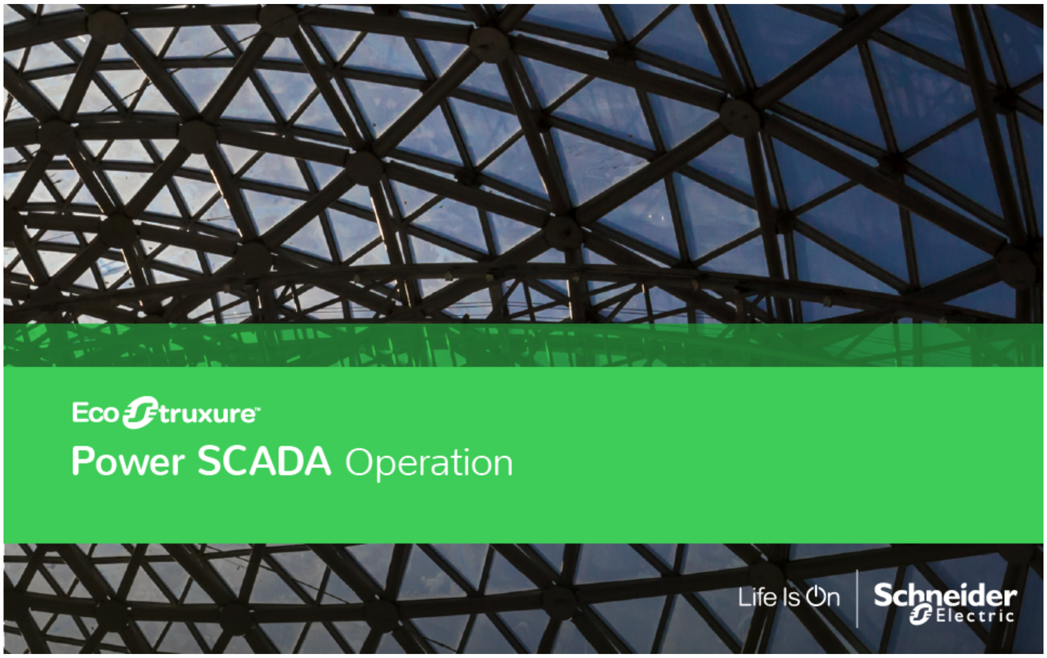
EcoStruxure™

Power SCADA Operation 2020 with Advanced Reporting and Dashboards

System Guide

7EN02-0440-03

04/2022



Legal Information

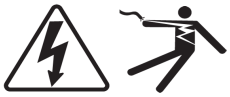
Electrical equipment should be installed, operated, serviced and maintained only by qualified personnel. No responsibility is assumed by Schneider Electric for any consequences arising out of the use of this material.

As standards, specifications and designs change from time to time, please ask for confirmation of the information given in this publication.

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, or equipment damage.

⚠ 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 2020 – 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:

- Plan
- Install and upgrade (includes Licensing)
- Configure
- Administer
- Operate
- Extend
- Troubleshoot
- Decommission
- Reference

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 mirrors the structure of the functional life-cycle stages.

Some tools, tasks, or functions are specific to a stage, others are part of various stages. For example, you set up Power SCADA projects during the Configuration stage. In contrast, basic reports have a Configuration component and an Operating component.

Content 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 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 built 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 the following competencies and training

- Power SCADA Operation with Advanced Reporting and Dashboards 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.
- 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 2020 with Advanced Reporting and Dashboards Revision 2 (R2) is a minor release that introduces several improvements. We highly recommend you upgrade your existing Power SCADA Operation system to version 2020 R2.

Highlights of the 2020 R2 CU 2 release – April 2022

- Users can now turn off credential requirements for individual control components, as per their discretion.

- The ability to configure TGML to read or write alarm properties using web graphics was added.

Previous releases

Highlights of the 2020 R2 release – September 2020

- Improved IT compliance supporting the latest Internet Protocol version 6 standard (IPv6).

Several improvements were made to the Web Applications. You can now:

- Improved IT compliance supporting the latest Internet Protocol version 6 standard (IPv6).
- Several improvements were made to the Web Applications. You can now:
  - Navigate from an alarm to its associated graphics page.
  - View real time and historical trends.
  - Open web pages from the Diagrams application.
  - Debug tags in Diagrams using the Advanced Tag Debugger TGML component.
  - Embed Basic Reports in Web Applications.
  - Configure how alarms are displayed in the Web Application banner.
  - Export and import alarm views.
- Updated Floating License Manager to version 2.5
- Updated Power SCADA Operation 2020 System Guide, including:
  - New interactive TGML graphics examples.
  - New linked TGML graphics examples.
  - New TGML snippet examples.
  - New Advanced Tag Debugger content.
  - "Navigate to associated graphics page" on page 778
  - Exporting and importing alarm menus.
  - Changing the alarms display in the Web Application banner.
  - Integrate EcoStruxure Building Operation in the Power SCADA Operation Web Applications.

Highlights of the 2020 CU1 release – April 2020

- Trends web application supports real time data.
- Control over web client.
- Graphical enhancements, including arc flash.
- PLS_Example available as TGML.
- Expanded device support Including Easergy P3 and T300.

- Updated Floating License Manager.
- Updated Power SCADA Operation 2020 System Guide, including:
  - new graphics workflows.
  - alarm count binding and filtering.
  - pop-ups.
  - graphics de-cluttering, and more.

Highlights of the 2020 release – February 2020

- New native web client with modern graphics and alarms applications. See the Web Applications Configuring and Operating sections for more information.
- Simplified graphics tooling and multi-source one-line coloring with the new Graphics Editor. See Diagrams configuration and the Graphics Appendices for more information.
- French version of the user interface, System Guide, and embedded Help.
- Expanded device support: See Supported power devices for details.
- Cybersecurity updates, including TLS 1.2 encryption capabilities.
- Windows Server 2019 support.
- Power SCADA Operation 2020 is built on Citect SCADA 2018 R2. See the Citect SCADA 2018 R2 help for more information on the new features, including TLS 1.2 encryption.

Resources

Download Center

The following Power SCADA Operation 2020 with Advanced Reporting and Dashboards documents are available on the Schneider Electric Download Center:

- *Power Monitoring Expert 2020 System Guide* (English) – Document number 7EN02-0426
- *Power Monitoring Expert 2020 Web Applications Guide* (Multilingual) – Document number 7EN02-0427

Exchange Extranet

- EcoStruxure Power SCADA Operation portal (product demos, videos, and other product content)
- Power SCADA Operation 2020 with Advanced Reporting and Dashboards Design & Quote tools, including:
  - PSO Software Assurance Calculator
  - PSO Disk Sizing Calculator
  - PSO Commissioning Time Tool

- Power SCADA Operation 2020 with Advanced Reporting and Dashboards document:
  - *Power SCADA Operation 2020 with Advanced Reporting and Dashboards – IT Guide* (English) – Document number 7EN42-0169 (Design & Quote)
- Power Monitoring Expert 2020 documents:
  - *Power Monitoring Expert 2020 – IT Guide* (English) – Document number 7EN42-0193 (Design & Quote)
  - *Power Monitoring Expert 2020 System Guide* (English) – Document number 7EN02-0426 (Install & Maintain)

Power SCADA Anywhere

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

Manuals

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

- Citect SCADA Help
- Release Notes: Includes information specific to this release of the product

Help files

In addition to the help file released with this product, there are several related help files. They are in `C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020 R2\bin`

Glossary

For a list of commonly used terms, see the Glossary.

Other

- Schneider Electric Cybersecurity Support Portal – The latest cybersecurity news. Includes security notifications, and where you can report a vulnerability, or a security or data privacy event.

  NOTE: The Schneider Electric Cybersecurity Portal lets you register for automatic email notification of the newly released or updated Security Notifications.

- Schneider Electric Knowledge Base
- Schneider Data Privacy and Cookie Policy

Use the following links to obtain support if you can't find what you're looking for in this help or on the Schneider Electric Exchange:

- Schneider Electric - Contact Support (Technical Support)
- mySchneider app

  24/7 support. Mobile catalog. Access to expert help.
- Software Licensing Support

  Offline license activation, license returns
- Software Registration Centers

  Global contact information. Contact a Software Registration Center (SRC) if you exceed the license return limit, or if a license has become untrusted. Do not contact an SRC for troubleshooting license issues or to get new licenses. They are not able to help with these issues.

.

Cybersecurity

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

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

Cybersecurity awareness

Cybersecurity features

Recommended actions

Cybersecurity awareness

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 the first step to prevent cyber intrusions. Review the following resources to increase your cybersecurity awareness:

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

To find out about the latest cybersecurity news, sign up for security notifications, or to report a vulnerability, visit the Schneider Electric Cybersecurity Support Portal. See Resources for link information.

RECOMMENDATION: Sign-up for security notification emails on the Schneider Electric Cybersecurity Support Portal.

NOTE: To prevent CSV-injection attacks, use the latest version of Microsoft Excel.

Cybersecurity features

Power SCADA Operation includes features that help to secure your network, servers, and user access:

- Support for McAfee Application Control software to help protect against zero day attacks.
- Windows Active Directory integration, role-based access control, and two-factor authentication using YubiKey.
- Power SCADA Runtime user partitioning (8 levels of user privilege) and user event monitoring (log in, log out, shutdown, control, etc.).

Recommended actions

You must take steps to help secure your system at every stage of the project life-cycle. The following table lists the actions we recommend to help secure your system, organized by life-cycle stage. Follow the links in the table to find the content you are looking for:

NOTE: Create strong passwords when configuring projects in the Power SCADA Studio.

Life-cycle Stage	Recommended Actions
Plan	"Encryption" on page 79 "Windows Active Directory" on page 82 "Two-factor authentication" on page 83
Configure	"McAfee Application Control" on page 592 "Two-Factor Authentication" on page 593
Administer	"Windows Updates" on page 692
Decommission	"Decommission" on page 832

Life-cycle Stage	Recommended Actions
Configure	"McAfee Application Control" on page 592 "Two-Factor Authentication" on page 593
Administer	"Windows Updates" on page 692
Decommission	"Decommission" on page 832

Plan

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 following table to find the content you are looking for:

Topic	Content
"Overview" on page 45	An overview of Power SCADA Operation and how it differs from non-power SCADA applications.
"Components and single-site architectures" on page 47	Design considerations and sample architectures for the Power SCADA Operation components.
"Multi-site architectures" on page 63	Design considerations for multi-site architectures to monitor multiple systems from a single location.
"Connected devices and protocols" on page 66	Detailed information on supported drivers (native, 3rd party, other devices, and Citect drivers), supported protocols, and waveform file share access.
"Computer requirements" on page 71	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.
"Cybersecurity" on page 78	Recommendations on how to help secure your system from a malicious cyber-attack.
"Localization and translation" on page 85	Design considerations for localizing Power SCADA .
"Licensing" on page 86	Licensing information and options.
"Commercial references" on page 89	Power SCADA Operation commercial references.
"Integrating with Advanced Reporting and Dashboards" on page 91	Customizing advanced reports and design considerations for device communication in Power SCADA Operation with Advanced Reporting and Dashboards.
"Interoperability" on page 97	Integrating Power SCADA Operation with other systems (EBO, 3rd party) and for extending and customizing your system (Cicode, CtAPI).
"Power SCADA Anywhere" on page 103	Design considerations for use of Power SCADA Anywhere.
"OFS system time stamping" on page 647	Architectural guidelines for implementing system time stamping.

