data acquisition parameters" on page 152.
- Add INI edits to the standby server `citect.ini` file. Other settings from the primary server citect.ini file, such as I/O device parameters and any other customizations, will need to be added to the standby server `citect.ini` file.
- Configure the notifications. See "Notifications in a redundant system" on page 334 for details.

Restore the Power SCADA Studio project

To restore the project:

1. In Power SCADA Studio, click **Projects**  .
2. Click the **Backup** drop down and then click **Restore**.
3. Beside the **Backup file** text field, click **Browse**, and then browse to the location of the project file you will use to restore.
4. (Optional) Click **Select all included projects**.
5. In the **To** area, click **Current Project**.
6. In the **Options** area:
   a. Click **Configuration files** to restore backed up INI files and the TimeSyncConfig.xml file (used to store time synchronization settings).
   b. Click **Select sub-directories**. The sub-directories included in the earlier backup will be listed.
7. Click **OK**.

Import the One-Time Password

When you import password settings into another server, you will overwrite any password settings that already exist there. You are not simply adding the new password settings to the existing ones.

1. Open the Application Configuration Utility:
   a. In Power SCADA Studio. click **Projects**.
   b. Form the **Power Applications** drop down, click **Application Config Utility**.
2. Expand **Security** and then click **One-time Password**.
3. Click **Import**.
4. Browse to the location where you earlier placed the XML file.
5. Click **Open** and accept the XML file.

Import the Advanced One-Line AES Encryption File

To import the Advanced One Line encryption file:

1. Open the Application Configuration Utility:
   a. In Power SCADA Studio. click Projects.
   b. Form the **Power Applications** drop down, click **Application Config Utility**.
2. Expand **Applications** and then click **One Line Engine**.
3. Click **Redundancy**.
4. Click **Import Key**, navigate to and select the AES file, and then click **Open**.

After you access the AES file copy the `AdvOneLine.ini.txt` file to the Power SCADA Secondary server. You will now be able to access and use it.

Administering

Use the information provided in this chapter to administer a deployed, running Power SCADA Operation system.

Use the links in the table to find the content you are looking for:

Topic	Description
"Updating a running system" on page 464	Lists the changes that you can make to a live, running system.
"Add and modify user accounts" on page 359	How to modify user access to a deployed Power SCADA Operation system.
"Prerequisites" on page 420	How to edit a PSO to PME ETL job when devices change in either Power SCADA Operation or Power Monitoring Expert.

Updating a running system

This section describes configurations that you can make to the system while it is running (you do not have to restart the system).

Adding I/O devices, variable tags

In an architecture that has redundant or multiple I/O servers, new I/O devices and variable tags can be added to the project while it is running and online. Simply add the devices and tags, re-compile the project and restart just the associated I/O server processes to which the I/O devices were added. In the redundant architecture, this means you should restart just the associated primary I/O server process, while the associated standby I/O server remains up and running. After the primary I/O server is running again, restore the updated project on the secondary Power SCADA server machine and restart the associated standby I/O server process. In this manner, project uptime is not lost when you add I/O devices and variable tags to the running system.

Alarms, trends, reports

In the project, design an administration page containing a button used to execute the ServerReload Cicode function. Ensure that this page is only accessible by a logged-in user with the highest Administrator privileges. While the project is running, ensure that in the Citect.INI file located on the target Power SCADA Operation server machine, the [LAN]AllowRemoteReload parameter is set to "1". You can use the administrative ServerReload button to load subsequent changes to alarms, trends, and reports. (For a list of supported changes to alarms/reports/trends fields, see the Power SCADA Operation PC-based help file, "Server Side Online Changes" topic). Keep in mind that a "server reload" is not restarting the Alarm/Trend/Report server processes, nor is it rebooting the physical server machine. It simply re-loads the configuration databases into the running alarm/trend/report server processes.

Graphics pages

After modifying a graphics page, save the page and re-compile the project. In the HMI client, reload the page by navigating away from it and then returning to it. The updates to the page can then be seen in the HMI client, all while the project remains running.

New graphics pages

After adding new graphics pages, save the pages and re-compile the project. Restart the HMI client only. It is not necessary to restart any other server processes.

Other changes to project configurations

Changes to other configurations such as users, roles, menus, and Cicode require a full system restart of all server processes.

Debug logging

The following PWRMODBUS driver parameters can be changed without needing to restart the associated I/O server:

- DebugCategory
- DebugLevel
- DebugUnits

Assign and control user privileges

You need to give users appropriate levels of access, depending on the work they will do. For safety reasons, only advanced users should be given access to such features as controls and resets.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

- Do not use the software or devices for critical control or protection applications where human or equipment safety relies on the operation of the control action.
- Do not use the software to control time-critical functions.
- Do not use the software to control remote equipment without proper access control and status feedback.

Failure to follow these instructions can result in death or serious injury.

Because Power SCADA Operation lets you set user permissions on runtime graphical objects, thoroughly test the deployed project to ensure that permissions are applied as intended.

⚠ WARNING

POTENTIAL COMPROMISE OF SYSTEM AVAILABILITY, INTEGRITY, AND CONFIDENTIALITY

- Use cybersecurity best practices when configuring user access.

Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

Cybersecurity policies that govern user accounts and access – such as least privilege and separation of duties – vary from site to site. Work with the facility IT System Administrator to ensure that user access adheres to the site-specific cyber security policies.

For cybersecurity purposes, use Windows Authentication when you create user accounts.

Use Windows Integrated Users

You can incorporate Power SCADA Operation users and security options with the standard Windows security system. Using the integrated Windows security feature, the Windows user can log on to Power SCADA Operation runtime with runtime privileges and areas configured within the project. For a Windows user to be able to log on to runtime, it must be linked to a Power SCADA Operation "role," which is defined in the project with associated privileges.

To link a Windows user to a Power SCADA Operation role, add the "role" that specifies the Windows security group of which the Windows user is a member.

The pre-existing AutoLogin capability is extended to include the client, when the user is a Windows user, having an associated Power SCADA Operation role.

To invoke this functionality for a Windows user, you need to set the `[Client]AutoLoginMode` parameter in the `Citect.ini` file.

Instead of using auto-login when the system starts up, users can also log in to Power SCADA Operation using any Windows user credential that is a member of the linked group.

When the name of a Power SCADA Operation user also has the same name as a Windows user, the Power SCADA Operation user takes priority at runtime. However, if a valid Power SCADA Operation user login fails for some reason, the Windows user credentials will not be checked and an alert will be generated to advise that the login was not effective.

For more information, see Windows Security Usage Scenarios in the Citect SCADA help file (C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin).

Integrate with the Schneider Electric Security Access Module

If the Schneider Electric Security Access Module (SAM) is a part of the customer's solution, any domain/users created in the SAM can be used in Power SCADA Operation in the same manner as described above in Use Windows Integrated Users. To do this, add a role to the Power SCADA Operation project, and use the name of the SAM security group in the role's "Windows Group" field.

Operating

Use the information provided in this chapter to use the Power SCADA Runtime.

Use the links in the table to find the content you are looking for:

Topic	Description
"Log on to the Power SCADA Runtime" on page 467	Log into the Power SCADA runtime to access system features and tools.
"View the interface" on page 468	How to navigate the Power SCADA Runtime
"View the Alarms/Events Page" on page 470	How to view alarms. Includes information on how to filter alarms and change the form by adding or removing "Event/Alarm Log Columns Table" on page 473
"Use Security Viewer" on page 477	How to use the Security Viewer to monitor your system, and how to use the "Security Viewer Filter" on page 479.
"Use the Analysis Page" on page 480	How to use the Analysis Page to view trend data.
"Use the Equipment Pop-Up Page" on page 481	How to use the Equipment pop up page to see the detailed status of a particular device and to control the device.
"View the Tag Viewer" on page 487	Customizing advanced reports and design considerations for device communication in Power SCADA Operation with Advanced Reporting and Dashboards.
"Basic reports" on page 488	A description of the Power SCADA Operation reports.
"Use basic reports" on page 491	How to use basic reports as well as how to create "Rapid access labels (QR codes)" on page 499.

Log on to the Power SCADA Runtime

1. Launch the Power SCADA Runtime.
2. In the upper right corner, click **Login**.
3. Enter your user ID and password.

The features that are available will vary, depending on your user level.

Log in With a Programmed YubiKey and One-Time Password

Use this procedure to log in to Power SCADA Operation sing a YubiKey.

Prerequisites

The YubiKey is programmed and associated with a user in Power SCADA Operation, and the YubiKey is enabled.

To log into the system using YubiKey:

1. Insert the programmed YubiKey into a USB port of the Power SCADA Operation server.
2. Launch Power SCADA Operation Runtime, or access runtime using a remote Web Client.
3. Run the project you want to view.
4. In the upper right corner of the Startup screen, click **Login**.
5. In the Power SCADA Studio login screen, enter your name and password and then click **OK**.

   The One-time Password screen appears.
6. Press the button on the YubiKey.

The one-time password is generated. The key and software communicate behind the scenes to verify the uniqueness of the one-time password and to click OK.

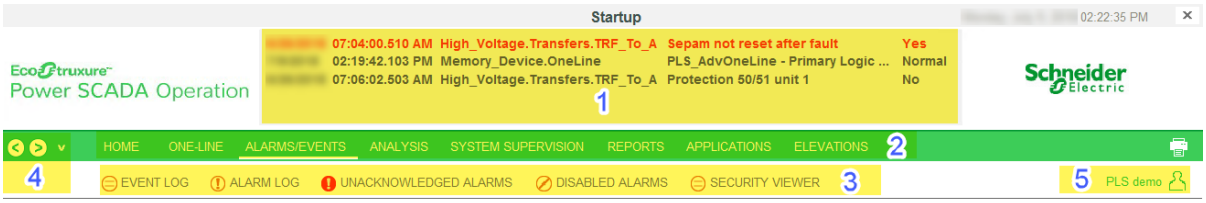
You can start using Power SCADA Runtime.

View the interface

After you log in to Power SCADA Operation, and you launch the Power SCADA Runtime, you see the individual landing page that has been created for this project. The Power SCADA Runtime includes a banner and a variety of tabs that open graphics pages.

NOTE: The graphics pages that appear in the Power SCADA Runtime are configurable and can vary greatly among projects. The pages that appear are defined by the Menu Configuration file for this project. If you need to change the appearance of tabs and menus, see "Use menu configuration to edit pagemenu.dbf (Change the graphics page appearance)" on page 270.

If your runtime is based on the Normal template from the pls_include_1 library, the Power SCADA Runtime banner consists of the following elements:



1 The alarm banner. It lists the last five active alarms.

2 Tabbed-style menu. Its contents are determined by the information entered in the Menu Configuration tool: "Add Pages to Project Menu Configuration" on page 301. If there are more links available than the ones that fit on the page, a small arrow displays at the right side of the row. Click the arrow to display a pop-up menu of the remaining links. Click a link in the menu to shift the contents of the row to make it visible for selection.

3 The upper row is typically used for organizing pages into several topics (or tabs). A typical system would include topics for one-line diagrams, alarms/events, analysis (for trends), and system supervision (allows you to view the network connection topics).
The lower row lists the links/pages under the topic that is currently selected in the upper row. If you select the one-lines topic on the upper row, the lower row displays all of the links to individual one-line pages.

4 These two arrows allow you to go back and forward one page in your navigation history. To see the history of visited pages, click the drop down arrow next to the right arrow. This displays a listing of visited pages (the current page is checked). To jump to a page in this list, click it in the menu.

5 The project name. The name of the user who is currently logged in.

Viewing one-lines

If the busbars and circuit breakers do not display as expected, it could be that a custom genie is not set up correctly. See "Create a new genie" on page 285 for details on customizing genies.

⚠ DANGER

EQUIPMENT ELECTRIC SHOCK, EXPLOSION, OR ARC FLASH

- Do not rely solely on the display of the graphic on the one-line.
- Verify that the device is physically locked out/tagged out before you work on the equipment or any downstream equipment.
- Ensure that all safety regulations and procedures have been followed before you work on the equipment.

Failure to follow these instructions will result in death or serious injury.

See "Enable lockout/tagout" on page 289 for instructions on enabling the lockout/tagout feature.

Communications loss

When there is a communication loss for a device, the genie or any part of the genie on the one-line page should have cross-hatches (gray dots) over the affected area, and a communication loss indication displays on the genie. An alarm should also annunciate. The color state before communication was lost will remain unchanged.

However, the indication of loss of communications does not filter through the entire bus animation: the downstream part of the drawing may still appear as if communication is working. When any part of a one-line drawing loses communication, do not continue to trust downstream readings until you address the loss of communication.

View the Alarms/Events Page

To view the alarms or events, click **Alarms/Events**, then click **Alarm Log** or **Event Log**.

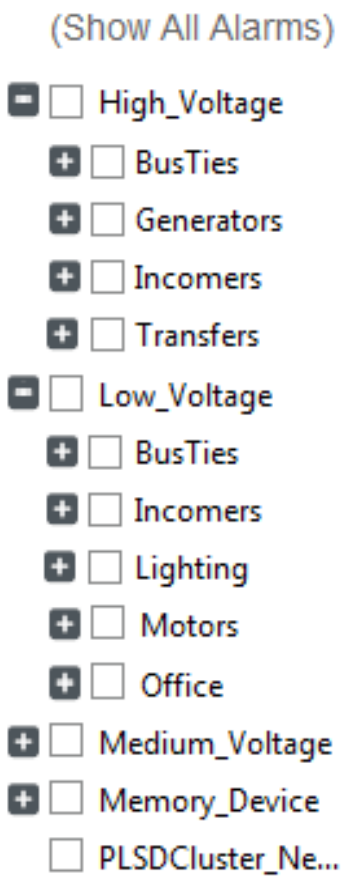
The Event Log displays all alarms and events that have occurred. The Alarms Log displays enabled alarms.

NOTE: If the alarms are not displaying correctly, see "When alarms do not display correctly" on page 509.

See The Alarm Log, below, for descriptions of color codes.

Equipment column

On the left side of the page, there is an equipment column. To hide or display this column, click the splitter:



All of the equipment in the project is listed. Most of the equipment is grouped by voltage level. By default, none of the names are checked, which means that information for all of them will display. To list alarms and events for a shortened list of equipment, check the box(es) to the left of the equipment name(s).

The number to the right of the equipment name is the number of active alarms for that equipment.

Filter information

To filter the information that displays, click **Filter** (just above the Date column). From the Alarm Filter window, you can select from a variety of filters. See "Alarm/Event filter form" on page 474 for more information.

Remove, insert, and move columns

To remove a column from the list:

Right-click its header and then click **Remove Column**.

To insert a column:

Right-click a column header, click **Insert Column**, and then from the dropdown list click the name of the column you want to insert .

The new column displays to the left of the column you right-clicked. If you right-click the white area to the right of existing columns, you will insert the column to the right of the last column.

To move a column:

Click the column that you want to move and then drag the column to the new position.

Sort by column

To sort on the information in a single column (such as the Equipment column), double-click the column header. It will toggle between ascending and descending order.

Event log

The Event Log lists alarm/event activity, most recent first (provides sequence of events information). The time is reported to the millisecond. You can display the Message column to see the most detail (such as, "Alarms disabled" and "Alarm xxx acknowledged").

Alarm log

To filter the alarms that display, click Filter (just above the Date column.) You can filter by date range, by text matches for various attributes, or by alarm type. See instructions on using the filter option in "Alarm/Event filter form" on page 474.

Notice the alarm colors:

- Acknowledged active alarms display in a normal red font.
- Unacknowledged active alarms display in a **bold red font**.
- Acknowledged inactive alarms display in a normal gray font.
- Unacknowledged inactive alarms display in a **bold gray font**.

Each alarm provides additional options. To view these options, right-click the alarm. Then you can do the following. Note that these changes will remain only until you leave the page. To set the order, use the parameters,

- Acknowledge or disable the alarm
- View alarm detail (similar to the genie status page in the one-lines of the runtime environment)

- view waveforms: (If the [equipment name Waveform] option does not display, there are no waveforms for this alarm.) Waveforms can display only if the device is set to “acquire on event,” and the waveform option is checked in the Profile Editor (see "Enable Waveforms" on page 210).

  When the waveform is available for viewing, the Search Waveform dialog displays. From this dialog, click Time Range, and then select the appropriate times; or click All Available to see all waveforms for this equipment. Click OK to display a list of waveforms that fit the date criteria. Highlight the waveform and click View.

  After the selected waveform displays, you can view a PDF file that describes the operation of the waveform viewer. Access this file (WaveWeb.pdf in the Citect Bin folder (64-bit example: C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin).

  Waveforms must be correctly set up before they will display. See "Enable waveforms for onboard alarms" on page 260 for more information. See "Use the Equipment Pop-Up Page" on page 481 for instructions on viewing waveforms.

  When you select the waveform option, you may see a message telling you “please try again after waveform has been acquired.” This means one of two things:

  - The alarm has been acquired at the device, but it has not yet been passed to Power SCADA Operation
  - The device was not set to acquire a waveform, and the waveform option was checked in the Profile Editor.

  NOTE: If there are multiple waveforms captures for this alarm, and if there is a disturbance waveform, it is the only one that is available here. If there are both an adaptive and transient, but no disturbance, the one with the earliest time stamp displays.

Unacknowledged alarms and disabled alarms

As with the Alarm Log, these logs display either unacknowledged alarms or disabled alarms. The sort and filter options operate as they do in the Alarm Log.

Alarm and events logging

Alarms from the Event Log can be saved to a file on the Alarm Server, thus protecting them from being lost when the FIFO size is passed. This feature is disabled by default, but it can be enabled by setting the FileFormat INI parameter.

[PLSEventLog] FileFormat: Determines the file format to be used for logging alarm/event data to disk files.

Allowable Values:

- 0 - (Disable)
- 1 - (CSV)
- 2 – (XML)
- Default value: 0

Acknowledge, silence, and print

Each of the logs includes these buttons:

Button	Description
	**Acknowledge Current Page of Alarms**: Click to acknowledge all of the alarms that display on the current page. **NOTE**: You can acknowledge individual alarms in this way: Right-click the alarm that you want to acknowledge, then choose Acknowledge. On a touch screen, tap twice on the alarm row to display the menu, then tap "Acknowledge."
	**Silence Alarms**: Click to silence all active alarms. This does not clear unacknowledged alarms or make alarms inactive; it only stops the audible portion of the alarm.
	**Print/Export Alarms**: Click to begin printing or exporting part or all of the log. Select All or the number of pages, then choose whether to print or export (to HTML file, which can then be opened in Excel or OpenOffice). When printing, the default location is: ..\ProgramData\Schneider Electric\Power SCADA Operation 9.0\Data **Notes:** • When printing: To avoid truncating data, choose the Landscape orientation. • When using Internet Explorer 8 and a dot matrix printer, you might have problems with overlapping columns in the printout. To solve this, either switch to Internet Explorer 7 or select a "square" matrix (e.g., 180 x 180 DPI).

See also: "Use the Equipment Pop-Up Page" on page 481

Event/Alarm Log Columns Table

This table lists the Citect column headings that are available for use when viewing the event log and alarm logs. To add a column to the table, right-click the column-heading row, then select Insert Column and choose the column from the list. The column displays to the left of where you right-clicked. To move columns left or right, drag and drop them. To insert a column to the right of the table, right-click the white space next to the existing columns.

To remove a column from the table, right-click its header and select Remove Column.

Column Name	Description
AlmComment	alarm log only: these entries come from the time-stamped digital alarm window “Comment” fields
Area	area, value needs to be set between 0 and 255
Category	event; or high, medium, or low alarm

Column Name	Description
Change	alarm logs only: when the alarm changes state: first state, second state
Cluster	cluster name to which the alarm belongs
Comment	alarm log only: displays comments from the alarm
Custom3 through 8	custom filters
Date	date (MMDDYYYY) that the event occurred or that the alarm annunciated
Description	description of alarm e.g., Sag Vcn or Under Voltage B-C
Equipment	default equipment name displays; used for alarm filtering and viewing
Help	help page
Location	Onboard or PC-Based
LogState	alarm logs only: The last state that the alarm passed through.
Millisec	alarm logs only: time (MS) that alarm annunciated
Operator	user name from the Citect users list
Priority	the alarm category's priority
Priv	privilege = security level
State	event log: state of the entry in the event log.. event alarm log: disappearance, appearance
Tag	alarm tag
Time	time (HH:MM:SS:MS) that alarm annunciated
Time Quality	This column displays the quality (accuracy) of the time stamp for alarms/events. Use the "Time Sync" filter to display only data that has confirmed time quality in the log (see "Alarm/Event filter form" on page 474 for instructions on enabling the filter). When there is no SER data, this column reads "No Time Sync Information." When the filter is set to Yes, the view displays only the available time sync information from SER devices. Note: If there is no SER data from any device and the filter is set to Yes, the entire log will be blank.

Alarm/Event filter form

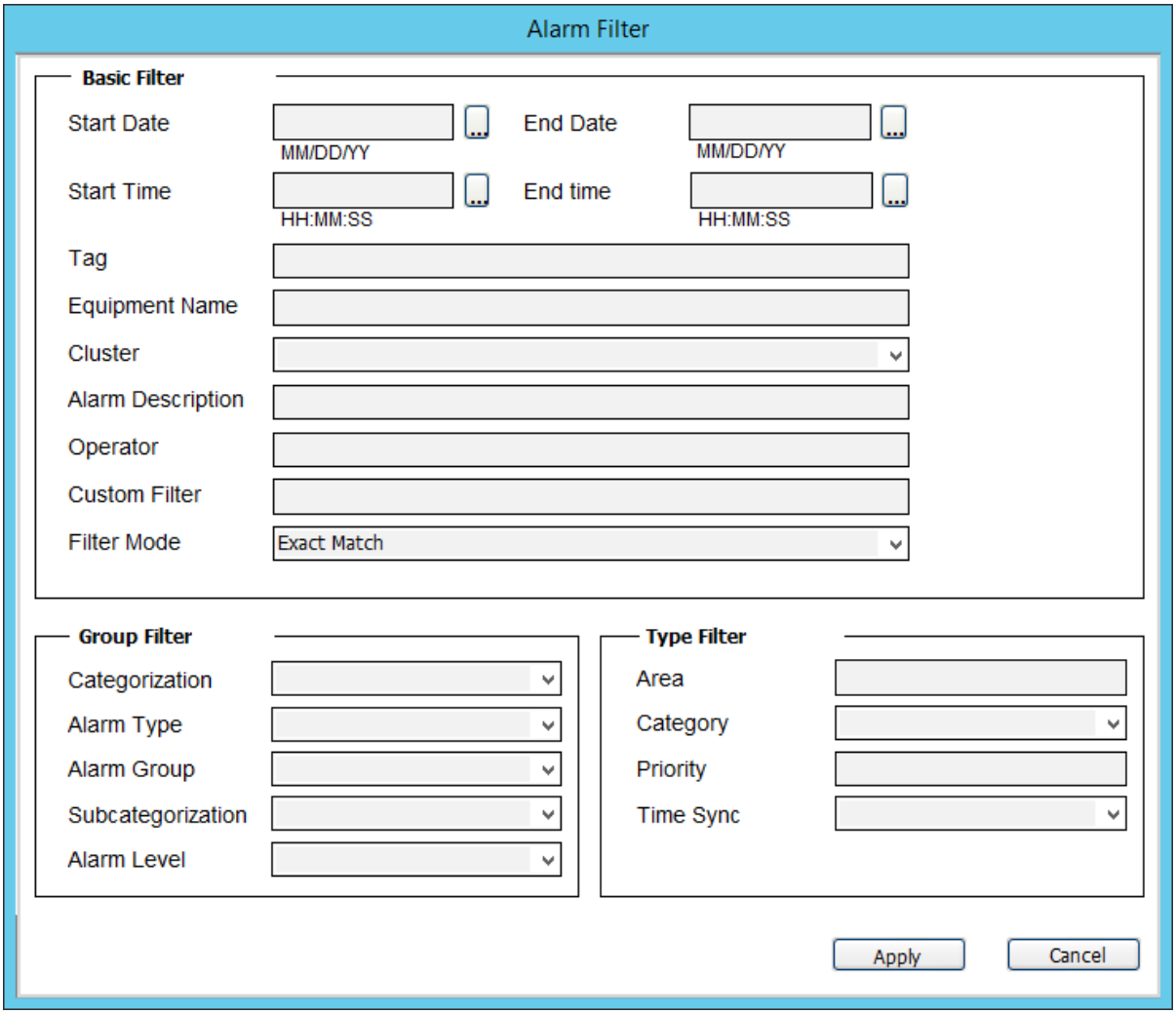
This topic describes the PLSCADA filter form. The information in the Citect filter form is the same, but is presented differently on the page. To change the filter form, use the UsePLSFilter parameter:

[Alarm] UsePLSFilter

default: 1 (use PLSCADA filter form)

Change to 0 to use the Citect filter form.

To filter for the information that displays in the alarm logs and the event log, click **Filter** (in the upper left corner of the screen). The Advanced Alarm Filter screen displays:



The table below describes its settings.

Filter Option	Description: Display all alarms for:
Basic Filter box:	
Start Date/End Date	a date range. Choosing only a start date displays alarms from that date to the current date. Choosing only an end date displays alarms for the past year up to that date. For example, to display alarms only for today's date, enter only a start date.
Start Time/End Time	a time range. Choosing only a beginning time displays alarms from that time through the end of the day (23:59:59 or 11:59:59 p.m.). Choosing only an ending time from the start of the day (00:00:00 or 12:00:00 a.m.) through the time selected.

Filter Option	Description: Display all alarms for:
Tag	a single tag; use tag name only, do not include equipment name. For example, enter MMXU1\A\phsA, not MainCM4\MMXU1\A\phsA. To filter on tag and equipment, enter the tag here and the equipment in the Equipment Name field.
Equipment Name	a device (entered when using the Profile Wizard or Automation Interface; (listed in Citect Explorer > System > Equipment)
Cluster	a single cluster, which was added when setting up the project (listed in Project Editor > Servers > Clusters)
Alarm Description	Alarm Desc from Time Stamped Digital Alarms: a customized on and off text description, such as “active” and “inactive”
Custom Filter	There are eight custom filters, which can be assigned by the customer in each alarm. A group of alarms in a specific location could have the same name in CUSTOM8 so that custom filtering can be easily applied. Custom8 has a default assignment of “Equipment.” To change custom filter assignments, use the AlarmFormat parameter (Project Editor > System > Parameters). This is the only means available for filtering on a custom field. When viewing the log, you can use the new custom filter by typing it into the Custom Filter field.
Group Filter box:	
Categorization	These “alarm filters” are created in the Profile Editor when alarms are created.
Alarm Type	
Alarm Group	
Subcategorization	
Alarm Level	
Type Filter box: These are advanced topics; see Power SCADA Operation help for more information.	
Area	the area associated with the alarm
Category	This is the alarm category. There are four predefined categories (high, medium, low, and event).You can assign alarms to their own categories by changing the equipment profiles and then re-generating the database. See the following table (Categories and Priorities) for a list of the categories and their defaults. Keep in mind that alarms that are categorized as events need to keep the category of _PLS_ALM_EVENT (category 1004).

Filter Option	Description: Display all alarms for:
Priority	This is the priority of the alarm category; not used in the default PLS_ Include project. As with the category, priority has defaults (see Categories and Priorities table below). You can change these settings in the equipment profiles. *However, be sure that you use priority 1 for events.*
Time Sync	Yes = in the Alarm or Event Log, only events/alarms with time quality information will be listed. The time sync data displays in the Time Quality column of the log. Data displays to the accuracy recorded at the device. Default: no

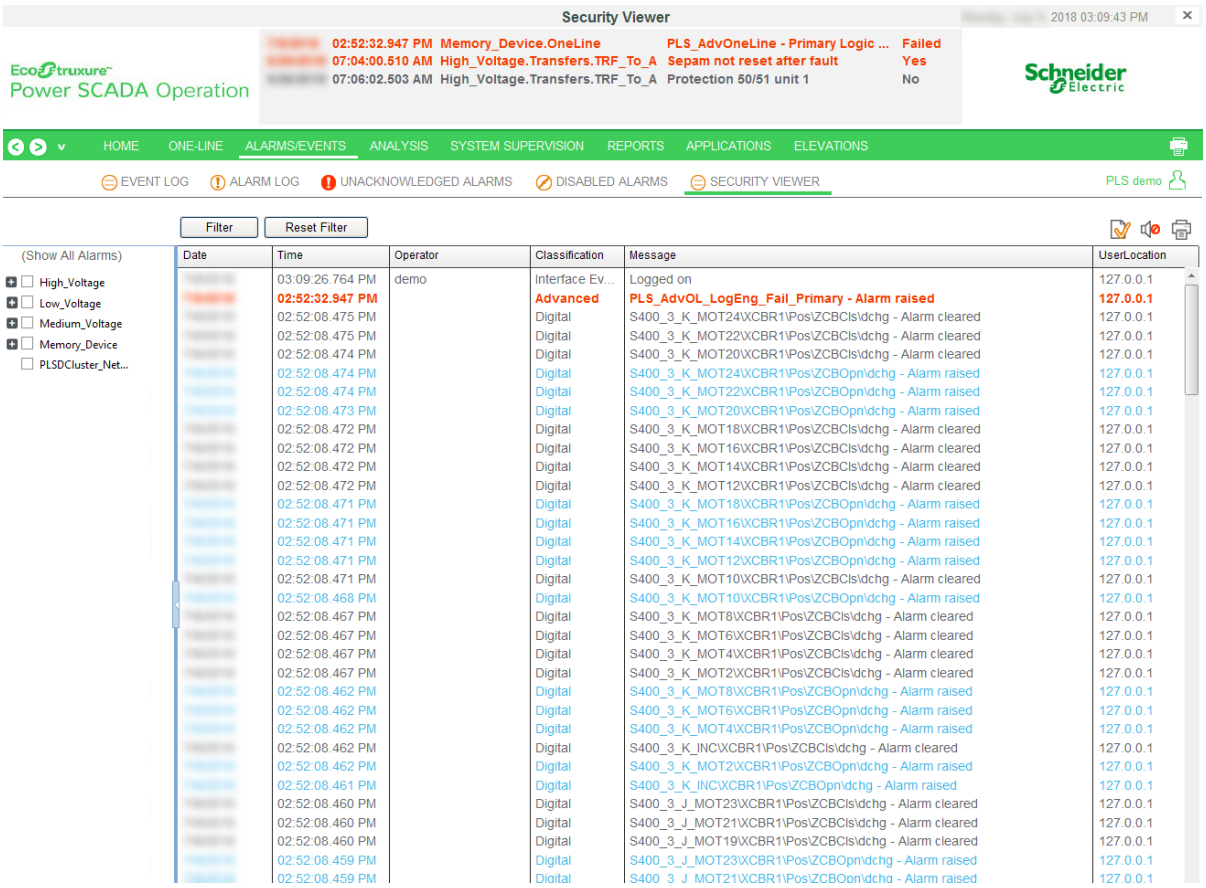
Category Label	Category Number	Priority Number
_PLSALM_HIGH	1001	1
PLSALM MEDIUM	1002	2
_PLSALM_LOW	1003	3
_PLSALM_EVENT	1004	0

Use Security Viewer

The Security Viewer lets you view user activity within your system. This screen lists all user actions that are captured in the Event Log.

To open the Security Viewer:

In the Power SCADA Runtime, click the **Alarms/Events** tab, and then click **Security Viewer**.



The screen displays a table with the following default columns:

Date	The date that the activity was logged
Operator	User name from the Citect users
Time	The time that the activity was logged
Classification	The class of the event.
Message	From the Message field in the Alarm Log
UserLocation	URL of the computer at which the activity occurred

For more information on these fields, see **Alarm SOE fields** in the Citect SCADA help file (...\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin\Help\Citect SCADA).

To change the view of the log, you can use any of the sort or filter features that are available in the Event Log.

There are 3 ways to filter information:

1. To the left of the log, check one or more of the devices in the system. This filters information to include data only for those devices. When nothing is checked, all devices are included.
2. You can insert and remove columns.

   To add a column:

   Right-click in the header area of the log, then choose **Insert Column**. From the list that appears, check an additional column title. The new column displays to the left of the column you clicked.

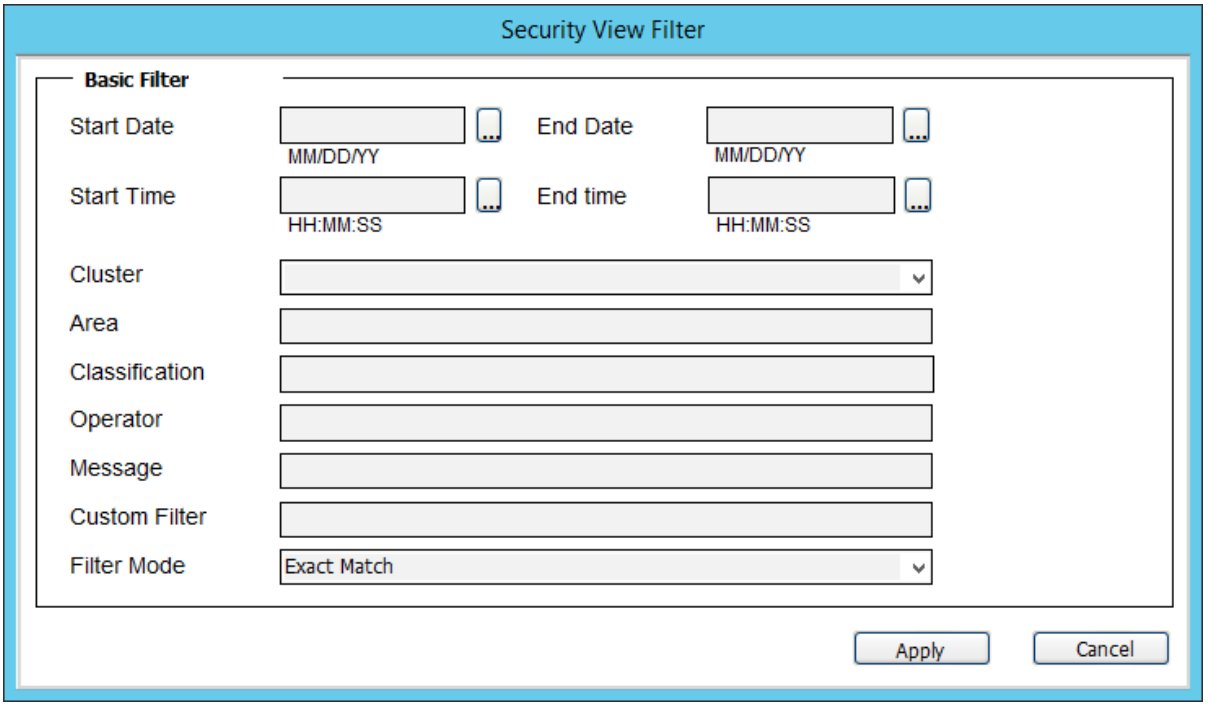
To remove a column:

Right-click on the header of the column you want to delete and then click **Remove Column**.

3. You can filter that data that is included. To do this, use the Security Viewer filter. For instructions on filtering the columns in the log, see "Security Viewer Filter" on page 479.

Security Viewer Filter

To filter for the information that displays in the security viewer log, click **Filter** (in the upper left corner of the screen). The Security View Filter screen displays.



The following table describes the Security View Filter settings:

Filter Option	Description: Display all alarms for:
Basic Filter box:	
Start Date/End Date	Choosing only a start date displays alarms from that date to the current date. Choosing only an end date displays alarms for the past year up to that date. For example, to display alarms only for today's date, enter only a start date.
Start Time/End Time	Choosing only a beginning time displays alarms from that time through the end of the day (23:59:59 or 11:59:59 p.m.). Choosing only an ending time from the start of the day (00:00:00 or 12:00:00 a.m.) through the time selected.
Cluster	This is a single cluster, which was added when setting up the project (listed in Project Editor > Servers > Clusters)
Area	Area (Between 0 and 255). See Alarm SOE fields in the Citect SCADA help file (...\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin\Help\Citect SCADA).

Filter Option	Description: Display all alarms for:
Classification	The class of the event. See Alarm SOE fields in the Citect SCADA help file (...\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin\Help\Citect SCADA).
Operator	The user ID of the person who has logged on Power SCADA Operation.
Message	This comes from the Message field in the Alarm Log.
Custom Filter	There are eight custom filters, which can be assigned by the customer in each alarm. A group of alarms in a specific location could have the same name in CUSTOM8 so that custom filtering can be easily applied. Custom8 has a default assignment of “Equipment.” To change custom filter assignments, use the AlarmFormat parameter (Project Editor > System > Parameters). This is the only means available for filtering on a custom field. When viewing the log, you can use the new custom filter by typing it into the Custom Filter field.

Use the Analysis Page

The Analysis Page offers 2 options for viewing data trends. In both options you must select the tags that are to be included. To be available for viewing in trends, a tag must be included in a device profile, and it must have the **Trend Tag** box checked.

Trend data is automatically logged when you check **Trend Tag** for tag and then add it to the project. If too many tags are chosen as trend tags, it could cause the hard drive to fill up.

NOTE: The maximum number of tags (pens) that will display correctly on the screen is ten. If you exceed ten pens, labels for these pens will not display correctly. Use one of these methods to correct this issue:

1. Enlarge the window to accommodate the extra pens/labels.
2. Write custom code to cause the labels to always be in the same position, overlapping each other when the trend pen is created. The user can then move the label around for better viewing.
3. As with option 2, control the label positions with code; but then, move the labels back to that same spot when a user selects the trend pen again.

There are 2 methods of calculating disk space usage: scaled and floating point. For more information on these calculations, see Calculating Disk Storage in the Citect SCADA help file (...\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin\Help\Citect SCADA).

Trending: Use this option to view historical trends. To select tags, click **Add Pen** on the toolbar:

Then associate the pen with a tag. By default, most trend data is polled every 15 minutes, and it is stored for one year in the trend tags, or until it is FIFO’d out. Some tags are polled every 5 seconds and are stored for two weeks. These tags are:

- Current A
- Current B
- Current C
- Apparent Power Total
- Reactive Power Total
- Real Power Total
- Voltage A-B
- Voltage B-C
- Voltage C-A
- Frequency
- Power Factor Total

Instant Trend: Use this option to view real-time trends. This allows viewing of data that is not set up for storage. To select tags for this trend, click **Instant Trend Selector** on the toolbar: 

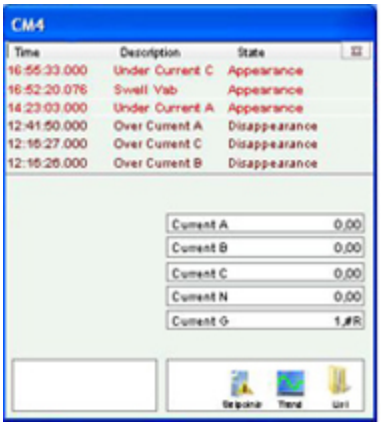
NOTE: If one of the pens returns a value of "1.#R,' the tag selected was not valid; no number could be reported for it. None of the values for any of the pens in the trend will be updated. To solve this issue, close the trend and open it again. This time, do not include the pen that gave the invalid return.

For either trending option, click **Help** for help using the tool: 

Use the Equipment Pop-Up Page

The pop-up page displays when you click on a device symbol/genie on a one-line page. This page shows a detailed status for a particular device. Some controls on this page are available only to users with certain privilege levels (see "Add and modify user accounts" on page 359 for user access levels).

One of two status pages displays. The page on the left illustrates the status page for a meter genie. The page on the right illustrates the status page for a circuit breaker genie.



At the top of the page, the most recent alarms and events are listed (racked in/out, Comms Loss, and so on). To view details about an individual alarm or event, right-click the alarm. You can view:

- A waveform. (If you do not see “Waveform” in the list when you right-click the alarm, there are no waveforms for this alarm.) Waveforms can display only if the device is set to “acquire on event,” and the waveform option is checked in the Profile Editor (see "Enable Waveforms" on page 210).

  When the waveform is available for viewing, it displays when you click this link. For information about how the waveform viewer works, see the WaveWeb.pdf file in the Citect bin folder (64-bit example: C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\9.0\bin).

  Waveforms must be correctly set up before they will display. If there are multiple waveforms, you must select from the list that displays (by default, the waveform search returns all waveform files acquired within the 24 hours prior to the time of the alarm). See "Enable waveforms for onboard alarms" on page 260 for more information.

  When you select the waveform option, and no waveforms are returned, one of two things is likely:

  - the alarm has been acquired at the device, but it has not yet been passed Power SCADA Operation
  - the device was not set to acquire a waveform, and the waveform option was checked in the Profile Editor

- Details about the device (currents, voltages, powers, resets, others.)
- You can acknowledge or disable the alarm. Acknowledged and disabled alarms are moved to their own sub-tabs.

On the left side of the of the status page, status messages display, based on the tags defined for equipment referenced in this genie. The list varies, depending on the device. Possible tags are:

- XCBR1\Pos Position (circuit breakers only)
- XCBR1\CCBRkdPos Racked Out (circuit breakers only)
- XCBR1\CBRkdPos Matching Fault/Trip Circuit Supervision (circuit breakers only)
- XCBR1\Loc Local/Remote (circuit breakers only)
- XCBR1\ESwPos Earth Switch (circuit breakers only)
- PTRC1\Op Tripped
- LPHD1\EEHealth Communication Failure

NOTE: For MicroLogic Type P devices, circuit breaker status fields will display #COM if the device does not have a CCM. Thus, you should not add any tags that refer to the CCM, such as Racked In/Racked Out.