Overview

Power SCADA Operation is uniquely designed to let you take advantage 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 using 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 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 built 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:

- HTML5 native graphics, alarms, and waveforms
- Event Notification Module
- 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 single-site architectures

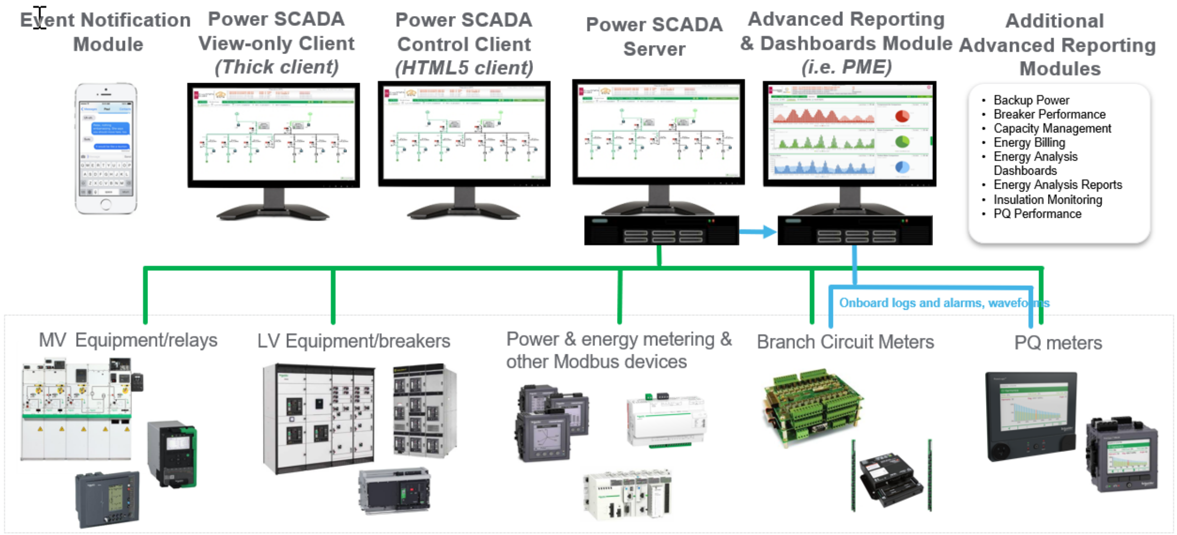
This section provides information on the design considerations for Power SCADA Operation 2020 with Advanced Reporting and Dashboards components as well as component architectures.

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

Topic	Content
"Components overview" on page 48	An overview of the Power SCADA Operation with Advanced Reporting and Dashboards components and data flows.
"Time synchronization" on page 48	Time synchronization requirements for Power SCADA systems.
"Power SCADA Server component" on page 49	Server component purpose, licensing options, design considerations, and how points are calculated.
"Server component architecture" on page 50	Standalone and redundant server component architectures and data flows.
"Power SCADA Control Client component (HTML5 client)" on page 52	Client component purpose, licensing options, and design considerations.
"Control Client component architecture" on page 53	Client component architectures and data flows.
"Power SCADA View-only Client component (thick client)" on page 55	View-only client component purpose, licensing options, and design considerations.
"Power SCADA View-only Client architectures" on page 56	View-only client component architectures and data flows.
"Event Notification Module component" on page 56	Event Notification Module (ENM) purpose, licensing options, and design considerations.
"Advanced Reporting and Dashboards component" on page 58	Advanced Reporting and Dashboards component purpose, licensing options, and design considerations. Architectures and data flows. Additional module information.

Components overview

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


Event Notification Module
Power SCADA View-only Client (Thick client)
Power SCADA Control Client (HTML5 client)
Power SCADA Server
Advanced Reporting & Dashboards Module (i.e. PME)
Additional Advanced Reporting Modules
Backup Power
Breaker Performance
Capacity Management
Energy Billing
Energy Analysis Dashboards
Energy Analysis Reports
Insulation Monitoring
PQ Performance
Onboard logs and alarms, waveforms
MV Equipment/relays
LV Equipment/breakers
Power & energy metering & other Modbus devices
Branch Circuit Meters
PQ meters

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

Time synchronization

⚠ WARNING

INACCURATE DATA RESULTS

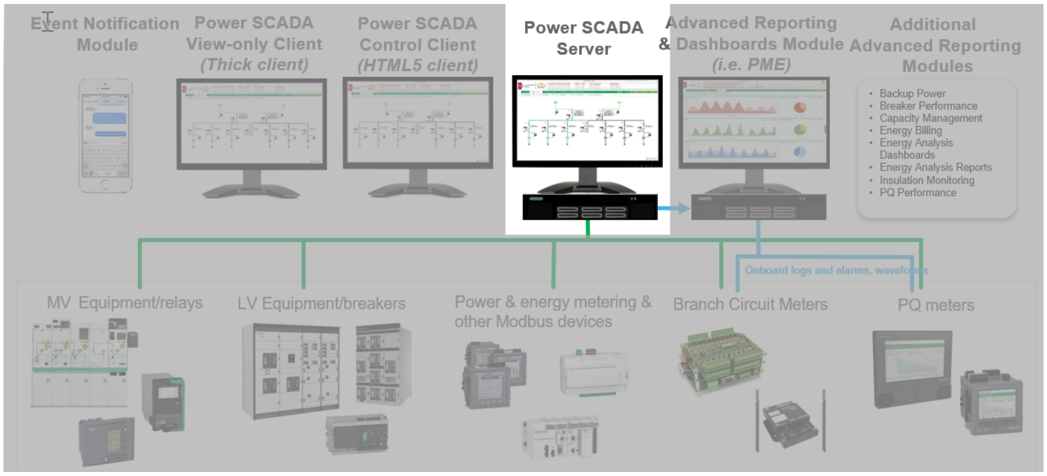
- Do not incorrectly configure the software or the devices.
- Do not base your maintenance or service actions solely on messages and information displayed by the software.
- Do not rely solely on software messages and reports to determine if the system is functioning correctly or meeting all applicable 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 are synchronized to the same NTP server (public or private). If you do not synchronize time across Power SCADA components, alarms and notifications may be delayed.

Do not confuse time synchronization with 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...

Power SCADA Server component


Event Notification Module
Power SCADA View-only Client (Thick client)
Power SCADA Control Client (HTML5 client)
Power SCADA Server
Advanced Reporting & Dashboards Module (i.e. PME)
Additional Advanced Reporting Modules
Backup Power
Breaker Performance
Capacity Management
Energy Billing
Energy Analysis Dashboards
Energy Analysis Reports
Insulation Monitoring
PQ Performance
Onboard logs and alarms, waveforms
MV Equipment/relays
LV Equipment/breakers
Power & energy metering & other Modbus devices
Branch Circuit Meters
PQ meters

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/ EBO, CtAPI)
  - Each Server supports up to 10 concurrent OPC, 10 concurrent OLED8, and 10 concurrent CtAPI connections.
- 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 or tags (options include: 500, 1500, 5000, 15000 and Unlimited tags). For more information on licensing, see "Licensing" on page 86.

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 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 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:

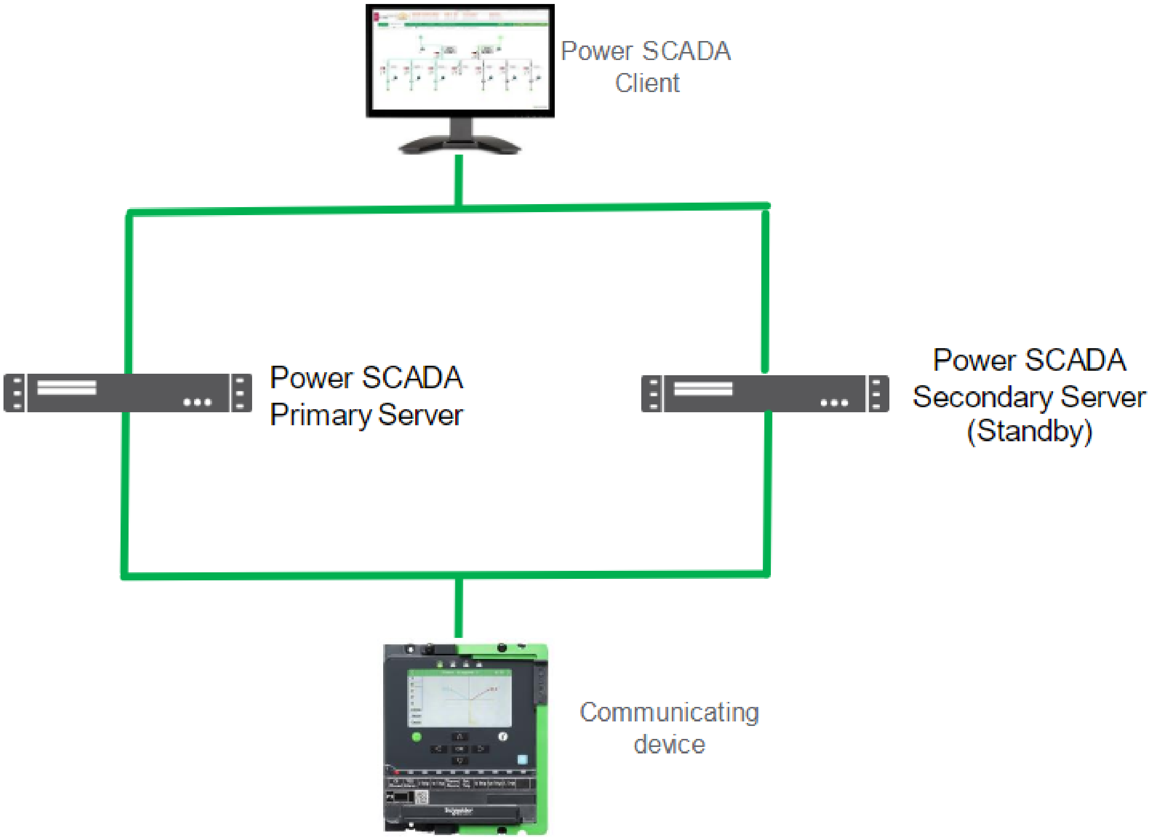
- I/O point count is now tag based not address based. For example, two tags that use the same PLC address will be counted twice. If two trend tags use the same variable tag, it will be counted once. The same applies to alarms.
- For the multi-process mode, each server component will accumulate its own 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 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.


Power SCADA Client
Power SCADA Primary Server
Power SCADA Secondary Server (Standby)
Communicating device

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:

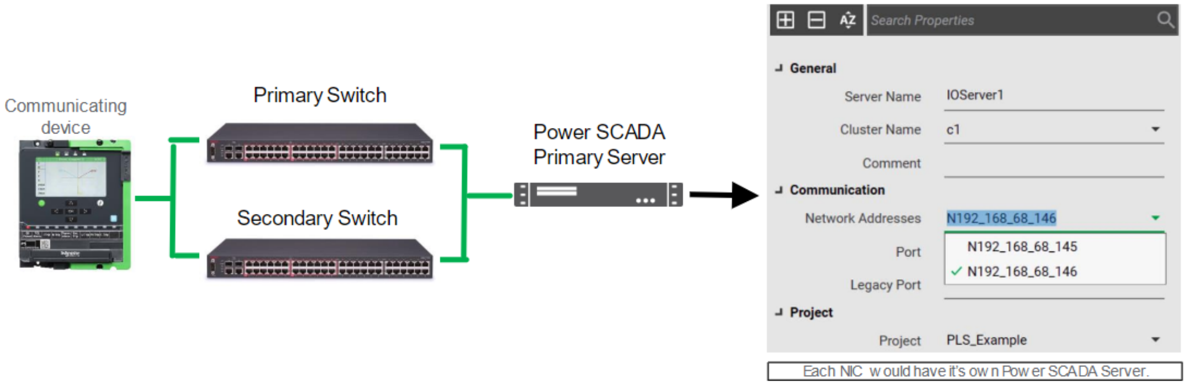
1. Making a change on Secondary Server
2. Restarting Secondary Server
3. Making the updated Secondary Server the Primary Server

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



General

Server Name	IOServer1
Cluster Name	c1
Comment	

Communication

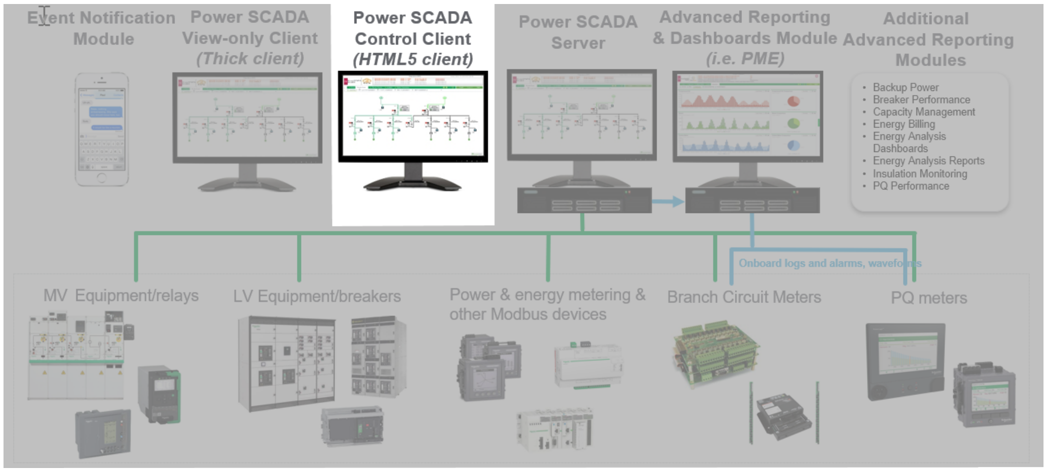
Network Addresses	N192_168_68_146
Port	N192_168_68_145 ✓ N192_168_68_146
Legacy Port	

Project

Project	PLS_Example

Each NIC would have it's own Power SCADA Server.

Power SCADA Control Client component (HTML5 client)


Event Notification Module
Power SCADA View-only Client (Thick client)
Power SCADA Control Client (HTML5 client)
Power SCADA Server
Advanced Reporting & Dashboards Module (i.e. PME)
Additional Advanced Reporting Modules
Backup Power
Breaker Performance
Capacity Management
Energy Billing
Energy Analysis Dashboards
Energy Analysis Reports
Insulation Monitoring
PQ Performance
Onboard logs and alarms, waveforms
MV Equipment/relays
LV Equipment/breakers
Power & energy metering & other Modbus devices
Branch Circuit Meters
PQ meters

Purpose

The Control 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 a native HTML5 web client. The Control Client can be used to perform control and/or acknowledge alarms. Control Clients allow up to 2 concurrent CtAPI connections. Customers can have a mix of Windows desktop clients and native HTML5 web clients.

NOTE: At the time of PSO 2020 release, control functionality will not be available in the web client. It will be available in a future PSO 2020 cumulative update.

Licensing options

The Control Client is licensed by number of points/tags (options include: 500, 1500, 5000, 15000 and Unlimited tags). 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 86.

Design considerations

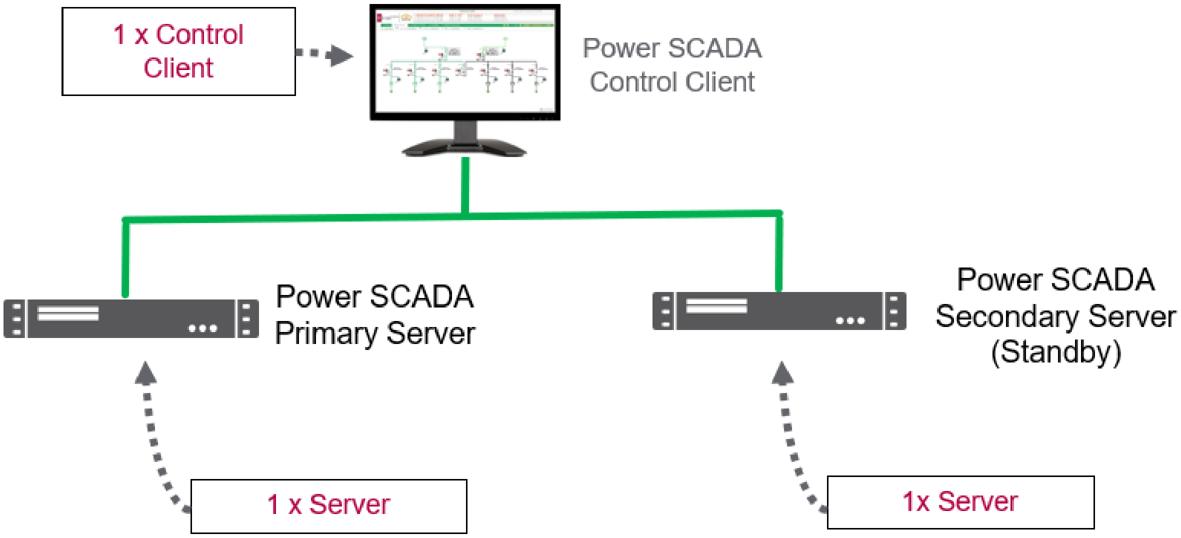
Clients can use one of the following license models:

- Floating license model: limited number of licenses can be shared between a number of concurrent users and computers .
- Static license model: license reserved for set number of computers.

Control Client component architecture

Architecture #1: Server redundancy with static Control Client

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


1 x Control Client
Power SCADA Control Client
Power SCADA Primary Server
Power SCADA Secondary Server (Standby)
1 x Server
1x Server

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

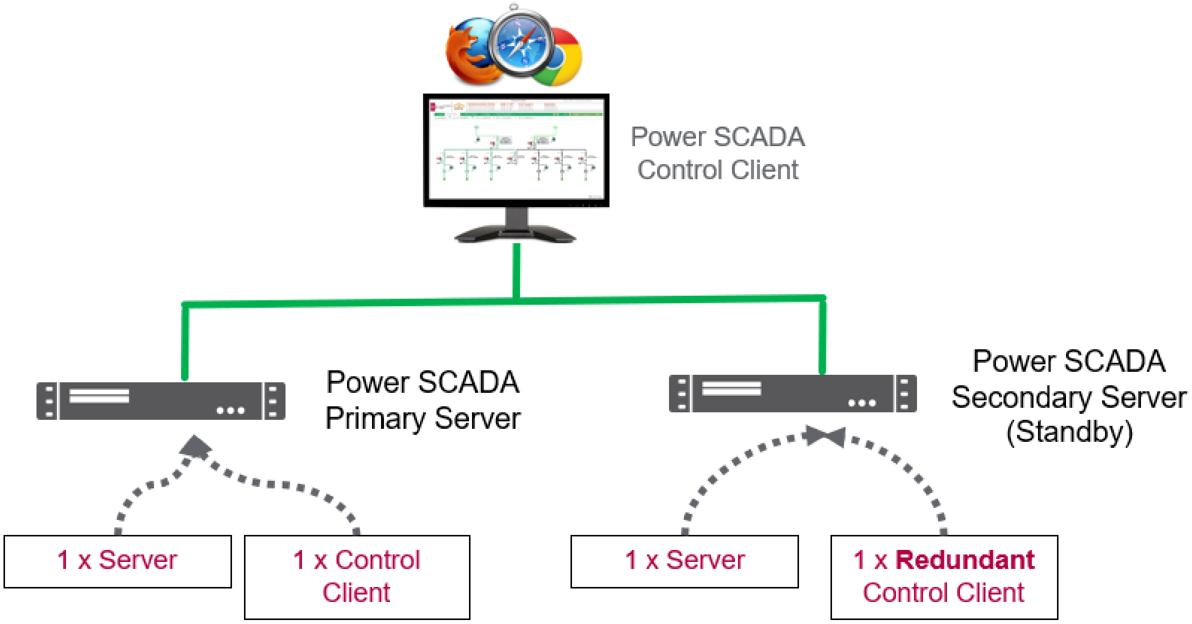
Server software and licenses are installed on the Primary and Secondary Server machines.

Control Client software and license is hosted on Client machine.

Placing the Client license on the Client machine limits the Control Client to Windows desktop application access only, instead of web client access if the static license model is 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 and Secondary Server machines.

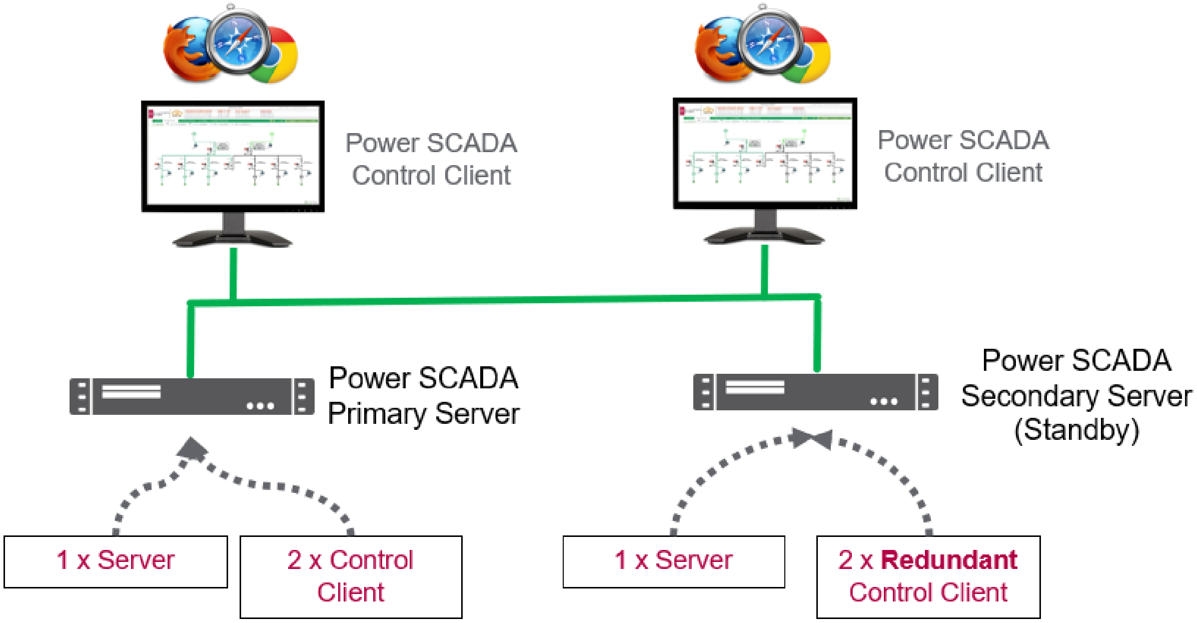
By placing Client license on Server machine (floating license model), Control Client could be accessed via native web client or Windows desktop application.