On the right side of the page, real-time values will display for the tag type that you chose in the **Value** field when you added the genie in the design-time mode. For example, if you enter MMXU1\A\phsA as the value, you will see real-time currents here, as illustrated above. If you did not enter anything in the Value field when adding the genie, this area will be blank.

At the bottom left corner of the circuit breaker status page, Open, Close, Reset (for circuit breakers).

At the bottom right corner, are the Setpoints, Trend, and List options. See the following sections for descriptions.

Perform IEC 61850 advanced control

To begin using the advanced control feature, click **Check** in the lower left section of the window. See "Set up IEC 61850 advanced control" on page 313 for information on setup. See "Perform IEC 61850 advanced control" on page 485 for information on performing this advanced control.

View waveforms

After you select a waveform for viewing from the genie status page, the external waveform viewer displays it. For instructions on using the tool's analysis feature, see WaveWeb.PDF, located in the Bin folder of the Power SCADA Operation 9.0 Bin folder (example: C:\Program Files (x86)\Schneider Electric\9.0\Bin.

Enter setpoints for alarms

NOTE: Any time you change setpoints, you should immediately restart the project. Otherwise, setpoints will not be properly read (they will be truncated and either rounded down or up to a whole integer).

To add setpoints for alarms:

1. View the page, then click the genie for which you want to enter setpoints. A status window displays with the name of the genie.
2. Click **Setpoints**, then choose **Analog**, **Digital**, or **All**. When the Alarm Setpoints screen displays, select the first value you want to change. At the “keypad” screen (see below), enter the new value. Click **OK** to save it. Do this for each setpoint that you want to change.

Based on these setpoints, alarms can begin to display both in the alarms window at the top of the runtime screen and on the Alarms/Events tab (assuming you have set one up for this installation).

When there is a comms loss for a device, the last state before the loss happened is displayed on the screen.

The indication of loss of communications does not filter through the entire bus animation: the downstream part of the drawing may still appear as if communication is working. When any part of a one-line drawing loses communication, do not continue to trust downstream readings until you address the loss of communication.

View real-time trends

This option displays an historical trend. The data that displays is determined by the value that was selected in the Value Type field when this genie was added to the one-line page.

To view a trend:

1. From the one-line page in the runtime environment, click a genie to view its status window.
2. Click **Trend**, in the lower right corner. The Analyst screen displays for that trend.

You can select the timeframe for the trend. You can also uncheck phases to remove them from the trend, or highlight a phase to bring it to the front of the trend. For detailed information about the buttons on the screen, click “?” at the top of the page.

View lists of real-time information for the genie

To view lists of real-time currents, voltages; powers; resets and controls; and miscellaneous readings, click List, in the lower right corner, then click an item from the list: Currents, Voltages, Powers, Resets, or Others.

For resets and controls, which are interactive, you should assign users a high level of security. For a list of the default user levels, see "Add and modify user accounts" on page 359. That link also includes information on creating unique users.

When you click an item from the list, individual tag readings display for that tag type (depending on the tags that you have chosen for this device type). When you click any item in that list, the tag pop-up menu displays with these options: Trend, Override Tag, Control Inhibit Tag, and Tag Status. See Override Tag Status, below, for details.

Override tag status

From the list, you can right-click individual tags and override status settings. To access this feature, the user account must be at least level 4.

Trend: This link allows you to view a trend for the tag that you clicked.

Override Tag: You can use this feature to override a real-time value that is incorrect, or to test graphics. Enter the value that you want the system to "read" for this tag in the Override Value line. When you click **Apply**, the tag is highlighted. When you have finished the test, return to this list to remove the override.

Control Inhibit Tag: When this feature is ON, you will not be able to process writes for this tag. To enable this inhibit, click Apply for this tag from the list. The tag reading is highlighted. To disable this feature, return to the list view of this tag; click Remove.

You can perform control inhibit on an entire device. To do this, you will use the IODeviceControl Cicode function. For more information, see the **I/O Device Properties** topic in the Citect SCADA help file (...\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin\Help\Citect SCADA).

Tag Status: This screen views the status of the display value, override status, control inhibit status, and field value. You can also change the override status and control inhibit status on this screen.

Changing background colors: Default colors are assigned for the tag override and control inhibit. Change the default background colors in the parameters, not in the ini file.

To change the color for tag overrides, use OverrideTextBackgroundColor. To change the color for control inhibits, use ControlInhibitTextBackgroundColor. For detailed help, see Page Parameters in the Parameters.chm help file (Start > Programs > Schneider Electric > Power SCADA Operation 9.0 > Power SCADA Operation web-based help).

See also: "View the Alarms/Events Page" on page 470

Perform IEC 61850 advanced control

The advanced control window provides these options for IEC 61850 IEDs:

- Run synchro check on the selected equipment
- Run interlock check on the selected equipment
- Send a command to open or close the equipment

You can either check the features without sending an open/close command, or you can send an open/close command without running the checks.

NOTE: Only users who have privilege level of Engineer or Admin can perform these checks or operate the equipment.

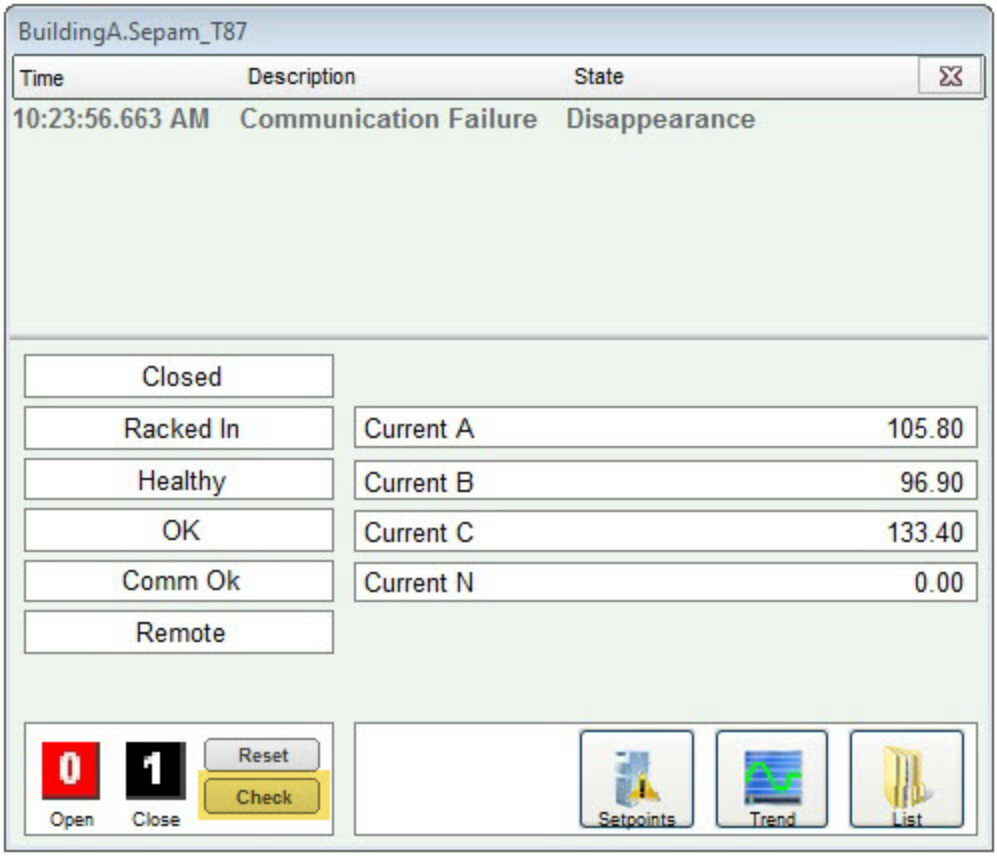
⚠ WARNING

INACCURATE DATA RESULTS

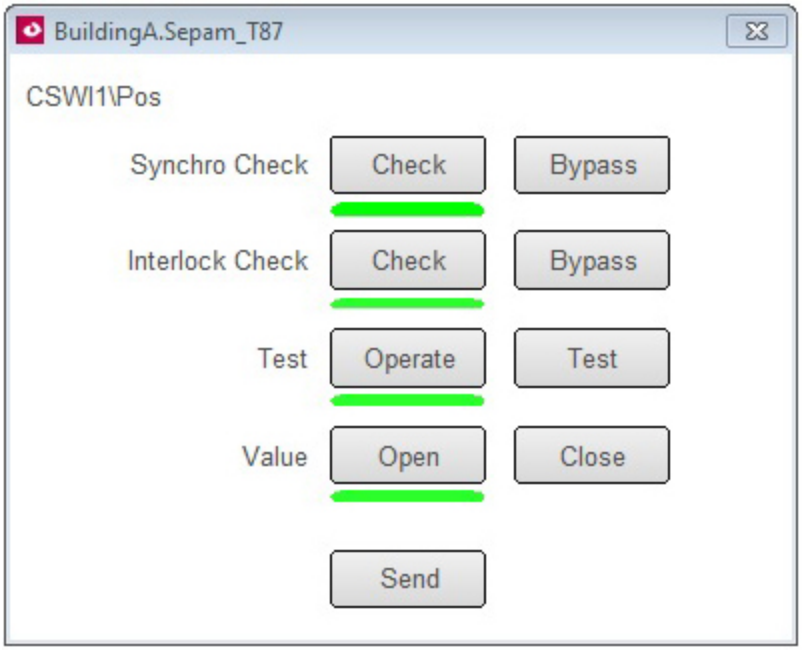
- Do not incorrectly configure the tag.
- Ensure that you understand the effects of using the "bypass" option so you do not shut down critical equipment.
- Do not rely solely on software messages and reports to determine if the system is functioning correctly or meeting all applicable standards and requirements.
- Consider the implications of unanticipated transmission delays or failures of communications links.

Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

To access the advanced control window, open the equipment genie pop-up page on the one-line. Note that after you enable this feature, there is a **Check** button on the lower left:



Click **Check**. The advanced control window opens:



Synchro Check: Use synchro check to verify that the waveforms for the equipment's power factor, voltage, and current are all aligned.

On the Synchro Check line, click **Check** to perform the synchro check, or click **Bypass** to ignore the synchro check. Default: Check.

Interlock Check: Use interlock to verify that there are no blocking conditions that need to be considered before switches are opened or closed.

On the Interlock Check line, click **Check** to perform the interlock check option, or click **Bypass** to ignore the interlock check. Default: Check.

Test: Click **Operate** if you want to send the command to the equipment and to complete the "value" setting. Click **Test** if you want to send the command to the equipment, and to verify the synchro and/or interlock statuses, but not complete the "value" setting. Default: Operate.

If you choose **Check** for the synchro and/or interlock checks and **Operate** for the Test line, the open/close operation will not occur if the equipment fails the checks.

Value: Choose the command that you want to send to the equipment: open or close. Default: Closed if the breaker is open; otherwise, Open.

Send: Click to send the command to the device to perform the action(s) that you selected.

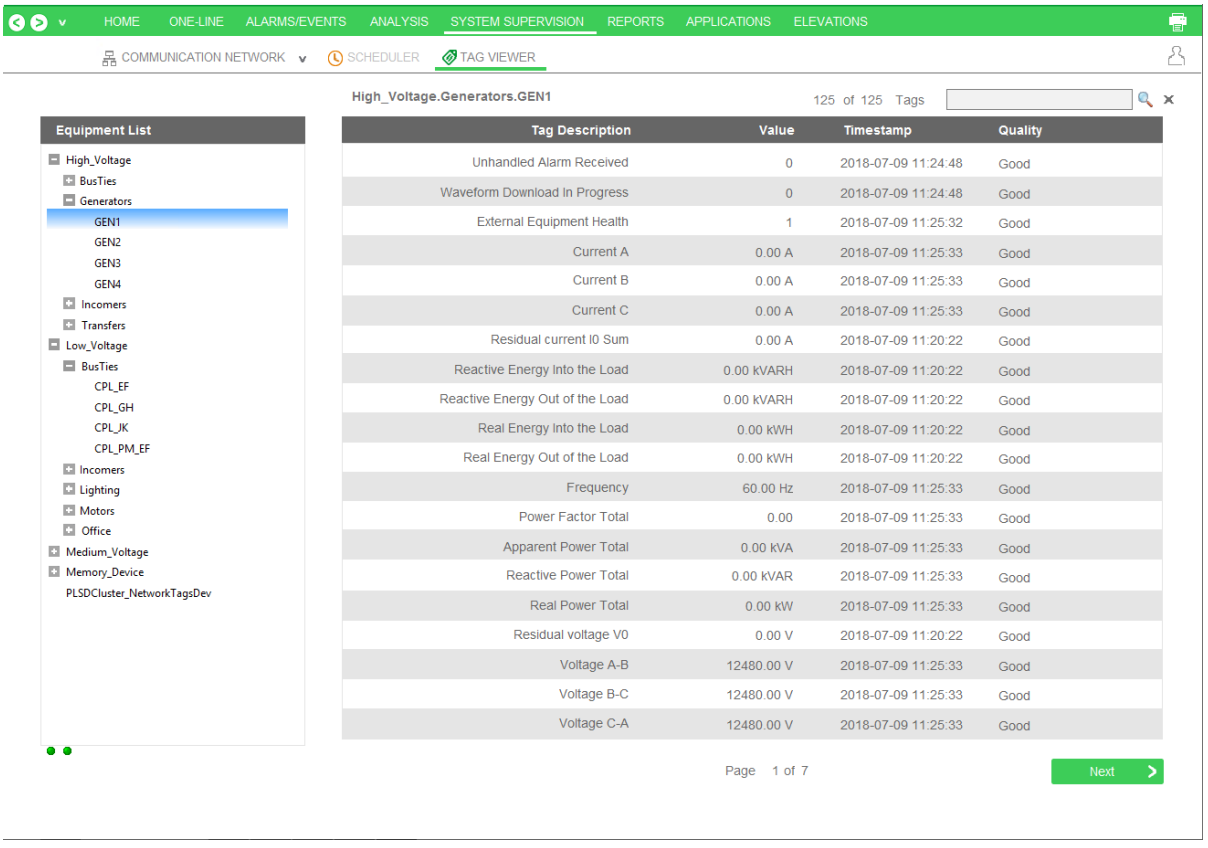
View the Tag Viewer

Use the Tag Viewer to learn the status of all of your project tags. This can provide information that you need to troubleshoot the project.

You can filter the tags that you view by individual equipment included in the project. You can also filter on strings that are part of the tag description or tag name. The tag viewer will work in all supported screen resolutions.

To view tags:

Click the tab for the page that was used when setting up the tag viewer, then select Tag Viewer. The viewer displays in a screen similar to this:



Note the following features:

Filter by equipment: The left-hand pane gives you the option to filter by equipment name. Most equipment is grouped by voltage level. You can select one equipment node, and you will view the tags for that equipment.

Filter by tag: In the upper right corner of the screen, type the tag name. You can type a string, such as "power factor," and you will retrieve a list of tags that have "power factor" in their tag description or tag name.

NOTE: Any time you display a tag, you add to the dynamic point count. See "Dynamic-point Count Licensing" in the Citect SCADA help file (default location: Program Files > Schneider Electric >Power SCADA Operation > v8.2 > bin) for more information about point counts.

The viewer includes the following columns:

- **Tag Description/Tag Name:** the description and name used when the equipment was added to Power SCADA Operation.
- **Value/Timestamp:** The real-time value that was read at the date/time shown.
- **Quality:** The data quality (for example, Good or Bad) of the tag from Power SCADA Studio.

Use **Previous** and **Next** to scroll through multiple pages.

Basic reports

You can create, view, save, and print basic reports in the Power SCADA Runtime.

Prerequisites

Before you can create and view basic reports, the following requirements must be met:

- You must set up reporting in the Power SCADA Runtime. See "Set up the Power SCADA Runtime for basic reports" on page 306.
- There must be data logged for the project. See "Use the Analysis Page" on page 480 for help.

NOTE: If the Schneider Electric CoreServiceHost has not been refreshed after devices or topics have been added, you should clear the cache and refresh the platform in order to access the new devices or topics. See "Clear cache and refresh platform" on page 373 for instructions.

After you have logged trend information, you can create and view basic reports. In the Power SCADA Runtime, click the **Reports** tab and then choose the basic report type you want to create:

- "Single Device Usage reports" on page 489
- "Multi Device Usage reports" on page 489
- "Tabular reports" on page 490
- "Trend reports" on page 491
- "Rapid access labels (QR codes)" on page 499

Single Device Usage reports

A Single Device Usage Report displays historical energy data from a single device and multiple topics. A single device report includes only usage and consumption topics.

NOTE: The report is optimized for up to five topics. If you choose too many topics, the chart legend can become unreadable.

To set up a Single Device Usage Report:

1. Browse to the Single Device Usage Report in the reporting web application. When prompted, enter your Power SCADA user account information. Click **Login**.
2. At the next screen, complete the following:
   a. Type a report title.
   b. In **Reporting Period**, choose the date range for this report, for example, *last week*.
   c. If you choose *Custom...*, the *Start Date/Time* and *End Date/Time* fields display. Enter the date and hour:minutes:AM/PM. (The date/time fields do not apply for the other reporting periods.)
   d. From **Period Grouping**, choose the interval by which you want to see the data reported. (The options here vary, depending on the date range selected.)
   If you leave the default *By Interval*, you will get every data point in the selected date range.
   e. Highlight the name of the device that you want for the report.
   f. Check the topics to be included.
3. Click **Generate Report**.

After the report is generated, it displays on the screen. It includes a usage summary table, and a graph and table for each topic you selected. You will probably have to page forward in the report to see all of the information.

For information about reading, exporting, printing, or editing reports, see "Working with basic reports" on page 494.

Multi Device Usage reports

A Multi Device Usage Report displays historical energy data for multiple devices and one topic. A multi device usage report includes only usage and consumption topics.

NOTE: If you choose too many topics, the chart legend can become unreadable.

To set up a Multi Device Usage Report :

1. Browse to the Multi Device Usage Report in the reporting web application. When prompted, enter your Power SCADA user account information. Click **Login**.
2. At the next screen, complete the following:
   a. Type a report title.
   b. In **Reporting Period**, choose the date range for this report, for example, *last week*.

c. If you choose *Custom...*, the *Start Date/Time* and *End Date/Time* fields display. Enter the date and hour:minutes:AM/PM. (The date/time fields do not apply for the other reporting periods.)

d. From **Period Grouping**, choose the interval by which you want to see the data reported. (The options here vary, depending on the date range selected.)
If you leave the default *By Interval*, you will get every data point in the selected date range.

e. Click the names of the devices for the report.

f. Highlight the topic to be included.

3. Click **Generate Report**.

After the report is generated, it displays on the screen. It includes a usage summary, a value table by interval for all of the devices selected, and a pie chart. You will probably have to page forward in the report to see all of the information.

For information about reading, exporting, printing, or editing reports, see "Working with basic reports" on page 494.

Tabular reports

A Tabular Report displays a system's historical data in a table format. Tabular reports can include one or more devices and one or more topics. A Tabular Report can include all available topics.

NOTE: The report is optimized for up to five topics. If you choose too many devices or topics, the chart legend can become unreadable.

To set up a Tabular Report:

1. Browse to the Tabular Report in the reporting web application. When prompted, enter your Power SCADA user account information. Click **Login**.
2. At the next screen, complete the following:
   a. Type a report title.
   b. In **Reporting Period**, choose the date range for this report, for example, *last week*.
   c. If you choose *Custom...*, the *Start Date/Time* and *End Date/Time* fields display. Enter the date and hour:minutes:AM/PM. (The date/time fields do not apply for the other reporting periods.)
   d. From **Period Grouping**, choose the interval by which you want to see the data reported. (The options here vary, depending on the date range selected.)
   If you leave the default *By Interval*, you will get every data point in the selected date range.
   e. Click the name(s) of the device(s) for the report.
   f. Click the topic(s) to be included.
3. Click **Generate Report**.

After the report is generated, it displays as a table on the screen. It lists data for all of the tags according to their timestamps. You will probably have to page forward in the report to see all of the information.

For information about reading, exporting, printing, or editing reports, see "Working with basic reports" on page 494.

Trend reports

A Trend Report displays a system's historical data in a trend (line) and table formats. Trend reports can include one or more devices and one or more topics. A Trend Report can include all available topics.

NOTE: The report is optimized for up to five topics. If you choose too many topics, the chart legend can become unreadable.

To set up a Trend Report:

1. Browse to the Trend Report in the reporting web application. When prompted, enter your Power SCADA user account information. Click **Login**.
2. At the next screen, complete the following:
   a. Type a report title.
   b. In **Reporting Period**, choose the date range for this report, for example, *last week*.
   c. If you choose *Custom...*, the *Start Date/Time* and *End Date/Time* fields display. Enter the date and hour:minutes:AM/PM. (The date/time fields do not apply for the other reporting periods.)
   d. From **Period Grouping**, choose the interval by which you want to see the data reported. (The options here vary, depending on the date range selected.)
   If you leave the default *By Interval*, you will get every data point in the selected date range.
   e. Click the name(s) of the device(s) for the report.
   f. Click the topic(s) to be included.
3. Click **Generate Report**.

After the report is generated, it displays on the screen. It includes a trend for each topic included (selected data points over the period of the trend) followed by a table with every timestamp in the period selected. You will probably have to page forward in the report to see all of the information.

For information about reading, exporting, printing, or editing reports, see "Working with basic reports" on page 494.

Use basic reports

You can use the following tasks within the reporting application to create, view, and email basic reports:

- "Create and view basic reports" on page 492
- "Configure email settings to send basic reports" on page 308

- "Email basic reports" on page 496
- "Read, Export, Print, and Edit Basic Reports" on page 500

Create and view basic reports

Create basic reports and save report configurations using a Web browser such as Internet Explorer.

For information on interacting with the reporting Web application in the Power SCADA Runtime, see "Set up the Power SCADA Runtime for basic reports" on page 306.

You can create basic reports in two ways:

1. Run a new report by entering parameters
2. Run a report from a saved configuration

If you plan to view a basic report using "Rapid access labels (QR codes)" on page 499, you must save a configuration. After it is saved and you generate a rapid access label, do not change the configuration name. If the configuration name is changed, you must generate a new rapid access label.

NOTE: For Windows 2008 R2, Windows 7, or Windows XP operating systems, additional formatting might be required. For more information, see "URL routing for basic reports" on page 312.

Run a new basic report

There are two ways to run a new basic report:

1. Browse to the report URL using the following format:

   `http://<ServerName>/Reporting/Report/<ReportName>`

   where:

   `<ServerName>` = the name or IP of the reporting server

   `<ReportName>` = the name of the report you want to view (MultiDeviceReport, SingleDeviceReport, TabularReport, TrendReport)

   OR

2. Browse to the default reporting URL, and click the report you want to view using the following format:

   `http://<ServerName>/Reporting/`

   where:

   `<ServerName>` = the name or IP of the reporting server

Run a basic report and save its configuration

To create and save a basic report configuration:

1. Browse to the build configuration URL of the report you want to create, using the following format:

`http://<ServerName>/Reporting/Report/<ReportName>/BuildConfiguration`

where

`<ServerName>` = the name or IP of the reporting server

`<ReportName>` = the name of the report you want to view (MultiDeviceReport, SingleDeviceReport, TabularReport, TrendReport)

2. Enter the report query parameters.

   After the report runs, a text box displays at the bottom containing the XML of your saved report configuration.

   NOTE: If you enter a fixed date range, all reports that you generate with this configuration will use that date range. The best practice is to use one of the relative date ranges, such as "last month."

3. Copy the entire contents of the text box into a text editor of your choice.
4. Save this new file to the `Reporting\ReportConfigurations\` directory, located on the application root install directory (which is also the physical directory behind the reporting web application's virtual path in IIS).

   Example (64 bit):
   `C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\Power SCADA Operation Reporting\Reporting\ReportConfigurations\`

   The file name must be in the following format:

   `<ReportName>_<ConfigurationName>.cfg`

   where:

   `<ReportName>` = the name of the report you want to view (MultiDeviceReport, SingleDeviceReport, TabularReport, TrendReport)

   `<ConfigurationName>` = a name for this configuration (alphanumeric only)

   NOTE: If you use Notepad, ensure that you apply the correct file extension (.cfg), not the default (.txt).

View a basic report using a saved configuration

Viewing a basic report with a saved configuration runs the report directly with the saved configuration (you cannot change the parameters).

To view a basic report with a saved configuration:

1. Browse to the URL of the report and specify the configuration using the following format:

   `http://<ServerName>/Reporting/Report/<ReportName>/<ReportConfiguration>`

   where

<ServerName> = the name or IP of the reporting server

<ReportName> = the name of the report you want to view (MultiDeviceReport, SingleDeviceReport, TabularReport, TrendReport)

<ReportConfiguration> = the name of the saved configuration to use

Modify and view a basic report using a saved configuration

To modify a previously saved configuration:

1. Browse to the show configuration URL for the report that you want to modify using the following format:

   http://<*ServerName*>/Reporting/Report/<*ReportName*>/<*ReportConfiguration*>/ShowConfiguration

   where

   <ServerName> = the name or IP of the reporting server

   <ReportName> = the name of the report you want to view (MultiDeviceReport, SingleDeviceReport, TabularReport, TrendReport)

   <ReportConfiguration> = the name of the saved configuration to use

2. Run the report as you normally would, editing selections on the parameter entry page as necessary.

   After the report runs, a text box displays at the bottom containing the new XML of your saved report configuration.

3. Copy and paste this new XML into your saved configuration file (overwriting the old XML).

Remove a saved configuration

To remove a saved configuration, delete the saved configuration file from Reporting\ReportConfigurations\ directory.

Example (64 bit):
C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\Applications\Reporting\ReportConfigurations

Working with basic reports

After you create a basic report, you can:

- Change its appearance (Page Setup)
- Change its view: HTML or PDF
- Scroll to the beginning, previous page, next page, or last page
- Export it to a variety of formats
- Print it
- Email it

The report toolbar options are described in the following table:

Report Toolbar Option	Description
Parameters/Report	Toggle between viewing the parameters (setup) page and the report.
Parent/Child reports	Not currently used.
Hide/Show	Not currently used.
Page Setup	Click this link to open the Page setup window, where you can determine paper size, and page orientation and margins.
Print	Click this link to print the report. NOTE: For best formatting of the report, export it to PDF, and then print.
Print Preview	In HTML mode, click this link to view the print output.
Viewer Mode	You can view in HTML or PDF mode. Select the mode, then click Viewer Mode to change the view.
Viewer Mode Set ()	Click to confirm the choice of viewer mode.
Pagination 	Click the left and right arrows to page backward and forward in the report. Or type the page number you want to see.
Select a format	For exporting, choose the format (not HTML) that you want.
Export ()	See instructions below for exporting a report.
Email ()	Click this link, and then enter the requested information. Click **Send**. For other ways to email reports, see "Email basic reports" on page 496

Export a basic report

You can export to the following file types:

- PDF
- Web Archive
- Word Document
- XML
- XLS Document

NOTE: Before you can print a basic report, you must export it into a format that can be printed.

To export a basic report:

1. While viewing the report, select a format, then click **Export** ( ).
2. Type the location at which you want to save the file.
3. Set any other properties you wish.
4. Click **Export**.

Edit the basic report appearance

With the report displayed, you can:

- Change the paper size
- Change the paper source
- Change the page orientation
- Change the page margins
- Change the number of pages per sheet
- Add a watermark

Email basic reports

Before you can email Power SCADA Operation basic reports, configure the SMTP server and email list(s). See "Configure email settings to send basic reports" on page 308 for details.

There are 3 ways to email basic reports:

1. The Report Viewer email button
2. Visit a Specific URL
3. Use Cicode via ReportMailer

Report Viewer email button

Use this method to send a customized one-time email to an individual or group of email addresses.

1. Run the report as normal.
2. In the Report Viewer, click (**Email**) .
3. Enter the requested information in the pop-up dialog.
4. Click **Send**.

Visit a Specific URL

NOTE: Each visit to a URL causes the email to be sent. Be sure that you have the correct report and email list before you visit this URL/send the email. Also, you should secure this URL using the web.config file. For information on modifying/using the web.config file, see http://support.microsoft.com, and search on kb 815179.

To send a basic report to an existing email list, visit the following URL:

`http://<`*`ServerName`*`>/Reporting/Report/<`*`ReportName`*`>/<`*`ReportConfiguration`*`>/Email/<`*`EmailList`*`>`

where:

- <ServerName> = the name or IP of the reporting server
- <ReportName> = the name of the report you wish to view
- <ReportConfiguration> = the name of the saved configuration to use
- <EmailList> = the name of the email list you wish to use

You must use a saved configuration (see "Create and view basic reports" on page 492 for instructions). You cannot change report parameters from this URL.

No progress bar or update will display, as these interfere with some scheduling clients.

Use Cicode via ReportMailer

You can use a utility called ReportMailer to email basic reports. This command line utility is located in the PLS_Include project. It can be called by Cicode. You can create a button on the graphics page and have it call the Cicode function or use a scheduled process to trigger an email.

Before you can use ReportMailer, you need to create or edit the file called `ReportMailer.ini` file that is in your project (not in PLS_Include). The `ReportMailer.ini` file must include the text listed in the following table:

Text Field	Required Setting	Description
LoginUsername	demo	Username for logging in to reporting system for emailing reports
LoginPassword	demo	User's password, will be encrypted on the first run
IsEncrypted	False	Flag that indicates if the password is encrypted. If you change the password, edit the field (replacing the unreadable encrypted entry, if one exists). Then change this value to False. The new password will be encrypted at the next startup cycle, and this field will be updated to True.
ScadaBinPath	C:\Program Files (x86)\Schneider Electric\Power SCADA Operation 9.0\Bin	The bin path of Power SCADA Operation
LogOnUrl	http://SCADASERVER/Reporting/LogOn.aspx	The URL of the logon page(this is an example; use your own server name)
ReportServerName	SCADASERVER	The name or IP address of the server running the reporting application

Text Field	Required Setting	Description
LogLevel	All	The level of logging you want in the report mailer application. This log is saved to a ReportMailerLog.txt file in the running project's directory. Possible settings are ALL, DEBUG, ERROR, WARN.

After this file is configured, run the `ReportMailer.exe` with the following syntax:

`ReportMailer.exe <ReportName> <ReportConfiguration> <EmailList> <ScadaProjectPath>`

where:

- <ReportName> = the name of the report you wish to view
- <ReportConfiguration> = the name of the saved configuration to use
- <EmailList> = the name of the email list you wish to use
- <ScadaProjectPath> = the full path to your SCADA project

This command line application may be called from Cicode using the following example:

```
FUNCTION
PLS_EmailReport()
ErrSet(1);
STRING FilePath = ParameterGet("CtEdit","User","") + "\PLS_Include\
ReportMailer.exe " + "MultiDeviceReport SampleConfiguration SampleList
" +
"^"C:\ProgramData\Schneider Electric\Power SCADA Operation\User\PLS_
Example^"";
Exec(FilePath);
END
```

NOTES:

- The SCADA project path must be enclosed in escaped quotes ("^").
- This is an asynchronous (non-blocking) call. While the EXEC() method will return immediately, it may take a few moments to run and email the report. See the web.config timeout value (see option 2 above) for more information.
- You can also call the ReportMailer application directly from a command line. In this case, you can add the term "blocking" to the command line (as a fifth parameter). This causes ReportMailer to act in a synchronous state (block the call) and to return any error messages to the console. Never use the "blocking" parameter by Cicode, as it could prevent EXEC() from returning in a timely fashion.

Scheduling basic reports

You can schedule the emailing of basic reports by executing the above Cicode as an action from a timed event. For more information, see **Configuring Events** in the Citect SCADA help file (...\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin\Help\Citect SCADA).

You can also use the Windows Task Scheduler to send these reports. Refer to Microsoft's documentation on Using the Task Scheduler (Microsoft Docs).

Rapid access labels (QR codes)

Use this report to create quick-response code (QR code) stickers that can be placed on your system equipment to provide quick access to Power SCADA Operation standard reports and LiveView table views. You can also generate a label for any URL. After you create and print the code stickers, you can read them with a smartphone or QR code reader.

Before you begin

Make sure that you have completely configured your system. This includes:

- Set up all servers, equipment, and addressing
- Create the reports and LiveView views that you want to see. Note that the report configuration names and LiveView view names cannot be changed after you create the QR stickers, as the links would be broken to the reports/table views.
- To view a report, you must first save the report configuration. See Run a Report and Save its Configuration in "Create and view basic reports" on page 492 for instructions.
- All devices that will be used to scan QR codes must be on the same network with the server.
- Purchase the label stock paper for the labels you will print. Use **Avery 6578 Label Stock**, or equivalent. Other label stock may not be spaced correctly, which could result in the labels not printing correctly.

Create the sticker

To create the sticker:

1. Ensure that you have a laser printer set up and available for printing.
2. At the printer that you will use to print the labels, insert the blank label stickers. Use **Avery 6578 Label Stock** or equivalent.
3. Browse to the Rapid Access Labels report in the reporting web application. When prompted, enter your Citect user account information. Click **Login**.

   The Rapid Access Labels screen displays.
4. From **Server Address**, choose the IP address that is connected to the same network as the wireless access points.
   Do not use an IP provided by DHCP, as the IP address can change frequently.
   If your network supports DNS, we recommend that you use the machine name of the server.
5. In the **Port** box, accept the default "80" or, if necessary, enter a different port.
6. In the **Select items to generate labels** box, check the report configuration(s) and LiveView table(s) for which you want to print stickers.
7. (Optional) You may want to print a sticker for a different URL (such as a corporate website). To do this, enter the URL in the **URL** line of the **Manual URL Entries** box (the site name automatically displays in the upper box).
8. (Optional) On the **Caption** line, you can type any text that you want to have printed above the QR code on the sticker. If you want the output table or report to have a title, enter it here.

9. Click **Generate Report**.

> **NOTE**: To print correctly, use the icon on the report control bar, not the one from the browser (which would add a header and footer, and throw off alignment).

Read the sticker

Stickers print at the designated printer. Each sticker has a title that is one of the following:

- A report configuration name
- A LiveView name
- User-entered text from the *Caption* text box

Place each sticker in the desired location, such as next to the device that is being monitored.

To read a sticker, use a smartphone or QR code reader. The reader must have access to the network and server. We recommend that you use the QR Droid application if you are viewing reports/tables from an Android phone.

Troubleshooting

If you cannot read the QR code, verify the following:

- Your smartphone or reader has access to the wireless network, and the server can be reached by the IP you selected when generating labels.
- The server address and port name are correct.
- The report configuration name or LiveView table name are correct, and have not been changed or deleted.

Read, Export, Print, and Edit Basic Reports

After you create a basic report, you can:

- Change its appearance (Page Setup)
- Print it
- Change its view: HTML or PDF
- Scroll to the beginning, previous page, next page, or last page
- Export it to a variety of formats
- Email it

The toolbar options table below describes these options.

Option	Description
Parameters/Report	Toggle between viewing the parameters (setup) page and the report
Prent/Child reports	Not currently used.
Hide/Show	Not currently used.
Page Setup	Click this link to open the Page setup window, where you can determine paper size, and page orientation and margins.

Print	Click this link to print the report. NOTE: For best formatting of the report, you should export to PDF, and then print.
Print Preview	In HTML mode, click this link to view the print output.
Viewer Mode	You can view in HTML or PDF mode. Select the mode, then click Viewer Mode to change the view.
Viewer Mode Set	Click to confirm the choice of viewer mode.
Pagination	Click the left and right arrows to page backward and forward in the report. Or type the page number you want to see.
Select a format	For exporting, choose the format (not HTML) that you want.
Export	See instructions below for exporting a report.
Email	Click this link, and then enter the requested information. Click Send. For other ways to email reports, see "Email basic reports" on page 496

Export a Basic Report

Before you can print a basic report, you must export it into a format that can be printed.

You can export to:

- Acrobat (Pdf) File
- Web Archive
- Word Document
- XML File
- XLS Document

To export a report:

1. While viewing the report, select a format, then click Export.
2. Type the location at which you want to save the file.
3. Set any other properties you wish.
4. Click Export.

Edit the Basic Report Appearance

With the report displayed, you can:

- Change the paper size
- Change the paper source
- Change the page orientation
- Change the page margins
- Change the number of pages per sheet
- Add a watermark

Troubleshooting

This section contains hints and instructions for correcting issues with your project.

- For diagnosing and troubleshooting problems with I/O device communications and data quality use the I/O Device Settings, see "Diagnostics Overview" on page 502.
- Use the One-Line Configuration Utility to repair problems with equipment on graphics pages: see "One-Line Errors and Warnings" on page 507.
- General troubleshooting questions: see "Frequently Asked Questions (FAQs)" on page 510.

To learn the status of all of your project tags, see "View the Tag Viewer" on page 487 .

Application Services Logging

Logging Level:

This feature turns on extra diagnostic information that can be useful when diagnosing problems that occur in application services or its hosted applications (such as LiveView). Choose the level of logging to be used in all applications. Debug and Verbose increase the amount of information that is logged during runtime for applications such as Basic Reports and LiveView.

- Normal: Use when the project is live.
- Debug: includes additional logging statements (in the Windows event log named PowerLogic). This logging should not affect performance in the system during runtime.
- Verbose: releases additional diagnostic information, such as large lists, that could affect system performance.

Service Inventory:

This is a read-only list of Web services hosted by the Schneider Electric CoreServiceHost, details about them, and whether they are running.

Diagnostics Overview

Diagnostics provides visibility into your Power SCADA Operation system to help you understand how the system is organized, monitor performance, and troubleshoot issues.

Offering several views of the components that make up your system, Diagnostics displays general information, settings, live data, issues, and more for system components including:

- Machines
- Servers
- Ports
- Devices

Designed to simplify system navigation and troubleshooting, you can use Diagnostics to identify and address issues that could negatively impact system performance or data integrity.

Offline and Online Mode

Diagnostics includes 2 modes: **Offline** and **Online**. Each mode provides unique functionality:

- **Offline** – Save or print a timestamped configuration settings report.
- **Online** – View the health of a running system and set the data refresh rate.

You can also make changes to component configuration settings in both modes.

NOTE: For redundant systems: The primary and secondary systems remain in sync; any changes made to devices on one system must also be made on the other system.

For more information on the views available in both modes, see Navigating Diagnostics.

Generating a Timestamped Configuration Settings Report

To save or print a configuration settings report in offline mode:

1. In the **Application Configuration Utility**, select your project, and then select **Diagnostics** > **I/O Device Settings**.
2. To save or print a timestamped
   a. System configuration settings report, click **Display All Settings**.
   b. Component configuration settings report, select the component and click **Display Selected Settings**.
3. Click **Copy** to save the report to the desktop, or **Print** to print it.

Setting the Data Refresh Rate

To set the data refresh time in online mode:

1. In Power SCADA Operation Runtime, click **Analysis** > **Diagnostics**.
2. To set the refresh rate, click and then select:
   a. **Manual** to limit data refreshes to a manual click of the**Update Now** button.
   b. One of the time value options between 5 seconds and 5 minutes.
3. Click to save your selection and close the window.

For more information, see Navigating Diagnostics.

Navigating Diagnostics

Navigating your system in Diagnostics helps you understand how a Power SCADA Operation system is organized, as well as the various components that comprise the system.

Diagnostics includes 2 modes: **Offline** and **Online**. Each mode provides unique functionality:

- **Offline** – Save or print a timestamped configuration settings report.
- **Online** – View the health of a running system and set the data refresh rate.

You can also make changes to component configuration settings in both modes.

NOTE: For redundant systems: The primary and secondary systems remain in sync; any changes made to devices on one system must also be made on the other system.

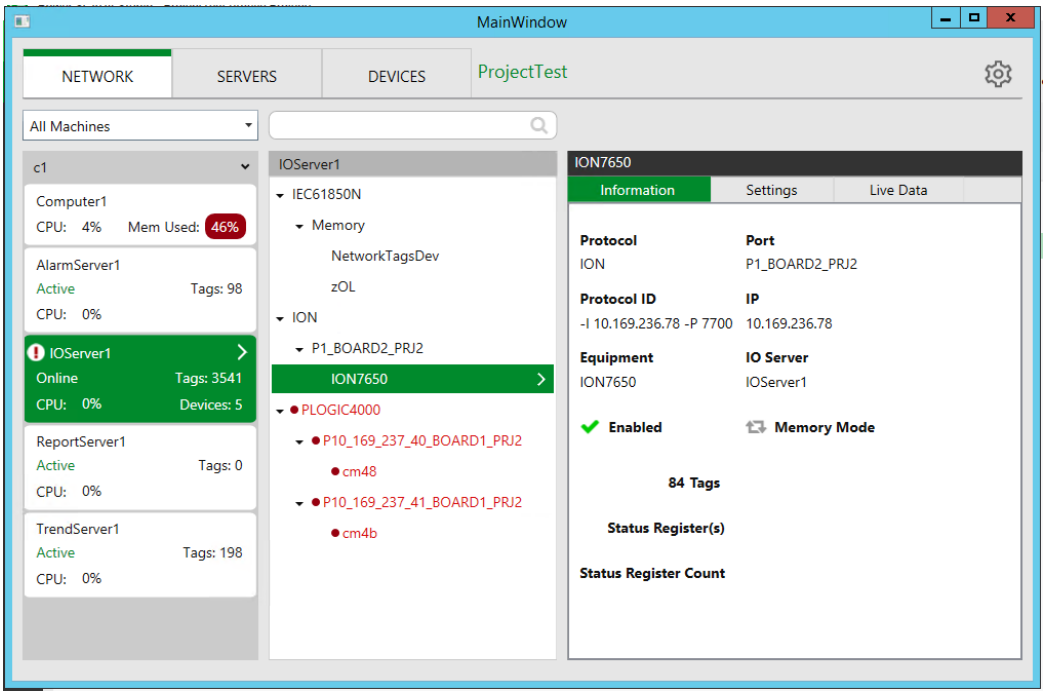
For more information, see "Offline and Online Mode" on page 503.