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

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

Architecture #3: Server redundancy with two floating Control Clients

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



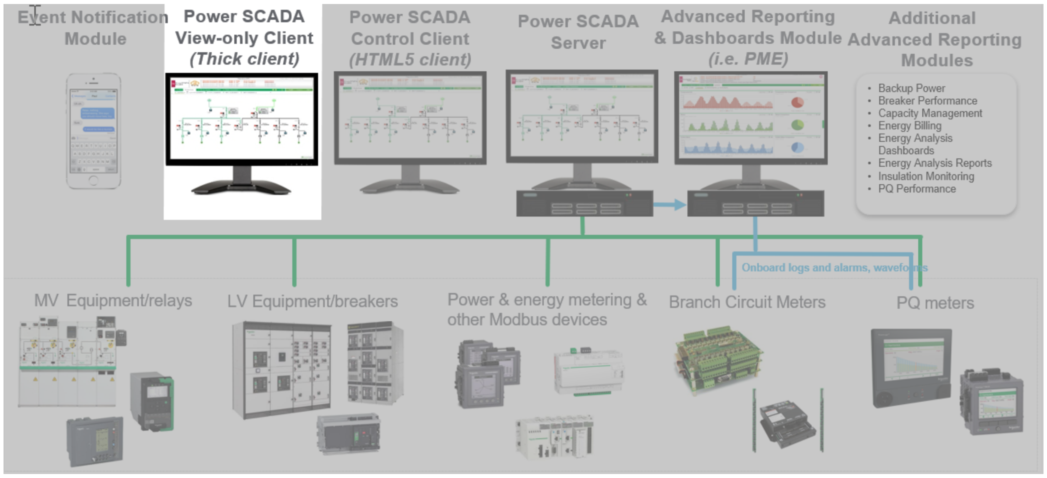
Server software and licenses are installed on the Primary and Secondary Server machines.

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 two simultaneous connections due to having two Client licenses.

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

Power SCADA View-only Client component (thick client)


Event Notification Module
Power SCADA View-only Client (Thick client)
Power SCADA Control Client (HTML5 client)
Power SCADA Server
Advanced Reporting & Dashboards Module (i.e. PME)
Additional Advanced Reporting Modules
Backup Power
Breaker Performance
Capacity Management
Energy Billing
Energy Analysis Dashboards
Energy Analysis Reports
Insulation Monitoring
PQ Performance
Onboard logs and alarms, waveforms
MV Equipment/relays
LV Equipment/breakers
Power & energy metering & other Modbus devices
Branch Circuit Meters
PQ meters

Purpose

The View-only Client is an optional component that allows operators to access the Power SCADA runtime from a machine other than the Server machine. View-only Clients can only be run as a Windows desktop application. Native web client support is not available using the View-only Client. The View-only Client can only view the runtime (no control or alarm acknowledge rights).

> **NOTE**: Customers can have a mix of Control and View-only Clients.

Licensing options

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 86.

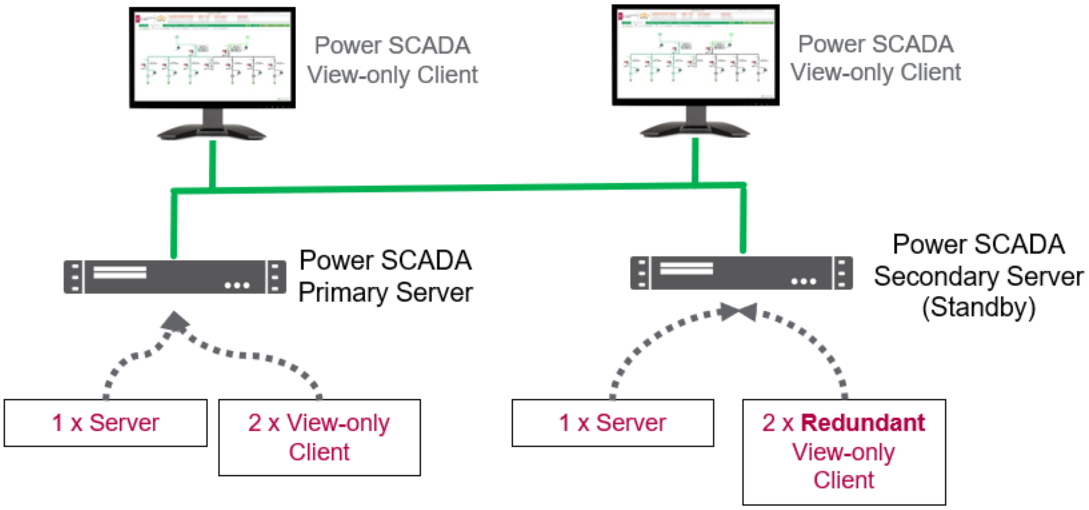
Design considerations

Clients can use one of the following license models:

- Floating license model: limited number of licenses can be shared between a number of concurrent users and computers .
- Static license model: license reserved for set number of computers.

Power SCADA View-only Client architectures

Architecture #1: Server redundancy with two floating View-only Clients



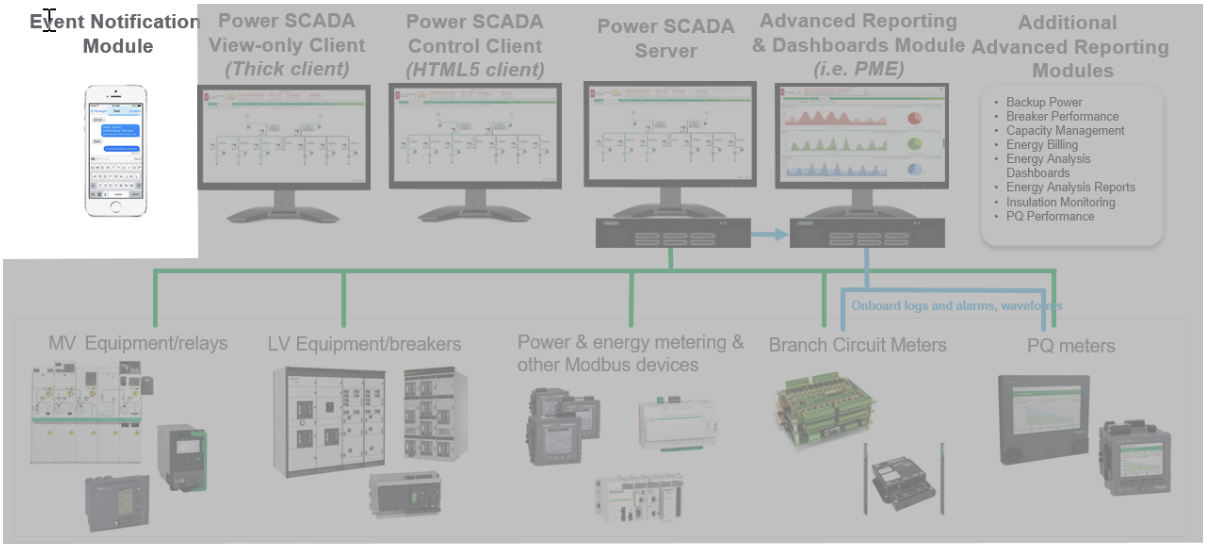
Server software and licenses are installed on the Primary and Secondary Server machines.

By placing View-only Client license on Server machine (floating license model), View-only Client could be accessed via Windows desktop application.

Client connectivity limited to two simultaneous connections due to having two Client licenses.

NOTE: The secondary Server hosts two redundant View-only Client licenses instead of the standard View-only Client license.

Event Notification Module component


Event Notification Module
Power SCADA View-only Client (Thick client)
Power SCADA Control Client (HTML5 client)
Power SCADA Server
Advanced Reporting & Dashboards Module (i.e. PME)
Additional Advanced Reporting Modules
Backup Power
Breaker Performance
Capacity Management
Energy Billing
Energy Analysis Dashboards
Energy Analysis Reports
Insulation Monitoring
PQ Performance
Onboard logs and alarms, waveforms
MV Equipment/relays
LV Equipment/breakers
Power & energy metering & other Modbus devices
Branch Circuit Meters
PQ meters

Purpose

Event Notification Module (ENM) is an optional component. ENM delivers event/alarm information from the Power SCADA Server to users via SMTP (email) or SMS (text message). It can be configured to send notifications upon specific events/alarms occurring and to specific users.

Licensing options

Single license. For systems with Server redundancy, a second ENM license is required for stand-by Server. For more information on licensing, see "Licensing" on page 86.

Design considerations

ENM configuration can be run from thick Power SCADA clients (Control and View-only) or from Power SCADA Anywhere. Configuration not available from ‘native’ web client.

For SMTP support, requires access to SMTP Server (not sold with PSO).

For SMS, factory tested serial modem models should be used. See "Factory tested serial modems" on page 57 for more information.

NOTE: As of Power SCADA Operation 9.0, ENM is no longer a separate software package with separate Windows Services, databases, etc. It is an integrated component installed with the Power SCADA Server.

Factory tested serial modems

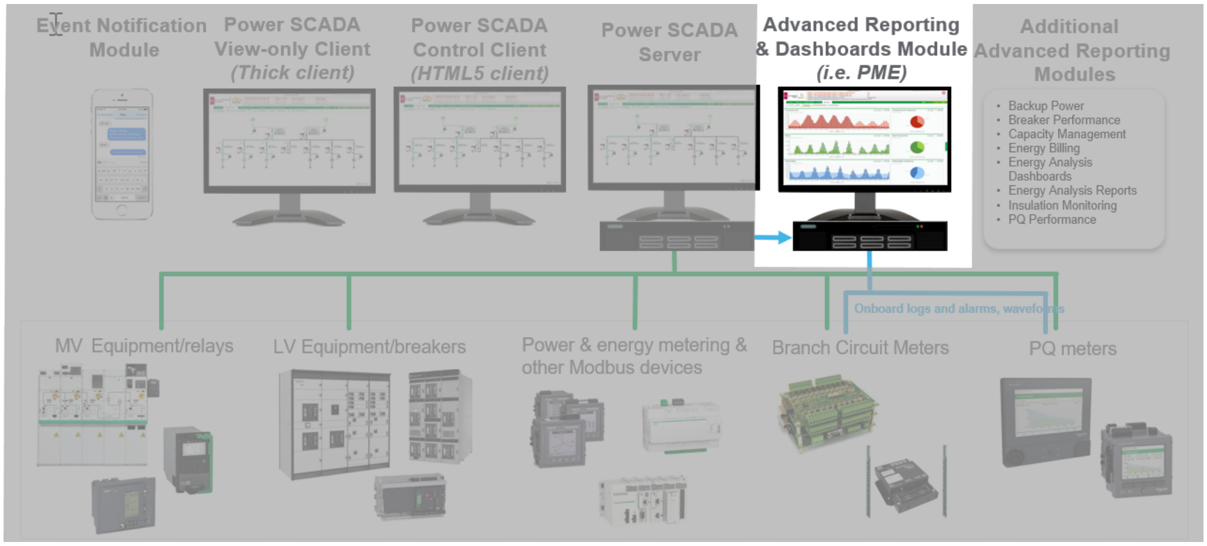
The following are factory tested SMS serial modems:

- Multi-Tech MTC-H5-B03-KIT
- Multi-Tech MTR-H5-B07-US-EU-GB
- Multi-Tech Quick Carrier USB-D USB Cellular Modem, MTD-H5

Other serial modem models may work with Event Notification Module. Teams should test other serial modem models before deploying to customer.

IP modems are not currently supported.

Advanced Reporting and Dashboards component


Event Notification Module
Power SCADA View-only Client (Thick client)
Power SCADA Control Client (HTML5 client)
Power SCADA Server
Advanced Reporting & Dashboards Module (i.e. PME)
Additional Advanced Reporting Modules
Backup Power
Breaker Performance
Capacity Management
Energy Billing
Energy Analysis Dashboards
Energy Analysis Reports
Insulation Monitoring
PQ Performance
Onboard logs and alarms, waveforms
MV Equipment/relays
LV Equipment/breakers
Power & energy metering & other Modbus devices
Branch Circuit Meters
PQ meters

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 86.

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

Design considerations

Advanced Reporting and Dashboards (PME) component does not support redundancy.

Redundancy of the historical data acquisition can largely be achieved using PSO trending capabilities along with using an ETL (Extract Transform Load) task to pass this PSO trend data to the PME databases.

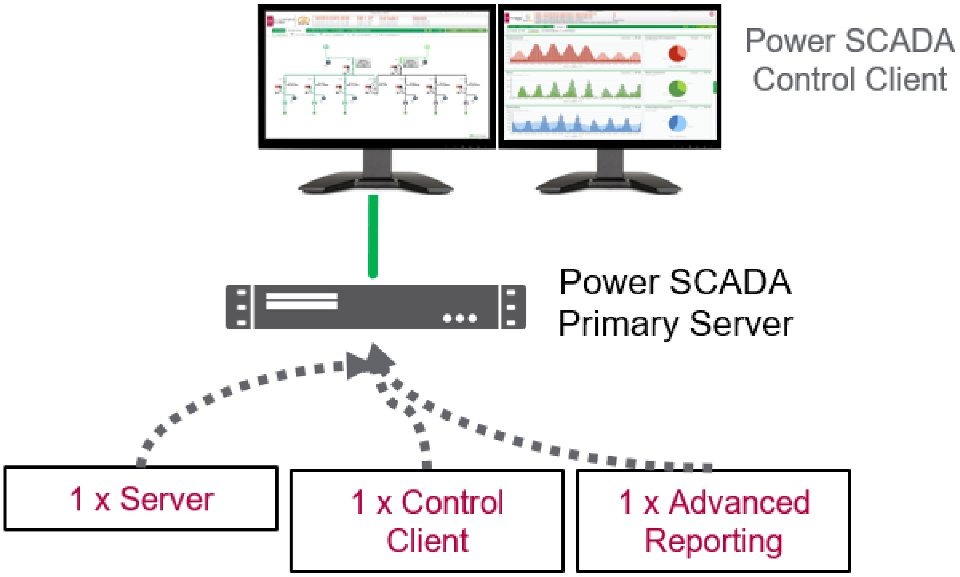
See "Integrating with Advanced Reporting and Dashboards" on page 91.

Advanced Reporting and Dashboards architectures

Architecture #1: Simple system without redundancy

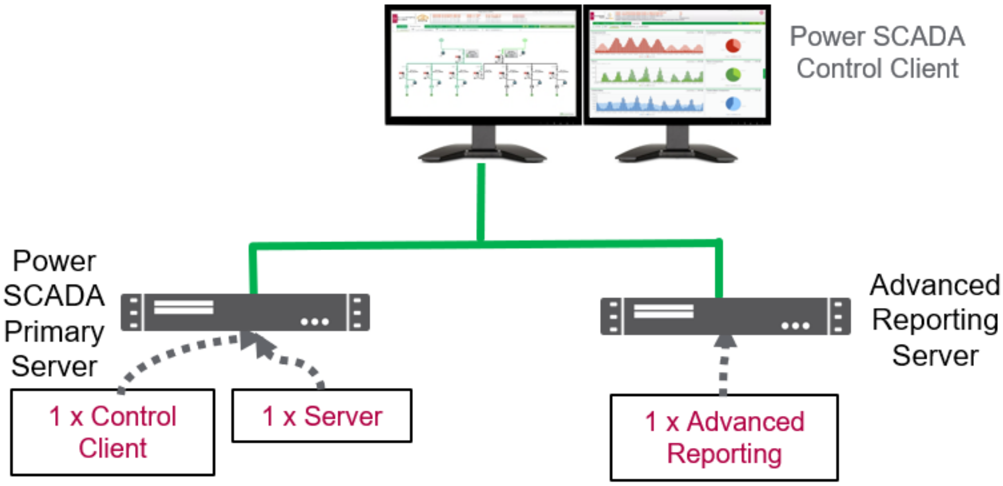
The following example architecture illustrates the Advanced Reporting & Dashboards Module in a system with a single Power SCADA Server.

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



Architecture #2: Large system without redundancy

The following example architecture illustrates the Advanced Reporting and Dashboards Module installed on a separate server from the Power SCADA Server.

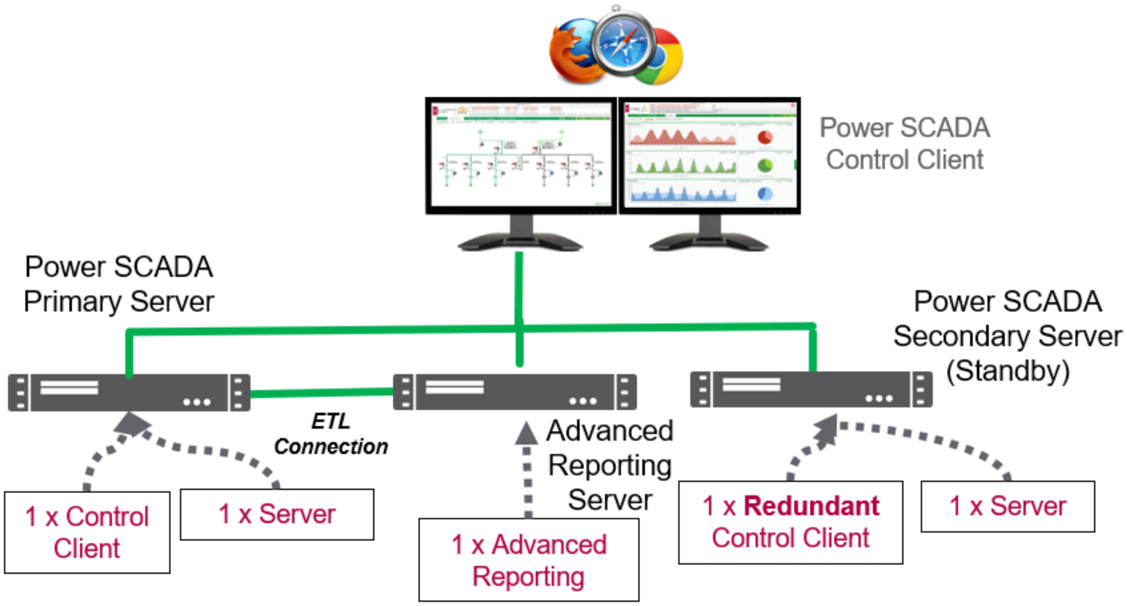

Power SCADA Control Client
Power SCADA Primary Server
1 x Control Client
1 x Server
Advanced Reporting Server
1 x Advanced Reporting

This architecture is typically used for performance reasons. Systems with over 150,000 tags or 600 devices should have Advanced Reporting and PSO Server on separate machines.

Architecture #3: Advanced Reporting with Server redundancy

NOTE: This is the recommended Advanced Reporting architecture.

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



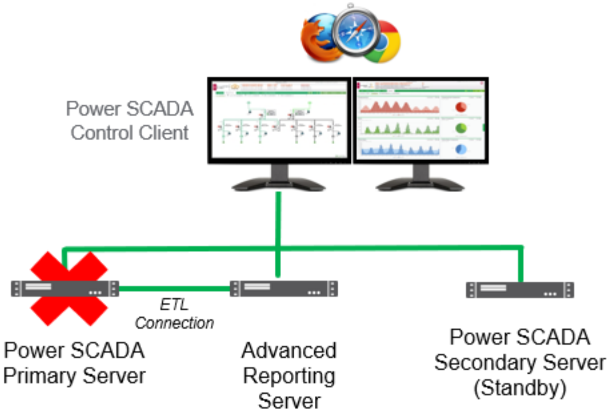
The Advanced Reporting Server contains both the Advanced Reporting software (PME) and the software key.

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

NOTE: 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.

Redundancy Scenario: ETL used for Server redundancy

The following image shows an example where the Primary Server becomes inoperable and the Advanced Reporting ETL is not reconfigured to point to the Secondary Server. In our example, the Primary Server becomes inoperable on June 1 and is restarted on June 3.



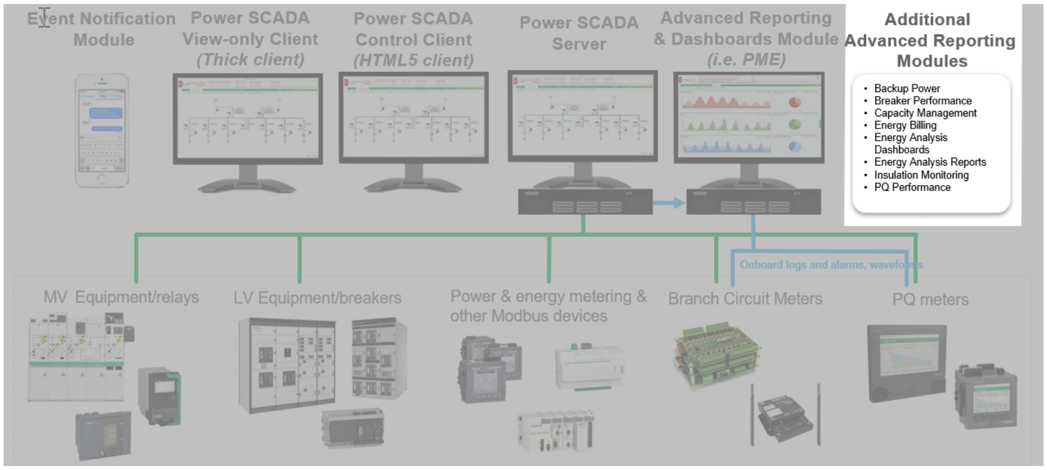
The Secondary Server has taken over the alarming and one-line diagram visualization. Since Advanced Reporting is still running with reports, dashboards, and WebReach diagrams, the functionality of Power SCADA+PME would largely remain active from June 1 to 3.