Diagnostics provides 3 views: Network, Servers, and Devices.

Network View

Groups a Power SCADA Operation system into machines, clusters, servers, protocols, ports, and devices:

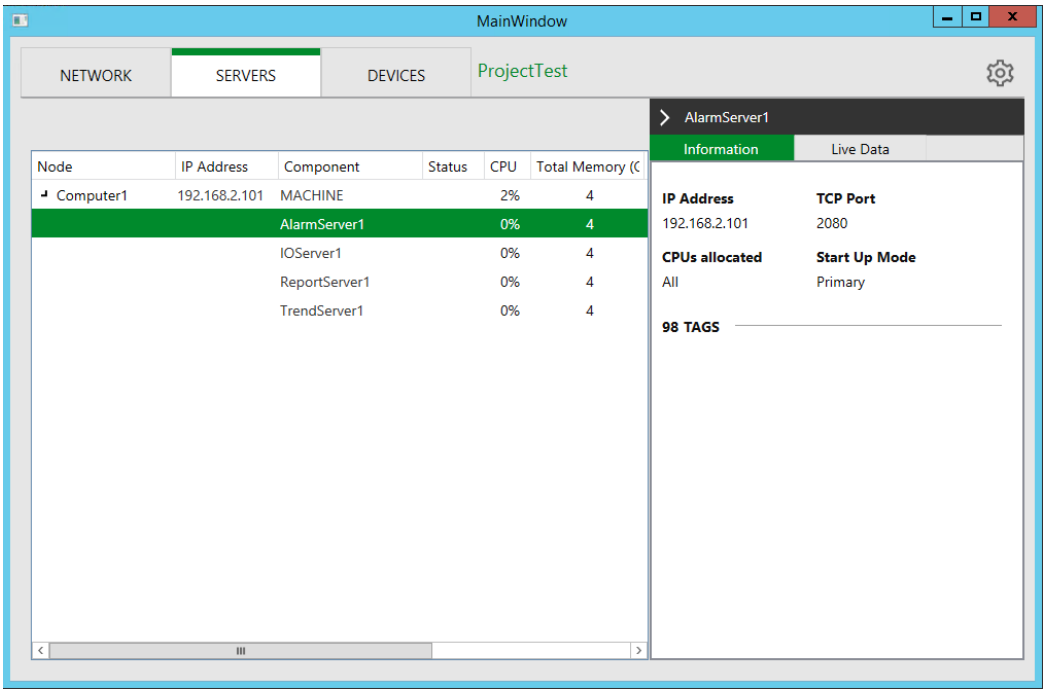
- Select a machine from the drop-down list to view the servers running on the selected machine. If you click on a server with associated devices, its devices display in the middle pane, grouped by protocol and port.
- Click any server, protocol, port, or device to view its **Information**, **Settings**, and **Live Data** ("Offline and Online Mode" on page 503 only) in the far-right pane.
- Click ⚙ to set the refresh rate.



Servers View

Groups servers by machine and individual servers:

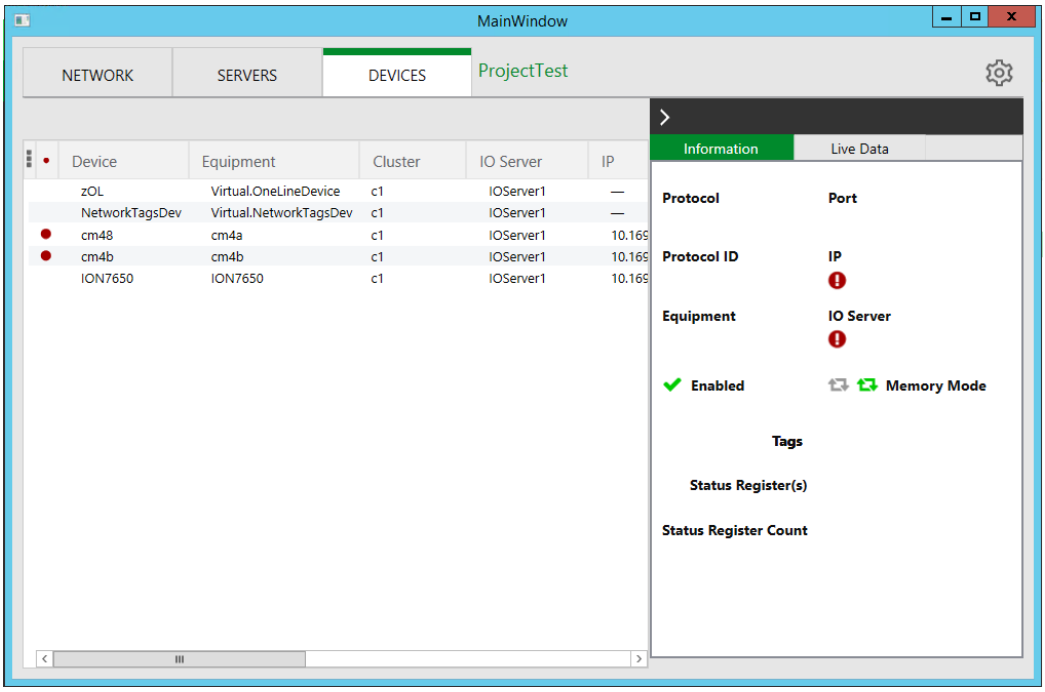
- Click the > to expand a machine and see the individual servers.
- Click any server to view its **Information** and **Live Data** ("Offline and Online Mode" on page 503 only) in the right pane.
- Click ‹ to open and › to close the right pane.
- Click ⚙ to set the refresh rate.



Devices View

Displays devices in a spreadsheet view to help you investigate and troubleshoot issues.

- Click to select or deselect the columns to be displayed.
- Click to sort by devices with known errors.
- Click a column heading to filter or sort.
- Drag column headings to reorder.
- Click any device to view its **Information** and **Live Data** ("Offline and Online Mode" on page 503 only) in the right pane.
- Click to open and to close the right pane.
- Click to set the refresh rate.



Diagnostics page

The Diagnostics page provides a quick view of the I/O device INI settings for all protocols, clusters, servers, ports, and devices. Use this information as the first step in troubleshooting device/communication issues in your system.

When you first click the Diagnostics tab, a short description and help link display on the right-hand side. The basic steps you follow are below. See the section after the steps for the logic behind how data displays.

1. To begin viewing data, click the Diagnostics link on the left. The I/O Device Settings link displays. Click that link.
2. Notice that the right-hand pane displays a link for the project name.
3. Click the Project Name drop-down box and choose the project for which you want to view data.
4. Power SCADA Operation loads the server information for this project. Note that you could have multiple servers: I/O, alarm, and trend
5. From the View pane, click a cluster and then a server.
6. (I/O servers only): From the Protocol column, choose a protocol and then the port and device.
7. View the data for that device:
   - The top row of the Settings are a "breadcrumb," showing the device information from cluster to device.
   - The second row, just above Effective Settings, displays details about the device, equipment name, number of tags, address, priority, memory mode and startup mode.

- The remaining sections display information only if there have been changes made to the default settings.

Data Selection Logic

Depending on the server type that you select, the Settings on the right display with different information.

All three server types display settings that include:

- Default Settings: the device default before any changes have been made; displays above the Effective Settings section.
- Effective Settings: the current settings, including any changes made, combining changes from the protocol, port, and device. In all cases, port changes will override protocol changes, and device changes will override port changes.
- Protocol Settings/Port Settings/Device Settings: If there are any overrides to the default settings, they display here.

I/O servers alone display a separate column: When you select an I/O server, the Protocol column displays beside it. You will select protocol, port and device.

Data is retrieved in this manner: protocol settings are retrieved from the Param.dbf file and then from the citect.ini file. These two lists of settings are merged. If there are duplicate settings, the citect.ini file changes take precedence. Finally, device settings are retrieved and merged. If there are duplicate settings, the device settings take precedence.

For example, for a given device:

The Protocol Setting for CacheRefreshTime = 2000.

The Port Setting = 1000.

There is no additional Device Setting to override the other settings.

The Effective Setting uses the CacheRefreshTime of 1000 from the Port Setting.

One-Line Errors and Warnings

Typical one-line errors are:

- CSV formatting errors
- Files required by the logic engine are locked or open in another process
- Non-existent tags are specified in CSV conditions
- Not running the Computer Setup Wizard for the runtime project

Communication Errors

When communication errors occur, the object that has lost communications gives an "unknown" status, which is graphically represented in the one-line animation.

Objects in the one-line should be defined to display the communication errors as a different color. The errors are calculated using the quality of a tag. If a tag or point becomes invalid, it is assumed that the communication is also offline. When this occurs, the graphical objects (buses, breaker, and sources) should change to the pre-set "unknown status" color; the array position 255 in the graphic.

Error Logging

The most common CSV file errors are logged to the Run project in a file named `AdvOneLineStatusLog.txt`. The file can contain several messages. The following table lists these errors and their descriptions:

Error Message	Description
Main Execution Loop Unexpected Failure	The main logic loop has thrown an exception that has not been handled by other error messages.
AdvOneLineDebugBus.Csv is locked	Another process or user has this required CSV file locked. Ensure that you do not have the file open.
Power SCADA Operation Running Project Path: "PATH" Does not Exist. Please Shutdown your Project and Try Running your Computer Setup Wizard	The Citect.ini "Run" parameter has an invalid project path that does not exist. Run the Computer Setup Wizard, and this path should be corrected.
Power SCADA Operation Running Project Path Not Specified. Please Shutdown your Project and Try Running your Computer Setup Wizard	This problem is almost exclusively caused by not running the Computer Setup Wizard.
PLSCADA is not in runtime	You must have your project running before you execute the AdvOneLine.exe file.
Failed to Establish Connection with CTAPI. PLSEngine.establishPLSConnection (FAILED CONNECTION)	This error message indicates the PLS API connection has unexpectedly been disconnected.
Required CSV file is locked	The CSV file specified (AdvOneLine.csv) is locked by another process or user. Ensure that you do not have the file open.
Invalid prefix located in CSVParser.FormatCSVData	The CSV parser has detected an invalid component prefix. This error message should not occur.
ERROR: Duplicate Component Name Encountered	Check the CSV file to ensure that you do not have two sources, meters, or breakers with the same component number.
ERROR: Invalid Node1 Number Encountered	In the Bus1 column, you have a node that is not a number between 1 and 1000.
ERROR: Invalid Node2 Number Encountered	In the Bus2 column, you have a node that is not a number between 1 and 1000.

Error Message	Description
ERROR: Node Not Specified	You have a component without a Bus1 and/or Bus2 specified.
ERROR: Invalid Condition String Encountered (MESSAGE)	You have a syntax error in your condition column. Read the message. It will give details about the syntax error, the line on which it occurred, and (if applicable) the character at which it occurred.
One or more of the tags specified in your CSV file do not exist in your Runtime SCADA project	Examine your CSV file. Either add the tags listed above the error message, or remove the tags from the CSV

By default, only exceptions are logged.

When alarms do not display correctly

Alarms may display incorrectly for a variety of reasons. The following table lists some common issues and resolutions:

Issue	Cause	Resolution
Alarm Log and Event Log do not display any data.	If there are two alarm servers, primary and redundant (standby), they may be synchronizing. This causes data to display slowly.	Data will display; but it could take several minutes.
Alarms display in Alarm Log, but not in Event Log or Banner	The missing alarm(s) were triggered while the runtime graphics page was not running.	These alarms will only display in the Alarm Log unless they are triggered again while the runtime graphics page is running. This will only affect alarms that were triggered before the runtime screen was running.

Issue	Cause	Resolution
PC-based and onboard alarms do not appear or disappear as expected.	This is due to the difference between way the two alarm types are handled: When an alarm is enabled, the system processes alarms for that tag. If the alarm is disabled, the system cannot process alarms for that tag. For the PC-Based alarm, the condition for this is, for example, IA > 80; if the tag value for IA is > 80, the appearance will show. The tag is constantly scanned, so the condition triggers the alarm once it is enabled. For the Onboard alarm, the condition for this is a digital tag, which is set by the driver when a new alarm record on the device is read. If the alarm was disabled, the driver cannot set the digital tag. When the alarm is enabled, nothing happens because the alarm was already "processed" by the driver and will never get reprocessed. Thus, there is no resolution.	
The number of alarms that display is fewer than the limit set by Alarm Summary length parameter.	This happens when the number of alarms exceeds 1000 and the system has multiple clusters.	Use one or more of these procedures: Set alarm filtering in the alarm viewer to reduce the number of alarms that can display. Only support a one-cluster system. If a multiple-cluster system is necessary, display a separate alarm page for each cluster.
Cannot filter on categories for alarms.	The new categories do not display in the list when you want to select them.	Use Custom Filter 8 instead. Currently, it is the only means available for adding custom filtering to alarms.
Page Down button causes an empty page to display.	The last alarm was on the previous page. When there are no more alarms, pressing Page Down displays a blank page.	Click Page Up to return to the previous page (and the last alarms for the system).

Frequently Asked Questions (FAQs)

The following items provide information about topics that generate frequent questions.

If I don't use PowerLogic drivers, how do I create device profiles?

Create a device type using a non-PowerLogic driver (like MODNET).

1. Using that device type, create a device profile.
2. You need to change the addressing of the new device type. Copy the addressing from a known device type, and then make the necessary changes for the new device type.

How should we manage categories and subcategories?

We recommend that each integration team decide in advance which categories and subcategories they will use. The I/O Device Manager requires the entire Profile name (which uses the category and subcategory as part of its name). Thus, you must be consistent in naming if the profiles are going to be shared and re-used.

1. Category should be used for a vendor.
2. Subcategory should be used to describe a type of device.
3. From the master computer that has the Profile Editor installed, create the categories and sub-categories that you plan to use.
4. Copy the DeviceTypeCategories.xml file (located in the OS-specific data directory: Data/Profile Editor/ Vx.x) to every computer being used to create profiles.

When should I create a device type rather than device profile?

Create a new device type, instead of a profile, when the addressing for a specific tag needs to change. For example:

The integration team can choose the Input to which they will wire circuit breaker status and position. In this case, the tags for circuit breaker status and position would have different addressing, based on how that particular circuit breaker is wired. We recommend a new device type in this case.

How do we synchronize a new PC with the master Profile Editor PC?

To synchronize a new machine with the latest device types and profiles from your master Profile Editor PC, you can:

- Use the Import feature to import tags, device types, and profiles from either an existing project or from SCL files. See "Import files into the Profile Editor" on page 234 for details.
- On the source PC: From the OS-specific Data/Profile Editor/ Vx.x directory, copy the entire OS-specific Data/Profile Editor/ Vx.x directory to the corresponding directory on the destination machine.

What do I do before I add or remove devices in the I/O Device Manager?

- Close all open DBF files.

  If you are removing a device:

- Click **pack database after removal** on the last page of the wizard.

  NOTE: Any changes that you made inside the Power SCADA Studio (such as setpoints or data type modifications) are lost when you delete the device from Power SCADA Operation.

What are the requirements for device names?

Device Name:

Keep Device name ≤ 16 characters. Use _ as a separator.

If you use a naming convention that incorporates location, you will be able to do filtering on alarm location.

For example, Site_Building_Panel_device would be named Sx_Bx_Px_Device. (Site1_Building1_Panel1_CM41 — S1_B1_P1_CM41).

The fewer levels you have, the more characters you can have in each level.

Device Comment:

Use this field as an alias for the device name.

This comment will be placed in the Equipment database, which is accessible from Cicode.

How do I troubleshoot device communications issues?

Power SCADA Operation drivers provide default communication settings that work with most devices. However, in cases when communication losses occur, use this checklist for finding the issues.

Initial checks, if the device is attached via a gateway:

☐ Ensure that all communication settings are correct on the gateway and device.

☐ Check the gateway timeout. A setting that is too low will cause many timeouts to occur. A setting that is too high will impact performance. We recommend a 3 second timeout, because most devices work well with this setting. Some devices may require a higher timeout (5 seconds).

In all communication setups (also see the driver help for parameters):

☐ Ensure that the Power SCADA Operation driver timeout is correct. We recommend that you set this to:

gateway timeout x number of clients + padding

Example: If the gateway timeout is 3 seconds and there are 3 clients, set the timeout in Power SCADA Operation to 10 seconds.

☐ Check the maximum block read size. Some devices do not handle large block reads well. When you lower the maximum block read size, the requests are smaller and faster. The downside is that more requests will be sent to the device, and tags will refresh more slowly.

☐ Check the device to see if there are registers that it cannot read. Some devices do not allow access to all registers.

Example: Data is in register 100-125 and 130-150. Power SCADA Operation will perform one read from 100-150. If 126-129 do not allow reading, this packet will return an exception. Use the appropriate logic code to mark these registers as invalid reads.

☐ If there are still timeout/no response issues, enable retries on exception. Some devices may not respond if they are performing other functions. In this case, a0x0A or 0x0B exception will be returned to Power SCADA Operation, which will cause a communication loss. Enabling the "retry on exception" will re-try the request.

How do I use Modbus communications methods?

Modbus TCP/IP via Gateway: Use this for any device that is not speaking TCP/IP natively. These devices connect through a gateway device such as an EGX or ECC.

Modbus TCPIP: Use this for any device that can speak TCP/IP natively. This includes CM4 or PM8 devices that have an ECC card installed.

How can I add more than one device at a time?

The I/O Device Manager requires that the profiles have already been exported from the Profile Editor to the project.

If the CSV file you use to add multiple devices attempts to add a device that is already present in the project, an error will be thrown.

In the event that an error is thrown (for invalid profiles, communication parameters, etc), the row containing the error will display in Excel. To prevent duplicate device entries from being attempted, you must remove any rows above the row indicated in the error message.

If you need to keep a record of the devices added to the system, then keep each of the spreadsheets that was used to install devices in a known location for that customer.

The Setup Sheet needs to be modified for each project. Specify the entire path for each file.

The Input Sheet requires the following:

The entire path name for each profile. The path name for a profile is based on the category and subcategory from the Profile editor.

Example: Schneider Electric.Monitoring Device.Branch Circuit Monitor Full

What are the naming conventions for servers and clusters?

There is no enforced naming convention for server and cluster names, other than the restriction that each server name and cluster name must be unique. Cluster names must be a maximum of 16 characters, contain no spaces, and cannot begin with a number.

Each team should come up with a naming convention for the servers and clusters. Consistent naming makes it easier to edit or create the automation spreadsheet used for device addition.

How and when do I create users for the Runtime environment?

New projects do not have any users created by default.

The default graphics objects (such as circuit breakers and alarm pages) are constructed using a pre-defined set of user privileges the security grid). During development, you must have users of various privilege levels for testing purposes. Create users for each of the various levels according to the security grid. To make the best use of these privileges, we recommend that you use this security grid when adding users as you create new projects.

See "Default User Access Settings (Privileges)" on page 357. For additional information, see **Using Security** in the `citectSCADA.chm` help file (...\Program Files (x86)\Schneider Electric\Power SCADA Operation\v9.0\bin\Help\Citect SCADA).

How do I manage projects in the Power SCADA Studio of Power SCADA Operation?

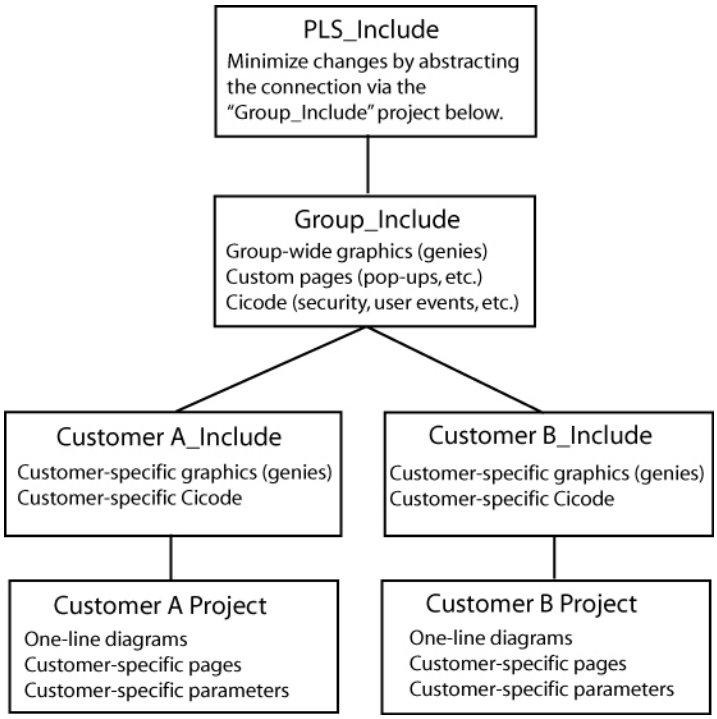
Although the Project Designer might want to organize each project in a particular way to suit customers' needs, the following is a recommended best practice:

- Keep original ‘Master’ copies of the PLS_Example and the PLS_Include projects for reference.
- The Services Group may develop a group-wide “include” project that will act as a conduit between the PLS_Include project and all customer projects (for example: "Group_Include"). This will make the upgrading of PLS_Include much easier, as it will be the only project that must be modified to be compatible with the new version in the group-wide include project.

  Any changes made to the PLS_Include project should be made at the Group_Include project level. This would involve removing portions of the code from the PLS_Include project, modifying the code and saving it in the Group_Include project. By removing (or commenting out) the original code and placing the new code in the Group_Include project, a layer of abstraction is preserved, further simplifying the upgrade process. In other words, the only changes to PLS_ Include should be code removal.
- When a new customer project is started, also create a customer-level “include” project.

  Always backup and restore the customer project and its associated include projects together.

  Always restore include projects before restoring the customer (or top-level) project.



- Upgrading PLS_Include:

  Document all changes to PLS_Include. This is absolutely necessary when upgrading to a new version of the PLS_Include project.

  Minimize changes to the PLS_include project.

Abstract as many changes to the PLS_Include project as possible. Use multiple include projects as shown in the diagram above.

New versions of PLS_Include will include a detailed description of each change, allowing you to merge the old and new versions.

New versions of PLS_Include will be backward compatibility, where possible.

On the Graphics page, what do I need to know about creating genies?

Creating a new genie

The easiest way to create a new genie is to use an existing genie from the library. This ensures that the new genie is compatible with the system, and that it preserves this feature:

A sizing guide (a dotted rectangle) is included; it displays during graphics edit mode. This guide ensures that new genies can be swapped with existing genies without the need to recreate portions of the drawing. Save the new genie in the appropriate project (do not overwrite the provided genies).

Save the new genie in the appropriate project (do not overwrite the provided genies).

Copying a genie to another project

Open the genie in the graphics editor, and do a <save as> into another project/library.

Genie Form Files

Any new genie (copied or created) will not have a FRM file entry associated with it. While the new genie is functional, it will show a cryptic unformatted properties box in the Graphics Editor. You can create your own FRM file with the needed entries by following the instructions available in the Power SCADA Studio Knowledge base.

If you want to use the FRM dialog box that belongs to the genie you copied, go to the PLS_Include library; locate the CTM and FTM files. Each library has its own CTM/FTM files that include the description for every genie in the library. (This is an ASCII text file that you can open in any text editor.) Find the genie that you copied (or on which you're basing the new form). Copy the portion that matches the copied genie, and create a FRM file that has the desired library name on it. Copy in the text from the FRM file. Restart Power SCADA Studio, or it will not detect the new FRM.

Genie Sizing

The provided genies come in two sizes: size 1 and size 2. When making a new genie for reuse among multiple projects, it will be beneficial to create a genie for both sizes. Follow the same steps for both sizes (sizing guides are provided for both sizes).

How do we customize existing templates?

Template Editing

All objects on the page contain one or more Animation Numbers (ANs). Symbols take one AN while genies may take tens to hundreds of ANs. Placeholder ANs allow you to add objects to a template that is used on existing pages.

Some default templates contain ANs that have associated Cicode functions that rely on the animation number to remain a fixed number. For this reason, we have pre-allocated a set of ANs for the default templates. The base normal template uses ANs 1–263, and it has placeholder ANs from 264–500. When customizing this template, you should use the placeholder ANs as required.

You can place an AN (or a placeholder AN) on the page by using the “System 3.x/4.x tools available in the Graphics Builder under Tools< Options.

The default template uses ANs 1–263 and it has placeholder ANs from 264–500.

New objects added to a page or template will take the next available ANs. Any previously used (and now abandoned) ANs will be reused.

To add an object on the template:

1. Open the template.
2. View the page properties and change the page width to 2000. This will reveal the hidden placeholder ANs on the page. You may have to change the width to a wider dimension for widescreen templates.
3. Determine how many ANs the new object requires. (You can place the new object on a blank page and then view the object in the object browser.)
4. Remove exactly the amount of ANs to allow the new object to be placed on the template. Remove ANs beginning with the lowest available placeholder AN (in the default template, this would be 264).
5. Place the object on the template.
6. Save the template.
7. Create a new page based on this template.
8. Drop a numeric object on the page.
   - This object’s AN should be 502 (501 is reserved for placing the template on the page).
   - If the object has an AN less than 502 then you have unused AN(s) on the template. This must be resolved. (Place additional ANs on the template to rectify this situation.)
   - If the object has an AN greater than 502 then you have too many ANs on the template (a AN on the template is going beyond the 500 limit). You must find the culprit (via the object browser) and rectify the situation using the steps above.

How do I change the default pickup/dropout text for alarms?

The default ‘pickup/dropout’ text is shown as Appearance/Disappearance.

To change globally:

This text can be changed by configuring INI parameters in the citect.ini file. For more information, see the Power SCADA Operation 9.0 Help Manual(Graphics Library Parameters).

This is the *global fallback text* that will be used if pickup/dropout text is not specific on a per-alarm basis in the Device Profile. You can specify the per-alarm pickup/drop-out text on the profile tab in the Profile Editor.

To change on an individual basis:

See the Power SCADA Operation 9.0 Help Manual (Viewing Device Profiles: "Alarm On Text" and "Alarm Off Text").

What can I modify during runtime?

See the Power SCADA Operation 9.0 Design Guide, "Updates to the System While Online," for a list of items you can modify during runtime.

Why do the browser navigation buttons not work?

If the browser navigation buttons do not work when you are viewing the runtime window, you have probably added a new page, but have not done the following:

Added the startup page to the Page parameter. See "Set a new page as the project startup page" on page 268 for help.

Left the INI settings at <default>. In the Computer Setup Wizard, General Options Setup screen, do not change the StartupPage field; leave it as <default>.

What can I set up in logging and archiving?

Event Logging and Archiving:

Event fields that are logged to disk may be configured by adjusting the AlarmFormat parameter.

There is no automatic maintenance performed on the log files. It is important that the log/waveform data be cleared out periodically (to prevent the hard drive from filling up; this does not affect performance).

How do I create and configure busbars?

When drawing one-line diagrams:

Analyze the drawings at a customer site.

Number the busbars consistently on the one line diagram(s). If busbar 14 spans across multiple pages, it should be numbered busbar 14 on all pages. Label the voltage level (0–3) on each busbar.

Uses for Line Active:

Page Connections: Many one-line diagrams will span multiple pages. To connect these pages together, you must use the line active field of the ‘incomers’ of the second and subsequent pages. Set the line active field of the incoming busbars on these pages to an expression that references the nearest device on the same busbar of the previous page.

Metered Busbar: Many busbars are metered. It is more accurate to allow these metering devices to dictate state than to rely solely on the simulation (see Expressions below).

Configuration of Line Active:

Simulation: If the Line-Active field is left blank, the busbar state will be determined by surrounding devices.

Expressions:

A Cicode expression in the form of Device\Tag > Nominal Voltage (I.E., S1_B1_P1_CM41\MMXU1\PhV\zavg > 120).

If the expression is TRUE, the ACTIVE color will be shown. The active color is determined by the voltage level assigned.

If the expression is FALSE, the DE-ENERGIZED color will be shown.

Hard-Coded

If no upstream devices are available (in the event of an incomer, for example), you may have no other choice than to 'hard code' this field to a '1'. This forces the busbar to always be ACTIVE.

What INI parameters should I use for debugging?

We recommend that you contact Technical Support before performing any debugging.

Parameter: [PowerLogicCore]
DebugCategory = All
DebugLevel = All (or Error)
LogFileArchive = Deprecated; no longer used. Use *[Debug]SysLogArchive* instead.
LogFileSize = Deprecated; no longer used. Use *[Debug]SysLogSize* instead.

Parameter Details:

DebugCategory defines which message categories to log. (See table below).

DebugLevel defines debug levels of messages to be logged. (See table below).

Debug Levels

The following debug levels are accepted by PowerLogic driver core library:

- WARN: log all warning level messages
- ERROR: log all error messages
- TRACE: log all trace messages
- DEBUG: log all debug messages
- ALL: include all level messages

Debug Categories

PowerLogic core library and driver messages are grouped in categories. Each of these categories can be enabled independently from others in any combination.

- ALL: enables all categories
- ALARM: messages related to alarms, regarding collection and detection
- COMMAND: messages related to commands
- CORE: core events that do not fall into driver-specific logic
- DATAPOINT: debug messages related to data points
- ENTRY: trace messages produced when driver API entry points are called
- MISC: miscellaneous messages that do not all into any other category
- MODBUS: TCP/MODBUS messages
- PORT: traces related to the port events
- REAL: messages related to real-time data collection
- REPLICATION: messages produced by replication subsystem
- STATE: messages related to internal object-state changes

- STATISTICS: enables driver statistics data output
- UNIT: traces related to specific unit events
- WAVE: messages related to waveforms -- waveforms download, processing
- WAVETOALARM: not used

Parameter: [Debug]
Menu = 1

Parameter Details:

The Menu parameter determines whether the Kernel option is displayed on the control menu of the runtime menu. This can also be enabled using the Computer Setup Editor.

How do I tune my system for best performance?

There are several parameters you can use to enhance your system's performance:

Driver-tuning parameters:

Parameter (Back Polling Rate): [SEPAM40]
CacheRefreshTime = 1000
InitUniCheckTime = 120
Retry = 3
Timeout = 1000

Parameter Details:

The CacheRefreshTime parameter controls the maximum rate at which the driver will attempt to repopulate its cache. If the driver cannot refresh its cache within the time specified, it will collect data as fast as the network allows.

This back polling rate can be global to all devices or tuned up to a specific I/O device.

The InitUniCheckTime parameter controls how long the driver will wait before attempting to bring a device online after it has gone offline. This value can be decreased to bring offline devices back into service in a shorter period of time. In a multi-drop scenario, this time should be relatively long, to prevent init unit requests from stalling communications to the rest of the devices on that port.

The Retry parameter defines the number of retry attempts for specific MODBUS requests. Retries will only occur in response to the MODBUS errors which are defined below.

The Timeout parameter controls how long the driver will wait for a response from a device before setting that device as offline. This value should be greater than the device/gateway timeout period.

Parameter: [Device]
WatchTime = 5000

Parameter Details:

Device WatchTime is the frequency that Power SCADA Operation checks devices for history files and flushes logging data to disk.

Default: 5000
Range: 1000–3600000 milliseconds.

Miscellaneous Parameters

Parameter: [Kernel]
Task = 20000

Parameter Details:

Kernel Task is the number of tasks. Increasing the number of kernel tasks is used when “Out of Kernel Task” message is received. The change will be likely for large systems.

Default Value: 256
Range: 50–32767

Parameter: [Page]
ScanTime = 250

Parameter Details:

Page ScanTime determines how often the Animator refreshes a graphics page at runtime.

Default: 250
Range: 1–60000 milliseconds

Parameter: [ALARM]
ScanTime = 500

Parameter Details:

Alarm ScanTime determines the rate at which alarms are scanned and processed.

Default: 500
Range: 0–60000 milliseconds

If a tag is configured, how is it polled in the device?

In other words, is a tag only polled on demand when it is requested by a client; for example, when the operator displays a page with the tag on it? Or are all configured tags polled all the time, with the relative polling rates/communications bandwidth carefully managed?

The ModNet driver polls real-time tags on a user demand basis (when a user opens a page with the tags on it). Therefore, the time to retrieve data will vary, depending not only on the communications bandwidth, but on the amount of data being requested. This can vary significantly, depending on which pages are displayed by the operators at any particular time.

The PWRMODBUS driver polls all configured tags; however, different types of tags can be polled at different relative rates, and the available communications bandwidth is carefully managed. This approach means that tag update rates are not subject to the scalability issues associated with operator actions (as is the case for the ModNet driver). It is also advantageous in that performance issues associated with communications bandwidth or I/O device response times can be determined at SAT/time of implementation and are not subject to significant change during operation.

The different tag types can be allocated relative importance in data requests, expressed as a percentage. (See Bandwidth Allocation Parameters in Performance Tuning Parameters, in the Power SCADA Operation 9.0 Help Manual. Keep in mind that any unused bandwidth allocation (from, for example, events retrieval) is made available for other data types to use. If the event does not need the default 25% allocation, it will be made available to the other parameters (real-time tag retrieval, etc). This potentially increases the update rate of real-time tags.

Additionally, the real-time tag relative scan rate based on priority can be set to three different levels. (See "Tag Scan Rate Parameters" in Performance Tuning Parameters, in the Power SCADA Operation 9.0 Help Manual.) This means that, if some real-time tags are more important that others, you can set their relative priorities. For example, configuration tags vs. important real-time tags vs. normal real-time tags.

Device popup from a one-line: Why do the fields overlap?

This is controlled by a parameter entry:

Section: Page

Name: EquipDetailDescLength (the total number of characters in a single row of this popup)

Default = 48. The problem will occur with a larger font or if the window is resized. The default value of 48 can be changed or the window and associated genies can be resized.

Can I change the %CLUSTER% name in the I/O Device Manager?

No. If you change the placeholder %CLUSTER% to any other name in the I/O Device Manager, the system will be unable to find the actual cluster to which it refers.

A device can prevent writes to its registers: how do I ensure that writes are successful?

Power SCADA Operation cannot provide feedback about whether a write to a device register is successful. If a device is capable is preventing (blocking) writes to its registers (for example, Sepam), you need to verify that its “block” feature is not enabled. Do this at the device.

In Cicode, you can also use the tagwrite function in blocking mode, i.e., bSync parameter = true; Check the return code: 0 = success, anything else = error. For more information, see the Cicode Programming Reference help file.

How do I prevent Power SCADA Operation from accidentally making invalid areas in memory available to reads and writes?

Power SCADA Operation normally optimizes its packets for greatest performance. This optimization can sometimes include invalid areas of memory in devices. These invalid areas can be specifically defined and excluded from optimization packets created by Power SCADA Operation. For more information, see "Advanced Tag Blocking Capabilities" in Performance Tuning Parameters, in the Power SCADA Operation 9.0 Help Manual.

How do I create an audit in the Event Log for user logins and logouts?

```
//LOGOUT

FUNCTION
PLSLoginUser()

//INT iPage = PageInfo(1);
INT iPage = WinNumber();
IF     mbLoginFormShown[iPage] = TRUE THEN
```

```
RETURN;                //form already shown
END
//prevent multiple forms
mbLoginFormShown[iPage] = TRUE;
IF (UserInfo(0) <> "0") THEN
// Confirm User Action
IF (0 = Message(StrToLocalText("@(Confirm)"), StrToLocalText("@
(Logout)"), 1+32)) THEN
PLSAlmDspEventAdd(0, 0, 1, "User Logout", Name(), "Logout", "");
Logout();
END
mbLoginFormShown[iPage] = FALSE;
RETURN;
END
IF (0 = LoginForm())
PLSAlmDspEventAdd(0, 0, 1, "User Login", Name(), "Login", "");
END
mbLoginFormShown[iPage] = FALSE;
END
```

Why am I seeing #COM for circuit breaker status in the genie status page?

If this is a Micrologic P device, and it does not have a CCM, you will not be able to view data referring to circuit breaker status, e.g. racked in/racked out. When there is no CCM, the device profile should not have tags that refer to the CCM.

Why can't I acquire waveforms in the waveform viewer?

The "acquire" feature (the "A" button on the waveform viewer) does not work in Power SCADA Operation. You can, however, view waveforms from device onboard waveform files. To do this:

At the device or in the meter configuration software, add the appropriate alarm, and enable automatic capture of the waveform when the alarm occurs.

In the Profile Editor (Create Device Profiles tab), check the Waveform box for the alarm you added.

When the alarm occurs, the waveform is captured. You can view the waveform in the Alarm Log. You can also view alarms/waveforms from a drawing in the runtime environment. Click the genie for the device; right-click the alarm to view the waveform.

Note that, in very large systems, it could take as much as an hour for the waveform to appear.

Why won't the Excel DBF Add-In toolbar install?

When you are installing the Excel DBF Add-In toolbar, you may see this error: "Error 1308. Source file not found....."

You can click "ignore" at this error, and the install will finish. The next time you open Excel, the DBF toolbar will display.

What causes the "First dbf record" error message? How do I keep it from happening?

The error message "First dbf record" tells you that a project is not found. This happens when you add a project, and then rename it or delete it. Then, when you try to create a new project, you see this error message.

To resolve this issue, simply shut down and then restart the Power SCADA Studio.

Why is my device in comms loss?

When you bring your system on line, and you find that Power SCADA Operation has lost communications with a device, check the following:

Verify that the physical correction is correct and secure.

Verify the IP address.

Verify the Modbus address.

Check the statusRegister, statusRegistersCount, and statusRegisterType (see for details)

How do I set up select before operate?

For systems in which you can determine that a single user is selecting a device prior to sending an open/close command, you can add a "select before operate" button.

To do this:

1. Locate the Select Before Operate tag in the variable tags.
2. Append `\str` to the end of the tag name.
3. Change the data type to STRING.
4. Click **Add**.

This creates the SBOw tag for the IEC 61850 advanced control screen. For more information about advanced control, see:

- "Set up IEC 61850 advanced control" on page 313
- "Perform IEC 61850 advanced control" on page 485

Why am I getting 'Out of licenses' notifications in the FlexNet Publisher?

These notifications simply warn that all the available licenses hosted in the Floating License Manager are currently in use and there are no spare licenses left.

So, in a normal working condition where Power SCADA Operation uses all the available licenses (Server, Clients, View-Only Clients, etc..), it is normal to have these alerts.

It is possible to disable these notifications. Log in to the Administration page of the FlexNet License Administrator portal, click Alert Configuration, and then un-check the following options:

- Out of activatable licenses
- Out of concurrent licenses

- Activatable threshold exceeded
- Concurrent threshold exceeded

Reference

The Reference chapter contains detailed reference information related to planning, installing and upgrading, configuring, administering, or using Power SCADA Operation. This information is referenced in the other chaptersof this guide.

Use the links below to find the content you are looking for:

Section	Description
"Upgrading Reference" on page 525	Detailed information on the Cicode functions, Citect INI settings, and upgrade information specific to previous versions of Power SCADA Operation.
"Configuring Reference" on page 577	Detailed information on the Citect INI parameters, logic code definitions, the default genie library, deadbands and ignored devices, Power SCADA Operation configuration tools, engineering units, LiveView tables, and notifications.
"Glossary" on page 656	A list of commonly used terms and acronyms, and their definitions.

Upgrading Reference

The topics in this section contain detailed reference information that pertains to upgrading to Power SCADA Operation 9.0

Upgrade Information

Refer to the upgrade information on the steps you may need to perform before and after the upgrade process.

- "General Upgrade Information" on page 526
- "Upgrade Information for versions 8.1 and 8.0 SR1" on page 527
- "Upgrade Information for versions 7.40 and 8.0" on page 531
- "Upgrade Information for Version 7.30" on page 532
- "Upgrade Information for Version 7.20" on page 535

NOTE: Review the information up to and including the version to which you are upgrading.

Cicode Functions

Refer to the following topics for detailed information on the Cicode functions that were added for each release:

Citect.ini Parameters

Refer to the following topics for detailed information on the Citect.ini parameters that were added for each release:

General Upgrade Information

Refer to the information below on the steps you may need to perform before and after the upgrade process.

NOTE: Also review the information up to and including the version to which you are upgrading.

The information below should be reviewed and is not version specific.

Command Execution and User Management Security

A new field "Allow Exec" has been added to the Role form which determines if a user role can invoke the "Exec" Cicode function.

A new field "Manage Users" has been added to the Role form which determines if a user role is authorized to manage user accounts.