However, when running reports during June 1-3 while the Primary Server is down, reports and dashboard data would NOT be present for this time period. Data previous to June 1 would be present.

Once the Primary Server is recovered on June 3, the Secondary Server will fill the Primary Sever with the missed trend and historical data.

The Advanced Reporting ETL would start pulling data from the Power SCADA's Primary trend file system. Depending on system size, this June 1-3 data would eventually be available in the reports and dashboards.

Additional Advanced Reporting Modules component


Event Notification Module
Power SCADA View-only Client (Thick client)
Power SCADA Control Client (HTML5 client)
Power SCADA Server
Advanced Reporting & Dashboards Module (i.e. PME)
Additional Advanced Reporting Modules
Backup Power
Breaker Performance
Capacity Management
Energy Billing
Energy Analysis Dashboards
Energy Analysis Reports
Insulation Monitoring
PQ Performance
Onboard logs and alarms, waveforms
MV Equipment/relays
LV Equipment/breakers
Power & energy metering & other Modbus devices
Branch Circuit Meters
PQ meters

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 one Advanced Reporting license. For more information on licensing, see "Licensing" on page 86.

Design considerations

See "Integrating with Advanced Reporting and Dashboards" on page 91 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

EcoStruxure Power application	Advanced Reporting 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 2020 – System Guide* contains detailed information on how to configure the Advanced Reporting modules.

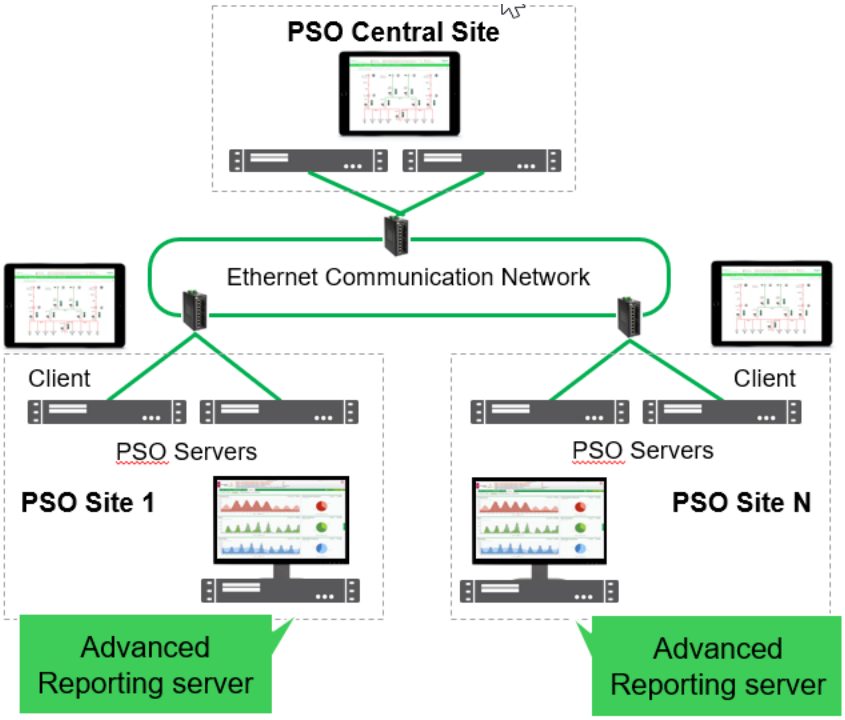
Advanced Reporting module descriptions

The following table provides short descriptions for the Advanced Reporting software modules.

Category	Advanced Reporting module	Description
Efficiency and Reliability	Energy Analysis Reports Module	Improve Operational Efficiency, Energy Performance and help achieve ISO50001 compliance.
	Energy Analysis Dashboards Module	Advanced analysis and visualization gadgets. Sankey, heatmap/carpet, pareto and ranking.
	Energy Billing Module	Flexible rate engine and reports for cost allocation, bill verification and tenant billing.
Reliability and Safety	Insulation Monitoring Module	Monitor insulation levels for power Isolated panels (IEC and ANSI).
	Capacity Management Module	Monitor the capacity loading of electrical equipment (UPS, Generators, multi-circuits).
	PQ Performance Module	Simple, global overview of the impact of power quality on your facility’s operations.
Asset Compliance and Reliability	Breaker Performance Module	Breaker status diagrams and reports including electrical ageing and mechanical wear, for proactive maintenance.
	Backup Power Module	Monitor the parameters of your generator, ATSs and UPSs. Automated results for emergency power supply systems.

Multi-site architectures

A multi-site architecture (or multi-clustered system) allows you to scale your system as your needs evolve. It gives you the ability to monitor multiple systems from a central location. You can roll up data, graphics, and controls under a central HMI and you can add servers and clusters of servers to expand or distribute systems.


PSO Central Site
Ethernet Communication Network
Client
PSO Servers
PSO Site 1
Advanced Reporting server
Client
PSO Servers
PSO Site N
Advanced Reporting server

Monitor and control multiple independent systems from single runtime client for geographically co-located customers.

System organized into separate sites (also known as clusters).

Each site is controlled by local operators and supported by local redundant PSO servers.

From central control site, one can simultaneously manage all the sites by viewing ‘federated’ data from multiple PSO servers.

Licensing options

When using native web clients, you need to have a full PSO Server license at Central Site.

When using thick clients, you need to have a Control Client or View-only Client license at Central Site.

For more information on licensing, see "Licensing" on page 86.

Design considerations

Geographically distributed systems

Power SCADA requires a constant, high bandwidth (for example: Ethernet), and reliable connection.

We recommend against doing real-time control from the central control room in this architecture without having first performed a Final Acceptance Test (FAT) prior to customer hand-off.

NOTE: Ensure a stable communication between PSO clusters and connected products that is always-connected and has sufficient bandwidth.

Devices in multiple time zones

Power SCADA Servers can contain different devices that can be distributed across several sites or time zones.

Instead of attempting to connect devices directly via a remote connection, a PSO Server 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 between PSO clusters and connected products that is always-connected and has sufficient bandwidth.

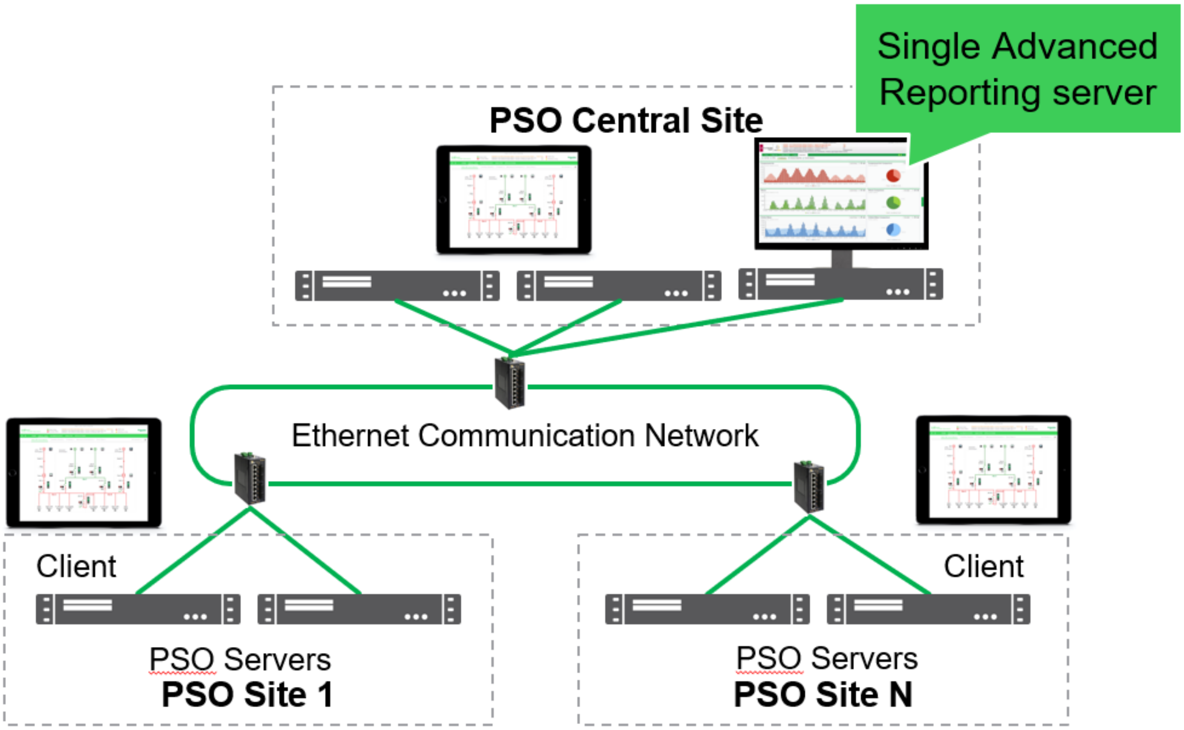
Maintaining/upgrading the system

All PSO sites must be using the same version of the PSO software for communication between systems to occur.

For example Central Site cannot be running PSO 9.0 with underlying sites running PSO 2020.

Advanced Reporting and software modules

Architecture option #1 - Single Advanced Reporting server

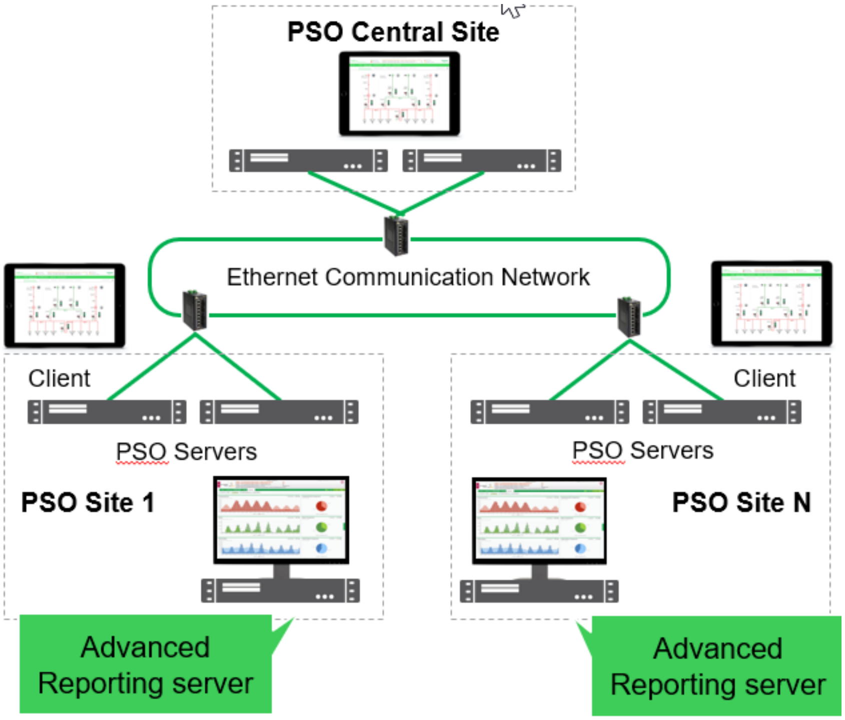


Install Advanced Reporting at central site and connect directly to devices located at individual sites.

Allows for central historical reporting across all sites.

Ensure in this architecture that you do not exceed device limits on Advanced Reporting.

Architecture option #2 - Advanced Reporting at each individual site


PSO Central Site
Ethernet Communication Network
Client
PSO Servers
PSO Site 1
Client
PSO Servers
PSO Site N
Advanced Reporting server
Advanced Reporting server

Install Advanced Reporting at each individual site.

Architecture would be considered if overall system size was beyond scale of single Advanced Reporting server.

Connected devices and protocols

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

Power SCADA Operation 2020 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 2020 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 that are compatible with Power SCADA, see the **Connectivity Hub** page in the AVEVA Knowledge & Support Center (requires login.)

Power SCADA Tool Suite

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 built 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.

Deploy projects faster with the Power SCADA Tool Suite, including tools that are unique to Power SCADA and critical to project success.

The suite includes:

- Simple SCADA Project Setup
- Centralized SCADA Project Deployment
- Power SCADA Studio
- Power Device management
- IO Devices Comms Optimization

The following Power SCADA features are supported only using Power SCADA tools and workflows:

- HTML5 native graphics, alarms, and waveforms
- Event Notification Module
- Interoperability with Advanced Reporting (specifically ETL for PME)
- Interoperability with EBO (specifically PSO EWS implementation for EBO)
- LiveView
- Basic Reports
- One-line configuration
- Power SCADA power graphics libraries (genies)
- I/O Device Manager (both UI and Excel)
- The ION and Power Modbus drivers are complex to setup without the use of Power SCADA tooling

Waveform file share access and permissions

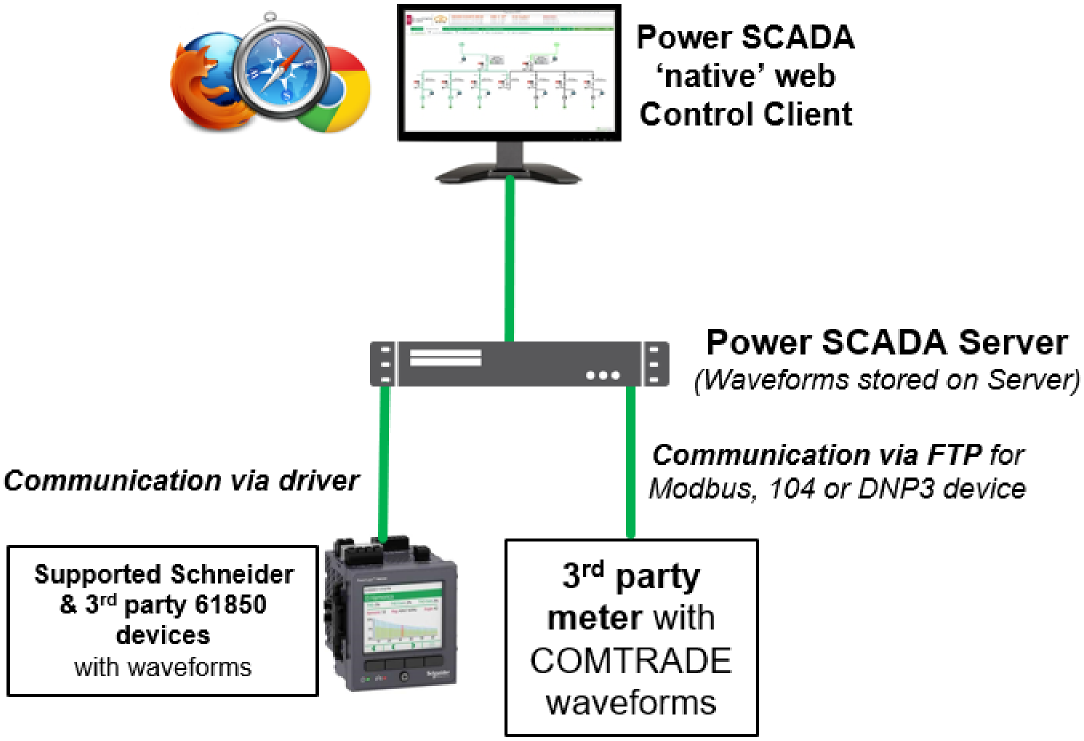
Waveforms are stored in a file share repository on the Power SCADA Server.

The following waveform file share permissions are required:

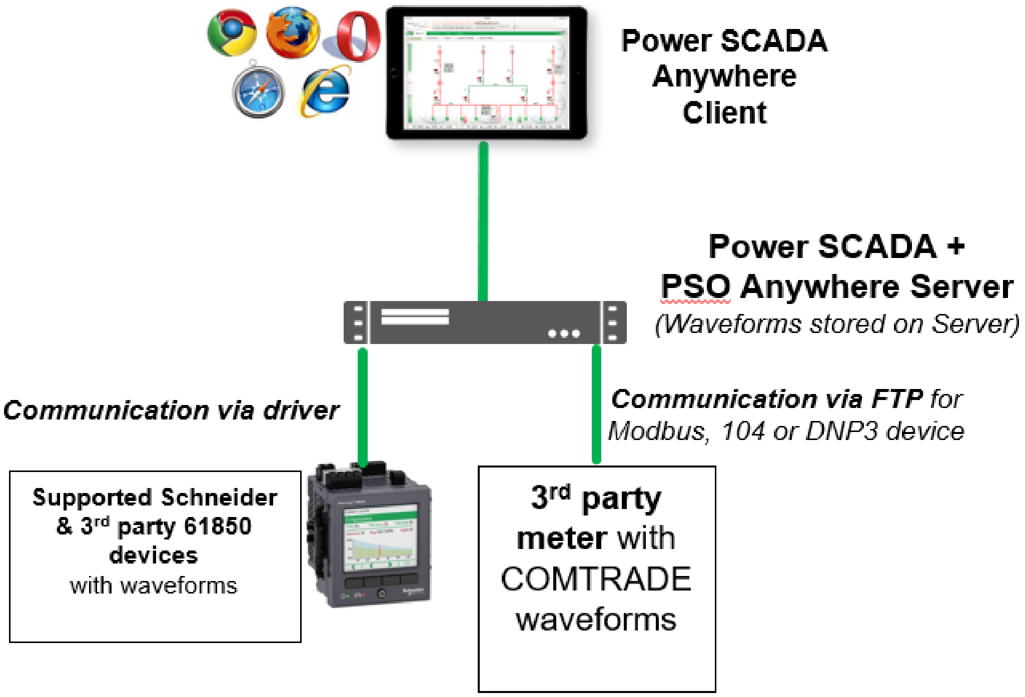
1. The account running Citect on Power SCADA Server requires Full permission (Read/Write/Modify)
2. Windows user accounts that are used to log-into remote machines using ‘native’ web clients require Read Only permission.
3. Windows user accounts must be linked to Power SCADA user roles that allow Remote 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



Supported power devices

Native and other supported devices

Power SCADA offers native support for many Schneider Electric devices to simplify the set up and acquire power event data types:

- Power event waveforms (sag, swell, transients, etc)
- Onboard alarms and time-stamps, events, and data logs

Examples of Schneider Electric devices natively supported:

- Protective relays: Sepam series 20, 40, 80, 2000
- LV breakers: Micrologic series including Masterpact MTZ
- Advanced Power Quality meters: ION8650, ION7650, ION7550, PM8000
- Sequence of Events recorders: Cyber Sciences SER2408 & SER3200
- Power & energy meters: PM5000 series, iEM3000, etc.
- Branch circuit monitors: BCPM
- Motor management: Tesys T
- PLC: Quantum, Momentum, Premium, M340

For complete details on the devices that Power SCADA supports, refer to the Power SCADA Device Support Matrix spreadsheet.

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.

IEC-61850 file formats supported

When integrating with 61850 IEDs:

- PSO 2020 supports integrating with IEC-61850 Edition #1 and Edition #2 IEDs via import of
  - CID files or
  - ICD files

PSO 2020 does not support consuming SCD files for integrating with IEC-61850 systems.

3rd party device support resources

Browse to 3rd Party Device Support (Box link) for information on how to integrate the following devices with Power SCADA Operation:

- 3rd party Modbus devices
- 3rd party IEC-61850 devices
- DNP3 devices
- SNMP v.2 devices
- Cyber Sciences devices

Citect drivers

Citect drivers can be used with Power SCADA Operation. For a complete list of Citect drivers that are compatible with Power SCADA, see the **Connectivity Hub** page in the AVEVA Knowledge & Support Center (requires login.)

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.

Computer 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 following table to find the content you are looking for:

Link	Description
"Server CPU and RAM requirements" on page 71	Server CPU and RAM recommendations for various system architectures
"Client CPU, RAM, and disc requirements" on page 73	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 74	Required disk space without Advanced Reporting and calculating disk storage size
"Network requirements" on page 75	Network requirements for Ethernet and serial communications
"Supported operating systems" on page 75	Supported Operating Systems
"Supported browsers" on page 76	Supported browsers for web clients, Power SCADA Anywhere
"Supported SQL Server versions" on page 77	Supported Microsoft SQL Server versions for the Advanced Reporting and Dashboards Module
"Virtualization" on page 77	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 server (medium and large systems)

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 two numbers. For example, if you have a system using 120,000 tags with 300 devices, use six CPU cores and 12 GB of RAM.

NOTE: These are minimum requirements. We recommend that you consider doubling the RAM requirements listed.

Use the larger figure below	CPU PassMark Score	# CPU Cores	RAM (GB)
1,500 tags or 50 devices	1,800	1	4
15,000 tags or 100 devices	4,500	4	8
50,000 tags or 200 devices	8,000	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	10,000	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.

NOTE: These are minimum requirements. We recommend that you consider doubling the RAM requirements listed.

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 2020 – 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 and/or View Clients used as Windows desktop thick clients have the following minimum requirements:

- CPU PassMark: 2000
- 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

Minimum requirements:

- DirectX 9 or later with WDDM 1.0 Driver
- 128 MB of dedicated VRAM (for systems of any size)

Server disk storage

The main consumers of historical data in PSO are:

1. Advanced Reporting and Dashboards Module (PME) historical data for display in reports and dashboards.
2. PSO historical data stored for PSO alarm viewer, trend viewer, and native basic reports.

Advanced Reporting and Dashboards data is stored in Microsoft SQL databases. PSO data is stored in file system flat files (no SQL required).

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 two 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) used by PSO native reports and trend viewer. 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 your 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

Network requirements

Use Ethernet whenever possible. For best system performance with devices, we recommend minimum 1 Gb Ethernet communication.

If you are using serial communication, use a minimum baud rate of 19.2K.

Supported operating systems

The following table lists the compatible operating systems for Power SCADA Operation, ENM, and Advanced Reporting. Columns for version 2020, 9.0, and 8.2 represent super-set of all PSO components including Servers, Clients, and Advanced Reporting and Dashboards (PME).