RPC server-side security

A new field "Allow RPC" has been added to each of the server forms which determines if a server can accept remote MsgRPC and ServerRPC calls.

Earliest Legacy Version

If you are performing an online upgrade, use the Citect.ini parameter [LAN]EarliestLegacyVersion to specify the minimum legacy version from which the new version will accept connections.

NOTE: You should reset this parameter to its normal setting when an upgrade is complete.

Setup the Development Environment

The existing version may have project configuration [INI] file settings that are related to the configuration and may be required to compile the project. The specific project configuration settings can be added to the new version using the Computer Setup Editor.

The previous settings can be migrated to the new version by replacing the new configuration [INI] file with the previous version of the INI file. When doing this, the following parameters would then need to be updated / added to reflect the new installation settings in the previous configuration file. Any old settings that are no longer used are removed from the file when the Computer Setup Editor is used to save the file the first time.

. Path: [CtEdit] Bin, [CtEdit] User, [CtEdit] Data, [CtEdit] Log, [CtEdit] Config

These settings should be copied from the new version configuration file (INI) to the current / previous file. The paths need to be set to the product application directories for the new version installation.

MsgRPC and ServerRPC server-side security

A new field "Allow RPC" has been added to the Role form which determines if a user or group of users can perform remote MsgRPC or ServerRPC calls. On upgrading projects this field will be left blank which will raise the following compiler warning message.

'Allow RPC' permission is not defined (defaulting to FALSE)

For existing users can continue to use MsgRPC and ServerRPC, you need to manually change the value of “Allow RPC” to TRUE.

If these functions are used in your project, the roles that execute the functions will also need to have the permissions enabled.

Upgrade Information for versions 8.1 and 8.0 SR1

Password security

If you are performing an online upgrade, wait until all PowerSCADA Expert nodes have been upgraded to 8.1 or 8.0 SR1 before changing your user passwords.

You will also need to update the CTAPI DLLs on any CTAPI clients before you change any user passwords, otherwise any legacy CTAPI clients will not be able to connect to the system.

For increased security, it is recommended users change their password on a regular basis.

Changes to alarm limits are not retained when upgrading if [Alarm]UseConfigLimits is set to 0

In PowerSCADA Expert 7.20, if you have modified an alarm limit via Cicode, on upgrading the alarm to version 8.1 or 8.0 SR1 the limit will need to be manually changed to the new value. For example, if you changed HIGHHIGH from 95 to 90 (in version 7.20) and you upgrade this alarm to version 8.1 or 8.0 SR1, its HIGHHIGH limit in 8.1 or 8.0 SR1 would be reverted to 95 (the original value from

7.20). You will need to manually change the value back to 90. However, if in version 7.20 [Alarm]UserConfigLimits=1 at the time of the limit change, the alarm will be migrated to version 8.1 or 8.0 SR1 with the HIGHHIGH limit set to 90 (that is, the newest value).

[Alarm]StartTimeout parameter used when upgrading from Version 7.20 to 8.1 or 8.0 SR1

The [Alarm]StartTimeout parameter has been reinstated only for those performing an online upgrade from Version 7.20 to PowerSCADA Expert 8.1 or 8.0 SR1. This parameter sets the timeout period (default 120 seconds) for loading each packet from the version 7.20 alarms server. When a version 8.1 or 8.0 SR1 alarm server starts and is connected to a version 7.20 server, it tries to retrieve the current alarm states and the historical alarm data. This parameter determines how long to wait for a reply from the 7.20 server. If the data has not been fully retrieved from the 7.20 alarms server by the end of the timeout period, the 8.1 or 8.0 SR1 alarms server either loads the saved data or reads the alarm data (from the I/O devices).

If the alarms server is timing out, you will see the message "Timeout from RndAlarm Server" in the PowerSCADA Expert Kernel window and the alarm server syslog file. This timeout should only occur if you have a large number of alarms, typically greater than 10,000. If you see this message, increase this parameter until the message no longer displays at startup.

Upgrade from version 7.20 to 8.1 or 8.0 SR1 requires a clean alarms database

Before you upgrade a version 7.20 alarms database to PowerSCADA Expert 8.1 or 8.0 SR1, please ensure that the 8.1 or 8.0 SR1 database is clean and has no existing records. If this is not the case, the alarm server will not be able to synchronize with the version 7.20 server.

Extend ArchiveAfter parameter before upgrading

Before upgrading to PowerSCADA Expert 8.1 or 8.0 SR1 from version 7.20, 7.30, 7.40 or 8.0, you need to extend the setting of the INI parameter [Alarm.<ClusterName>.<ServerName>]ArchiveAfter so that it will capture all the data you would like to migrate.

When an upgrade to 8.1 or 8.0 SR1 occurs, any data that is older than the time range specified in the ArchiveAfter parameter may be lost when migration occurs. For example, if you have set the ArchiveAfter parameter to 8 (weeks), then any non-active data that is older than eight weeks may be lost and not available for archiving.

A number of checks have been implemented to help avoid this situation. If the ArchiveAfter parameter is not set, you can expect the following behavior:

- If you are performing an online upgrade and [LAN]EarliestLegacyVersion has been set to less than 7500, you will be prompted to set the ArchiveAfter parameter. The alarm server will not start until a setting is detected.
- If you are attempting to migrate alarm data from a legacy .dat file (used in version 7.20), the file will be checked for any data that could potentially be lost. If any is detected, the alarm server will not start.

- If you are trying to migrate alarm summary data from a version 7.40 or 8.0 database, the data will be checked for any data that could potentially be lost. If any is detected, the alarm server will not start. (Also see Migrating alarm summary data from Version 7.40.)

If the INI setting is removed before upgrading, and there is data detected beyond the ArchiveAfter period, you will receive the following error in the Runtime Manager:

"Earliest alarm event date is [day month timestamp] please adjust [Alarm.<ClusterName>.<ServerName>]ArchiveAfter parameter."

Adjust the ArchiveAfter setting to cover all the data (use the earliest date from the error message). After the migration is complete, you can then archive your data and return this parameter to its normal setting.

Removal of alarm save files

When upgrading from version 7.30 or 7.40 or 8.0, please ensure that the alarm save files (named "<project_cluster>_ALMSAVE.DAT" and "<project_cluster>_ALMINDEXSAVE.DAT") are removed from the 8.1 or 8.0 SR1 project folders.

Change in behavior for AlmSummaryDelete

Browse cursor automatically moves to the next summary record on AlmSummaryDelete(). Previously in 7.20, AlmSummaryNext() needed to be called to move to the next summary record.

Change in behavior for Alarm Summary time fields

In PowerSCADA Expert 8.1 or 8.0 SR1, if any of the following Alarm Summary fields have not been set, the Cicode functions AlarmGetDSP, AlmSummaryGetFields, and AlarmSumGet will return "".

- OffDate
- OffTime
- OffMilli
- AckDate
- AckTime
- DeltaTime

In version 7.20, '0' would have been returned.

Change in behavior for reinstated AlarmSum* Cicode functions

1. The index passed to AlarmSum* functions needs to be current. That is, either:
   - the index returned by latest call to AlarmSumFirst/Last/Find/Next/Prev

     or
   - the index returned by latest call to AlarmSumAppend/Split.

   Otherwise, error 561 is raised (AlarmSum index not current).
2. The alarm sum session needs to be initialized by calling AlarmSumFirst/Last/Find before AlarmSumNext/Prev can be called.

Otherwise, error 562 is raised (AlarmSum not initialized).

3. AlarmSum* functions should not be called from multiple concurrent Cicode tasks. If the AlarmSum session is busy in another task, error 563 (AlarmSum busy) is raised.

Fields no longer supported on Sequence Of Events page

The following fields are no longer supported on the Sequence of Events page:

- AckTime
- OffTime
- OnTime
- DeltaTime

These fields are only available on the Alarm Summary page.

Alarm Summary can be archived

The Alarm Summary can now be archived. Use the existing functions SOEArchive, SOEMount, and SOEDismount.

AlmSummaryOpen Query Timeout

If your system generates a lot of alarm summary records (aproximately 100k records or more within an hour), AlmSummaryOpen() will return -1 after a lengthy timeout of 90 seconds or more.

Use multiple browse sessions filtered by time range of small intervals. The size of interval should be smaller than an hour, and the exact size should depend on the density of alarm summary records in the history.

```
   FUNCTION OpenAlarmSummaryTimeRange(TIMESTAMP tEndTime, INT
iDurationSec, INT iInterval)
       INT session;
       TIMESTAMP tQueryStartTime;
       TIMESTAMP tQueryEndTime;
       STRING t0 ;
       STRING t1;
       INT iRemaining = iDurationSec;
       //
       tQueryStartTime= TimestampSub(tEndTime,iDurationSec,5);
       t0 = IntToStr(TimestampToTimeInt(tStartTime));
       WHILE iRemaining > 0 DO
               IF iRemaining > iInterval THEN
                  tQueryEndTime = TimestampAdd(tQueryStartTime,iInterval,5)
                  iRemaining = iRemaining - iInterval;
                  t1 = IntToStr(TimestampToTimeInt(tQueryEndTime));
               ELSE
                  tQueryEndTime = TimestampAdd(tQueryStartTime,iRemaining ,5)
                  t1 = IntToStr(TimestampToTimeInt(tQueryEndTime));
```

```
                    iRemaining = 0
                END

                session = AlmSummaryOpen("OnTime >= " + t0 + " AND OnTime < "
t1, "");
                IF (session >= 0) THEN
                AlmSummaryFirst(session);

                // Do something with the browse session
                // ...

                AlmSummaryClose(session);
                tQueryStartTime = tQueryEndTime;
                END
        END

        FUNCTION QuerySummaryOneHour()

        //Query the summary from the current time back one hour in 20 minute
intervals
        OpenAlarmSummaryTimeRange(TimestampCurrent(),3600,1200);
        END
```

Sorting on the Alarm Summary and SOE Pages

In PowerSCADA Expert 8.1 or 8.0 SR1, performance of the SOE and summary pages has been improved. When sorting either the Alarm Summary or SOE pages by any field other than 'TIME' or 'ONTIME', or when applying a heavy filter to either of these pages, it is recommended that you apply a time based filter.

A new mode has been added to AlarmGetInfo() to detect if a timeout occurred and as a result no records were returned. You may then add an animation to the Alarm Summary / SOE page to notify users that a timeout has occurred.

Functional limitations with alarm data during online upgrade

If you are performing an online upgrade, you may notice some functional limitations while your alarms servers and clients are running different versions. For example, the Alarm Summary page may appear blank if the server is running version 8.1 or 8.0 SR1 and the client is still on version 7.40/8.0. This situation is temporary, and all data will be restored when the upgrade is complete.

Upgrade Information for versions 7.40 and 8.0

Running a Mixed Version System

Running a system with mixed version servers is only recommended during the upgrade procedure. It is not advisable to run in a mixed version server environment for any longer than necessary.

Equipment.Item

In v7.40/8.0 you can reference a variable tag using associated equipment name and item name (equipment.item syntax). In this release, referencing trend tags and alarm tags using this syntax is not supported. After upgrading some existing equipment / item names may no longer be accepted due to new compiler rules, for example, root equipment names cannot be a reserved word and item names cannot be tag extension keywords.

You can also insert Equipment.item references into expression fields using the **insert tag** option available when configuring objects in the Graphics Builder; however if no equipment has been configured in your system the list will be empty by default. You will need to configure equipment or deselect the option '**Display equipment items when populating tag list**' in the Project Editor Options dialog to populate the list with available tags.

EcoStruxure Web Services Server

To invoke EWS Service Methods from EWS client requires certificate and user credentials authentication.

The user of EWS Service should be a valid Citect user that is defined in **System**->**Users form**.

The EWS Service uses ctapi call to access variable tags, as such , INI parameter [CtAPI]Remote should be set to 1 if PowerSCADA Expert is running in single process or multi-process mode.

Upgrade Information for Version 7.30

If upgrading to a more recent version, all upgrade procedures starting from the following procedures for v7.20 to the desired upgraded version should be reviewed.

ADO Support

The SQL engine for database query was updated in v7.30, as a result the Cicode function SQLNoFields was removed.

Alarm Enhancements

- [Alarm]SummaryLength
  The maximum value of the [Alarm]SummaryLength parameter has been changed from 4096000 to 100000.
- Migrating alarm event history

  Version 7.30 introduced a new historical alarm storage repository. The existing historical data is automatically migrated to the new repository, once, on first start of your alarm server.

  INI parameter [Alarm]SummaryTimeout should be set to -1 if the existing historical data remains in the alarm summary queue.
- Alarm Server Upgrade

  There are a number of changes to the way alarm servers are configured (including ports, paths and redundancy architecture). Alarm Server and legacy alarm client interoperability options have changed. When doing a live migration, older alarm clients will not connect to a new alarm server process.

If you are running multiple Alarm Servers on the machine, the unique Database Port number should be configured (Extended forms fields in the Alarm Servers dialog window). The default TCP/IP Port for the Alarm Server Database Port is 5482.

These port numbers cannot conflict with any other TCP ports on the same PC.

If two alarm servers are configured on the same machine with both database ports left as empty or configured with the same port (i.e. default to 5482) project compilation would not be successful.

If the configured database port is used by another external application or is blocked by firewall on the same PC, alarm server will not be functional at runtime.

- The Computer Setup Wizard Alarm Server Properties Setup page has been removed.
- [Alarm]UseVisibleTimeAsAlarmActiveTime

  Enables / disables the update of timestamps on multi-digital alarms when being unsuppressed.
- AlarmSetQuery

  Users using custom Cicode for filtering (implemented with AlarmSetQuery) will need to re-engineer their code to use the new filter functions.
- AlarmRec Functions

  Version 7.30 required that the cluster be explicitly specified in multi-cluster systems when using these functions. Multi-cluster systems need to re-engineer Cicode using these functions. The compiler is not able to identify that change to code is required.
- Summary Page Behavior Change

  The new SOE view of historical alarm records is designed to replace the alarm summary view. The alarm summary page will no longer dynamically update whilst displayed. Existing alarm summary pages will need to be redisplayed in order to retrieve the latest data. Comments can no longer be added or deleted directly from the summary page. Comments can only be added from the new SOE page. Some Alarm Summary Cicode functions have also been removed.

Batch Icons Removed

The Batch toolbar icons have been removed from the Project Editor: Batch Build, Batch Simulate, and Batch Execute.

Cluster Replication

Version 7.30 has cluster replication off by default. If this was used previously, compiler errors may occur. To enable the system to replicate clusters (similar to version 7.20), the cluster replication parameter has been added: [General]ClusterReplication.

Computer Setup Wizard

Using the CPU Configuration page you can now assign a CPU to a component.

Default Trend Storage

- Storage Method

  Trend records now have to explicitly define their trend storage method on the trend tag configuration form. The compiler will raise an error if not defined. In previous versions, the default when not defined was 8 byte. When upgrading, customers need to set the trend storage method for blank entries to match the default from the previous version.

Graphics

- Disable style behavior correction

  Disable style behavior has been corrected in v7.30. It is recommended when using a style other than "embossed" to check the disable style of all groups, genies, and symbols sets at runtime.

Introduction of Equipment

- Equipment field has been added to tags, alarms and trends as a new feature.

Internet Display Client

- IDC

  Support for the Internet Display Client (IDC) has been removed from this release. It is recommended you consider the use of the Web Client or the Single File Runtime-only Install. The Single File Runtime-only Install should be used in conjunction with the Run/Copy configuration (INI) settings to have similar behaviour to IDC.

Localization

- Alarm string translation changes

  Alarm server records in this release only support a single translation per field. If translations have been used in a previous version, some changes will be required. Changes have also been made to the available formatting.

- Using local language as native

  Languages need to be explicitly defined in your project before they may be used.

- SetLanguage Cicode Function

  Runtime language switching is now achieved using the Login() Cicode function. The existing SetLanguage cicode function has been removed. If using SetLanguaage Cicode function the compiler will raise an error.

OPC Server

- OPC servers need to be explicitly defined in the server section of the project configuration.
- The Program ID has changed. The OPC DA server, program ID is SchneiderElectric.SCADA.OPCDaServer.1. (Old name Citect.OPC.1 and Citect.OPCRemote.1). The DCOM setting needs to be updated based on the new program ID.

Scheduler

- Introduction of Time Scheduler as a new component of the report server (this was also available in 7.20 service pack 3). The Scheduler allows events to be triggered based on states defined for

equipment.

System Migration

- Hardware Requirements

  In 9.0 the minimum and recommended hardware requirements have increased. Load test your system as part of your upgrade procedure to check that the hardware in use is adequate.

- Alarm Server Upgrade

  When doing a live migration, older alarm clients will not connect to a new alarm server process.

- Historian

  Customers using Historian should upgrade their version to Historian v4.40 before upgrading to v7.30.

Security

- Reserved User Names

  Additional reserved user names were introduced in 9.0. When adding users to Power SCADA Operation these reserved names should not be used.

- User Name Restriction

  User Names with a dot in the name are invalid.

Upgrade Information for Version 7.20

- Client Connection Control

  Version 7.20 has introduced the ability to control the client connection to the alarm, trend and report servers. Two new configuration parameters have been added:

  - [ServerType.ClusterName.ServerName] Priority
  - [ServerType.ClusterName.ServerName] DisableConnection

- Persisted I/O Memory Mode

  It is recommended that data assigned to disk I/O devices be migrated to the new persisted memory I/0 mode.

- Super Genies and Environment Variables

  Prior to upgrading to PowerSCADA Expert 7.20, identify and record Super Genie instance page environment variables. After the upgrade, reinstate the Super Genie instance page environment variables. If not, existing Super Genie template environment variables will override the variables, due to synchronization.

- Launch Power SCADA Operation

  An automatic upgrade of your projects will occur when you initially start Power SCADA Operation

- Configure Tags to Use Clustering

  Alarms, reports, trends, SPC tags, and accumulators can now be configured to run in a specific cluster.

- Run the Migration Tool

  The automatic update that occurs when you initially launch PowerSCADA Expert does not fully upgrade your projects, as such it needs to be followed by running the Migration Tool.

- Creation of roles for existing users

  The migration tool will update all existing user definitions to use roles. In 7.20, both users and Windows groups use roles as a common base for security definition. When the migration tool updates the users, an existing role will be used if it matches the configuration of the user, otherwise a new role will be created, such as Role_1, Role_2 etc.

- Copy of XP_Style menu items

  The migration tool will copy any existing XP_Style menu entries to the new menu configuration database. The menu configuration database is a new feature in version 7.20. It is supported by default in the Tab_Style templates and the menu configuration can be accessed using the new menu Cicode functions.

- "Compile the Project" on page 300

  Once you have configured your project, compile it and verify that there are no errors.

- Run the Computer Setup Wizard

  Run the Computer Setup Wizard for each computer running the project. At each stage of the Wizard, configure the appropriate settings for that computer.

- Super Genies

  Performance improvements in v7.20 remove the page display delay which was in previous versions.

  The page properties for a graphics page have a new tab added for associations. This can be used to document existing SuperGenie associations used on pages or SuperGenies. 7.20 allows for associations to support names in place of a numbered index.

  By default, the ability to open and edit an instantiated SuperGenie is not allowed as SuperGenies should be edited via the library. The parameter [CtDraw.RSC] AllowEditSuperGeniePage can be used to enable access to directly edit the instantiated page if required.

  When upgrading from a previous version, existing Super Genie template environment variables will override Super Genie page environment variables. Any manual updates that were made to Super Genie page environment variables prior to the upgrade will be lost.

  Graphics enhancements have been added in version 7.20. Any existing Super Genie Cicode will function as in previous versions. In version 7.20, Super Genies can be launched using meta-data to remove the need for Cicode functions to be created. Super Genie associations support name references and can have properties defined via the page property form.

- System Migration

  Version 7.20 has added trusted network authentication between SCADA servers. The Computer Setup Wizard will allow a system password to be set on each server on your network. Servers that have been configured with the same password will be able to participate in the trusted network for inter-server communication. There is now a requirement to have at least one user defined.

  A compile error will be raised if no user is defined within the project. Version 7.20 installs with the multi-process configuration parameter set to use multi-process. For upgraded projects, this setting should be confirmed when using the Computer Setup Wizard.

- Value, Quality and Timestamps

  Animation that does not have a tooltip will automatically receive a new tooltip that shows the value, quality and timestamp of the variables. This behavior can be disabled using the parameter [Page] EnableQualityToolTip.

Cicode Functions in version 8.2

Some Cicode functions have been introduced. The following sections detail the changes made to these functions:

New Functions

Alarm Functions

AlarmCountEquipment	Counts the available alarms for the given equipments in conjunction with the selected filter criteria.

Modified Functions

No functions have been modified for PowerSCADA Expert 8.2.

Reinstated Functions

No functions have been re-instated for PowerSCADA Expert 8.2.

Deprecated Functions

No functions have been deprecated for PowerSCADA Expert 8.2.

Removed Functions

No functions have been removed for PowerSCADA Expert 8.2.

Cicode Functions in versions 8.1 and 8.0 SR1

Some Cicode functions have been introduced. The following sections detail the changes made to these functions:

New Functions

Net Functions

DllClassDispose	Use this function to clean up resources used by the .Net object and any other .Net objects created via the use of the object.
DllClassCreate	Use this function to instantiate a new .Net object by specifying the path, class and arguments required for the matching constructor of the class.
DllClassGetProperty	Use this function to get a property of the .Net object.
DllClassIsValid	Use this function to validate class. Uses the handle of the class returned from DllClassCreate.
DllClassCallMethod	Use this function to call a method of a .Net object, passing in the method name and any arguments required for the matching prototype of the method.
DllClassSetProperty	Use this function to set a property of the .Net object. The property may be of any type or an object itself.

Modified Functions

Reinstated Functions

Alarm Functions

AlarmDelete	Deletes alarm summary entries that are currently displayed.
AlarmSplit	Splits an alarm summary entry which has no Off time.
AlarmSumAppend	Appends a new blank record to the alarm summary.
AlarmSumCommit	Commits the alarm summary record to the alarm summary device.
AlarmSumDelete	Deletes alarm summary entries.
AlarmSumFind	Finds an alarm summary index for an alarm record and alarm on time.
AlarmSumFirst	Gets the oldest alarm summary entry.
AlarmSumGet	Gets field information from an alarm summary entry.
AlarmSumLast	Gets the latest alarm summary entry.
AlarmSumNext	Gets the next alarm summary entry.
AlarmSumPrev	Gets the previous alarm summary entry.
AlarmSumSet	Sets field information in an alarm summary entry.
AlarmSumSplit	Duplicates an alarm summary entry.

AlarmSumType	Retrieves a value that indicates a specified alarm's type.

Deprecated Functions

No functions have been deprecated for these versions.

Removed Functions

No functions have been removed for these versions.

Cicode Functions in 7.40 and 8.0

Some Cicode functions have been introduced. The following sections detail the changes made to these functions.

New Functions

Security Functions

GetLanguage	Gets the language currently used on the display client.

Page Functions

PageListCount	Gets number of pages in the page list of the current window.
PageListCurrent	Gets index of the current page in the page list of the current window.
PageListInfo	Gets information of a page at the specific index in the page list of current window.
PageListDisplay	Displays a page at the specific index in the page list of the current window, and moves the current index to the page. When a page is recalled, the original parameters (such as cluster context, super genie associations, PageTask arguments if applicable) used to display the page will be restored.
PageListDelete	Deletes a page at the specific index from the page list of the current window.

XML Functions

XMLClose	Deletes an XML document in memory
XMLCreate	Creates a new XML document in memory
XMLGetAttribute	Retrieves the attribute value of the node from an XML document in memory

XMLGetAttributeCount	Retrieves the number of attributes (properties of a node. Each attribute has a name and a value) within an XML document in memory
XMLGetAttributeName	Retrieves the name of an attribute (property of a node. Each attribute has a name and a value) within an XML document in memory
XMLGetAttributeValue	Retrieves the value of an attribute (property of a node. Each attribute has a name and a value) within an XML document in memory
XMLGetChild	Retrieves the child node for the specified parent node in XML document in memory
XMLGetChildCount	Retrieves the total number of child nodes for the specified parent node in an XML document in memory
XMLGetParent	Retrieves the parent node within the contents of an XML document in memory
XMLGetRoot	Retrieves the root node of an XML document in memory
XMLNodeAddChild	Creates an element node with the specified Name and Namespace and appends the node to the end of the list of child nodes of specified parent node in the XML document.
XMLNodeFind	Selects a single node from the contents of an XML document in memory
XMLNodeGetName	Retrieves the name of the specified node
XMLNodeGetValue	Retrieves the value of a node from the contents of an XML document in memory
XMLNodeRemove	Removes specified XML node from its parent and XML document
XMLNodeSetValue	Sets the value of the specified node.
XMLOpen	Loads an XML file from disk
XMLSave	Saves an XML file to disk
XMLSetAttribute	Sets the value of specified attribute of the node in the XML document. If the attribute does not exist, it will be created.

Modified Functions

Alarm Functions

AlarmGetInfo	Gets data on the alarm list displayed at a specified AN. A new type of 13 was added to return the ready state of the data on an alarm display view.

Super Genie Functions

Ass	Associates a variable tag with a Super Genie. Now supports Equipment.Item.

Tag Functions

Ass	Associates a variable tag with a Super Genie. Now supports Equipment.Item.
Ass	Associates a variable tag with a Super Genie. Now supports Equipment.Item.
Ass	Associates a variable tag with a Super Genie. Now supports Equipment.Item.
Ass	Associates a variable tag with a Super Genie. Now supports Equipment.Item.
Ass	Associates a variable tag with a Super Genie. Now supports Equipment.Item.
Ass	Associates a variable tag with a Super Genie. Now supports Equipment.Item.
Ass	Associates a variable tag with a Super Genie. Now supports Equipment.Item.
Ass	Associates a variable tag with a Super Genie. Now supports Equipment.Item.
Ass	Associates a variable tag with a Super Genie. Now supports Equipment.Item.
Ass	Associates a variable tag with a Super Genie. Now supports Equipment.Item.
Ass	Associates a variable tag with a Super Genie. Now supports Equipment.Item.
Ass	Associates a variable tag with a Super Genie. Now supports Equipment.Item.
Ass	Associates a variable tag with a Super Genie. Now supports Equipment.Item.
Ass	Associates a variable tag with a Super Genie. Now supports Equipment.Item.
Ass	Associates a variable tag with a Super Genie. Now supports Equipment.Item.

Reinstated Functions

No functions have been reinstated for 7.40 SP1.

Deprecated Functions

No functions have been deprecated for 7.40 SP1.

Removed Functions

No functions have been removed for 7.40 SP1.

Cicode Functions in 7.30

Some Cicode functions have been introduced, modified, deprecated or removed. The following sections detail the changes made to these functions:

New Functions

Alarm Functions

AlarmAckTag	Acknowledge a specified alarm.
AlarmCount	Counts the available alarms for the selected filter criteria.
AlarmCountList	Counts the available alarms for the selected alarm list (selected by its animation).
AlmBrowseAck	Acknowledges the alarm tag at the current cursor position in an active data browse session.
AlmBrowseClose	Closes an alarm tags browse session.
AlmBrowseDisable	Disables the alarm tag at the current cursor position in an active data browse session.
AlmBrowseEnable	Enables the alarm tag at the current cursor position in an active data browse session.
AlmBrowseFirst	Gets the oldest alarm tags entry.
AlmBrowseGetField	Gets the field indicated by the cursor position in the browse session.
AlmBrowseNext	Gets the next alarm tags entry in the browse session.
AlmBrowseNumRecords	Returns the number of records in the current browse session.
AlmBrowseOpen	Opens an alarm tags browse session.
AlmBrowsePrev	Gets the previous alarm tags entry in the browse session.
AlarmFilterClose	Removes named filter from memory.
AlarmFilterEditAppend	Appends the provided expression to the current filter session content without any validation.
AlarmFilterEditClose	Removes the session from the memory.

AlarmFilterEditCommit	Validates the filter built in this session and, if valid, applies the filter to the list associated with the session.
AlarmFilterEditFirst	Retrieves the first part of the filter.
AlarmFilterEditLast	Retrieves the last part of the filter.
AlarmFilterEditNext	Retrieves the next part of the filter.
AlarmFilterEditOpen	Creates a session for the historical list associated with the provided animation number (aN).
AlarmFilterForm	Displays a form for specifying filtering criteria for either an alarm list or a named filter.
AlarmFilterOpen	Creates a named filter.
AlmFilterEditPrev	Retrieves the previous part of the filter.
AlmFilterEditSet	Replaces the current filter session content by the provided expression without any validation.
AlarmGetFilterName	Retrieves the name of the linked filter for the supplied AN.
AlarmResetQuery	Clears the filter of the specified filter source. Used to reset the filter set up by the Cicode function AlarmFilterForm().
LibAlarmFilterForm	Displays a generic alarm filter pop-up for specifying filtering criteria for either an alarm list or a named filter.

Equipment Functions

EquipSetProperty	Sets the property of an item of equipment.
EquipStateBrowseClose	Terminates a browsing session and cleans up the resources used by the session.
EquipStateBrowseFirst	Places the data browse cursor at the first record.
EquipStateBrowseGetField	Returns the value of the particular field in a record to which the data browse cursor is currently referencing.
EquipStateBrowseNext	Places the data browse cursor at the next available record.
EquipStateBrowseNumRecords	Returns the number of records that match the current filter criteria.
EquipStateBrowseOpen	Initiates a new session for browsing the equipment states configured.

EquipStateBrowsePrev	Places the data browse cursor at the previous record.

Page Functions

PageSOE	Displays a category of sequence of events (SOE) entries on the SOE page.

Scheduler Functions

SchdClose	Terminates a browsing session and cleans up the resources used by the session.
SchdConfigClose	Terminates a browsing session and cleans up the resources used by the session.
SchdConfigFirst	Places the data browse cursor at the first record.
SchdConfigGetField	Returns the value of the particular field in a record to which the data browse cursor is currently referencing.
SchdConfigNext	Places the data browse cursor at the next available record.
SchdConfigNumRecords	Returns the number of records that match the current filter criteria.
SchdConfigOpen	Initiates a new session for browsing the schedules configured.
SchdConfigPrev	Places the data browse cursor at the previous record.
SchdFirst	Places the data browse cursor at the first record.
SchdGetField	Returns the value of the particular field in a record to which the data browse cursor is currently referencing.
SchdNext	Places the data browse cursor at the next available record.
SchdNumRecords	Returns the number of records that match the current filter criteria.
SchdOpen	Initiates a new session for browsing the runtime schedules.
SchdPrev	Places the data browse cursor at the previous record.
SchdSpecialAdd	Adds a new special day group to the scheduler engine.

SchdSpecialClose	Terminates a browsing session and cleans up the resources used in the session.
SchdSpecialDelete	Deletes an existing special day group.
SchdSpecialFirst	Places the data browse cursor at the first record.
SchdSpecialGetField	Returns the value of the particular field in a record to which the data browse cursor is currently referencing.
SchdSpecialItemAdd	Adds a new special day to the scheduler engine.
SchdSpecialItemClose	Terminates a browsing session and cleans up the resources used in the session.
SchdSpecialItemDelete	Deletes an existing schedule.
SchdSpecialItemFirst	Places the data browse cursor at the first record.
SchdSpecialItemGetField	Returns the value of the particular field in a record to which the data browse cursor is currently referencing.
SchdSpecialItemModify	Modifies an existing special day.
SchdSpecialItemNext	Places the data browse cursor at the next available record.
SchdSpecialItemNumRecords	Returns the number of records that match the current filter criteria.
SchdSpecialItemOpen	Initiates a new session for browsing the special days.
SchdSpecialItemPrev	Places the data browse cursor at the previous record.
SchdSpecialModify	Modifies an existing special day group
SchdSpecialNext	Places the data browse cursor at the next available record.
SchdSpecialNumRecords	Returns the number of records that match the current filter criteria.
SchdSpecialOpen	Initiates a new session for browsing the special day groups.
SchdSpecialPrev	Places the data browse cursor at the previous record.
ScheduleItemAdd	Adds a new schedule to the scheduler engine.
ScheduleItemDelete	Deletes an existing schedule.
ScheduleItemModify	Modifies an existing schedule.

ScheduleItemSetRepeat	Adds recurrence information for an existing schedule to the scheduler engine.

Sequence of Events Functions

SOEArchive	Archives event journal.
SOEDismount	Use to dismount archive volume.
SOEEventAdd	Inserts a new event into the event journal.
SOEMount	Use to mount archive volume.

SQL Functions

SQLCall	Executes an SQL query on a database
SQLClose	Closes a SQL connection between the DB connection object specified by the function's parameter and a database
SQLCreate	Creates an internal DB connection object and returns a handle to the object for use by the other DB functions
SQLDispose	Disposes the DB connection object
SQLGetRecordset	Executes an SQL query on a database
SQLGetScalar	Executes an SQL query on a database
SQLIsNullField	Checks presence of null value in field from a recordset
SQLNumFields	Gets the number of fields or columns that were returned by the last SQL statement
SQLOpen	Opens an SQL connection between the DB connection object
SQLParamsClearAll	Turns on a debug trace
SQLParamsSetAsInt	Adds or replaces a parameterized query's parameter as integer and its value in the specified connection
SQLParamsSetAsReal	Adds or replaces a parameterized query's parameter as real and its value in the specified connection
SQLParamsSetAsString	Adds or replaces a parameterized query's parameter as string and its value in the specified connection
SQLPrev	Gets the previous database record from an SQL query.
SQLQueryCreate	The function creates a new query and returns its handle

SQLQueryDispose	The function disposes the query which handle is given as the argument
SQLRowCount	Gets the number of rows in the recordset.

Timestamp Functions

StrToTimestamp	Converts timestamp in a STRING format into a TIMESTAMP format

Tag Functions

TagBrowseClose	Close an existing browsing session
TagBrowseFirst	Move to the first record
TagBrowseGetField	Get the specified field of a record.
TagBrowseNext	Move to the next record
TagBrowseNumRecords	Get the number of records for a given browsing session.
TagBrowseOpen	Opens a new browsing session.
TagBrowsePrev	Move to the previous record

Modified Functions

Alarm Functions

AlarmAck	Acknowledges an active alarm.
AlarmCatGetFormat	Returns the display format string of the specified alarm category. Type has been extended to include SOE format.
AlarmDisable	Disables an alarm.
AlarmDsp	Displays alarms.
AlarmDspNext	Displays the next page of alarms. Works with new SOE display type.
AlarmDspPrev	Displays the previous page of alarms. Works with new SOE display type.
AlarmEnable	Enables a disabled alarm.

AlarmFirstTagRec AlarmFirstCatRec AlarmFirstPriRec AlarmFirstQueryRec AlarmNextTagRec AlarmNextCatRec AlarmNextPriRec AlarmNextQueryRec AlarmAckRec AlarmEnableRec AlarmDisableRec AlarmGetDelayRec AlarmSetDelayRec AlarmGetThresholdRec AlarmSetThresholdRec AlarmGetFieldRec	Alarm 'Rec' functions listed are now executed in the client process, with the function MsgRPC no longer required when called remotely to the Alarm Server.
AlarmGetDsp	Retrieves field data from the alarm record that is displayed at the specified AN. Works with new SOE display type.
AlarmGetInfo	Retrieves data on the alarm list displayed at a specified AN. New type 12 added.
AlarmSetInfo	Controls different aspects of the alarm list displayed at a specified AN. Supports automatic refresh of the new SOE display type.
AlmSummaryGetField	Gets the field indicated by the cursor position in the browse session. Now supports Equipment field.
AlmSummaryOpen	Opens an alarm summary browse session. Now supports Equipment field. Will not return data for 'NODE' field name.
AlmTagsGetField	Gets the field indicated by the cursor position in the browse session. Now supports Equipment field.
AlmTagsOpen	Opens an alarm tags browse session. Now supports Equipment field. Will not return data for 'NODE' field name.

Accumulator Functions

AccumBrowseGetField	Gets the field indicated by the cursor position in the browse session. Now supports Equipment field.
AccumBrowseOpen	Opens an accumulator browse session. Now supports Equipment field.

Equipment Functions

EquipBrowseGetField	Gets the field indicated by the cursor position in the browse session. Now supports Parent and Composite fields.
EquipBrowseOpen	Opens an equipment database browse session. Now supports Parent and Composite fields.
EquipGetProperty	Reads a property of an equipment database record from the EQUIP.DBF file. Now supports Parent and Composite fields.

Format Functions

FmtOpen	Opens a format template. mode has been extended to include SOE format.

Security Functions

Login	Logs a user into the Power SCADA Operation system, using Power SCADA Operation security and gives users access to the areas and privileges assigned to them in the Users database. New sLanguage parameter added.
UserLogin	Logs a user into the Power SCADA Operation system, using either Windows security or Power SCADA Operation security and gives users access to the areas and privileges assigned to them in the Users database. New sLanguage parameter added.

Server Functions

ServeGetProperty	Returns information about a specified server and can be called from any client.
ServerReload	Reloads the server specified by cluster and server name.

Super Genie Functions

AssGetProperty	Gets association information about the current Super Genie from the datasource.
AssInfo	Gets association information about the current Super Genie.
AssInfoEx	Gets association information about the current Super Genie.

SQL Functions

SQLGetField	Gets field or column data from a database record.
SQLInfo	Gets information about a database connection. No longer supports type 3 and 4.
SQLNoFields	Gets the number of fields or columns that were returned by the last SQL statement.

Tag Functions

TagGetProperty	Gets a property for a variable tag from the datasource. Now supports Equipment field.
TagInfo	Gets information about a variable tag. Now supports Equipment field.
TagInfoEx	Gets information about a variable tag. Now supports Equipment field.

Trend Functions

TrnBrowseGetField	Gets the field indicated by the cursor position in the browse session. Now supports Equipment field.
TrnBrowseOpen	Opens a trend browse session. Now supports Equipment field.

Reinstated Functions

No functions have been reinstated for 7.30.

Deprecated Functions

AlmTagsEnable	Enables the alarm tag at the current cursor position in an active data browse session.
AlmTagsDisable	Disables the alarm tag at the current cursor position in an active data browse session.
AlmTagsNext	Gets the next alarm tags entry in the browse session.
AlmTagsAck	Acknowledges the alarm tag at the current cursor position in an active data browse session.
AlmTagsClear	Clears the alarm tag at the current cursor position in an active data browse session.
AlmTagsClose	Closes an alarm tags browse session.
AlmTagsFirst	Gets the oldest alarm tags entry.
AlmTagsGetField	Gets the field indicated by the cursor position in the browse session.
AlmTagsNumRecords	Returns the number of records in the current browse session.

AlmTagsOpen	Creates a session for the historical list associated with the provided animation number (aN).
AlmTagsPrev	Gets the previous alarm tags entry in the browse session.

Removed Functions

AlmBrowseClear	Clears the alarm tag at the current cursor position in an active data browse session. Now obsolete.
AlarmClear	Clears acknowledged, inactive alarms from the active alarm list.
AlarmClearRec	Clear an alarm by its record number. Now obsolete.
AlarmDelete	Deletes alarm summary entries that are currently displayed. Now obsolete.
AlarmSetQuery	Specifies a query to be used in selecting alarms for display. Now Obsolete. Use the new Alarm Filter Edit functions.
AlarmSumAppend	Appends a new blank record to the alarm summary. Now obsolete.
AlarmSumCommit	Commits the alarm summary record to the alarm summary device. Now obsolete.
AlmSummaryCommit	Commits the alarm summary record to the alarm summary device. Now obsolete.
AlarmSplit	Duplicates an alarm summary entry where the cursor is positioned. Now obsolete.
AlarmSumDelete	Deletes alarm summary entries. Now obsolete.
AlarmSumFind	Finds an alarm summary index for an alarm record and alarm on time. Now obsolete.
AlarmSumFindExact	Finds the alarm summary index for an alarm specified by the alarm record identifier and alarm activation time.
AlarmSumFirst	Gets the oldest alarm summary entry. Now obsolete.

AlarmSumGet	Gets field information from an alarm summary entry. Now obsolete.
AlarmSumLast	Gets the latest alarm summary entry. Now obsolete.
AlarmSumNext	Gets the next alarm summary entry. Now obsolete.
AlarmSumPrev	Gets the previous alarm summary entry. Now obsolete.
AlarmSumSet	Sets field information in an alarm summary entry. Now obsolete.
AlmSummarySetFieldValue	Sets the value of the field indicated by the cursor position in the browse session. Now obsolete.
AlarmSumSplit	Duplicates an alarm summary entry. Now obsolete.
AlarmSumType	Retrieves a value that indicates a specified alarm's type. Now obsolete.
QueryFunction	The user-defined query function set in AlarmSetQuery. Now obsolete.

Miscellaneous Functions

SetLanguage	Sets the language database from which the local translations of native strings in the project will be drawn, and specifies the character set to be used. Now obsolete. Use the Login(), UserLogin() and LoginForm() to set the preferred language.

Cicode Functions in 7.20

Some Cicode functions have been introduced, modified, deprecated or removed. The following sections detail the changes made to these functions:

New Functions

Alarm Functions

AlarmCatGetFormat	Returns the display format string of the specified alarm category.
AlarmDspClusterAdd	Adds a cluster to a client's alarm list.
AlarmDspClusterInUse	Determines if a cluster is included in a client's alarm list.
AlarmDspClusterRemove	Removes a cluster from a client's alarm list.

Display Functions

DspAnGetMetadata	Retrieves the field value of the specified metadata entry.
DspAnGetMetadataAt	Retrieves metadata information at the specified index.
DspAnSetMetadata	Non-blocking function, that sets the value of the specified metadata entry.
DspAnSetMetadataAt	Sets the value of a metadata entry.
DspPopupConfigMenu	Displays the contents of a menu node as a pop-up (context) menu, and run the command associated with the selected menu item.

Format Functions

FmtGetFieldCount	Retrieves the number of fields in a format object.
FmtGetFieldName	Retrieves the name of a particular field in a format object.
FmtGetFieldWidth	Retrieves the width of a particular field in a format object.

Menu Functions

MenuGetChild	Returns the handle to the child node with the specified name.
MenuGetFirstChild	Returns the handle to the first child of a menu node.
MenuGetGenericNode	Returns the root node of the default menu tree.
MenuGetNextChild	Returns the next node that shares the same parent.
MenuGetPageNode	Returns the Base menu node of a specific page.
MenuGetParent	Returns the parent node of the menu item.
MenuGetPrevChild	Returns the previous node that shares the same parent.
MenuGetWindowNode	Returns the handle of the root menu node for a given window.
MenuNodeAddChild	Dynamically adds a new item to the menu at runtime.
MenuNodeGetProperty	Return the item value of the specified menu node.
MenuNodeHasCommand	Checks whether the menu node has a valid Cicode command associated with it.

MenuNodeIsDisabled	Checks whether the menu node is disabled by evaluating its DisabledWhen Cicode expression.
MenuNodeIsHidden	Checks whether the menu node is hidden by evaluating its HiddenWhen Cicode expression.
MenuNodeRemove	Remove the menu node from the menu tree.
MenuNodeRunCommand	Run the associated command for a menu node.
MenuNodeSetDisabledWhen	Set the DisabledWhen expression for a newly added node.
MenuNodeSetHiddenWhen	Set the HiddenWhen expression for a newly added node.
MenuNodeSetProperty	Set the item value of the specified menu node.
MenuReload	Reload base Menu Configuration from the compiled database.

Miscellaneous Functions

GetLogging	Gets the current value for one or more logging parameters.
SetLogging	Adjusts logging parameters while online.
ProductInfo	Returns information about the Power SCADA Operation product.
ProjectInfo	Returns information about a particular project, which is identified by a project enumerated number.

Page Functions

PageBack	Displays the previously displayed page in the Window.
PageForward	PageForward() restores the previously displayed page in the window following a PageBack command.
PageHistoryDspMenu	Displays a pop-up menu which lists the page history of current window.
PageHistoryEmpty	Returns whether page history of the current window is empty.
PageHome	Displays the predefined home page in the window.
PagePeekCurrent	Return the index in the page stack for the current page.

PageProcessAnalyst	Displays a Process Analyst page (in the same window) preloaded with the pre-defined Process Analyst View (PAV) file.
PageProcessAnalystPens	Displays a Process Analyst page (in the same window) preloaded with the pre-defined Process Analyst View (PAV) file and specified trend or variable tags.
PageRecall	Displays the page at a specified depth in the stack of previously displayed pages.
PageTask	Used for running preliminary Cicode before displaying a page in a window.
PageTransformCoords	Converts Page coordinates to absolute screen coordinates.

Process Analyst Functions

ProcessAnalystLoadFile	Loads the specified PAV file to a Process Analyst object, which is identified by parameter ObjName.
ProcessAnalystPopup	Displays a Process Analyst page (in the same window) preloaded with the pre-defined Process Analyst View (PAV) file and specified trend or variable tags.
ProcessAnalystSelect	Allows a set of pens to be selected before displaying the PA page.
ProcessAnalystSetPen	Allows a new pen to be added to a PA display.
ProcessAnalystWin	Displays a Process Analyst page (in a new window) preloaded with the pre-defined Process Analyst View (PAV) file.

Quality Functions

QualityCreate	Creates a quality value based on the quality fields provided.
QualityGetPart	Extracts a requested part of the Quality value from the QUALITY variable.
QualityIsBad	Returns a value indicating whether the quality is bad.
QualityIsGood	Returns a value indicating whether the quality is good.
QualityIsUncertain	Returns a value indicating whether the quality is uncertain.
QualitySetPart	Sets a Quality part's value to the QUALITY variable.

QualityToStr	Returns a textual representation of the Power SCADA Operation quality.
QualityIsOverride	Returns a value indicating whether the tag is in Override Mode.
QualityIsControlInhibit	Returns a value indicating whether the tag is in Control inhibit mode.
VariableQuality	Extracts the quality from a given variable.

Server Functions

ServerBrowseClose	This function terminates an active data browse session and cleans up resources associated with the session.
ServerBrowseFirst	This function places the data browse cursor at the first record.
ServerBrowseGetField	This function retrieves the value of the specified field from the record the data browse cursor is currently referencing.
ServerBrowseNext	This function moves the data browse cursor forward one record.
ServerBrowseNumRecords	This function returns the number of records that match the filter criteria.
ServerBrowseOpen	This function initiates a new browse session and returns a handle to the new session that can be used in subsequent data browse function calls.
ServerBrowsePrev	This function moves the data browse cursor back one record.
ServerGetProperty	This function returns information about a specified server and can be called from any client.
ServerReload	This function reloads the server specified by cluster and server name.
ServerIsOnline	This function checks if the given server can be contacted by the client for giving the online/offline status of the server.

String Functions

StrCalcWidth	Retrieves the pixel width of a string using a particular font.
StrTruncFont	Returns the truncated string using a particular font (specified by name) or the specified number of characters.

StrTruncFontHnd	Returns the truncated string using a particular font (specified by font number) or the specified number of characters.

Super Genie Functions

AssMetadata	Performs Super Genie associations using the "Name" and "Value" fields.
AssMetadataPage	Uses the metadata information from the current animation point for the page associations for a new Super Genie page, and displays the new Super Genie in the current page.
AssMetadataPopup	Uses the metadata information from the current animation point for the associations for a new Super Genie page, and displays the new Super Genie in a new pop up window.
AssMetadataWin	Uses the metadata information from the current animation point for the associations for a new Super Genie page, and displays the new Super Genie in a new window.

Tag Functions

SubscriptionGetInfo	Reads the specified text information about a subscribed tag.
SubscriptionGetQuality	Reads quality of a subscribed tag.
SubscriptionGetTag	Reads a value, quality and timestamps of a subscribed tag.
SubscriptionGetTimestamp	Reads the specified timestamp of a subscribed tag.
SubscriptionGetValue	Reads a value of a subscribed tag.
TagSetOverrideBad	Sets a quality Override element for a specified tag to Bad Non Specific.
TagSetOverrideGood	Sets a quality Override element for a specified tag to Good Non Specific.
TagSetOverrideUncertain	Sets a quality Override element for a specified tag to Uncertain Non Specific.
TagSetOverrideQuality	Sets a quality of Override element for a specified tag.

Task Functions

TaskCall	Calls a Cicode function by specifying the function name and providing an arguments string.

Timestamp Functions

TimestampToStr	Converts a TIMESTAMP variable into a string.
TimestampDifference	Returns a difference between two TIMESTAMP variables as a number of milliseconds.
TimestampCreate	Returns a timestamp variable created from the parts.
TimestampFormat	Format a TIMESTAMP variable into a string.
TimestampGetPart	Returns one part (year, month, day, etc) of the timestamp variable.
TimestampToTimeInt	Converts a TIMETSTAMP variable into a time INTEGER which is represented as a number of seconds since 01/01/1970.
TimeIntTo Timestamp	Converts a time INTEGER which is represented as a number of seconds since 01/01/1970 to a TIMETSTAMP
TimestampCurrent	Returns the current system date and time as a TIMESTAMP variable.
TimestampAdd	Adds time (in milliseconds) to a TIMESTAMP variable.
TimestampSub	Subtracts time (in milliseconds) from a TIMESTAMP variable.
VariableTimestamp	Extract the TIMESTAMP from a given variable.

Window Functions

MultiMonitorStart	Displays a Power SCADA Operation window on each of the configured monitors when a display client starts up.
WinSetName	Associates a name with a particular window by its window number.
WndMonitorInfo	Returns information about a particular monitor.

Modified Functions

Accumulator Functions

AccumBrowseOpen	Opens an accumulator browse session.

Alarm Functions

AlarmDsp	Displays alarms.
AlarmDspLast	Displays the latest, unacknowledged alarms.
AlmSummaryOpen	Opens an alarm summary browse session.
AlmTagsOpen	Opens an alarm tags browse session.

Display Functions

DspStr	Displays a string at an AN.
DspText	Displays text at an AN.

Format Functions

FmtOpen	Creates a format template.

Miscellaneous Functions

Shutdown	Ends Power SCADA Operation operation.

Page Functions

PageGetInt	Gets a local page-based integer.
PageGetStr	Gets a local page-based string.
PageInfo	Gets information about the current page.
PagePeekLast	Gets any page on the PageLast stack.
PageSetInt	Stores a local page-based integer.
PagesetStr	Stores a local page-based string.

Security Functions

Login	Logs an operator into the Power SCADA Operation system. Not available when logged in as Windows user.

Super Genie Functions

The following functions were updated to accept string identifiers for substitution parameters.

Ass	Associates a variable tag with a Super Genie.
AssGetProperty	Retrieves association information about the current Super Genie from the datasource.
AssGetScale	Gets scale information about the associations of the current Super Genie from the datasource (that is scale information about a variable tag that has been substituted into the Super Genie)
AssInfo	Gets association information about the current Super Genie (that is information about a variable tag that has been substituted into the Super Genie).
AssInfoEx	Retrieves association information about the current Super Genie (that is information about a variable tag that has been substituted into the Super Genie).

AssScaleStr	Gets scale information about the associations of the current Super Genie (that is scale information about a variable tag that has been substituted into the Super Genie).

Tag Functions

SubscriptionGetAttribute	Reads an attribute value of a tag subscription.
TagRead	Reads the value of a particular tag element.
TagWrite	Writes a tag element value for the tag elements which have read/write access.
TagSubscribe	Subscribes to a particular tag element.

Window Functions

WinNumber	Gets the window number of the active Power SCADA Operation window.
WndInfo	Gets the Windows system metrics information.

Reinstated Functions

Following functions have been reinstated for 7.20.

Time and Date Functions

TimeSet	Sets the new system time. Requires UAC to be disabled in order for the time to be set.

Citect.ini parameters in 8.2

New Parameters

The following parameters are new or have been altered in this release. For an entire list of the system parameters, refer to the Parameters documentation.

Deployment Parameters

[CtEdit]Deploy	The location where a project will be stored when a deployment package is received from the deployment server.
[Deployment]AskRestartArgs	Passes arguments to the Cicode function called by [Deployment]AskRestartFunc.
[Deployment]AskRestartFunc	Calls a Cicode function instead of a displaying a restart notification dialog when a prompted deployment occurs.
[Deployment]Enabled	Determines if Runtime Manager runs a project that has been deployed from the deployment server, or the Active Project.

Modified Parameters

[Win]Configure	Determines whether Name of environment and Graphics Builder options are displayed on the control many of the runtime system.

Removed Parameters

[Lan]SecureLogin	[LAN]SecureLogin is no longer supported.

Obsolete Parameters

[OID]Reset	Resets all OIDs (Object IDs) at compile. This parameter has been removed.
[CtEdit]MaxFields	The maximum number of fields that can display on a Citect Project Editor form.
[CtEdit]ShowToolbar	Shows / hides the toolbar in the Citect Project Editor.

Citect.ini parameters in 8.1 and 8.0 SR1

This topic lists the parameters that have changed in PowerSCADA Expert versions 8.0 SR1 and 8.1.

New Parameters

Alarm Parameters

[Alarm]AlarmListRequestTimeout	Specifies the length of time (in seconds) that an alarm display will wait to receive data from all clusters.
[Alarm]DBLogDBServer	Use to turn on logging for the ClearSCADA Database Server.
[Alarm]DBLogHistoric	When set to 119 this parameter provides logging of historic ClearSCADA data.
[Alarm]DBLogServerCore	Used to find redundancy and synchronization issues.
[Alarm]DeltaTimeUpdate	Determines if DeltaTime (duration) field is set on non-OFF alarms by calculating volatile duration between current time and the time when the alarm was activated.
[Alarm]DisableSOE	Used to turn off the processing for the event journal.
[Alarm]DisableSummary	Allows a user to turn off processing for the summary events in the alarm server.
[Alarm]IsolationDetectInterval	Sets the interval between ICMP packets to detect network isolation on alarm servers.
[Alarm]IsolationDetectIP1	Determines status of the disconnected alarm server when network communication has been interrupted.
[Alarm]IsolationDetectIP2	Determines status of the disconnected alarm server when network communication has been interrupted.

[Alarm]IsolationDetectRetryCount	Sets the ICMP retry count to detect network isolation on alarm servers.
[Alarm]MaxQueryExecuteTime	Creates a log entry if an internal SQL query takes longer than a specified amount of time.
[Alarm]MemoryWarningLimit	Value in Mb, of the threshold of the alarm server memory.
[Alarm]SummaryAutoRefreshMode	Represents the default value for AlarmSetInfo type 15.
[Alarm]SummaryTimeoutTolerance	The length of time from timeout after which an alarm summary entry is committed to Summary Device regardless the fact that Off Time is not set.

Alarm Process Parameters

[Alarm<ClusterName><ServerName>]IsolationDetectInterval	Sets the interval between ICMP packets to detect network isolation on alarm servers.
[Alarm<ClusterName><ServerName>]IsolationDetectIP1	Determines status of the disconnected alarm server when network communication has been interrupted.
[Alarm<ClusterName><ServerName>]IsolationDetectIP2	Determines status of the disconnected alarm server when network communication has been interrupted.
[Alarm<ClusterName><ServerName>]IsolationDetectRetryCount	Sets the ICMP retry count to detect network isolation on alarm servers.

CtEdit Parameters

[CtEdit] IncrementalEquipmentUpdate	Determines whether an incremental equipment update will occur.

DBClient Parameters

[DBClient]Enabled	Enables ODBC logging.
[DBClient]FileBase	Specifies a location for the ODBC log files.
[DBClient]MaxFiles	Specifies the maximum number of ODBC log files that are retained.
[DBClient]MaxSize	Specifies the maximum size for an ODBC log file (in kilobytes).
[DBClient]OldFiles	Specifies the maximum number of log file sets that can be retained.

LAN Parameters

[LAN]HeartbeatPeriod	Controls how frequently a tran channel sends a heartbeat packet to the other peer.
[LAN]HeartbeatTimeout	Controls how much idle time on network is accepted prior to dropping the tran connection.

Modified Parameters

[Alarm]DisplayDisable	In Power SCADA Operation 8.1, when you set this parameter to 1 (disabled alarms are suppressed), disabled alarm will now be listed on the Alarm Summary page.
[Debug]CategoryFilter	New alarm filters are now supported.
[LAN]EarliestLegacyVersion	The allowable values were updated and the default value is now "7500".

Reinstated Parameters

[Alarm]StartTimeout	Sets the timeout period for loading each packet from the primary Alarms Server. This parameter has been reinstated for v2015 only.

Obsolete Parameters

Alarm Parameters

[Alarm]ArgyleTagValueTimeout	Defines the length of time that the alarm server will wait for argyle tag values to become available (without error) before starting to scan for argyle alarms.
[Alarm]DefaultSOETimeRange	Applies a time range filter to all SOE queries.
[Alarm]SOERowLimit	Defined the maximum number of SOE rows per cluster that can be displayed on an SOE page.
[Alarm]SummaryLength	The maximum number of alarm summary entries that can be held in memory.
[Alarm]SumStateFix	Determined whether an alarm summary entry maintained its state information when the alarm changed to an OFF state.

LAN Parameters

[LAN]KeepAliveInterval	Sets the length of time between two keep alive transmissions by the client.
[LAN]KeepAliveTime	Sets the length of time between two keep alive transmissions in idle conditions.

Citect.ini parameters in 7.40 SP1

This topic lists the parameters that have been added or changed in PowerSCADA Expert version 7.40 SP1.

New Parameters

Alarm Parameters

[Alarm]WebClientUpdatePollPeriod	Sets the polling period in milliseconds for web client to get data updates.
[Alarm]ClientUpdatePollPeriod	Sets the polling period in milliseconds for the display client to get data updates.

Modified Parameters

No parameters were modified in 7.40 SP1

Obsolete Parameters

No parameters were made obsolete in 7.40 SP1

Citect.ini parameters in 7.40

This topic lists the parameters that have been added or changed in PowerSCADA Expert version 7.40:

New Parameters

CTEdit Parameters

[CTEDIT]DisplayEquipmentItem	Used to control the population of the variable tag list, or equipment item list in graphics builder.

General Parameters

[General]TagDBReloadOnChange	Determines whether the Variable Tags database is checked for changes and reloaded when a new page is displayed.

Page Parameters

[Page]MaxList	The maximum number of pages that can be placed on the page list stack.

Server Parameters

[Server]AllowAnonymousAccess	Determines whether the EWS Server will allow the EWS Client anonymous data access.

Modified Parameters

Code Parameters

[Code]Stack	The size of the Cicode stack. The default has been changed from 127 to 256.

General Parameters

[General]TagDB	Determines whether the Variable Tags database is loaded at runtime. The Variable Tags database needs to be loaded to allow tags to be referenced with the Equipment.Item syntax.

Citect.ini parameters in 7.30

This topic lists the parameters that have been added or changed in PowerSCADA Expert version 7.30

New Parameters

Alarm Parameters

[Alarm]DefaultSOETimeRange	Specifies the default time range, in days, for SOE views that have no other time-based filter.
[Alarm]DefSOEFmt	Specifies an SOE display format to use if the SOE Display Format field is blank (in Alarm Categories).
[AlarmFilterRuleList.Active]Rule<n>	Defines the name of rules to appear on the Simple Rule dropdown list of the active alarm filter form.
[AlarmFilterRuleList.Disabled]Rule<n>	Defines the name of rules to appear on the Simple Rule dropdown list of disabled alarm filter form.
[AlarmFilterRuleList.SOE]Rule<n>	Defines the name of rules to appear on the Simple Rule dropdown list of alarm summary filter form.
[AlarmFilterRuleList.Summary]Rule<n>	Defines the name of rules to appear on the Simple Rule dropdown list of alarm summary filter form.
[AlarmFilterRules]<RuleName>	Defines the filter expression represented by the rule name.
AlarmFilterRuleList].Rule<n>	Defines the name of the common rules to appear on the Simple Rule dropdown list of all alarm filter form.

AlarmFilterRules Parameters

[AlarmFilterRuleList.Active]Rule<n>	Defines the name of rules to appear on the Simple Rule dropdown list of the active alarm filter form.

[AlarmFilterRuleList.Disabled]Rule<n>	Defines the name of rules to appear on the Simple Rule dropdown list of disabled alarm filter form.
[AlarmFilterRuleList.SOE]Rule<n>	Defines the name of rules to appear on the Simple Rule dropdown list of alarm summary filter form.
[AlarmFilterRuleList.Summary]Rule<n>	Defines the name of rules to appear on the Simple Rule dropdown list of alarm summary filter form.
[AlarmFilterRules]<RuleName>	Defines the filter expression represented by the rule name.
AlarmFilterRuleList].Rule<n>	Defines the name of the common rules to appear on the Simple Rule dropdown list of all alarm filter form.

Alarm Process Parameters

Alarm.<ClusterName>.<ServerName> ArchiveAfter	The archive after time (Event Journal) is the amount of time between each archive of Event Journal data.
Alarm.<ClusterName>.<ServerName> CacheSize	Defines the amount of memory (in megabytes) dedicated to the storage of event data.
Alarm.<ClusterName>.<ServerName> ClientConnectTimeout	Defines the amount of time, in milliseconds, in which the client can attempt to make a connection.
Alarm.<ClusterName>.<ServerName> ClientDisconnectTimeout	Defines the amount of time, in milliseconds, in which the client can attempt to terminate a connection to the server.
Alarm.<ClusterName>.<ServerName>ClientRequestTimeout	Defines the amount of time, in milliseconds, in which the client can request data from a server.
Alarm.<ClusterName>.<ServerName> FutureMessages	Event Journal records that have a time stamp with a date and time in the future can be stored historically.
Alarm.<ClusterName>.<ServerName> HeartbeatTimeout	Defines how long a server will wait before terminating a link that has been used for receiving heartbeat poll requests from its pair server, but is currently idle.

Alarm.<ClusterName>.<ServerName> KeepOnlineFor	The Event Journal Life is the amount of time for which the Alarm Server stores event messages on-line.
Alarm.<ClusterName>.<ServerName> MonitorConnectTimeout	Defines the amount of time, in seconds, that the server will wait for a monitor connection to occur.
Alarm.<ClusterName>.<ServerName> MonitorRequestTimeout	Defines the amount of time, in seconds, that the server will wait for a response from the other server in the pair.
Alarm.<ClusterName>.<ServerName>QueryCPUUsage	Defines the percentage of processor use you want to allocate to query searches.
Alarm.<ClusterName>.<ServerName> QueryRowLimit	Defines the maximum number of rows that can be returned in the result set for a single query.
Alarm.<ClusterName>.<ServerName>QueryTimeout	Defines the amount of time (in seconds) that is permitted for query searches.
Alarm.<ClusterName>.<ServerName> StreamSize	Defines the amount of data that is included in each event data file.
Alarm.<ClusterName>.<ServerName> SyncAllHistoricData	On multi-server systems, the Primary server and Standby server synchronize their data so that the Standby server contains an accurate, up to date backup of the Primary server's data.
Alarm.<ClusterName>.<ServerName>TransferConnectTimeout	Defines the amount of time, in seconds, that the Primary server will wait for a connection to occur.
Alarm.<ClusterName>.<ServerName> TransferInterleave	Controls how often the data synchronization is triggered by the Primary to the Standby Server.
Alarm.<ClusterName>.<ServerName>TransferInterval	Defines the number of seconds between each attempt to update the data on the Standby server.

BrowseTableView Parameters

[BrowseTableView]<BrowseType>.<ViewName>.ColWidths	Sets the column widths in pixels of the current data browse table.
[BrowseTableView]<BrowseType>.<ViewName>.Fields	Sets the field names of the columns in the current data browse table under the View Name configured on the page.

ClientParameters

[Client]PointCountRequired	Specifies what license point count a client requires.

General Parameters

[General]ClusterReplication	Controls whether tag will be replicated in a multi-cluster system.
[General]LicenseReservationTimeout	Specifies the number of seconds to reserve a license for a given IP address in cases where a remote client connection is lost.

Page Parameters

[Page]SOEPage	The name of the graphics page to display when you call up an sequence of events (SOE) page via the Cicode function PageSOE().

SQL Parameters

[SQL]MaxConnections	Defines the maximum number of DB connection objects.

Scheduler Parameters

[Scheduling]PersistPath	Directs where the configuration data for the scheduler is stored.
[Scheduling]StartDelay	Sets the delay from when the Scheduler's server components are initialized to the point when Scheduler begins processing active schedule entries.

Modified Parameters

Alarm Parameters

[Alarm]SavePrimary	This parameter is now used only to import alarm history from previous versions of Power SCADA Operation.
[Alarm]SaveSecondary	This parameter is now used only to import alarm history from previous versions of Power SCADA Operation .

[Alarm]SummaryLength	The maximum number of alarm summary entries that can be held in memory. The maximum number for this parameter has been modified from 4096000 to 100000.

Language Parameters

Language]LocalLanguage	Used to set the default language during start-up.

SQL Parameters

[SQL]QueryTimeout	Sets the timeout period for SQL queries globally.

Tab Style Template Parameters

[Format]FormatName	Define the display format by name.

Re-instated Parameters

None

Obsolete Parameters

[Alarm]Ack	Determined whether Power SCADA Operation acknowledges current alarms on startup.
[Alarm]AckHold	Determined whether alarms that have become inactive (and have been acknowledged) remain in the OFF ACKNOWLEDGED alarm list.
[Alarm]CacheLength	The maximum number of alarms that can be held in the cache of a client
[Alarm]FilterViewByPrivilege	If privilege is not checked, a user with no privilege (0) can browse and view trends and alarms that require privilege 1. The Power SCADA Operation behavior is the same as [Alarm]FilterViewByPrivilege = 0 in 7.20. The set of records returned from browse is now filtered by area.
[Alarm]SavePeriod	Set the path to the primary save file.
[Alarm]SaveStyle	Determines whether alarms records are identified by their record number or alarm tag.
[Alarm]StartTimeout	Sets the timeout period for loading data from the primary Alarms Server.

IIntl Parameters

[Intl]s1159	If a 12 hour clock is set (see [Intl]iTime), this parameter sets the format of the morning extension.
[Intl]s2359	If a 12 hour clock is set (see [Intl]iTime), this parameter sets the format of the evening extension.

Citect.ini parameters in 7.20

This topic lists the parameters that have been added or changed in version 7.20 of PowerSCADA Expert.

It includes:

- New parameters
- Modified parameters
- Re-installed parameters
- Obsolete parameters

New Parameters

The following parameters are new in version 7.20 . For an entire list of the system parameters, refer to the Parameters documentation.

Alarm Parameters

[Alarm.ClusterName.ServerName]DisableConnection	Specifies if a client will not connect to a server.
[Alarm.ClusterName.ServerName]Priority	Specifies the client priority for the server connection.
[Alarm]ReloadBackOffTime	Back-off time configured to control the pace of the reload on an alarm server.

Client Parameters

[Client]AutoLoginClearPassword	When set to 1 the cache is cleared of any client login credentials for consistency with the [Server]AutoLoginClearPassword ini parameter.
[Client]DisableDisplay	Sets whether to allow the client process to run in the background without a visible window.
[Client]EvictTimeout	Sets the amount of time a tag reference is cached before it is evicted.
[Client]PartOfTrustedNetwork	Tells a Client process to attempt to authenticate using the stored server password. It is automatically set by the Setup Wizard.
[Client] StalenessPeriod	Number of seconds to use for tag staleness period.
[Client]StalenessPeriodTolerance	Staleness period tolerance

CtAPI Parameters

[CtAPI]RoundToFormat	Indicates to the user if values rounded to format.

CtDraw.RSC Parameters

[CtDraw.RSC]AllowEditSuperGeniePage	When set enables the user to choose whether or not to open and edit a Super Genie page.

CtEdit Parameters

[CtEdit]CompileSuccessfulCommand	Indicates to the compiler an optional command, script or batch file to execute after a successful compile.
[CtEdit]CompileUnsuccessfulCommand	Indicates to the compiler an optional command, script or batch file to execute after an unsuccessful compile.
[CtEdit]Starter	Specifies the directory where the starter projects are located.

Debug Parameters

[Debug]ArchiveFiles	Archives log files once the size specified by [Debug]MaximumFileSize is reached.
[Debug]CategoryFilter	Allows you to filter logging messages by component category.
[Debug]CategoryFilterMode	Enables logging of categories declared by the [Debug]CategoryFilter value.
[Debug]EnableLogging	Enables or disables the logging mechanism.
[Debug]MaximumFileSize	Sets the maximum size for a log file.
[Debug]Priority	Allows you to filter logging messages according to their priority.
[Debug]SeverityFilter	Allows you to filter logging messages according to their severity.
[Debug]SeverityFilterMode	Enables logging of severities declared by the [Debug]SeverityFilter value.

General Parameters

[General]MiniumlUpdateRate	Specifies the time period (sec) at which a DataSource will send tag update value notifications to the subscription clients.
[General]StalenessPeriod	Specifies the time period (sec) after which a tag value is considered to be “stale” if it was not updated during this period.

IOServer Parameters

[IOServer]EnableEventQueue	Enables the event queue.
[IOServer]MaxEventsDrop	Sets the number of events that are dropped when too many are queued.
[IOServer]MaxEventsQueued	Sets the total number of events that can be queued.
[IOServer]MaxTimeInQueueMs	Sets the total time for which an event can be queued.

LAN Parameters

[LAN]AllowRemoteReload	Enables remote reloading of servers from a client.
[LAN]ClientRetryTime	Sets the length of time between connection attempts by a client.
[LAN]EarliestLegacyVersion	Specify the minimum legacy version from which the current version will accept connections.
[LAN]HighWaterMark	The number of messages waiting to be sent on a particular network connection at which the high water mark event will occur.
[LAN]KeepAliveInterval	Sets the length of time between two keep alive transmissions by the client.
[LAN]KeepAliveTime	Sets the length of time between two keep alive transmissions in idle conditions.
[LAN]ListenerRetryTime	Sets the length of time a server waits between attempts to listen for a client connection.
[LAN]LowWaterMark	After the high water mark has been reached on a particular network connection, the low water mark represents the number of messages waiting to be sent at which we will resume normal operations.
[LAN]NoSocketDelay	Switches off the delay on a socket caused by the use of the Nagle algorithm.
[LAN]ReadOnlyLegacyConnections	When set to 1 version 7.10 clients can only communicate in read-only mode.This parameter overrides 'EarliestLegacyVersion' .

Multi-Monitor Parameters (CSV Include project)

[MultiMonitor]DisableAutoStart	Disables the new multi-monitor functionality.

Page Parameters

[Page]AddDefaultMenu	Determines whether to add the default menu items to the tabbed menu bar.
[Page]BadDitheringColor	Sets the dithering color for graphics elements which are dithered if the value quality is “bad”.
[Page]BadDitheringDensity	Sets the dithering density for graphics elements which are dithered if the value quality is “bad”.
[Page]BadText	Text Objects can be displayed as #COM type errors, or as the text overlaid with a dithered pattern if the ‘display value’ expression has “bad” quality.

[Page]BadTextBackgroundColor	Sets the background color for numeric / text graphics objects to indicate “bad” quality.
[Page]EnableQualityToolTip	Set by default it controls the quality tooltip
[Page]ErrorDitheringColor	Sets the dithering color for graphics elements which are dithered if an internal error occurs.
[Page]ErrorDitheringDensity	Sets the dithering density for graphics elements which are dithered if an internal error occurs.
[Page]ErrorTextBackgroundColor	Sets the background color for numeric / text graphics objects to indicate an internal error.
[Page]IgnoreValueQuality	Defines the value quality handling by graphics pages.
[Page]OverrideDitheringColor	Sets the dithering color for graphics elements which are dithered if their values are override (“forced”).
[Page]OverrideDitheringDensity	Sets the dithering density for graphics elements which are dithered if an internal error occurs.
[Page]OverrideTextBackgroundColor	Sets the background color for numeric / text graphics objects to indicate that the value presented on the objects is override (“forced”).
[Page]ShowBadText	Text Objects can be displayed as #BAD text, or as the text overlaid with a dithered pattern if the "display value" expression has “bad” quality.
[Page]ShowErrorText	Text Objects can be displayed as #COM type errors, or as the text overlaid with a dithered pattern if the ‘display value’ expression has “uncertain” quality.
[Page]ShowUncertainText	Text Objects can be displayed as #UNC text, or as the text overlaid with a dithered pattern if the "display value" expression has “uncertain” quality.
[Page]Splash	Specify the name of the Splash Screen page.
[Page]SplashTimeout	Time in milliseconds for the Splash Screen to display.
[Page]SplashWinName	Specify the label of the Splash Window for use with the Cicode function WinNumber().
[Page]StartupDelay	Milliseconds between when Splash Screen and Start Screen are displayed.
[Page]StartupHeight	Height of the Start Page on main display monitor.

[Page]StartupMode	Mode of Start Page on main display monitor.
[Page]StartupWidth	Width of the Start Page on main display monitor.
[Page]StartupWinName	Specify the label of the Start Window for use with the Cicode function WinNumber().
[Page]StartupX	X coordinate of Start Page on main display monitor.
[Page]StartupY	Y coordinate of Start Page on main display monitor.
[Page]UncertainDitheringColor	Sets the dithering color for graphics elements which are dithered if the value quality is “uncertain”.
[Page]UncertainDitheringDensity	Sets the dithering density for graphics elements which are dithered if the value quality is “uncertain”.
[Page]UncertainText	Text Objects can be displayed as #COM type errors, or as the text overlaid with a dithered pattern if the ‘display value’ expression has “uncertain” quality.
[Page]UncertainTextBackgroundColor	Sets the background color for numeric / text graphics objects to indicate “uncertain” quality.
[Page]WaitForValidData	Specifies whether the animation system will attempt to wait for valid data from subscriptions necessary to draw a graphics page before it is animated.

Report Parameters

[Alarm.ClusterName.ServerName]DisableConnection	Specifies if a client will not connect to a server.
[Alarm.ClusterName.ServerName]Priority	Specifies the client priority for the server connection.

Runtime Manager Parameters

[RuntimeManager]AllowReload	Enables or disables the reload option in the Runtime Manager menu.

Security Parameters

[Security]DisableDEP	Set to turn off DEP protection for the Oower SCADA runtime.

Server Parameters

[Server]AutoLoginMode	Determines the auto login method the server will use when establishing connections to other servers.

Trend Parameters

[Trend]AcquisitionTimeout	Sets a timeout to stop a trend server infinitely acquiring a valid data sample from an I/O device.
[Trend.ClusterName.ServerName]DisableConnection	Specifies if a client should not connect to a server.
[Trend.ClusterName.ServerName]Priority	Specifies the client priority for the server connection.
[Trend]ReloadBackOffTime	Back-off time configured to control the pace of the reload on an Trend server.

Modified Parameters

CtEdit Parameters

[CtEdit]Copy	Supports runtime changes, it enables you to switch the SCADA node to use a new runtime configuration by pointing to a new location.

Re-instated Parameters

IOServer Parameters

[IOServer]BlockWrites	Determines whether Power SCADA Operation will try to block optimize writes to I/O devices.

Obsolete Parameters

AnmCursor Parameters

[IOServer]BlockWrites	Determines whether Power SCADA Operation will try to block optimize writes to I/O devices.

General Parameters

[General]TagAssMode	Validates the tag name in the Association Function. Refer to [General]TagDB instead.

LAN Parameters

[LAN]AllowLegacyConnections	Set to allow previous versions of client to connect to the server. Replaced with [LAN]EarliestLegacyVersion and the new trusted network authentication between SCADA servers. The Setup Wizard now allows a system password to be set on each server on your network.
[LAN]ServerLoginEnabled	Set to disable default server login. Replaced with [LAN]EarliestLegacyVersion and the new trusted network authentication between SCADA servers. The Setup Wizard now allows a system password to be set on each server on your network.

Page Parameters

[Page]BackgroundColour	Replaced with [Page]BackgroundColor. Specifies the color used to fill in the background when a page is smaller than the minimum width of a window.
[Page]ComBreak	Determines whether an error status is displayed on the screen if a communication error occurs. Replaced with new page quality settings such as [Page]IgnoreValueQuality, [Page]BadText, [Page]BadDitheringDenisty.
[Page]ComBreakText	Determines the display of text objects if a communication error occurs that affects the text. Replaced with new page quality settings such as [Page]IgnoreValueQuality, [Page]BadText, [Page]BadDitheringDenisty.
[Page]DynamicComBreakColour	Replaced with [Page]DynamicComBreakColor. Sets the color of the ComBreak dithering.
[Page]DynamicComBreakDensity	Sets the density of the ComBreak. Replaced with new page quality settings such as [Page]IgnoreValueQuality, [Page]BadText, [Page]BadDitheringDenisty.

Time Parameters

[Time]Deadband	The deadband time checked by the Time Server before it adjusts the time on the client(s).
[Time]Disable	Enables/disables the processing of time messages from the Time Server.
[Time]Name	Enables the time synchronization functionality.

[Time]PollTime	The period that the Time Server uses to synchronize other Power SCADA Operation computers on the network.
[Time]RTsync	Determines whether the Time Server will synchronize with the hardware clock.
[Time]Server	Determines whether this computer is a Time Server.

Trend Parameters

[Trend]CursorColour	Replaced with [Trend]CursorColor. Allows the cursor color to be specified.

Configuring Reference

The topics in this section contain detailed reference information that pertains to configuring Power SCADA Operation.

Use the links in the table to find the content you are looking for.

Topic	Content
"Citect INI Parameters" on page 577	A detailed listing of the Citect INI parameters you can use in your projects.
"Logic code definitions" on page 600	Design considerations and sample architectures for the Power SCADA Operation components
"Default Genie Library" on page 629	A detailed listing of the Power SCADA Operation PLS_* genie library and its naming conventions.
"Deadbands and ignored devices and topics" on page 637	Deadbands and ignored devices and topics let you limit information that you see in system queries and data acquisition from applications that use the Schneider Electric CoreServiceHost.
"Add engineering unit templates, units, and conversions" on page 638	Detailed information on how to set up and add engineering units and conversions.
"LiveView Tables" on page 646	Detailed information on the information contained in each LiveView table.
"Notifications Reference" on page 652	Detailed information on the notifications user interfaces (UIs)

Citect INI Parameters

There are a number of Citect INI parameters that you may use to configure driver parameters. These settings may be configured at the protocol level, cluster level, port level, or device level. More specific settings will override a general one. The order of precedence is:

Protocol Name > Cluster Name > Port Name > I/O Device Name

The level at which you want the INI settings to be in effect determines the name you define. For example:

To set the default timeout for all devices using the Micrologic protocol, use:

[MICROLOGIC]
Timeout = 2000
To override this default for cluster 'Cluster_1,' use:

[MICROLOGIC.Cluster_1]
Timeout = 1000
To override the default value for port 'Port_1' on cluster 'Cluster_1,' use:

[MICROLOGIC.Cluster_1.Port_1]
Timeout = 3000
To override the default value for I/O device 'CircuitBreaker_1' on port 'Port_1' on cluster 'Cluster_1,' use:

[MICROLOGIC.Cluster_1.Port_1.CircuitBreaker_1]
Timeout = 4000
Most settings can be configured to be specific to a particular I/O device. Exceptions are noted in the description for the individual parameter.

Parameters Database

All INI parameters described in the sections below can be set in the Parameters database. Using special syntax, you can access the parameters in the Project Editor (System < Parameters):

- The section name generally corresponds to the INI section name, although it includes the protocol name, cluster name, and primary device name only.
- The name is the INI value name.

If the parameter is set in the Parameters database, it becomes a new default for either protocol, cluster, or a concrete device (depending on the section name hierarchy).

Examples:

Section Name: [MICROLOGIC.Cluster_1.Breaker_1]
Name: Timeout
Value: 2000

This defines a new default timeout value for a redundant pair of MicroLogic devices (primary device is named Breaker_1 in Cluster_1).

Section Name: [PWRMODBUS.Cluster1]
Name: UseWriteMultiRegistersOnly
Value: 0

This sets UseWriteMultiRegistersOnly to 0 for all PWRMODBUS devices in Cluster 1.

The INI file is read after the parameter database is processed; thus the override options are set in the Parameters database.

In this section, you will find parameters organized into these categories:

"General Power SCADA Operation parameters" on page 579

"Performance Tuning Parameters" on page 585

"Waveform parameters" on page 593

"Sepam event reading parameters" on page 598

"MicroLogic modules configuration parameters" on page 597

"Data replication parameters" on page 595

General Power SCADA Operation parameters

The following parameters are common to all Power SCADA Operation devices.

watchtime

Controls how often the product will interrogate the driver to determine whether it is still online. This parameter can only be configured for an entire driver, and hence will have the driver dll name as its section name. Where another setting may be [PM870], to set this setting it must be [PLOGIC], as PLOGIC is the name of the dll. This is the only parameter whose section name is defined in this fashion.

Parameter type: secondsDefault Value: 2

Example: [SEPAM] watchtime = 5

kernelStatisticUpdateRate

Controls how frequently the statistics displayed in the driver kernel window are updated. This time period can be increased in order to decrease CPU load. This parameter can only be configured for the entire protocol (as with the watchtime parameter); it will have the driver dll name as its section name.

Parameter type: milliseconds
Default value: 5000

Examples:

[SEPAM40]
kernelStatisticUpdateRate = 20000

[SEPAM80]
kernelStatisticUpdateRate = 10000

UseWriteMultiRegistersOnly

Controls PWRMODBUS driver behavior when a single register is to be written. This parameter is set to 1 by default, enabling all writes to be made using "write multiple registers" MODBUS function. Setting this parameter to 0 allows driver to perform write using "write single register" function if (and only if) one MODBUS register is about to be written in current operation.

Parameter type: integer
Default value: 1

Examples:

[PWRMODBUS]
UseWriteMultiRegistersOnly = 1

[PWRMODBUS.MYCLUSTER.PORT_1.BCM1]
UseWriteMultiRegistersOnly = 0

timeout

Controls how long the driver waits for a response from a device before setting that device as offline. This value should be greater than the device/gateway timeout period. A timed out request will not be retried. The reason for this is that TCP is a guaranteed transport mechanism, and the lack of a response indicates that the device is offline or communication has been lost with that device. A device connected via a gateway should use the gateway's retry mechanism.

Parameter type: milliseconds
Default value: 5000

Examples:

[SEPAM40]
Timeout = 2000

[SEPAM40.MYCLUSTER.PORT_1.SLOW_SEPAM]
Timeout = 15000

retry

Defines the number of retry attempts for specific MODBUS requests. Retries may occur either when the request is timed out or certain MODBUS exception reply messages are received. The exact behavior is controlled by the RetryTimeout and RetryException parameters.

Parameter type: number of attempts
Default value: 3

Examples:

[SEPAM40]
retry = 1

[SEPAM40.MYCLUSTER.PORT_1.SEPAM_DEVICE]
retry = 5

RetryTimeout

When enabled (by default), the driver will re-try a timed-out MODBUS request.

Parameter type: long (Boolean)
Default value: 1

Examples:

[SEPAM40]
RetryTimeout = 1

[SEPAM40.MYCLUSTER.PORT_2.SEPAM_DEVICE]
RetryTimeout = 0

RetryException

When enabled (disabled by default), the driver will re-try a MODBUS request that has received MODBUS Exception messages. The number of retries is defined by the Retry parameter.

When Retry Exception is enabled, retry occurs when any of the following MODBUS exception messages is received:

- SLAVE_DEVICE_FAILURE_EXCEPTION = 0x5
- GATEWAY_PATH_UNAVAILABLE_EXCEPTION = 0xA
- GATEWAY_TARGET_DEVICE_FAILED_TO_RESPOND_EXCEPTION = 0xB
- SLAVE_DEVICE_BUSY_EXCEPTION = 0x6
- MEMORY_PARITY_ERROR_EXCEPTION = 0x8
- NEGATIVE_ACKNOWLEDGE_EXCPETION = 0x7

Parameter type: long (Boolean)
Default value: 0

Examples:

[SEPAM40]RetryTimeout = 1
RetryTimeout = 0

standbyRefreshRate

Controls how often a standby IO server attempts to poll a device to update its cache. This time period determines the maximum age that values may be when switching from a primary IO server to a standby. Decreasing this value degrades communications to the device.

Parameter type: seconds
Default value: 60

Examples:

[SEPAM40]
standbyRefreshRate = 30

[SEPAM40.MYCLUSTER.PORT_1.SLOW_SEPAM]
standbyRefreshRate = 120

standbyCheckTime

Controls how often the driver will inquire of Power SCADA Operation as to whether it is in standby or primary mode. This value can be increased to reduce CPU load.

Parameter type: milliseconds
Default value: 500

Examples:

[SEPAM40]
standbyCheckTime = 500

[SEPAM40.MYCLUSTER.PORT_1.SLOW_SEPAM]
standbyCheckTime = 1000

statusUnitCheckTime

This parameter defines how frequently the driver will try to re-establish the connection with a device that has gone offline on a port that is not disconnected. It sets the maximum rate at which the driver enquires of the device, to determine if it is still operational. If the "watchtime" parameter is set to a longer time, that value will be used instead.

If a network gateway has multiple devices connected to it, and one device is disconnected, the driver takes it offline and does not try to reconnect it according to this parameter's schedule. If the port is taken offline and then is reconnected, the driver will reconnect the devices immediately.

Parameter type: seconds
Default value: 5 (20 for MicroLogic)

Examples:

[SEPAM40]
statusUnitCheckTime = 5

[SEPAM40.MYCLUSTER.PORT_1.SLOW_SEPAM]
standbyCheckTime = 10

initUnitCheckTime

Controls how long the driver waits before attempting to bring a device online after it has gone offline. This value can be decreased to bring offline devices back into service in a shorter period of time. In a multi-drop scenario, this time should be relatively long, to prevent init unit requests from stalling communications to the rest of the devices on that port.

Parameter type: seconds
Default value: 120

Examples:

[SEPAM40]
initUnitCheckTime = 5

[SEPAM40.MYCLUSTER.PORT_1]
initUnitCheckTime = 120

initCacheTimeout

Controls how long the driver will spend attempting to populate the cache before bringing a device online. When a tag has been incorrectly configured, the device will come online after this period of time.

Parameter type: seconds
Default value: 60

Examples:

[SEPAM40]
initCacheTimeout = 60

[SEPAM40.MYCLUSTER.PORT_1.SLOW_SEPAM]
initCacheTimeout = 30

cacheRefreshTime

Controls the maximum rate at which the driver will attempt to repopulate its cache. If the driver cannot refresh its cache within the time period specified, it will collect data as fast as the network allows.

Parameter type: milliseconds
Default value: 500

Examples:

[SEPAM40]
cacheRefreshTime = 1000

[SEPAM40.MYCLUSTER.PORT_1.FAST_SEPAM]
cacheRefreshTime = 200

[SEPAM40.MYCLUSTER.PORT_1.UNIMPORTANT_DEVICE]
cacheRefreshTime = 5000

TimeSync

Enables/disables time synchronization for the PM5000S driver. On startup and on a 15-minute schedule, the driver reads each device clock. If a device clock is not within the specified 10-second drift, the driver sets the time on that device to the current system time.

Parameter type: Boolean
Default value: 0 (PM5000S) or 1 (PM5000S1)

This is a driver-level parameter, not a protocol-level parameter. All entries must be under the PM5000S section of the .ini file. By default, the PM5000S1 protocol enables time sync. For the PM5000S, it is disabled by default because most devices will have battery backup and GPS time sync availability.

Example:

[PM5000S] TimeSync = 1

StatusRegister

Defines a holding register that the driver reads to determine whether a device is responding to communication requests. The result of this read is not important, however it must be a valid register address within the device.

Parameter type: register address
Default value: 1100 (2 for Sepam) (PM1200 requires that this value be set to 3911)

Examples:

[PWRMODBUS]
statusRegister = 1000

[PWRMODBUS.MYCLUSTER.PORT_DEVICE_PM1200]
statusRegister = 3911

StatusRegistersCount

Defines the number of registers that the driver reads to determine whether a device is responding to communication requests. The result of this read is not important, however it must be a valid register address within the device.

Parameter type: number of registers
Default value: 1 (PM1200 requires that this value be set to 2)

Examples:

[PWRMODBUS]
statusRegistersCount = 2

[PWRMODBUS.MYCLUSTER.PORT_DEVICE_PM1200]
statusRegistersCount = 2

StatusRegisterType

Used together with StatusRegister; defines the type of the status register. Can only be configured for the PWRMODBUS driver. This parameter can have one of the following values:

- 0 - HOLDING register (default)
- 1 - INPUT register
- 2 - COIL register
- 3 - DIGITAL input (input coil) register

Any other value equals the default.

Parameter type: register type
Default value: 0

Example:

[PWRMODBUS]
statusRegister = 1000

[PWRMODBUS.MYCLUSTER.PORT_1.DEVICE_A]
statusRegister = 16000
statusRegisterType = 2

ModbusBase

Defines the base address for a device. Some MODBUS device registers are defined using a base address of 1. in this case, reading register 100 would actually require reading register 99. In other devices (such as the Sepam) the base address is 0. This parameter allows the base address to be configured according to the device.

Parameter type: integer
Default value: 0 for Sepam; 1 for all other drivers

Examples:

[PWRMODBUS]
ModbusBase = 1

[PWRMODBUS.MYCLUSTER.PORT_1.DEVICE_A]
ModbusBase = 0

RegMode

Specifies the order of bytes in a device register. It can only bet set for PWRMODBUS driver, and is supposed to be unit-specific. Value values are:

	RegMode	Order of bytes
Big endian (default)	0	1 0
Little endian	1	0 1

Any other value reverts to big endian.

Parameter type: integer
Default value: 0

Examples:

[PWRMODBUS]
RrMode = 0 # Default

[PWRMODBUS.MYCLUSTER.PORT_1.DEVICE_A]
RegMode = 1 # This device has little endian registers

timeZone

Time zone names are taken directly from the Windows registry database (case-insensitive), and will otherwise default to using the I/O server's local time zone. The Windows time zone database is located in the Windows registry in:

HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows NT\CurrentVersion\Time Zones\.

The examples of time zone names are:

- AUS Central Standard Time
- China Standard Time
- Pacific Standard Time (Mexico)

Use the general section [POWERLOGICCORE] to specify the time zone for all devices. For example:

[POWERLOGICCORE]
Timezone = Mountain Standard Time

This sets the default time zone for all devices (Sepam, PLogic, Micrologic, PWRMODBUS). Otherwise the time zone can be specified for each device with precedence taken as described in the start of this section.

Examples:

[PLOGIC870.Cluster1.Singapore_Port]
Timezone = Singapore Standard Time

[PLOGIC870.Aus_Cluster]
Timezone = Aus Central Standard Time

Not having a time zone specification means that the device is in the same time zone as the machine where the I/O Server is running. No time conversion will be done.

Performance Tuning Parameters

Several parameters are provided to allow tuning of the performance. These parameters fall into three broad categories; bandwidth allocation, packet blocking optimization, and tag scan rates.

Bandwidth Allocation Parameters

Bandwidth can be allocated for the different types of data as desired. The parameters to perform this are as follows:

[Parameter]	[Default Value]	[Parameter Type]
EventBandwidth	25	integer
WaveformsBandwidth	12	integer
CommandsBandwidth	13	integer
RealTimeBandwidth	50	integer

The percentage bandwidth allocated to each queue will be the ratio of an individual queue's value when compared to the total sum of defined bandwidths. The default values have a sum of 100 for ease of reference. Any unused bandwidth will be shared amongst the other categories.

Bandwidth can be configured at the port level, but not the device level.

Example:

[SEPAM40]
EventsBandwidth 30
WaveformsBandwidth 5
CommandsBandwidth 15
RealTimeBandwidth 50

[SEPAM40.MYCLUSTER.PORT_1]
EventsBandwidth 50
WaveformsBandwidth 30
CommandsBandwidth 10
RealTimeBandwidth 10

BandwidthAllocation

This parameter allows the ratio of bandwidth assigned to each device sharing a port to be configured. This parameter can only be configured at the device level.

Parameter type: integer
Default value: <equal split>

Example:

[SEPAM40.MYCLUSTER.PORT_1.DEVICE_A]
BandwidthAllocation 70

[SEPAM40.MYCLUSTER.PORT_1.DEVICE_B]
BandwidthAllocation 30

Packet Blocking Optimization Parameters

For all devices except the Sepam, parameters can be configured to optimize the MODBUS packets that are created for collection of data from the device. Sepam devices have pre-configured blocks that are already optimized.

The parameters that control the blocking are as follows:

enableScatteredReads

This causes the driver to use the ‘scattered read’ extension that can help improve blocking. This option should be enabled for devices that support this extension.

Parameter type: Boolean flag
Default value: 0 for generic Power MODBUS driver, 1 for PowerLogic driver

Example:

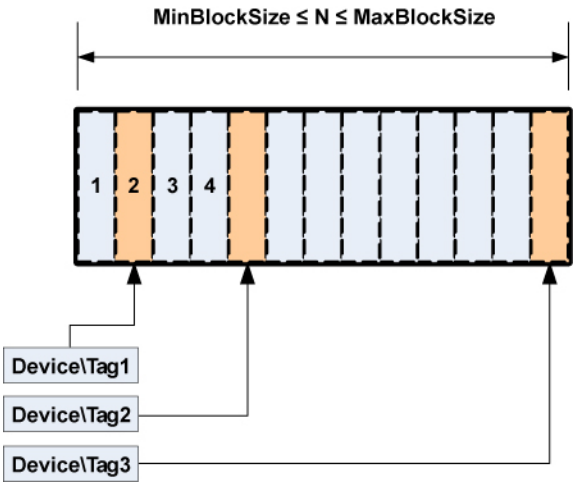
[PWRMODBUS.MYCLUSTER.PORT_1.DEVICE_A]
enableScatteredReads 1

[PWRMODBUS.MYCLUSTER.PORT_1.DEVICE_B]
enableScatteredReads 0

percentBlockFill

This parameter defines the maximum percentage of configured registers contained in a block before the drivers creates fixed blocks instead of scattered blocks. The following figure illustrates now a block of N registers can be constructed:

- If M<N registers are configured, the block builder can either create a scattered block or a multi-register block.
- If M/N*100% is less than PercentBlockFill, the block builder creates a scattered registers block.
- If the percentage of configured registers ≥ PercentBlockFill, the block builder creates a multi-register block.



Parameter type: percentage
Default value: 50

Example:

[PM870.MYCLUSTER.PORT_1.PM_DEVICE]
percentBlockFill 50

[CM4000.MYCLUSTER.PORT_1.CM_DEVICE]
percentBlockFill 80

maxBlockSize

This parameter defines the maximum number of registers that can be read in a single request. By default, this is 124, but some devices can read more than this.

Parameter type: integer
Default value: 124

Example:

[PWRMODBUS.MYCLUSTER.PORT_1.DEVICE_A]
maxBlockSize 1024

minBlockSize

This parameter defines the minimum number of registers to read as a fixed block before the block builder will instead add those registers to a scattered block. If latency is low, and scattered reads are expensive, this value should be lower. If latency is high, or scattered reads are inexpensive, it is better to set this value higher. Only applicable when scattered reads are enabled.

Parameter type: integer

Default value: 20

Example:

[PM870.MYCLUSTER.PORT_1.LOW_LATENCY_DEVICE]
minBlockSize 10

[CM4000.MYCLUSTER.PORT_1.HIGH_LATENCY_DEVICE]
minBlockSize 100

Tag Scan Rate Parameters

Each tag can be configured at a priority level from 1-3 where 1 is the highest. Parameters exist to adjust the relative scan rates of the high and low priority tags in comparison to the nominal tag scan rate.

HighScanRate

Parameter type: percent relative to nominal

Default value: 50

LowScanRate

Parameter type: percent relative to nominal
Default value: 200

Using the default parameters, the high priority tags will be refreshed twice as fast as the normal priority tags, and the low priority tags will be refreshed at half the rate of the normal priority tags. These parameters can be configured at the port level and higher.

Using the default settings and a nominal tag refresh rate of 1 second:

Low Priority Tag Refresh: 2000 ms
Normal Priority Tag Refresh: 1000 ms
High Priority Tag Refresh: 500 ms

Example:

[PM870.MYCLUSTER.PORT_1]]
HighScanRate 25
LowScanRate 500T

Advanced Tag Block Capabilities (Invalid Memory Access Blocks defined)

Some devices may restrict access to certain memory registers. Such registers may be available for read only, write only or may not be available at all, resulting in a MODBUS exception when the registers are addressed.

Definition: Blocks of registers that cannot be read or written to are referred as “invalid memory access blocks.”

These devices create a challenge for the PWRMODBUS driver. If the device has invalid blocks that do not support scattered reads (or they are disabled for this device), the driver may try to read registers in blocks that intersect with the registers that cannot be read. This can result in the whole block being invalidated and, in certain cases, may also result in device being taken offline. Figure 1 (below) illustrates an invalid block in the middle of an address space.

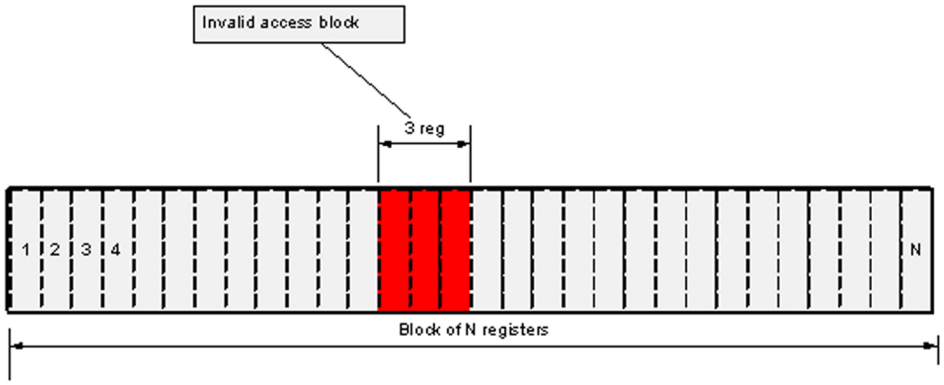
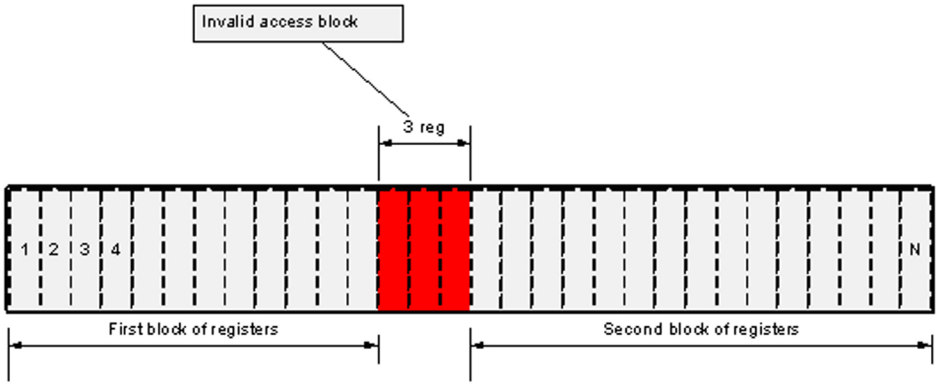


Figure 1 shows how the "invalid memory access block" affects MODBUS register blocking. In this situation, if the driver does not know that the block of 3 inaccessible registers exists, it will try to block all registers from 1 to N (depending on data that was requested by the real-time data collector). This block, however, will never be read successfully, as the device will respond with an exception to all attempts to read invalid registers.

If the configuration includes information about invalid memory access blocks, the driver will create two blocks instead of one, as shown in Figure 2:



In Figure 2, invalid registers were taken into account when the block was constructed. When configuring device that has invalid memory areas, it is especially important to define all blocks that may interfere with any of the tags.

Invalid Block Tag Definition Syntax

Invalid access memory areas are defined as variable tags, using the following format for the address:

```
T:IB;{m|i|c|s}:<start_register>:u<count>;E:1;L:P:0
```

where

- `m, i, c` and `s` define the type of MODBUS register
- `<start_register>` is the first register address of the invalid access block
- `<count>` defines the number of registers in the invalid access block

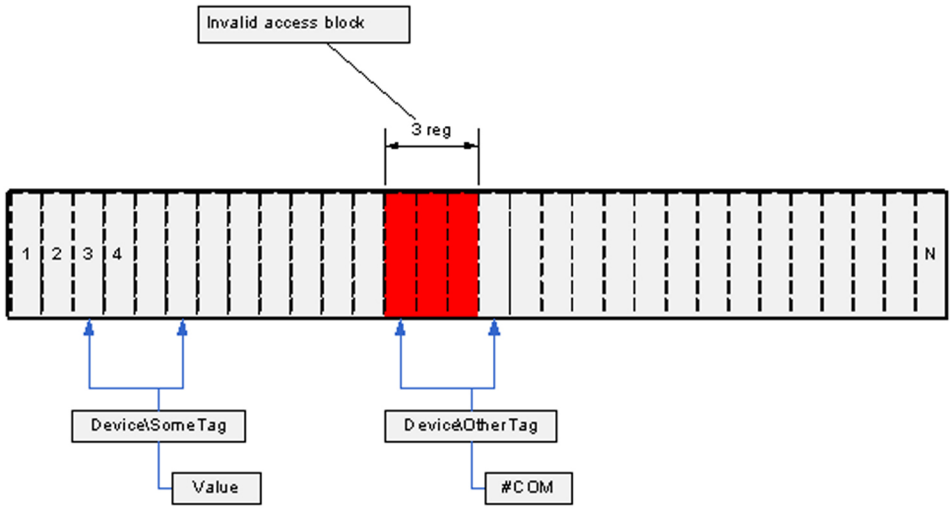
Example:

```
T:IB;m:300:u10;E:1;L:P:0
```

This defines an invalid access block of ten holding registers starting from register 300.

Configuration Notes

When one or more invalid access blocks is defined according to the syntax above, tags configured to read any of invalid registers will be affected by it. If any of the tag registers fall into an invalid memory access block, this tag will not be readable; any attempt to read its value will result in #COM, as shown on Figure 3:



However, such tags do not affect other tags, because the PWRMODBUS driver implements algorithms that prevent tags from being invalidated by invalid memory block logic.

Tags that try to use invalid registers are detected on startup and can be found by analyzing the log file. This is an example trace:

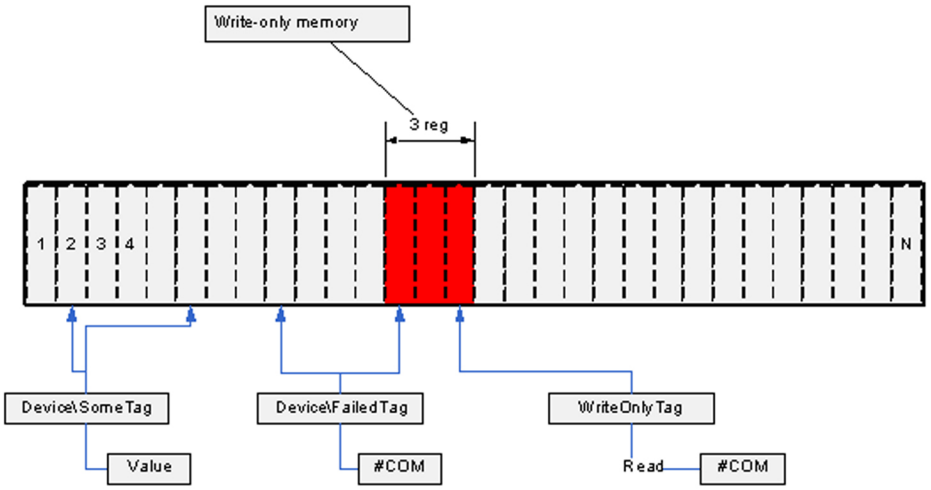
```
[DEBUG] [REAL][GeneralDriver::BaseDatapointBuilder::BuildDataPoints()]
 Adding datapoint. Tag - BCM1\H_QIVR34\SwlStr Address -
T:SS;m:283:2;E:1;L:P:26 Datapoint: class Datapoints::Status_SS
```

```
[DEBUG] [REAL][RealTimeData::DeviceCache::Subscribe()] Init Registers:
Polled Registers: Address:283 Type:3

[ERROR] [MISC] [RealTimeData::BlockBuilder::AddDataPoint()] Cannot add
datapoint, one or more invalid memory addresses fall into non-
splittable block

[ERROR] [MISC] [GeneralDriver::BaseDatapointBuilder::BuildDataPoints
()] Could not init datapoint. Tag BCM1\H_QIVR34\SwlStr Address
T:SS;m:283:2;E:1;L:P:26. Analyze other messages, this tag address may
contain invalid registers
```

Such output is expected when a holding register with address 283 is declared invalid. This trace helps figure out any configuration issues.

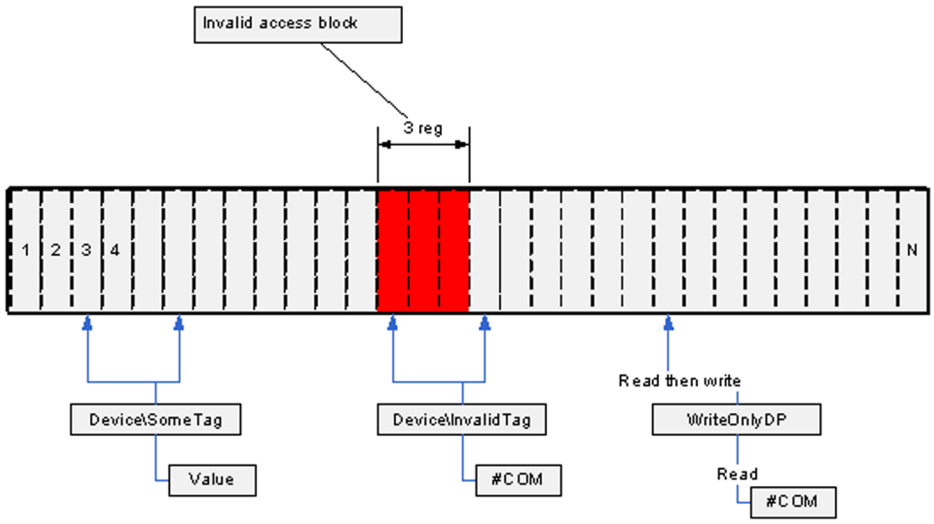
Write-only Memory

If a certain register range is accessible for write only, no additional configuration may be needed. However, to prevent the driver block optimizer from including these registers in a poll, they also must be configured by adding “invalid block” tags as described in the *Invalid Block Tag Definition Syntax*, described above. Declaring these registers invalid will not prevent drivers from trying to write to them. Figure 4 illustrates a write-only memory configuration:



Write-only registers should not be confused with write-only datapoints that internally read a register before attempting to write. Declaring the register they read invalid will result in a datapoint not working; such mistake should be avoided. Figure 4 shows “WriteOnlyDP” as an example; this tag cannot be read (it will result in #COM), but internally it needs to get the register value before writing into it. If this register was declared invalid, tag writes would also not succeed.

Figure 5 illustrates a write-only datapoint:



Tag Blocking Notes

The drivers support an advanced blocking mechanism for tags. That is, real-time tags are no longer blocked together with write-only tags.

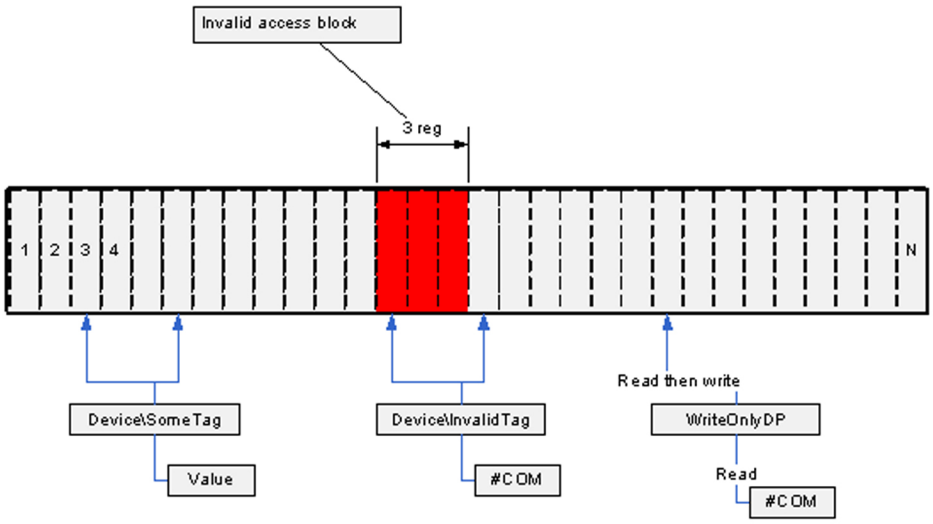
Tags found invalid, due to intersection with invalid memory areas, are not blocked with “good” real-time tags and will not therefore interfere with them.

Write-only Tags

Beginning with driver version 2.0.1.1, the write-only tags feature is fully supported.

There are no special logic codes or address formats for write-only tags. If a tag references memory that was declared invalid (see *Invalid block tag definition syntax*, above), and its datapoint has writing capabilities, the tag becomes write only. No preliminary checks are performed to verify that the memory can be written to, and no additional configuration is needed. It is assumed that, if the tag is configured to write into memory that has been declared “invalid,” the memory can actually be written to.

It is important to understand that scaled write tags (code 110) will become write-only tags, if that scale register can be read. Tag Device\TagN on Figure 6 explains this case: the datapoint needs to read the scale value from the scale register in order to write scaled value to write-only register. However, as long as the actual register belongs to the memory that can only be written to (and it is configured using T:IB tag syntax as explained in *Invalid Block Tag Definition Syntax* above), this tag cannot be read.



The fact that the tag mentioned before cannot be read will not affect other tags reads (see *Tag Blocking Notes*, above).

NOTE: The write-through feature of the device cache is disabled for write-only tags.

Security Parameters

Use the following security parameters to add system security.

EnterPasswordForControl

This parameter controls whether users must enter a password when they control a breaker. Regardless of whether the user is logged in, a setting of 1 (true) will require a password when the user initiates breaker control. When set to 0 (false), the password check is removed. In this case, no user will be required to enter a password to control a breaker.

Parameter type: integer
Default: 1 (true)

Waveform parameters

The following parameters configure the waveform downloading behavior. These parameters are only applicable for Sepam devices and PowerLogic devices that support waveforms.

[Parameter] [Default Value] [Parameter Type]
WaveformsDisable 0 Boolean value
WaveformMatchMargin 10 seconds
WaveformCheckTime 30 seconds (PM/CM)
WaveformZone 1 integer (Sepam)

WaveformsDisable

This parameter enables or disables waveform downloading for a particular device.

Parameter type: Boolean value
Default value: 0

Example:

[SEPAM40.MYCLUSTER.PORT_1.DEVICE_A]
WaveformsDisable 1 //Disable waveform downloading

NOTE: This INI setting is a global setting that sets the default at startup. You can set this for any set of devices (such as clusters, individual devices)

There is also a tag that will change an individual device's setting at runtime (it will reset to the default when you restart the project). This tag is LLNO\WaveformCollectionEnabled. 1 = True, 0 = False.

WaveformMatchMargin

Alarms are matched to waveforms by the timestamp of each. This parameter is the maximum difference between alarm timestamp and waveform timestamp for the product to consider it a match.

Parameter type: seconds
Default value: 10

Example:

[SEPAM40]
WaveformMatchMargin 2

WaveformCheckTime (PM, CM, and Sepam)

This parameter defines the time the driver will wait between checking for new waveforms.

Parameter type: seconds
Default value: 30

Example:

[SEPAM40.MYCLUSTER.PORT_1.DEVICE_A]
WaveformCheckTime 60 (checks every 60 seconds)

WaveformZone (Sepam)

This parameter defines the Sepam waveform zone that the Sepam driver will use to collect waveforms from the device. This allows two masters to extract waveforms from the same device. Valid values are 1 or 2.

Parameter type: integer
Default value: 1

Example:

[SEPAM40.MYCLUSTER.PORT_1.DEVICE_A]
WaveformZone 2

Alarm Parameters

The following parameters are used for alarms.

UsePLSFilter

Controls whether alarm/event filtering is done by the PLSCADA filter form or the Citect filter form. Both forms cause the same information to display on the page, but each is presented in a different format.

Parameter type: integer
Default value: 1 (PLSCADA filter form)

Example:

[ALARM] UsePLSFilter = 1

Data replication parameters

These parameters are used to configure the data directory paths of your servers. These settings are server wide, and must be added to the 'WaveformDB' area of the INI file.

Database root folder path

Waveform databases for all units will locate on the file system under the same common folder. The path to the root folder will be specified in the citect.ini file:

[WaveformDB]
LocalRoot = c:\path\to\the\database\root

This path must be specified as local path.

By default, the Power SCADA Operation[DATA] directory is be used as database root folder.

Database root UNC path

For waveform files to be accessible by the remote clients, the database root folder must be available as network shared folder. The UNC name of this folder must be specified in the INI file

[WaveformDB]
UNCPath = \\computerName\shareToTheLocalRootAbove

If the UNC path to the database root is not specified, all waveform file names returned by the library will be local file names for the I/O server, making viewing the waveforms on the remote clients impossible.

Replication destination configuration

In redundant scenario, the replication target folder must be specified for replication to work

[WaveformDB]
ReplicationDestinationRoot=\\OtherMachine\share\path

The destination path is the name of the network share on the redundant machine where its waveform database root is located. It must also allow write access.

No default value for it is assumed.

If not set or share is not accessible, no replication will be preformed.

Graphics library parameters

Maximum number of entries that can be held in Event Log

The Alarm Summary length parameter in Citect.ini defines the maximum number of entries that can be held in the Event Log (default = 5000 entries). You can view all events in the Event Log and alarms in the alarm logs (Alarm Log, Unacknowledged Alarms, Disabled Alarms).

Each event requires 256 bytes of memory, plus the length of the comment. 32,000 entries will require at least 8 MB of memory. If you have many events, you should ensure that there is enough memory to store them in RAM.

After the parameter number is reached, older events are FIFO'd out to storage in [Installed Project Directory]\Schneider Electric\9.0\Logs

Parameters for Alarm and Event States

[Alarm]
UseConfigLimits = 1
CacheLength = 2500
!Sound1 = <wave file name>
!Sound1Interval = <repeating interval in milliseconds>
!Sound2 = <wave file name>
!Sound2Interval = <repeating interval in milliseconds>
!Sound3 = <wave file name>
!Sound3Interval = <repeating interval in milliseconds>

[AlarmFormat]
EventLog=OnDate | Date, OnTimeMS | Time, Custom1 | Equipment, Name | Description, SumState | State | Custom2 | Location, UserName | User

[AlarmStateText]
ON=<default text for ACTIVE state>
OFF=<default text for INACTIVE state>
ACK=<default text for ACKNOWLEDGED state>
ENA=<default text for ENABLE state>
DIS=<default text for DISABLE state>
CLE=<default text for CLEAR state>

These parameters are read only when the system starts up. The user must restart Power SCADA Operation if they change these parameters.

- If you do not specify any value for these parameters, these default values will be used, in this order:
- Appearance
- Disappearance
- Acknowledge
- Enable
- Disable
- Clear

[General] IODevCheckStartupDelay
Delay time before the I/O server starts checking for I/O device status at start-up. The delay allows time for the I/O devices to come online. Otherwise, the I/O server would have triggered alarms to indicate that communication was not successful for the relevant equipment.

Allowed Values: ≥ 0
Default Value: 0

[General] IODevCheckInterval
The time interval in seconds that the I/O server repeats the I/O device status check.

Allowed Values: ≥ 2
Default Value: 2

Integration parameters

The following parameters deal with single sign-on and integration of Power SCADA Operation and Power Monitoring Expert.

[SSO]PSEHostName

If you do not have Power Monitoring Expert installed, and you want Power SCADA Operation reports, use this parameter. This parameter specifies the IP address for Power SCADA Operation.

Default: localhost

[SSO]HostName

This parameter specifies the IP address for Power Monitoring Expert.

Default: localhost

[SSO]RemoteCallHandlerServer

This parameter specifies the I/O server that will execute the call from a web client.

Default:N/A

[SSO]RemoteCallHandlerCluster

This parameter specifies the cluster of the I/O server that will execute the call from a Web client.

Default:N/A

MicroLogic modules configuration parameters

A MicroLogic unit consists of three or four modules, each acting as a separate MODBUS device; however the I/O server views MicroLogic as one I/O device. The communication control module (CCM) is optional for MicroLogic; its presence may be detected by the driver or specified in the INI file.

IFE/IFM

This parameter specifies whether the Micrologic device is connected through an IFE/IFM, or through the CCM (cradle comms module) or a Modbus Gateway.

0 - connection is through a Modbus Gateway

1 - connection is through an IFE/IFM

MicrologicType

This parameter, which indicates the Micrologic Type, enables/disables functionality that can increase system performance.

1 - Type A: Only the Circuit Breaker Manager (BCM) alarm file is read.

2 - Type E: Only the Circuit Breaker Manager (BCM) file is read.

3 - Type P: The Circuit Breaker Manager (BCM) and Protection Manager (PM) alarm files are read.

4 - Type H: The Circuit Breaker Manager (BCM), Protection manager (PM), and Metering Manager (MM) alarm files are read. Waveform files are also read.

CCM

The CCM parameter specifies whether a CCM is present on the device or if the driver should try to detect its presence ("auto mode"). Valid values are:

CCM not present - 0
CCM present - 1
Auto mode - 2 (default)

Any other value reverts to auto mode.

Parameter type: integer
Default value: 2

Example:

[Micrologic.MYCLUSTER.PORT_1.DEVICE_A]
CCM=1

Module-Specific Packet Blocking Optimization Settings

Due to different firmware versions, MicroLogic modules may require different blocking settings. This is especially true when MicroLogic contains a BCM that supports MODBUS "read multiple registers" requests for up to 124 registers, and an MM or a PM module that supports 21 register reads at max. The MicroLogic driver allows blocking optimization parameters to be overridden for each of the device's modules, as in the following example:

[Micrologic.MYCLUSTER.PORT_1.DEVICE_A]
maxBlockSize = 124

[Micrologic.MYCLUSTER.PORT_1.DEVICE_A.BCM]
maxBlockSize = 21

The parameter set for the device applies to all of its modules unless overridden in a module-specific section (e.g., [Micrologic.MYCLUSTER.PORT_1.DEVICE_A.BCM])

These parameters can be overridden:

- enableScatteredReads
- minBlockSize
- maxBlockSize
- PercentBlockFill

This applies to the BCM, CCM, MM, and PM modules.

MicrologicV INI Settings

The MicrologicV device driver includes these additional INI settings:

- Level3: This is the level 3 device password (4 digits), used by the driver when executing commands.
- Level4: This is the level 4 device password (4 digits), used by the driver when executing commands.

If you do not supply this parameter, the driver uses the default device passwords.

Sepam event reading parameters

EventTable

This parameter defines the Sepam event table that the Sepam driver uses to collect alarms from a device. This allows two masters to extract alarms from the same device. Valid values are 1 or 2.

Parameter type: integer
Default value: 1

Example:

[SEPAM40.MYCLUSTER.PORTO_1.DEVICE_A]
EventTable 2

EventIdle

This parameter defines the time that the driver will wait before requesting the next event from a Sepam device. It may be possible to reduce this value to increase the rate at which alarms can be retrieved from the device.

Parameter type: milliseconds
Default value: 500

Example:

[SEPAM40.MYCLUSTER.PORTO_1.DEVICE_A]
EventIdle 200

Sepam device driver INI configuration settings

Sepam devices support 2 event buffers, which enables 2 concurrent masters to read events. For all Sepam devices, the first buffer starts at register 0x40, and the second starts at register 0x70. By default, the first buffer is used; however, in certain configurations, there may be a need to tell the driver to use the second buffer. This can be done by adding the following section to citect.ini (see "Customize a project using Cicode" on page 371):

[Sepam]
[Parameter] [Default Value] [Parameter Type]
EventTable 1 //Valid values are 1 and 2.

Value 2 tells the driver to use event buffer starting at 0x70; any other value falls back to 0x40.

If the installation uses any other software—such as SMS, CET, or ION—the setting in that application should be buffer 2.

[Parameter] [Default Value] [Parameter Type]
EventIdle 500 Integer

'EventIdle' is the time the driver will wait before requesting the next event from the Sepam device. It may be possible to reduce this value to increase the rate at which alarms can be retrieved from the device.

Example.

[SEPAM40.MYCLUSTER.PORT_1.DEVICE_A]
EventIdle 200

See "Edit tag addresses" on page 190 for information about PowerLogic device driver addresses.

PLC Parameters

The following parameters are added to support device types as they are added to the system.

Quantum PLC time-stamped events

The PWRMODBUS driver supports Quantum time-stamped events. You must set the following INI parameter to enable time-stamped alarms downloading:

[PWRMODBUS]

TSEventsEnabled = 1
0 by default, valid values 1 or 0

TSMailboxAddress = 1104
1104 by default

TSAddrLost = 705
705 by default

Logic code definitions

The following table lists each logic code with its related information.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
READS						
Invalid Block (L:P:0)	IB	LONG	Up to 1,000 sequential registers	No	Generic only	Defines invalid blocks of memory in the device. The driver does not include these registers in block reads.
Date / Time (L:P:1) (3 register)	UT	LONG	3 sequential registers	No	Generic – if it fits	Register N: High byte = Month 1–12 Low byte = Day 1–31 Register N+1: High byte = Year 0–199 (+1900) Low byte = Hour 0–23 Register N+2: High byte = minutes 0–59 Low byte = seconds 0–59

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Date / Time (L:P:2) (6 register)	UT	LONG	6 sequential registers	No	Generic - if it fits	Register N: Seconds 0–59 Register N+1: Minutes 0–59 Register N+2: Hours 0–23 Register N+3: Day 1–31 Register N+4: Month 1–12 Register N+5: Year 0–199 (+1900)
Date / Time (L:P:3) (3 or 4 register -- Circuit Monitor/ Power Meter)	UT	LONG	3 or 4 sequential registers	No	CM/PM	Register N: High byte = Month 1–12, Low byte = Day 1–31 Register N+1: High byte = Year 0–199 (+1900) Low byte = Hour 0–23 Register N+2: High byte = minutes 0–59 Low byte = seconds 0–59 Register N+3: msec = 0–999 (unused)

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Date / Time (L:P:4) (3 or 4 registers SEPAM)	UT	LONG	3 or 4 sequential registers	No	SEPAM	Register N: Bits 0–6 = Year: 0 –70 (2000– 2070) 71 – 99 (1971–1999) Register N+1: Bits 8-11 = Month Bits 0-4 = Day Register N+2: Bits 8-12 = Hour Bits 0-5 = Minutes Register N+3: msec = 0-59,999 (seconds are ms/1000)
Date/Time (L:P:5) 3-register Micrologic	UT	LONG	3 sequential registers	No	micro	Register N: High byte = Month 1–12, Low byte = Day 1–31 Register N+1: High byte = Year 0–69 (+2000), Year 70–99 (+1900) Low byte = Hour 0–23 Register N+2: High byte = minutes 0–59 Low byte = seconds 0–59

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Date/Time (L:P:6) 4-register Micrologic	UT	LONG	4 sequential registers	No	micro	Register N: High byte = Month 1–12, Low byte = Day 1–31 Register N+1: High byte = Year 0–69 (+2000), Year 70–99 (+1900) Low byte = Hour 0–23 Register N+2: High byte = minutes 0–59 Low byte = seconds 0–59 Register N+3: msec = 0–999 (unused)
Date/Time (L:P:7) 3-register Argos	UT	LONG	3 sequential registers	No	Argos	The number of seconds since 01/01/2000 (00:00:00) register 1 = MSB register 2 = LSB register 3 = milliseconds

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Date/Time (L:P:8) 4-register IEC 870-5-4	UT	LONG	4 sequential registers	No	generic	Register N: Bits 0–6 = Year: 0 –127 (2000– 2127) Register N+1: Bits 8-11 = Month Bits 0-4 = Day Register N+2: Bits 8-12 = Hour Bits 0-5 = Minutes Register N+3: msec = 0-59,999 (seconds are ms/1000)
Modulo 10k (L:P:10)	BC	STRING	Up to 4 registers	No	generic	Result is a string representation. Range is 0 to 9,999,999,999,999,999 Each register has a range of 0 to 9,999 Result is: – R4*10,000^3 + R3*10,000^2 + R2*10,000 + R1

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Modulo 10k Val (L:P:11)	BC	REAL	Up to 4 registers	No	generic	Result is a string representation. Range is 0 to 9,999,999,999,999,999 Each register has a range of 0 to 9,999 Result is: – R4*10,000^3 + R3*10,000^2 + R2*10,000 + R1 NOTE: This logic code (and all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.
Modulo 10k Energy (L:P:12)	BC	STRING	Up to 4 registers	No	generic	Result is a string representation. Range is 0 to 9,999,999,999,999.9 Each register has a range of 0 to 9,999 Result is – (R4*10,000^3 + R3*10,000^2 + R2*10,000 + R1)/1000

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Modulo 10k Energy Val (L:P:13)	BC	REAL	Up to 4 registers	No	generic	Result is a string representation. Range is 0 to 9,999,999,999,999.9 Each register has a range of 0 to 9,999 Result is – (R4*10,000^3 + R3*10,000^2 + R2*10,000 + R1)/1000 NOTE: This logic code (and all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.
PL Digital Input SS (L:P:20)	SS	LONG	2 registers	No	CM/PM	First register (100–199 inclusive) indicates that this is a digital input register. Second register is masked to test for either one 1 or one 0. Result is: 0 = off and 1 = on. This result can be inverted.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
PL Digital Input DS (L:P:21)	DS	LONG	2 registers	No	CM/PM	Same as PL Digital Input SS except: Result is 0 = intermediate, 1 = off, 2 = on, 3 = bad-state. Inversion will invert only off and on states.
PL Digital Input TF (L:P:22)	SS	DIGITAL	2 registers	No	CM/PM	Same as PL Digital Input SS except: Result is: 0 = false and 1 = true. This result can be inverted.
PL Digital Output SS (L:P:23)	SS	LONG	2 registers	No	CM/PM	First register (200–299 inclusive) indicates that this is a digital output register. Second register is masked to test for either one 1 or one 0. Result is: 0 = off and 1 = on. This result can be inverted.
PL Digital Output DS (L:P:24)	DS	LONG	2 registers	No	CM/PM	Same as PL Digital Output SS, except: Result is: 0 = intermediate, 1 = off, 2 = on, 3 = bad-state. Inversion will invert only off and on states.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
PL Digital Output TF (L:P:25)	SS	DIGITAL	2 registers	No	CM/PM	Same as PL Digital Output SS except: Result is: 0 = false and 1 = true. This result can be inverted.
Status SS (L:P:26)	SS	LONG	Up to 4 registers	No	Generic	Each register is compared to a ones' mask. Optionally it can be compared to a zeros' mask. (Use the Edit Address screen in the Profile Editor to create masks for the user.) Result is: 0 = off and 1 = on. If there is only one register, the result can be inverted.
Status OR SS (L:P:226)	SS	LONG	2 to 4 registers	No	Generic	Each register is compared to a ones' mask. These results are OR'ed together. Optionally, it can be compared to a zeros' mask. (Use the Edit Address screen in the Profile Editor to create masks for the user.) Result is: 0 = off and 1 = on.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Status DS (L:P:27)	DS	LONG	Up to 4 registers	No	Generic	Same as Status SS except: Result is: 0 = intermediate, 1 = off, 2 = on, 3 = bad-state. Inversion will invert only off and on states.
Status OR DS (L:P:227)	DS	LONG	2 to 4 registers	No	Generic	Same as Status OR SS except: Result is: 0 = intermediate, 1 = off, 2 = on, 3 = bad-state.
Status TF (L:P:28)	SS	DIGITAL	Up to 4 registers	No	Generic	Same as Status SS except: Result is: 0 = false and 1 = true. This result can be inverted.
Status OR TF (L:P:228)	SS	DIGITAL	2 to 4 registers	No	Generic	Same as Status OR SS except: Result is: 0 = false and 1 = true.
Status Int (L:P:29)	BC	LONG	1 register	No	CM/PM	One register is bit anded with one mask. The result will be an integer that can be used to choose the appropriate enumeration.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Status Enumeration (L:P:229)	EN	LONG	1 to 4 registers	No	Generic	Each register is compared to a ones' mask. Optionally it can be compared to a zeros' mask. (Use the Edit Address screen in the Profile Editor to create masks for the user.) Result is a combination of the results for each register, using this formula: result for register 1 * 2^0 + result for register 2 * 2^1 + result for register 3 * 2^2 + result for register 4 * 2^3

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
PL Analog Input (L:P:30)	MV/CM	REAL	3 registers	No	CM/PM	First register (300–399 inclusive) indicates that this is a analog input register. Second register is treated as a signed value. Third register can contain a value from –3 to 3 and will be used to scale the second register (R2*10^R3). NOTE: This logic code (and all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.
Breaker Rack Status (L:P:230)	EN	LONG	2 to 3 registers	No	Generic	Register 1 = breaker racked in Register 2 = breaker racked out Register 3 = breaker in test (optional) Results: 0 = racked in 1= racked out 2 = test 3 = error 4= in between positions

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
PL Analog Output (L:P:31)	MV/CM	REAL	2 registers	No	CM/PM	First register (400–499 inclusive) indicates that this is a analog output register. Second register is treated as a signed value. NOTE: This logic code (and all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Scaled Register Signed (L:P:32)	MV/CM	REAL	1 or 2 registers	Either (optional)	Generic	For a single register: treated as a signed value from –32,767 to +32,767. (-32768 will result in a NA) For two registers: the registers will be concatenated together, the first register filling bits 16–32 and the second register filling bits 0–15. Values will range from –2,147,483,648 to –2,147,483,647. Values can be scaled using a fixed scale or a scale register. NOTE: This logic code (and all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Scaled Register Unsigned (L:P:33)	MV/CM	REAL	1 to 4 registers	Either (optional)	Generic	For a single register: treated as an unsigned value from 0 to 65,535. For two registers: the registers will be concatenated together, the first register filling bits 16–32 and the second register filling bits 0–15. Values will range from 0 to 4,294,967,295. Values can be scaled using a fixed scale or a scale register. NOTE: This logic code (and all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.
Scaled Register Signed No NA (L:P:34)	MV/CM	REAL	1 or 2 registers	Either (optional)	Generic	Same as Scaled Register except that a single register with value -32768 is acceptable and will be reported as such. NOTE: This logic code (and all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Scaled Register Signed SEPAM A (L:P:35)	MV/CM	REAL	2 registers	Either (optional)	Generic	Same as Scaled Register except that 0xFFFFFFFF or 0x00007FFF will be NA. NOTE: This logic code (and all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.
Scaled Register Signed SEPAM B (L:P:36)	MV/CM	REAL	2 registers	Either (optional)	Generic	Same as Scaled Register except that 0xFFFFFFFF will be NA. NOTE: This logic code (and all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.
IEEE 32 Real (L:P:37)	MV/CM	REAL	2 sequential registers	No	Generic	Uses the IEEE standard for floating-point arithmetic (IEEE 754); register 1 is MSB, register 2 is LSB NOTE: This logic code (and all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Scaled Register Signed SEPAM 2000 Format B (L:P:38)	MV/CM	REAL	1 register	Either (optional)	Generic	For a single register: treated as a signed value from –32,767 to +32,767: From the value of the unsigned register, subtract 32768; then apply the scale. 0000 or FFFF will be NA. NOTE: This logic code (and all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.
PL String (L:P:39)	ST	STRING	1 to 10 sequential registers	No	Generic	Each register can represent up to two ASCII characters.
Sum Registers (L:P:40)	MV/CM	REAL	1 to 4 registers	Either (required)	Generic	Result is: R1 + … + Rn * 10^scale NOTE: This logic code (and all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Divide Registers (L:P:41)	MV/CM	REAL	3 registers	Either (required)	Generic	Result is: R1/R2 * R3 * 10^scale If R2 is zero, result will be #COM NOTE: This logic code (and all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.
Multiply Registers (L:P:42)	MV/CM	REAL	1 to 4 registers	Either (required)	Generic	Result is: R1 * … * Rn * 10^scale NOTE: This logic code (and all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.
Average Registers (L:P:43)	MV/CM	REAL	1 to 4 registers	Either (required)	Generic	Result is: Avg(R1 … Rn) * 10^scale NOTE: This logic code (and all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Average Registers WF (L:P:44)	MV/CM	REAL	2 to 4 registers	Either (required)	Generic	Result is: Avg(R1 … Rn-1) * Rn * 10^scale NOTE: This logic code (as with all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.
Sum with Scale (L:P:45)	MV/CM	REAL	2 registers	Either (required)	CM/PM	Result is: (R1 * 10^scale) + R2 NOTE: This logic code (as with all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.
Sum with Scale Unsigned (L:P:46)	MV/CM	REAL	2 registers	Either (required)	CM/PM	Result is same as above, except unsigned. NOTE: This logic code (as with all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Vector Math (L:P:47)	MV/CM	REAL	2 registers	Either (required)	Generic	Result is: sqrt (R1^2 + R2^2) x scale NOTE: This logic code (as with all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.
Vector Math IEEE (L:P:48)	MV/CM	REAL	4 registers	Either (required)	Generic	Result is: sqrt ([R1 R2]^2 + [R3 R4]^2) x scale where [ ] indicates IEEE32 representation NOTE: This logic code (as with all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Multiply Registers 32-bit (L:P:49)	MV/CM	REAL	3 or 4 registers	Either (optional)	Generic	Result is: [R1R2] * [R3(R4)], meaning Regs 1 and 2 are a 32 bit number. The number is multiplied by Reg 3 (if 16 bit) or Reg 3 and 4 (32 bit number) NOTE: This logic code (and all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.
CM4 Power Factor IEEE (L:P:50)	MV/CM	REAL	1 register	No	CM4	Returns the IEEE power factor.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
PM8 Power Factor IEEE (L:P:51)	MV/CM	REAL	1 register	No	PM8	Returns the IEEE power factor (converted from IEC mode as necessary). The device may be in IEEE or IEC mode if the device firmware version is 11.6 or higher. If the device firmware version is below 11.6, IEC mode is not supported. NOTE: This logic code (as with all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.
SP40 Power Factor IEEE (L:P:52)	MV/CM	REAL	1 register	No	SEPAM 40	Returns the IEEE power factor (converted from IEC mode). NOTE: This logic code (as with all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
ML Power Factor IEEE (L:P:53)	MV/CM	REAL	2 registers	No	ML	Returns the IEEE power factor (converted from IEC mode as necessary). The second input register must be the associated Reactive Power for the Power Factor requested. NOTE: This logic code (as with all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.
Generic Power Factor (L:P:54)	MV/CM	REAL	2 registers	No	Generic	R2/sqrt (R2^2 + R1^2) where: R2 = real power R1 = reactive power NOTE: This logic code (as with all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Generic Power Factor - IEEE32 variation (L:P:55)	MV/CM	READ	4 registers	No	Generic	[R3 R4]/ sqrt ([R3 R4]^2 + [R1 R2]^2) where: R3 = real power IEEE32 MSR R4 = real power IEEE32 LSR R1 = reactive power IEEE32 MSR R2 = reactive power IEEE32 LSR
SP2000 Power Factor IEEE (L:P:56)	MV/CM	REAL	1 register	No	SEPAM 2000	Returns the IEEE power factor (converted from IEC mode). NOTE: This logic code (as with all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.
Scaled Register Signed 64-bit (L:P:57)	MV/CM	REAL	4 registers	Either (optional)	Generic	Reads a 64-bit signed integer and returns a REAL value. NOTE: This logic code (as with all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Power Factor IEEE (L:P:58)	MV/CM	REAL	2 registers	No	Generic	Takes a 4 quadrant power factor (IEEE32 real) and returns an IEEE power factor
IEEE 64-bit double (L:P:59)	MV/CM	REAL	4 registers	No	Generic	Uses the IEEE standard for floating-point arithmetic (IEEE 754); returns the value as 32-bit REAL. NOTE: This logic code (as with all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.
IEEE 64-bit double (L:P:60)	MV/CM	STRING	4 registers	No	Generic	Uses the IEEE standard for floating-point arithmetic (IEEE 754); returns the value as 64-bit STRING. NOTE: This logic code (as with all REAL logic codes) has an accuracy of 15 digits. Anything longer than 15 digits should not be considered accurate.
WRITES (these are write-only; see below for Read/Write codes) NOTE: If the device is capable of preventing (blocking) writes to its registers, verify that the "block" feature is disabled before you implement the write.						

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Status Write Register (L:P:101)	SS	LONG	1 register	No	Generic	If you input 1 to this tag it will write the MASK value to the register.
Status Write Register AND (L:P:102)	SS	LONG	1 register	No	Generic	If you input 1 to this tag it will read the register and AND the MASK with the register (This puts a 0 wherever there is a 1 in the mask and leaves the rest alone).
Status Write Register OR (L:P:103)	SS	LONG	1 register	No	Generic	If you input 1 to this tag it will read the register and OR the MASK with the register (This puts a 1 wherever there is a 1 in the mask and leaves the rest alone).
Write Register Unsigned (L:P:110)	MV/CM	REAL	1 register	Either	Generic	This will take the input value read in and divide out the scale factor and the conversion factor. It will then round to the nearest whole number and if it is a value from 0 to 65535 it will put this value in the register. NOTE: This logic code (and all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Write Register Signed (L:P:111)	MV/CM	REAL	1 register	Either	Generic	This will take the input value read in and divide out the scale factor and the conversion factor. It will then round to the nearest whole number and convert the signed value to an unsigned value from 0 to 65535. It will put this value in the register. NOTE: This logic code (and all REAL logic codes) has an accuracy of seven digits. Anything longer than seven digits should not be considered accurate.
READ/WRITES						
Read/Write Holding Register (L:P:120)	MV/CM	LONG	1 register	No	Generic	You can write any value from 0 to 65535 and read an unsigned value from the same register.
Read/Write Coil Register (L:P:121)	SS	DIGITAL	1 register	No	Generic	You can write 0 or 1 and read a value from the same register.
READ						

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Command Read Date/Time (L:P:170)	CR	LONG	2 to 4 registers	No	MicrologicX	Register 1: <Command ID>:<Module> Register 2: <Register>:<# of registers> Register 3: <# of parameters>:<Parameter 1> Register 4: <Parameter 2>:<Parameter 3> If there are no parameters needed, omit registers 3 and 4. All registers formatted as <Decimal>:<Hexadecimal>
Command Read IEEE32 (L:P:171)	CR	REAL	2 to 4 registers	Fixed	MicrologicX	Register 1: <Command ID>:<Module> Register 2: <Register>:<# of registers> Register 3: <# of parameters>:<Parameter 1> Register 4: <Parameter 2>:<Parameter 3> If there are no parameters needed, omit registers 3 and 4. All registers formatted as <Decimal>:<Hexadecimal>

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
Command Read Scaled Signed (L:P:172)	CR	REAL	2 to 4 registers	Fixed	MicrologicX	Register 1: <Command ID>:<Module> Register 2: <Register>:<# of registers> Register 3: <# of parameters>:<Parameter 1> Register 4: <Parameter 2>:<Parameter 3> If there are no parameters needed, omit registers 3 and 4. All registers formatted as <Decimal>:<Hexadecimal>
Command Read Scaled Unsigned (L:P:173)	CR	REAL	2 to 4 registers	Fixed	MicrologicX	Register 1: <Command ID>:<Module> Register 2: <Register>:<# of registers> Register 3: <# of parameters>:<Parameter 1> Register 4: <Parameter 2>:<Parameter 3> If there are no parameters needed, omit registers 3 and 4. All registers formatted as <Decimal>:<Hexadecimal>

Default Genie Library

The genie library includes a number of general genies for objects such as motors and pumps. There are also genies that are specific to Power SCADA Operation. These genies use a particular naming convention, which is described in the table below. In the Power SCADA Operation library, each genie name begins with “pls,” and is followed by a description of the type of genie according to this table:

first	second	third	fourth
pls indicates Power SCADA Operation library	alarm = alarm	base = primitive genies	1 = small
	ansi = ANSI style	cb = circuit breaker	2 = large
	display = equipment details	sw = switchgear	
	gen = generic	cmd = control genies	
	iec = IEC style	eq = equipment (devices)	
	style = navigation		

Additionally, the actual genies have abbreviated names. When you highlight a genie name, the abbreviation displays above the top row of genie icons.

The following tables list and define the individual genies in each of the Power SCADA Operation libraries.

PLS_ALARM

This library includes genies that provide functionality to alarm displays.

Genie Abbreviation	Description
Field	data portion of an alarm row
Row	a genie with a collection of fields
Selector	for column resizing
Setpoint	a setpoint row

PLS_ANSI_BASE_1 / PLS_ANSI_BASE_2

These libraries include a variety of base symbols used to created genies for ANSI equipment.

1 = small size
2 = large size

Genie Abbreviation	Description
sl_battery_gen	single-cell battery
sl_battery_multi	multiple-cell battery
sl_capacitor	capacitor

Genie Abbreviation	Description
sl_capacitor_vari	variable capacitor
SL_Closed_HV	closed circuit breaker position
sl_closed_knife	closed knife switch
sl_closed_lv	closed low-voltage circuit breaker
sl_conductive_path_1	conductive path 1
sl_conductive_path_2	conductive path 2
sl_conductive_path_3	conductive path 3
sl_conductor_junction	conductor junction
sl_contact_nc_closed	contact break, closed
sl_contact_nc_open	contact break, open
sl_contact_no_closed	contact make, closed
sl_contact_no_open	contact make, open
sl_contact_term	contact terminal
sl_ct	current transformer
sl_in_cb_rack	incoming, circuit breaker racked out, plug
SL_In_Rack	incoming, circuit breaker racked out, socket
sl_in_sw_head	incoming switch head
sl_inductor	inductor
sl_inductor_adjust	adjustable inductor
sl_inductor_gen	general inductor
sl_inductor_magcore	magnetic core inductor
sl_inductor_vari	variable inductor
sl_open	open symbol
sl_open_fuse_sw	open isolating fuse-switch
SL_Open_HV	open high-voltage circuit breaker
sl_open_knife	open knife-type switch
SL_Open_LV	open low-voltage circuit breaker
sl_out_cb_rack	outgoing, circuit breaker racked out, plug
SL_Out_Rack	outgoing, circuit breaker racked out, socket
sl_pb_break	push-button, break
sl_pb_make	push-button, make
sl_pb_term	push-button, terminal
sl_pt	potential transformer
sl_relay	relay
sl_resistor	resistor
sl_resistor_adjust	adjustable resistor
sl_resistor_vari	variable resistor
sl_separable_con_closed	separable connector, closed
sl_separable_con_open	separable connector, open

Genie Abbreviation	Description
sl_separable_con_plug	separable connector plug
sl_separable_con_socket	separable connector socket

PLS_ANSI_CB_1 / PLS_ANSI_CB_2

These libraries include genies for ANSI-type high-voltage and low-voltage drawout circuit breakers.

1 = small size
2 = large size

Additional definitions:

bus	=	busway
cb	=	circuit breaker
hv	=	high voltage
lv	=	low voltage
dr	=	drawout
nd	=	non-drawout
fd	=	earth at bottom (feeder)
inc	=	earth at top (incomer)
nes	=	no earth
nc	=	not remote control
c	=	remote control

Genie Abbreviation	Description
hv_cb_bus_dr_c	high-voltage drawout circuit breaker, remote control
hv_cb_bus_dr_nc	high-voltage drawout circuit breaker, not remote
hv_cb_bus_nd_c	high-voltage non-drawout circuit breaker, not remote
hv_cb_bus_nd_nc	high-voltage non-drawout circuit breaker, not remote
hv_cb_fd_dr_c	high-voltage, drawout circuit breaker, remote control, with earth at bottom
hv_cb_fd_dr_nc	high-voltage, drawout circuit breaker, no remote control, with earth at bottom
hv_cb_fd_nd_c	high-voltage, non-drawout circuit breaker, remote control, with earth at bottom
hv_cb_fd_nd_nc	high-voltage, non-drawout circuit breaker, no remote control, with earth at bottom
hv_cb_inc_dr_c	high-voltage, drawout circuit breaker, remote control, with earth at top

Genie Abbreviation	Description
hv_cb_inc_dr_nc	high-voltage, drawout circuit breaker, no remote control, with earth at top
hv_cb_inc_nd_c	high-voltage, non-drawout circuit breaker, remote control, with earth at top
hv_cb_inc_nd_nc	high-voltage, non-drawout circuit breaker, no remote control, with earth at top
hv_cb_nes_dr_c	high-voltage drawout circuit breaker, remote control, no earth
hv_cb_nes_dr_nc	high-voltage drawout circuit breaker, no remote control, no earth
hv_cb_nes_nd_c	high-voltage non-drawout circuit breaker, remote control, no earth
hv_cb_nes_nd_nc	high-voltage non-drawout circuit breaker, no remote control, no earth
lv_cb_bus_dr_c	low-voltage drawout circuit breaker, remote control, busbar-type with earth at bottom
lv_cb_bus_dr_nc	low-voltage drawout circuit breaker, no remote control, busbar-type with earth at bottom
lv_cb_bus_nd_c	low-voltage non-drawout circuit breaker, remote control, busbar-type with earth at bottom
lv_cb_bus_nd_nc	low-voltage non-drawout circuit breaker, no remote control, busbar-type with earth at bottom
lv_cb_fd_dr_c	low-voltage drawout circuit breaker, remote control, earth on load side (bottom of drawing)
lv_cb_fd_dr_nc	low-voltage drawout circuit breaker, no remote control, earth on load side (bottom of drawing)
lv_cb_fd_nd_c	low-voltage non-drawout circuit breaker, remote control, earth on load side (bottom of drawing)
lv_cb_fd_nd_nc	low-voltage non-drawout circuit breaker, no remote control, earth on load side (bottom of drawing)
lv_cb_inc_dr_c	low-voltage drawout circuit breaker, remote control, earth on feeder (top of drawing)
lv_cb_inc_dr_nc	low-voltage drawout circuit breaker, no remote control, earth on feeder (top of drawing)
lv_cb_inc_nd_c	low-voltage non-drawout circuit breaker, remote control, earth on feeder (top of drawing)
lv_cb_inc_nd_nc	low-voltage non-drawout circuit breaker, no remote control, earth on feeder (top of drawing)
lv_cb_nes_dr_c	low voltage drawout circuit breaker, no earth, remote control
lv_cb_nes_dr_nc	low voltage drawout circuit breaker, no earth, no remote control
lv_cb_nes_nd_c	low voltage non-drawout circuit breaker, no earth, remote control
lb_cb_nes_nd_nc	low voltage non-drawout circuit breaker, no earth, no remote control

PLS_ANSI_SW_1 / PLS_ANSI_SW_2

These libraries include ANSI-style switches:

1 = small size
2 = large size

Genie Abbreviation	Description
sw_fused	switch: feeder, fused
sw_fused_isolated	switch: feeder, fused, isolated
sw_general	switch: feeder, general
sw_knife	switch: knife type

PLS_DISPLAY

This library includes two genies that provide data row items for equipment.

Genie Abbreviation	Description
equiplistitem	data row for the equipment tag list
EquipValueItem	data row for the equipment popup

PLS_GEN_BASE_1 / PLS_GEN_BASE_2

These libraries include a variety of "parts" related to generators, motors, and transformers.

Genie Abbreviation	Description
chassis_ground	chassis ground
Dev_Base	device base
es_inc	earth switch, incomer
es_out	earth switch, feeder
Gen_1	generator, option 1
Gen_2	generator, option 2
gen_AC	generator: AC
gen_DC	generator: DC
genset	engine-generator
ground	ground
Motor_1	motor, option 1
Motor_2	motor, option 2
motor_ac	motor: AC
motor_dc	motor: DC
motor_synch	motor: synchronous
SL_Base	circuit breaker base symbol
sl_br_in	circuit breaker line in, non-drawout
sl_br_out	circuit breaker line out, non-drawout
SL_Bustie	bus tie
SL_CommLoss	comms loss
SL_Discrepancy	position discrepancy
sl_harmonic_filter_1	harmonic filter 1
sl_harmonic_filter_2	harmonic filter 2

Genie Abbreviation	Description
SL_In	incoming bus
SL_Local	local, rather than remote control
SL_Out	feeder
SL_Tripped	tripped
Test_CB_Control	health test for the circuit breaker control
transformer_1_in	transformer 1: general, on-line
transformer_1_in_y	transformer 1: star (wye), on-line
transformer_1_out	transformer 1: general, off-line
transformer_1_out_d	transformer 1: delta, off-line
transformer_1_out_y	transformer 1: star (wye), off-line
transformer_2_in	transformer 2: general, on-line
transformer_2_in_Y	transformer 2: star (wye), on-line
transformer_2_out	transformer 2: general, off-line
transformer_2_out_D	transformer 2: delta, off-line
transformer_2_out_Y	transformer 2: star (wye), off-line (no 2 IN D? or 1 IN D?)

PLS_GEN_CMD_1 / PLS_GEN_CMD_2

These libraries include genies that control display of popups and values:

1 = small size
2 = large size

Genie Abbreviation	Description
CmdDetail	provides access to the equipment detail popup
cmddetail_meter	provides access to the meter detail popup
Control	control in a circuit breaker
value	value section of a circuit breaker
value_meter	value section of a meter

PLS_GEN_EQ_1 / PLS_GEN_EQ_2

These libraries include the general equipment used to make up generators, motors, and transformers:

1 = small size
2 = large size

Genie Abbreviation	Description
busbar_horz	horizontal busbar
busbar_vert	vertical busbar
gen_ac	generator: AC
gen_dc	generator: DC

Genie Abbreviation	Description
gen_nd_1	generator 1: no current designation
gen_nd_2	generator 2: no current designation
mot_ac	motor: AC
mot_dc	motor: DC
mot_nd_1	motor 1: no current designation
mot_nd_2	motor 2: no current designation
mot_syn	motor, synchronous
trans_nd_1	transformer 1: no connection designation
trans_nd_2	transformer 2: no connection designation
trans_sd_1	transformer 1: star-delta (wye-delta)
trans_sd_2	transformer 2: star delta (wye-delta)
trans_ss_1	transformer 1: star-star (wye-wye)
trans_ss_2	transformer 2: star-star (wye-wye)

PLS_IEC_BASE_1 / PLS_IEC_BASE_2

These libraries include a variety of symbols for IEC equipment:

1 = small size

2 = large size

Genie Abbreviation	Description
sl_cap_bank_ tuned_3	capacitor bank 3: tuned
sl_cap_bank_ tuned_4	capacitor bank 4: tuned
sl_capacitor	capacitor
sl_capacitor_vari	capacitor, variable
sl_closed	closed switch
sl_contact_nc	contact break
sl_ct	contact
sl_fuse_1	fuse, option 1
sl_fuse_2	fuse, option 2
SL_Head	head
sl_head_2	head
sl_in_cb_rack	incoming, circuit breaker when racked out, plug
SL_In_Rack	incoming, circuit breaker when racked out, socket
sl_in_sw_hd_isol	incoming, switch head, isolated
sl_in_sw_head	incoming, switch head
sl_inductor	inductor
sl_inductor_adjust	inductor, adjustable
SL_Open	open

Genie Abbreviation	Description
sl_out_cb_rack	feeder, circuit breaker when racked out, plug
SL_Out_Rack	feeder, circuit breaker when racked out, socket
sl_resistor	resistor
sl_resistor_adjust	resistor with adjustable contact
sl_resistor_vari	resistor, variable
sl_sw_static_1	static switch 1
sl_sw_static_2	static switch 2

PLS_IEC_CB_1 / PLS_IEC_CB_2

These libraries include high-voltage drawout circuit breakers:

1 = small size
2 = large size

Genie Abbreviation	Description
hv_cb_bus_dr_c	high-voltage drawout circuit breaker, remote control
hv_cb_bus_dr_nc	high-voltage drawout circuit breaker, no remote control
hv_cb_bus_nd_c	high-voltage non-drawout circuit breaker, remote control
hv_cb_bus_nd_nc	high-voltage non-drawout circuit breaker, no remote control
hv_cb_fd_dr_c	high-voltage, drawout circuit breaker, remote control, earth at bottom
hv_cb_fd_dr_nc	high-voltage, drawout circuit breaker, no remote control, earth at bottom
hv_cb_fd_nd_c	high-voltage, non-drawout circuit breaker, remote control, earth at bottom
hv_cb_fd_nd_nc	high-voltage, non-drawout circuit breaker, no remote control, earth at bottom
hv_cb_inc_dr_c	high-voltage, drawout circuit breaker, remote control, earth at top
hv_cb_inc_dr_nc	high-voltage, drawout circuit breaker, no remote control, earth at top
hv_cb_inc_nd_c	high-voltage, non-drawout circuit breaker, remote control, earth at top
hv_cb_inc_nd_nc	high-voltage, non-drawout circuit breaker, no remote control, earth at top
hv_cb_nes_dr_c	high-voltage drawout circuit breaker, remote control, no earth

Genie Abbreviation	Description
hv_cb_nes_dr_nc	high-voltage drawout circuit breaker, no remote control, no earth
hv_cb_nes_nd_c	high-voltage non-drawout circuit breaker, remote control, no earth
hv_cb_nes_nd_nc	high-voltage non-drawout circuit breaker, no remote control, no earth

PLS_IEC_SW_1 / PLS_IEC_SW_2

These libraries include IEC-style switches:

1 = small size
2 = large size

Genie Abbreviation	Description
sw_general	general switch
sw_isolated	isolated switch

PLS_METER

This library includes meter symbols.

Genie Abbreviation	Description
circuit monitor	Power SCADA Operation circuit monitor
egx	Power SCADA Operation EGX
generic_meter	generic meter
ion_7650	ION 7650 meter
micrologic	all Mircologic meters
power_meter	Power SCADA Operation power meter
quantum	Power SCADA Operation Quantum
sepam	all Sepam metersI

ITEM1

This library includes miscellaneous symbols.

Genie Abbreviation	Description
Item1	value type and units block for a circuit breaker
Item2	value type and units block for a circuit breaker
tab1	menu tab
Tab2	menu tab

Deadbands and ignored devices and topics

The following settings apply to applications that use the Schneider Electric CoreServiceHost: EcoStruxure Web Services and ETL.

The two features described allow you to limit information that you see in system queries and data acquisition. You set the limits for these features in the `Configuration.xml` file (C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\9.0\Applications\AppServices\bin\Configuration.xml).

Deadbands

```
<ConfigurationItem Key="Deadbands" Category="Platform Mapping" Application="CitectPlatform">
<Value />
</ConfigurationItem>
```

Use this line in `Configuration.xml` to reduce the sensitivity to minor changes in real-time data. You can set default deadbands for variable tags. To set a deadband, enter the following in the value field:

```
<Value>XX|NN;</Value>
```

where XX is the IEC 61850 tag name and NN is the percentage.

For example, to set Current A to 5% and Current B to 10%, you would enter the following:

```
<Value>mmxu1\A\phsA|5.0;mmxu1\A\phsB|10.0;</Value>
```

Ignored Devices/Ignored Topics

Use these two lines in the *Configuration.xml* to develop a list of devices and topics that you want to ignore in system queries/data acquisition. Typically, you will use this to exclude devices such as the memory device zOL. Ignored devices and topics will not appear in Reporting or LiveView. (EWS, ETL)

To set a value for ignored devices, type the Citect device names (semi-colon delimited) that you want to ignore.

For example, to exclude zOL (the one-line memory device) and the network tags device (for monitoring comms loss), type:

```
<Value>zOL;NetworkTagsDev</Value>
```

In the Ignored Topics list, type the topic names (semi-colon delimited) that you want to ignore. Do not include the device name prefix that displays in the Citect project tag names. For example, to exclude AlarmUnhandled and AlarmInvalidTimestamp, type:

```
<Value>AlarmUnhandled;AlarmInvalidTimestamp</Value>
```

Save your changes.

Add engineering unit templates, units, and conversions

An *engineering unit* is a part of a tag. Use engineering unit templates to simplify the conversion between base units and their conversions (such as inches to centimeters), and to provide consistency in recording data in reports and on-screen viewing. For example, in one project you

might want to see amperes reported as kiloamps. In another, you might want to see amperes as milliamps. You will use the Units screens to determine the conversion for standard units and custom units (tied to custom tags) that you create.

You can also create templates to organize user-created unit/conversion pairs. Each template will include all of the predefined engineering units and conversions, as well as the ones you assign to it. These templates can then be used in system projects (see the Set Up Project tab for creating projects).

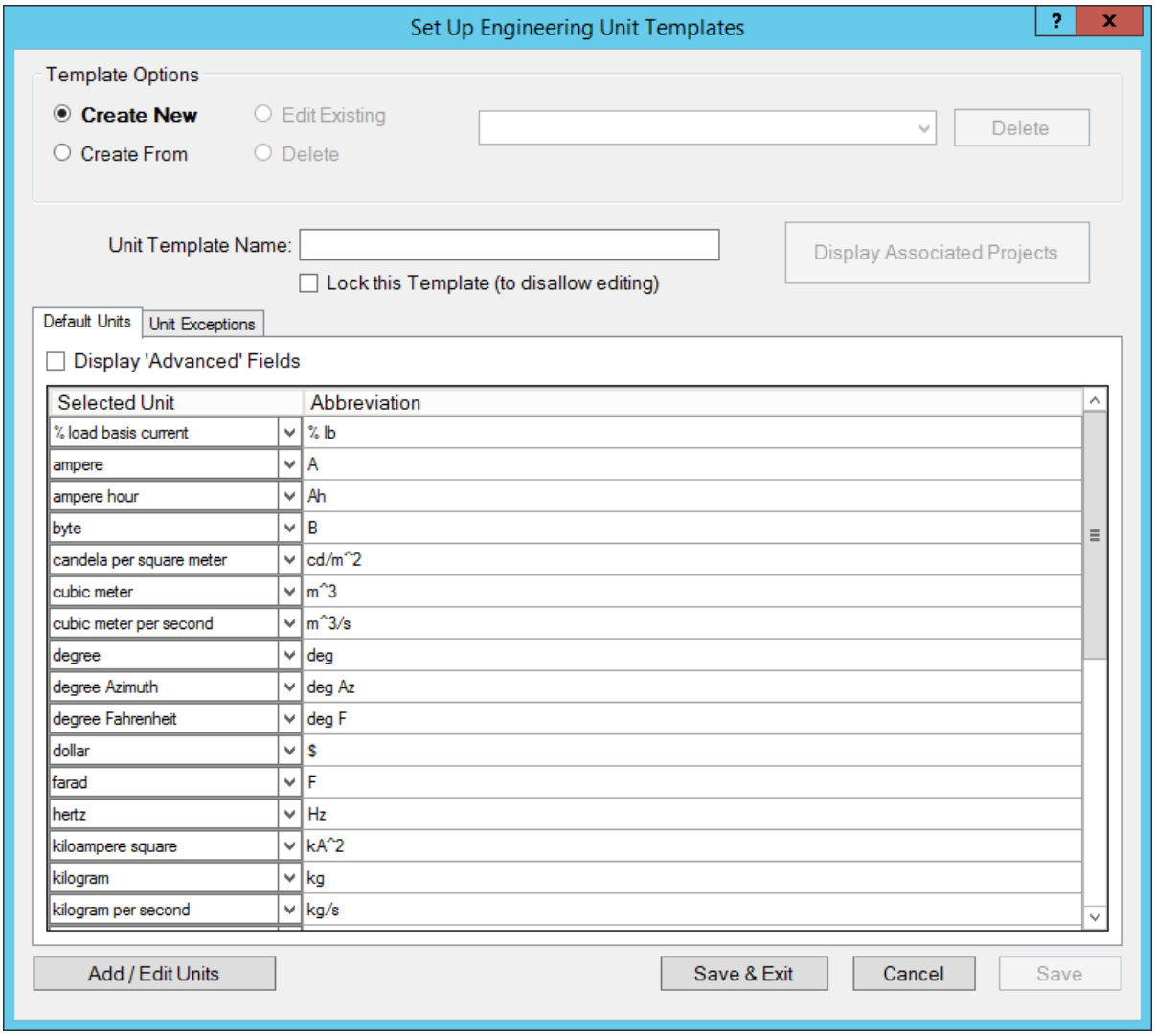
To configure engineering units or conversions, see:

- "Set up engineering templates and select conversions" on page 639
- "Add or edit a base engineering unit or conversion" on page 643

Set up engineering templates and select conversions

Use the Set Up Engineering Unit Templates screen when you want to add, edit, or delete an engineering unit template, or to make changes to how the unit is reported.

To view the Set Up Engineering Units screen, in the Profile Editor, click **Settings > Set Up Engineering Unit Templates**.



The following table describes the parts of the Set Up Engineering Unit Templates screen (it assumes that Display 'Advanced' Fields is checked). When you have finished making change, click Save & Exit.

Field Name	Valid Entries	Comments
Template Options box	Create New	Click to begin creating a new engineering unit template.
	Create From	Click to create an engineering units template that is based on an existing template.
	Edit Existing	Only available if you have added a template. Click to edit an engineering unit or its conversion.
	Delete	Only available if you have added a template. Click to begin deleting an engineering unit and its conversion. You cannot delete a locked template.

Field Name	Valid Entries	Comments
Unit Template to Create From	From the drop-down menu, select the template you wish to copy, in order to create a new template.	This field is live only when Create From is chosen as the option. The new template will initially include all of the units/conversions of the original; but you can add units and change the conversion settings.
Unit Template Name	This field is blank if you selected Create New or Create From; type the name of the new template. A name displays if you have selected a template to edit; you can change the name. A name displays, but it is greyed out if you selected a template to delete. Click *Save* to save the changes you make.	When creating a new template or creating from an existing template, type the name of the new template. To change the name of an existing template, choose it from the Unit Template to Delete menu, then change the name here.
Lock this Template	Click to prevent the template from being edited in the future.	The only way to “edit” a locked template is to delete it, and add back a new one with the edits entered.
Display Associated Projects	Live only when in “Edit” mode. Displays all projects that use this template.	You only need this if you want to delete a template that is associated with a project. Note the projects that display in the list, then go to the Set Up Project tab. For each project that you noted, change the unit template.
Display ‘Advanced’ Fields	Check this box to display additional columns of information about the template.	Unchecked: displays the unit and its abbreviation only. Checked: displays also the conversion, and its abbreviation and multiplier.
Default Units sub-tab		
Use this sub-tab to manage unit templates and to add global changes to a unit.		
Base Unit	n/a	Many standard units are pre-defined; they cannot be edited or deleted. To add a unit or edit a user-created unit, see "Add or edit a base engineering unit or conversion" on page 643.
Abbreviation	n/a	Added for the unit when the selected unit was created. To edit a user-created unit, see "Add or edit a base engineering unit or conversion" on page 643.

Field Name	Valid Entries	Comments
Selected Unit	Click the down arrow to display and select the preferred conversion for the unit.	Many conversions are pre-defined. To add or edit a conversion unit, see "Add or edit a base engineering unit or conversion" on page 643. Fahrenheit to Celsius temperature conversions must be handled by editing Cicode (Citect.ini).
Abbreviation	n/a	This is abbreviation for the selected unit. When the Selected Unit is changed, this field changes accordingly.
Multiplier	n/a	Added for the unit and for the conversion when the base unit was created. Pre-defined units/conversions cannot be changed. To edit a user-created unit, see "Add or edit a base engineering unit or conversion" on page 643.
Offset	n/a	Used for units that have more than one scale. For example, for temperature, if the base is degree Celsius, and you want to offset to Fahrenheit, you would type 32 here (and 1.8 in the multiplier).
Add/Edit Units button	Click to display the Add/Edit Units screen.	Use that screen to add units/conversions, or to edit user-created units/conversions.
Unit Exceptions sub-tab		
Use this tab to apply "exceptions" for individual tags, changing the way the unit is reported for the tag(s). This is most commonly used for WAGES tags.		
Tags	Choose an individual tag or tag subgroup.	This tag will be reported with the new settings.
Options	1. From the dropdown list, choose the unit you want to use for this tag/tag group. 2. Click the radio button for the exception to be made. 3. Either double-click the tag, or click the right arrow to move it to the Exception list.	1. If you choose Apply Unit Conversion, the tag will be reported according the unit you select. For example, if you want to report Air Volume in gallons, rather than cubic meters, choose "gallon" from the Select Unit dropdown list. 2. Click "Apply Unit Conversion" to convert and report the tag according to the unit you selected. Click "Apply Unit Name Only" to add the unit name to it, but not convert it, when it is reported.
Exception List	Review your changes.	You can check or uncheck tags here, changing them from one conversion option to the other. When you uncheck a tag, you do not remove it, you change it from being converted to simply being reported according the unit you selected.

Apply conversions

Use this screen to apply unit conversions to a template. To add a new conversion, see "Add or edit a base engineering unit or conversion" on page 643.

To apply a conversion:

1. From the main window of the Profile Editor, click **Settings > Set Up Engineering Unit Templates**.
2. Click **Edit Existing**, then select the template for which you want to select unit conversions.
3. In the Selected Unit column, click the down arrow and select the conversion you want to use. Fahrenheit to Celsius temperature conversions are handled by offsets (see "Add or edit a base engineering unit or conversion" on page 643).
4. Repeat step 3 for all units that you want to change.
5. Click **Save** to save the change, or click **Save & Exit** to save changes and close the screen.

Delete a template

You cannot delete the standard template nor a locked template.

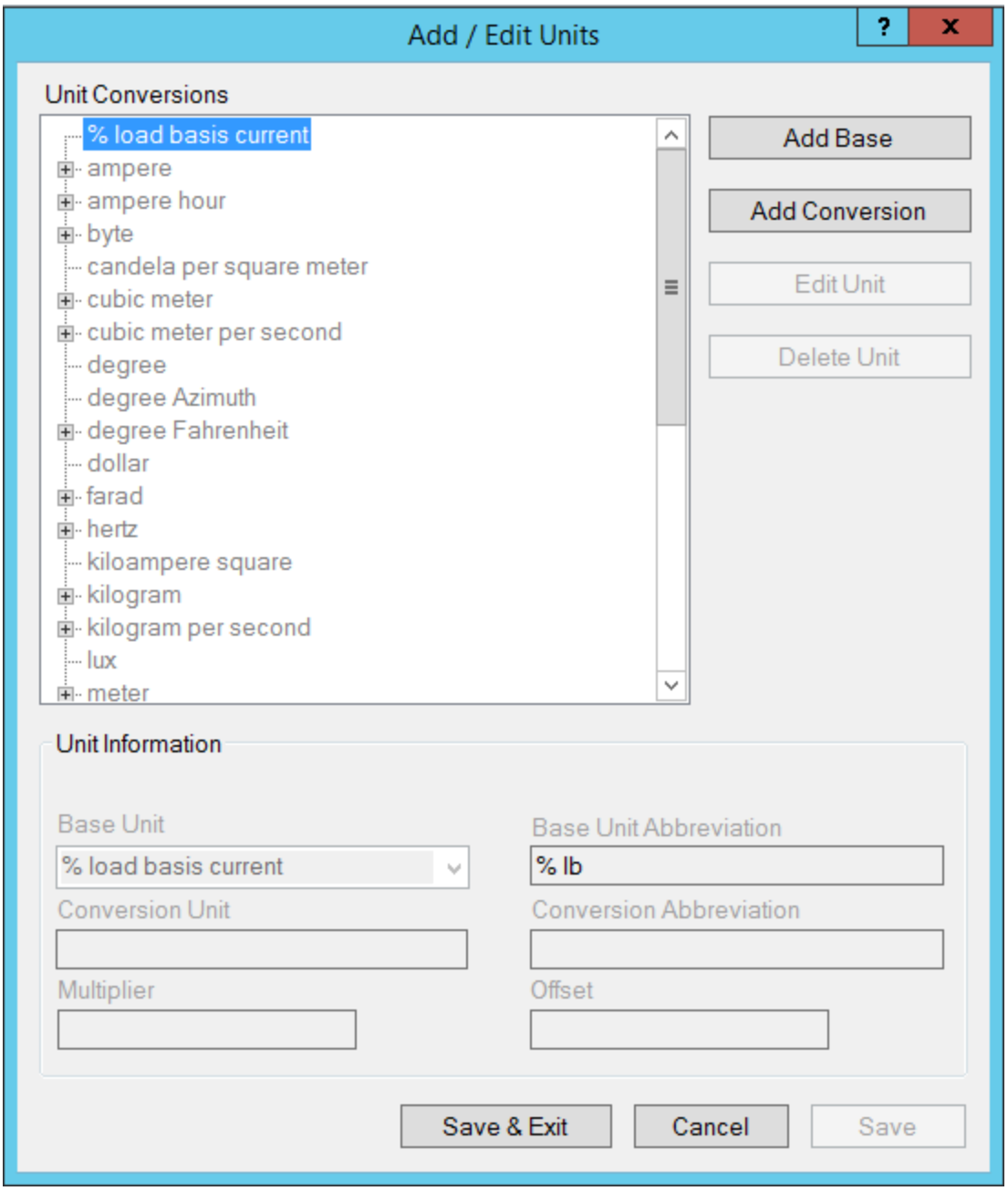
To delete a template:

1. From the **Define Device Type Tags** tab, click **Settings > Set Up Engineering Unit Templates**.
2. Click **Delete**, on the left, to delete a template.
3. Choose the template from the drop-down list.
4. Click Delete, on the right, to delete the selected template. At the Confirm Delete prompt. click **Yes**.

Add or edit a base engineering unit or conversion

Use the Add/Edit Units screen to add, edit, or delete base units and conversion units for custom tags. You cannot make any changes to predefined units (those that are grayed out).

Click Settings > Set Up Engineering Unit Templates. At the Set Up Engineering Units screen, choose the template you want to edit, and click Add/Edit Units.



The following table describes the fields of the Add/Edit Units screen. Instructions for editing and deleting units are after the table.

Field Name	Valid Entries	Comments
Unit Conversions	n/a for pre-defined units/conversions (grayed out) Select user-created units to begin edits.	All base engineering units and their conversions display. Grayed-out items are predefined; they cannot be edited or deleted. Note that predefined units can have custom conversions, which are editable.
Add Base	Click to begin adding a new base unit.	The Base Unit and Base Unit Abbreviation fields become live.

Field Name	Valid Entries	Comments
Add Conversion	Click to begin adding a conversion to a base unit.	The Base Unit field displays the unit you highlighted; the Conversion Unit, Conversion Abbreviation, and Multiplier fields become live.
Edit Unit/ Delete Unit	Click to either edit a custom unit/conversion, or to delete it.	These buttons are live when you select a custom unlocked unit.
Base Unit	When editing a unit/conversion, select the unit from this drop-down menu. When adding a new base unit, type the name.	Used in the Profile Editor only; not passed to projects for graphics viewing.
Base Unit Abbreviation	Type the abbreviation for the selected base unit.	If there is no conversion, this is passed to projects for viewing graphics.
Conversion Unit	Type the name of the conversion unit, such as milliamps, when amps is the base unit.	Becomes live only when you highlight a unit. Used in the Profile Editor only; not passed to projects for graphics viewing.
Conversion Abbreviation	Type the abbreviation for the conversion unit.	This is passed to projects for viewing graphics.
Multiplier	Use this field to determine the number of base units that are in the conversion unit. Type the multiplier "M," where Conversion Unit x M = Base Unit.	Example: There are 1,000 bytes in a kilobyte; so, the conversion unit multiplier is 1000, If you have 17.3 kB, 17.3 x 1,000 = 17300 bytes
Offset	Use this field to determine a numeric offset.	Example: If degrees Celsius is the base unit, and you are creating a conversion unit for Fahrenheit, you would enter a multiplier of 1.8 and an offset of 32.

Edit a base engineering unit or conversion

Changes are global, for all templates. You cannot change predefined engineering units or conversions (grayed out).

To edit a unit or conversion:

1. With the base unit or conversion highlighted, click **Edit Unit**.
   a. For a base unit: You can edit the base unit and base unit abbreviation.
   b. For a conversion: You can edit the conversion unit, abbreviation, and multiplier.

2. Click **Save** to save the changes or click **Save & Exit** to save the changes and close the screen.

Delete a base engineering unit or conversion

Deletions are global, for all templates. You cannot delete predefined units or conversions (grayed out).

To delete a unit or conversion:

1. With the base unit or conversion highlighted, click **Delete Unit**.
2. Click **Yes** to confirm the deletion.
3. Click **Save** to save the changes or click **Save & Exit** to save the changes and close the screen.

LiveView Tables

Click any of the following links to learn about the LiveView tables:

LiveView Basic Readings Summary

This summary displays real-time basic power information for a selected device or devices. After opening the basic readings summary template, you set the update interval for this summary. You can also add or remove devices from the summary.

Make your selections, and click Display Table.

The basic readings summary provides the following data:

- voltage A-B (V)
- current A (A)
- real power (kW)
- power factor

LiveView Power Flow Summary

This summary displays a power flow summary for your system devices. Use the information from this table to help optimize the system's power flow.

After opening the power flow summary template, you set the update interval for this summary. You can also add or remove devices from the summary.

Make your selections, and click **Display Table**.

The power flow summary provides this data:

- real power (kW)
- reactive power (kVAR)
- apparent power (kVA)
- demand average (kW)
- demand peak (kW)
- predicted demand (kW)

LiveView Energy Summary

This summary displays an energy summary for your system devices. Use the information from this table to help monitor the system's energy consumption.

After opening the energy summary template, you set the update interval for this summary. You can also add or remove devices from the summary.

Make your selections, and click Display Table.

The energy summary provides this data:

- real power (kW)
- reactive power (kVAR)
- apparent power (kVA)
- block demand real power (kW)
- thermal demand real power (kW)
- peak block demand real power (kW)
- peak thermal demand real power (kW)
- block demand real power predicted (kW)
- thermal demand real power predicted (kW)

LiveView Energy Readings

This table displays accumulated energy readings for a single device. Data is accumulated beginning with the last energy reset for the device.

Energy values, will be according to one of these accumulation methods:

Absolute (unsigned): The device stores positive energy values, regardless of the direction of power flow. The energy value increases, even during reverse power flow.

Signed: The device stores both positive and negative energy values. The energy value increases or decreases, depending on the direction of the power flow.

After opening the live view energy readings table, choose the device you want and set the update interval for this table. Click Display Table.

The live view energy readings table provides these accumulated readings:

- real energy (kWHr)
- reactive energy (kVARHr)
- apparent energy (kVAHr)

LiveView Fundamental Phasor Readings

This summary displays a fundamental phasor readings table for a single device, to confirm that the system is properly wired.

After opening the fundamental phasor reading template, choose the device for which you want readings and set the update interval for this table. Click **Display Table**.

The fundamental phasor readings table provides a phasor diagram that indicates current and voltage magnitudes and angles for each phase.

LiveView THD Current Summary

This summary displays a THD current summary for your system devices. Use the information from this table to monitor your equipment and system power quality.

After opening the THD current summary template, you set the update interval for this summary. You can also add or remove devices from the summary.

Make your selections, and click **Display Table**.

The THD current summary provides this data:

- phase A
- phase B
- phase C
- neutral

LiveView THD Voltage Summary

This summary displays a THD voltage summary for your system devices. Use the information from this table to monitor your equipment and system power quality.

After opening the THD voltage summary template, you set the update interval for this summary. You can also add or remove devices from the summary.

Make your selections, and click Display Table.

The THD current summary provides this data:

- THD voltage (%):
- Vab
- Vbc
- Vca
- Van
- Vbn
- Vcn

LiveView Uptime Summary

This summary displays an uptime summary for your system devices. Use the information from this table to view the number of seconds the equipment has been in uptime (defined as all three phases > 10% nominal), and to view the percentage of uptime vs. total time. The summary includes the last 12 months.

After opening the uptime summary template, you set the update interval for this summary. You can also add or remove devices from the summary.

Make your selections, and click Display Table.

The uptime summary provides this data for the past 12 months:

- Uptime %
- Uptime
- Downtime

LiveView Incremental Reactive Energy Summary

This summary displays an incremental reactive energy summary for your system devices. Use the information from this table to monitor transmission of energy beyond the previous baseline, to help achieve optimum loading.

After opening the incremental reactive energy summary template, you set the update interval for this summary. You can also add or remove devices from the summary.

Make your selections, and click Display Table.

The incremental reactive energy summary provides this data:

- incremental reactive energy into the load (kVARHr)
- incremental reactive energy out of the load (kVARHr)
- date/time of the last incremental energy update

LiveView Incremental Real Energy Summary

This summary displays an incremental real energy summary for your system devices. Use the information from this table to monitor the energy usage and production above the previous baseline, to help achieve optimum loading.

After opening the incremental real energy summary template, you set the update interval for this summary. You can also add or remove devices from the summary.

Make your selections, and click Display Table.

The incremental real energy summary provides this data:

- incremental real energy into the load (kVARHr)
- incremental real energy out of the load (kVARHr)
- date/time of the last incremental energy update

LiveView Harmonic Apparent Power Flows

The harmonic apparent power flows table displays real-time apparent power flow information. Use this information to help determine the impact of harmonics on system equipment.

After opening the harmonic apparent power flows template, select the device, then click Display Table.

In the upper right, you can set the update interval for this table.

You can set meter registers to enable frequency domain analysis of waveforms and the format used in analysis. Harmonics and trend tables reflect these register settings. For details about these settings, read the Notes to the right of the table.

The harmonic apparent power flows table provides this data:

- meter type
- wiring type
- FFT magnitudes
- FFT enable
- FFT status
- FFT hold time
- remaining hold
- total voltage harmonic distortion for all three phases
- total current harmonic distortion for all three phases
- harmonic components for all three phases:
  - power flow in from the utility
  - power flow out to the utility
  - apparent power

Magnitudes and angles are available for all odd harmonics from H1 through H31.

LiveView Harmonic Reactive Power Flows

The harmonic reactive power flows table displays real-time reactive power flow information. Use this information to help determine the impact of harmonics on system equipment.

After opening the harmonic reactive power flows template, select the device, then click Display Table.

In the upper right, you can set the update interval for this table.

You can set meter registers to enable frequency domain analysis of waveforms and the format used in analysis. Harmonics and trend tables reflect these register settings. For details about these settings, read the Notes to the right of the table.

The harmonic reactive power flows table provides this data:

- meter type
- wiring type
- FFT magnitudes
- FFT enable
- FFT status
- FFT hold time
- Remaining hold
- total voltage harmonic distortion for all three phases
- total current harmonic distortion for all three phases
- harmonic components for all three phases:
  - power flow in from the utility
  - power flow out to the utility
  - reactive power

Magnitudes and angles are available for all odd harmonics from H1 through H31.

LiveView Harmonic Real Power Flows

The harmonic real power flows table displays real-time real power flow information. Use this information to help determine the impact of harmonics on system equipment.

After opening the harmonic real power flows template, select the device, then click Display Table.

In the upper right, you can set the update interval for this table.

You can set meter registers to enable frequency domain analysis of waveforms and the format used in analysis. Harmonics and trend tables reflect these register settings. For details about these settings, read the Notes to the right of the table.

The harmonic real power flows table provides this data:

- meter type
- wiring type
- FFT magnitudes
- FFT enable
- FFT status
- FFT hold time
- remaining hold
- total voltage harmonic distortion for all three phases

- total current harmonic distortion for all three phases
- harmonic components for all three phases:
  - power flow in from the utility
  - power flow out to the utility
  - real power

Magnitudes and angles are available for all odd harmonics from H1 through H31.

LiveView Demand Current Summary

This summary displays a demand current summary for your system devices. Use the information from this table to help monitor the system's demand current.

After opening the demand current summary template, you set the update interval for this summary. You can also add or remove devices from the summary.

Make your selections, and click Display Table.

The demand current summary provides this data:

average demand current and peak demand (both in amps)

- Ia
- Ib
- Ic

Live View Demand Voltage Summary

This summary displays a demand voltage summary for your system devices. Use the information from this table to monitor the system's demand voltage.

After opening the demand voltage summary template, you set the update interval for this summary. You can also add or remove devices from the summary.

Make your selections, and click Display Table.

The demand voltage summary provides this data:

- demand voltage
- Vab
- Vbc
- Vca
- Van
- Vbn
- Vcn

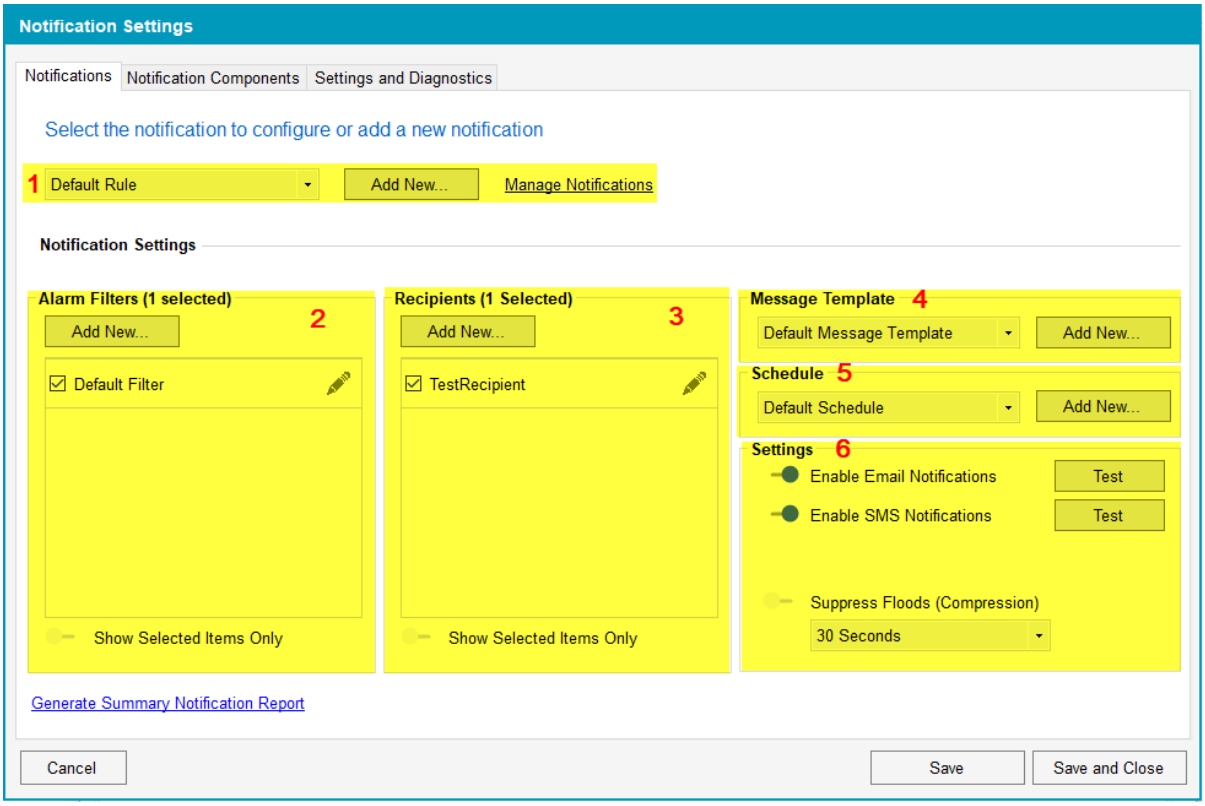
Notifications Reference

This section contains information on the Notifications Settings user interface (UI) as well as more detailed information on configuration options.

For detailed information on the notifications UIs, see the following topics:

Notifications UI

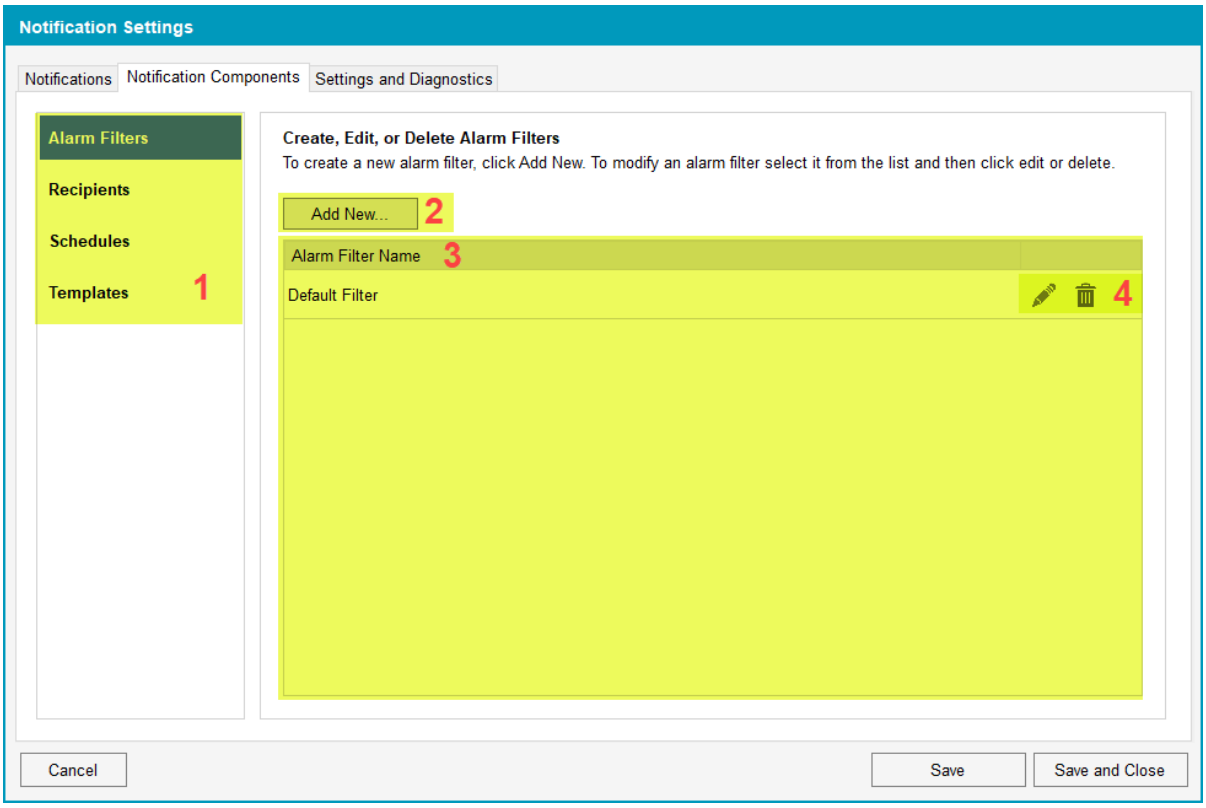
The **Notifications** pane lists all the system notifications and displays all the component information of a selected notification.



1	Create, edit, or manage your notifications. For more information on managing notifications, see
2	Edit or create alarm filters for the selected notification. For detailed information on alarm filters, see
3	Edit or create recipients for the selected notification. For detailed information on recipients, see
4	Select or create a message template for the selected notification. For detailed information on message templates, see
5	Select or create a schedule template for the selected notification. For detailed information on schedules, see
6	Set and test notification relays, and suppress floods.

Notifications Components UI

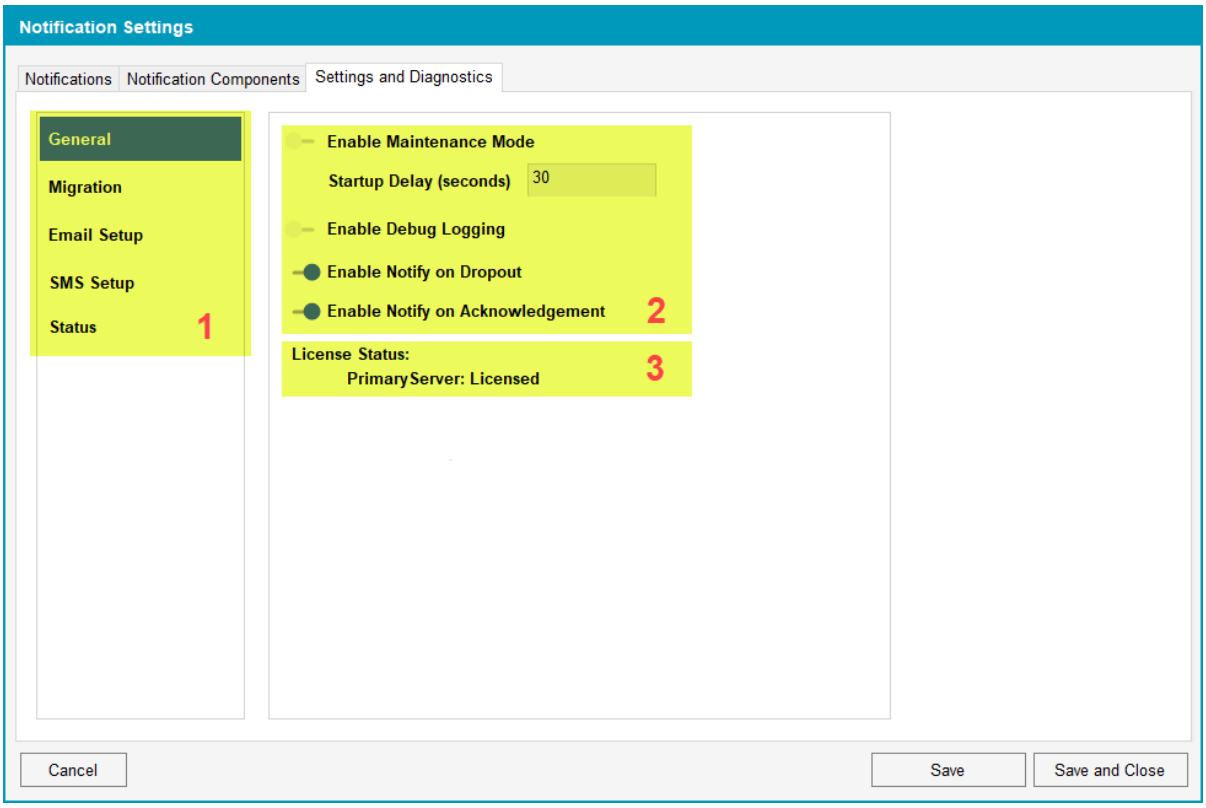
Notification Components consist of alarm filters, recipients, schedules, and templates. Use the **Notification Components** pane to manage notification components.



1	Notification component navigator pane. Click a component name to work with that component.
2	Create a new notification component.
3	Notification component list. This pane displays all the components that exist in the system.
4	Edit or delete an existing component.

Settings and Diagnostics UI

Settings and Diagnostics consists of the Notifications Settings configuration, migration, diagnostics, and licensing features and information.



1	Settings navigator pane. Click a setting category to work with that setting. For more information, see Configuring Notifications.
2	System diagnostics and settings. For more information, see Troubleshooting Notifications.
3	The server's license status.

Alarm Filter System Views

To help you create alarm filters, Notifications Settings displays all the system alarms using several views. A *view* logically groups alarms to help you quickly find the alarms you want to filter. When you select a view, the alarm tags are grouped by view and displayed in nodes.

The following table lists the alarm views upon which you can filter your alarms, and where these project values are stored in Power SCADA Studio:

System View	Power SCADA Studio Value
Equipment View	System Model > Alarms > Equipment > Equipment
Priority View	Setup > Alarm Categories > General > Priority
Category View	System Model > Alarms > General > Alarm Tag
Alarm Category View	Standards > Labels > Expression
Area View	System Model > Alarms > Security > Area
Tag View	System Model > Alarms > General > Alarm Tag

Glossary

address

The address contains all the information the SCADA system needs to get values from a known device, and to return a value determined by the values read from the device and the calculation rules defined in the address.

alarm categorization

Added when setting up custom tags, this is one of the alarm filters. which will be used for filtering and sorting alarms in the Alarm Log. Categories are: normal, over, over hs, rate of change, reversal, sag, swell, transient, under, and under hs.

alarm text (On/Off)

For onboard alarms, this is the text (added while adding a custom tag) that displays when the alarm is on or off. This text will display in the Alarm Log.

alarm filters

Setup in the Profile Editor, these filters help you filter and sort data that displays in the Alarm Log.

alarm groups

Added when setting up custom tags, this is one of the alarm filters. which will be used for filtering and sorting alarms. Groups are: frequencies, motors, power factors, powers, temperatures, time, and voltages.

alarm levels

Added when setting up custom tags, this is one of the alarm filters. which will be used for filtering and sorting alarms. Levels are: event, high, medium, and low.

alarm types

Added when setting up custom tags, this is one of the alarm filters. which will be used for filtering and sorting alarms. Types are: diagnostic, power quality, protection, and system.

Automation Interface

Used instead of the Profile Wizard, this tool allows you to add multiple devices at a time to a project.

bandwidth

The amount of space or processor resource being used by a part of the system. You can use the bandwidth allocation parameters to allocate bandwidth for different types of data.

baud rate

The speed of transmission of electrical signals on a line. This is often described in bits per second (bps), although the baud rate and bps are not truly interchangeable. The baud is actually the measurement of how frequently the sound changes on the line.

bitmask

A mask is defined as data that is used with an operation to extract information that is stored in another location of the code. A bitmask is the most common mask used. It extracts the status of certain bits in a binary string or number (a bit field or bit array).

Cicode

This programming language, which is similar to Visual Basic or "C," allows you to access and edit real-time data in the project. Although not difficult to use, the person working in Cicode must have received Cicode training.

cluster

A discrete group of alarms servers, trends servers, reports servers, and I/O servers. It would usually also possess local control clients. For a plant comprising several individual sections or systems, multiple clusters can be used, one cluster for each section.

CommsMethod (communications method)

This is the communication protocol, such as MODBUS/RTU via Gateway, that is being used by a device. When adding devices in the automation interface, you will need to specify the CommsMethod.

ComPort

(also COM port) The computer's communications port used to connect to devices, for sending and receiving serial data.

composite device type

A composite profile can be made from more than one device type. Each device type included in the composite profile can use its own protocol for communication. The composite device type allows the engineer to use two devices for one monitoring point, e.g., a breaker and a monitoring device. Power SCADA Operation combines the functionality of the two devices so that the end user only needs to consider one device when analyzing that location in their system.

configuration environment

(See design time environment.)

control

This is a command written to a device register that then causes an action within some equipment. There are a series of default control tags in Power SCADA Operation to achieve these actions. For example, in the Sepam 40, there are control tags to operate a circuit breaker and enable a recloser.

CRA

Remote I/O drop header

custom device type

This is a "new" device type that is added to a system. Although the Profile Editor includes many standard device types, it may be necessary to add a new device type that includes custom tags, or one that includes a different set of tags than the standard device types.

custom tag

This is a "new" tag that is added to the system. Although the Profile Editor includes many standard tags, you may need to add a tag for a third party device, or to edit an existing tag to change its attributes. In these cases, you need to add a custom tag. These tags are then added to a

customized device type to be made available in profiles and projects. The custom tag creation interface applies rules to the tag creation to help guide the user to making tags that will correctly retrieve the desired information from devices.

DataBits

This is the number of data bits used in data transmission. The I/O device and the ComPort must have the same value.

data type

Data types are restricted to these types that are supported by the SCADA system: digital, int, long, real, and string.

demo mode

This demonstration mode allows you to run the product without a hardware key. You can use all of the product features, but with limited runtime and I/O options.

design time environment

To be used only by the person who is creating and installing the project for the end user, this is the environment in which you add devices, profiles, and projects, as well as create genies and one-lines.

device category

Used in the Profile Wizard to logically group device profiles, to make them easier to locate. The default category is "Schneider Electric, and the default subcategories are "Monitoring Device," "PLC," and "Protective Device." Do not confuse these terms with:

- categorization and sub-categorization (alarm filters, used during runtime, to filter and sort alarm data)
- category type: real-time filters that provide metadata for future reporting

device profile

A subset of the device type; where the device type includes all of a device type's attributes, the device profile includes only the specific tags that are used by an individual customer. A device profile is set up like a device type, except that it is specially configured for a particular need. For example, a CM4000 that is being used to monitor the main at a given facility would have a different profile from the CM4000 that is used to monitor water and gas at a facility. The profile also allows you to designate that some tags will be used for trending and/or for PC-based alarming.

device type

Contains all the information for retrieving the available information from a given device type. This information is stored in the form of tags. Tags can be of these types: real-time, onboard alarms, controls, and Resets. Real Time tags can be further separated into groups such as Currents or Energies.

A device type has a name and has one or more drivers associated with it. It also has one or more tags associated with it; for each driver/tag combination, the device type can have an address.

device type drivers

Programs that allow Power SCADA Operation to interact with a device or series of devices. Power SCADA Operation includes several generic drivers (generic MODBUS, Sepam 40 Range, MicroLogic 5P and 6P, CM4000 series, and PM800 series) that interact with "standard" device types.

engineering unit templates

Used for conversions between base units and their conversions (for example, inches to centimeters or amperes to kiloamps).

enumeration (used for the circuit breaker status)

This is a single value (0-5) that defines a condition that is determined by multiple bits. They allow for dynamic contingencies, such as when you need to use multiple bits to describe the position of a circuit breaker.

Time stamping module

Time stamping module

format code

These codes define the attributes of the address field of a tag. See "Format code definitions" on page 202 for a list of format codes.

functional addressing

Creates addressing for a device that has data residing in different registers. Functional addressing dynamically addresses the device, based on its configuration (using C#, you can write code to account for user-defined variables). When you add the profile to a project, you will enable functional addressing. Then, when exporting to the Profile Wizard, you are prompted for the variable(s) related to these device types.

genie

A genie is a multi-layer graphic that is used on the Graphics page to indicate an object, such as a motor, generator, circuit breaker, or switch. Using genies, you only have to configure common behaviors of that object once. The default genie library includes a large number of pre-defined genies. A graphics page can contain any number of genies.

ICD file

IED capability description: This is the file that is imported into the Profile Editor from an IEC 61850 device. Editing for ICD files is limited to the ability to add/delete datasets and control blocks, and the ability to edit buffered and unbuffered control blocks that were created in the Profile Editor.

IEC tag name

The IEC 61850-compatible name that is created when a tag is created. This is the name that is used by the SCADA system. The tag names provided use an abbreviated form of the IEC 61850 naming convention. A tag name cannot include any special characters except (_ \ ). It can be a maximum of 32 characters.

IED

Intelligent electronic device

IID

Instantiated IED description: defines the configuration of one IED for a project; is used as the data exchange format. This file contains data for just the IED that is being configured.

logic code

Logic codes tell the program how to mathematically certain values in device registers, thus providing values that the user needs. Examples of logic codes are date and time for a circuit monitor or a Sepam device, digital inputs/outputs, and IEEE power factor.

metadata

Metadata provides data about other data. In Power SCADA Operation, metadata might include additional information about a custom tag: its category type, utility type, statistical type, or quantity. It is often used for reporting purposes.

multi-monitor support

This option allows you to view the runtime environment from multiple computer monitors. In Power SCADA Operation, this allows you to view a different startup page on each monitor.

OFS

OPC Factory Server

onboard alarm

Onboard alarms are alarms that are detected and stored in a device's data logs. If an onboard alarm is configured within a device, you can map it, via the Profile Editor, to a digital time-stamped alarm in Power SCADA Operation. These alarms and associated waveforms can be read and displayed in the Alarm Log.

PAC

Programmable Automation Controller

parity

Parity is used as a simple means of detecting error by verifying that the result is odd or even. In Power SCADA Operation, parity is required for the generic serial or MODBUS/RTU comms methods, when adding a device.

PC-based alarms

PC-based alarms are alarms that are detected from a device and are stored in the software. You can add them to the Profile Editor when you create the device profile. All PC-based alarms are analog by default.

PMCU

The Meter Configuration Help Utility. Use this application to set up the features within PowerLogic devices, and enabling such features as onboard alarms and waveforms. The information that is generated from PMCU is then available for use within Power SCADA Operation.

point (see SCADA tag)

polling priority

When adding a custom tag, this field determines the level of priority that Power SCADA Operation uses when reading data from the related device. Options are low, normal, or high.

Power SCADA Operation tag name library

This library includes electrical parameters, or measurements or topics. A tag name has three parts:

- an easy to read name (such as Current Phase A)
- a unique identifier
- meta data (attributes used to categorize the data for intelligent display/analysis)

Profile Editor

This tool allows you to create device type tags, device types, and device profiles. This information is then imported into Power SCADA Operation, for use in creating graphics pages.

Profile Wizard

This tool allows you add device profiles to, or delete them from, a project. From the Profile Editor, you export profile data into a file that can be used in the project. From there, you use the Profile Wizard to add the device profile into a project.

project

A project is made up of any number of profiles. Profiles that have been added to a project can be imported into the SCADA system and made available for setting up actual devices in the SCADA system.

A project name must match exactly between the Profile Editor and Power SCADA Studio.

Each project includes: a unit template, display name, and one or more instantiated device profiles (instantiated by choosing a device profile and specifying a name). The following is a simple example of how device profiles and projects inherit information from the device type.

- The device type myCM4Type can use either the Modbus driver or the IEC 61850 driver.
- The device profile myCM4Profile inherits this device type.
- The project myCM4Project instantiates the myCM4Profile and calls it myModbgusCM4, and it specifies that it uses the Modbus driver.
- When this project is imported into the SCADA system, Modbus addressing will be used.

register scaling

This is a conversion that is the result of applying a scaling multiplier to a register value.

resets

This feature allows you to reset data from a device. There are some pre-defined resets, such as device date/time and onboard data logs, You can also add custom resets.

reserved names

The following terms are reserved for use in the Include project. If you use them in projects that you create, they can cause compilation errors:

- IO_Server
- Report_Server
- Alarm_Server

- Trend_Server
- Client

runtime environment

This is where the end user views system information. This environment includes the one-line diagrams with interactive objects, alarm and event pages, and analysis pages (from which users can view trends and waveforms).

SCADA (Supervisory Control and Data Acquisition)

A system that collects data from various points, both local and remote, and then stores the data at a central location. A SCADA system also can control equipment at a facility.

SCADA tag (SCADA point)

A SCADA tag is an extension of the tag name. A SCADA tag is made up of five parts: two in addition to those already defined in the Power SCADA Operation tag library:

- an easy to read name (such as Current Phase A)
- a unique identifier
- an address (where on a device to read the raw data from)
- a formatting scheme (what to do with the data after it is read to scale it)
- meta data (attributes used to categorize the data for intelligent display/analysis).

SCL

Substation Configuration Language, the configuration description language for communication in electrical substations related to IEDs (defined by IEC 61850-6). This language is used when importing/exporting ICD files. SCL files are used in such devices as G3200 gateways.

SOE

Sequence of Event – a sequential set of state transitions recorded by an RTU. Each transition is represented by an event object, often recorded with the time of occurrence

StopBits

The number of bits that signals the end of a character in asynchronous transmission. The number is usually 1 or 2. Stop bits are required in asynchronous transmissions because the irregular time gaps between transmitted characters make it impossible for the server or I/O device to determine when the next character should arrive.

super-genie

Dynamic pages (usually pop-ups) to which the system can pass information when the runtime page displays. You can use super-genies for pop-up type controllers (for a very specific task that may not be always needed).

tag

Any quantity or measurement (topic) that is recorded by the device; for example, current A. All tag names will use the IEC61850 naming convention. The user can create custom tags; the naming convention will be in the following format:

<EquipmentName>\<PointName>

Where <EquipmentName> uses '_' (underscore character as a separator)

Where <PointName> uses '\' (backslash as a separator)

For example: SST_MV_BUSA_INC1\XCBR1\Pos

A tag contains a tag description, units, tag name, data type, and address.

Tags include the following (* indicates required information):

tag name*
display name*
group*
data type*
engineering units
Citect formatting
polling priority
alarm "on" text
alarm "off" text
category type
utility type
statistical type
quantity
alarming categorization
alarm type
alarm group
alarm subcategorization
alarm level

The tag's group determines the tag's class:

If the tag's group is onboard alarm, control, or reset, the tag's class is the same.

If the tag's group is anything else, the tag's class is real time.

tag address

This "address" includes everything you need to know about a tag (quantity/topic). Included are the data type, priority, and logic code; and how the tag is displayed in registers. You can change address attributes on the Edit Address screen. The full tag address displays on the Define Device Type Tags tab when "Advanced Properties" is selected.

tag description

The tag description is a human readable name which can include spaces and special characters (except for \ / : * ? < > |). The description can be a maximum of 32 characters long.

tag group

The basic groups include: real-time, of which there are several sub-groups (for example, currents, energies, frequencies and power factors); onboard; control; and reset.

units

Units are the standard measurement associated with the quantity measured by a tag. Units come in two types: base units and conversion units.

Some information is common to all units, and some applies only to conversion units:

Common Information: base unit name, base unit abbreviation

Conversion Unit Information: conversion unit name, conversion unit abbreviation, offset, multiplier

units template

The units template defines the conversion factor that must be applied to the standard units provided in order to give the user their information in their desired units. The units profile applies to an entire project. For example, If the standard unit for a device is MW, but the user wants their project to display KW, they must define this units conversion in the units template and then apply it to an entire project.

user privileges (user access, user rights)

This feature allows you to control the amount of access that each user has to the system. User privileges are password-protected. See"Default User Access Settings (Privileges)" on page 357 for more information.

vector math

Vector math and vector math IEEE are two logic codes. They are the result of math that use vectors, which are directed quantities.

zOL

A memory device that is used to drive one-line animation graphics. You must have at least one zOL device per project.

Schneider Electric
35 rue Joseph Monier
92500 Rueil Malmaison – France
www.schneider-electric.com

7EN02-0413-00 09/2018