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

Operating System	Power SCADA Operation Version					
	2020	9.0	8.2	8.1	8.0	7.40
Windows Server 2019	✓	–	–	–	–	–
Windows Server 2016	✓	✓	✓	–	–	–
Windows 10	✓ [4]	✓	✓	✓ [1]	✓ [2]	–

Operating System	Power SCADA Operation Version					
	2020	9.0	8.2	8.1	8.0	7.40
Windows Server 2012 R2	✓	✓	✓	✓	✓[3]	–
Windows 8.1		–	✓	✓	✓[3]	–
Windows Server 2012		✓	✓	✓	✓	✓
Windows Server 2008 R2		–	✓	✓	✓	✓
Windows 7		✓	✓	✓	✓	✓
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: Requires Windows 10 version 1607 and later (64-bit only)

Supported browsers

Web client access

Power SCADA has two 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:

	Chrome	Edge	Firefox
Power SCADA graphics pages including one-line diagram/engine using EcoStruxure graphics	Yes	Yes	Yes
PSO 2020 Alarms and Trends	Yes	Yes	Yes
PME reports and dashboards	Yes	Yes	Yes
Power SCADA Basic Reports	Yes	Yes	Yes
LiveView (Power SCADA Real Time Tables implementation)	Yes	Yes	Yes

NOTE: Graphics pages build using ‘classic’ Citect graphics will only run on Windows Desktop clients

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 2020 **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 2017 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

	Microsoft Hyper-V	VMWare vSphere
Power SCADA Anywhere (this refers to machine hosting Power SCADA Anywhere)	Yes	Yes
Advanced Reporting and Dashboards (Power Monitoring Expert)	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 2020 – 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.
- If you are using shared drive storage, use Fiber SAN storage. If you are not using Fiber SAN storage, use a direct attached, dedicated hard drive used by Power SCADA only.
- You must have a fixed-size disk virtual machine.
- Set host (for example: ESX host) power management to “High Performance”.

Additional virtual machine configuration guidelines vary by hypervisor.

Licensing considerations

When making changes to virtual machines (VM) (cloning a VM, moving a VM to another physical machine, changes to VM name, etc.), Power SCADA software key licenses will go untrusted and cause the Power SCADA software to shut down.

For complete details on changes that can cause software keys to go untrusted refer to the FlexNet whitepaper FlexNet Publisher White Paper: Understanding Virtualization Features in FlexNet Publisher.

When software licenses go untrusted, customers can call Schneider Electric tech support to reach Software Registration Centers (SRCs). SRCS can help reactivate untrusted software key licenses and restart Power SCADA with trusted licenses.

To avoid interruption due to the reactivation process, we recommend the use of USB dongles used with USB redirection software such as Digi AnywhereUSB® Plus.

Cybersecurity

This section provides information on how to help secure your system during the Planning phase.

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

Topic	Content
"IEC 62443" on page 79	About Power SCADA IEC 62443 compliance.

Topic	Content
"Encryption" on page 79	About for encryption between Power SCADA components.
"Encryption" on page 79	Planning for application whitelisting with McAfee Application Control.
"Windows Active Directory" on page 82	Planning for user access using Windows Active Directory.
"Two-factor authentication" on page 83	Planning for user access using two-factor authentication (YubiKey).
"Awareness and education" on page 84	Resources for understanding cyber threats and how to help protect against them.

IEC 62443

IEC 62443 is an international cybersecurity Operational Technology (OT) standard with various levels of robustness against Cyber threats.

Power SCADA Operation is SL1 certified to comply with IEC 62443 standard at the component level:

- IEC 62443-4.1: Assess a supplier’s product development lifecycle for industrial automation and control systems (IACS)
- IEC 62443-4.2: Defines the security requirements for components of an IACS

This SL1 certification involved process and product changes to achieve certification:

- Process: Secure Development Lifecycle (SDL). Cybersecurity staff training, requirements specification, design reviews, threat modeling, static code analysis, supply chain, penetration testing, etc.
- Product: Various levels of encryption, multi-factor authentication, role-based access control (RBAC), Active Directory support, object level security, audit logs, auto logoff mechanisms, whitelisting, digitally signed files, etc.

Encryption

PSO supports the ability to encrypt communication between PSO components via latest Transport Layer Security (TLS) version.

Communication is encrypted between:

- Server(s) and client(s)
- Server to server

NOTE: It is recommended that your system is configured to use the latest version of TLS, at least version 1.2.

Application whitelisting

Zero Day cybersecurity attacks take place before a software vendor is aware of a cybersecurity exploit. This means 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. Application whitelisting proactively blocks unauthorized executable files on the PSO Server than are not part of the whitelist, such as executable files, java apps, Active X controls, and scripts.

Power SCADA has been validated with the McAfee Application Control whitelisting application.

Whitelisting Design Considerations

- Power SCADA Servers, Control Clients, View-only Clients, and Advanced Reporting have been validated using McAfee Application Control.
- McAfee Whitelisting product documentation can be found on the McAfee Web site.

Configuring a System Management Server

Power SCADA needs post-installation configuration to use encrypted communications. Only one of the machines in the network can be identified and configured as the System Management Server.

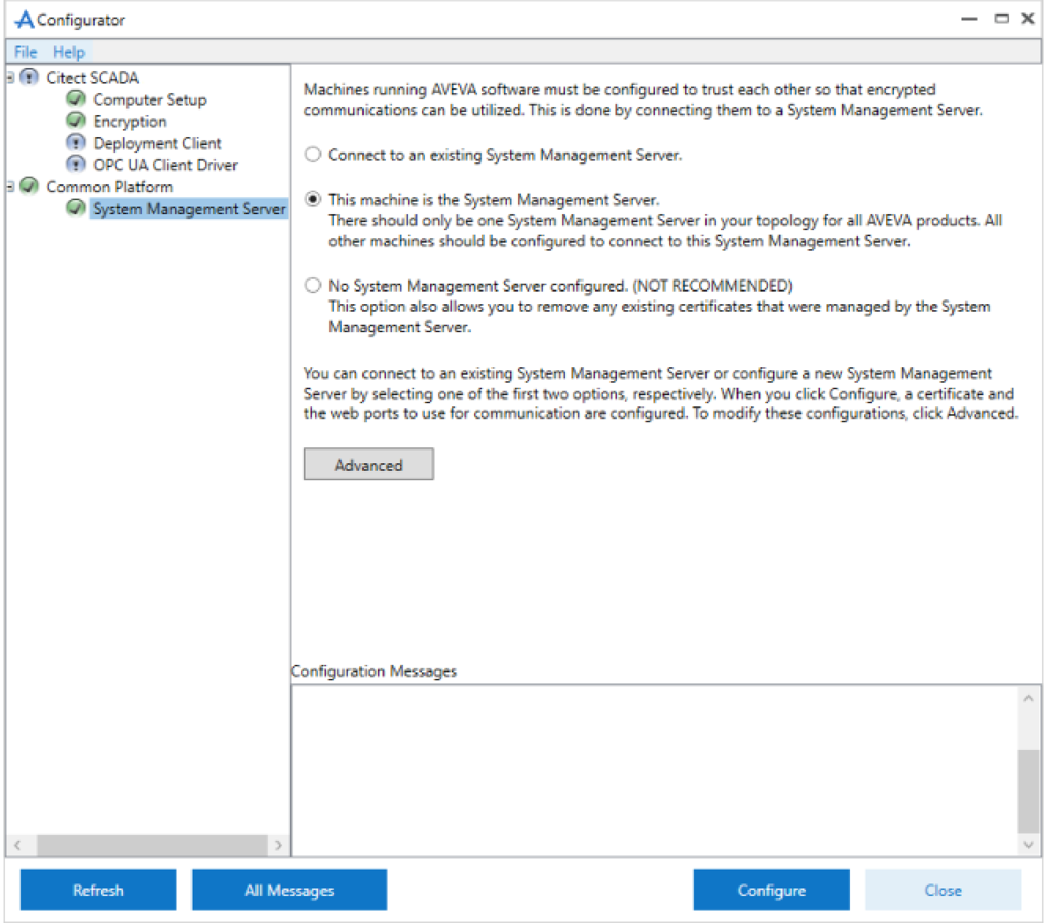
Use the Configurator to establish a trust relationship between one or more machines running Power SCADA. This configuration allows for encrypted communication between these machines, which is achieved through a common System Management Server on which a certificate is created and used to encrypt communications. Certificates may be generated automatically on the System Management Server or provided by the IT department.

NOTE: To connect to the System Management Server, you need to be a member of either the “aaAdministrators” or the “Administrators” group on the machine where the System Management Server is installed.

To configure the System Management Server:

1. Start the Configurator.

2. In the left pane, click **Common Platform** > **System Management Server**. The following is displayed:



Machines running AVEVA software must be configured to trust each other so that encrypted communications can be utilized. This is done by connecting them to a System Management Server.

○ Connect to an existing System Management Server.

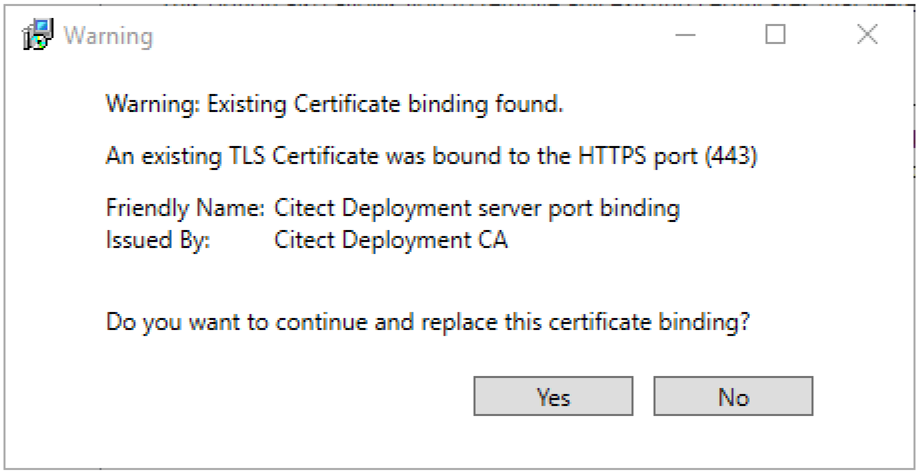
● This machine is the System Management Server.
There should only be one System Management Server in your topology for all AVEVA products. All other machines should be configured to connect to this System Management Server.

○ No System Management Server configured. (NOT RECOMMENDED)
This option also allows you to remove any existing certificates that were managed by the System Management Server.

You can connect to an existing System Management Server or configure a new System Management Server by selecting one of the first two options, respectively. When you click Configure, a certificate and the web ports to use for communication are configured. To modify these configurations, click Advanced.

3. Select **This machine is the System Management Server**. Review the notes on the screen before you start the configuration.

4. Click **Configure**. If an existing binding is found for the specified ports, the following message is displayed:



Warning: Existing Certificate binding found.

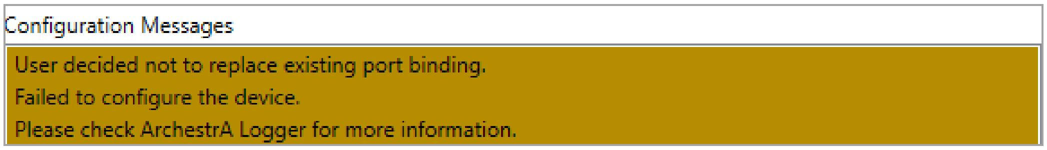
An existing TLS Certificate was bound to the HTTPS port (443)

Friendly Name: Citect Deployment server port binding
Issued By: Citect Deployment CA

Do you want to continue and replace this certificate binding?

5. Click **Yes** if you wish to replace the binding. The Configurator will start configuring the System Management Server.

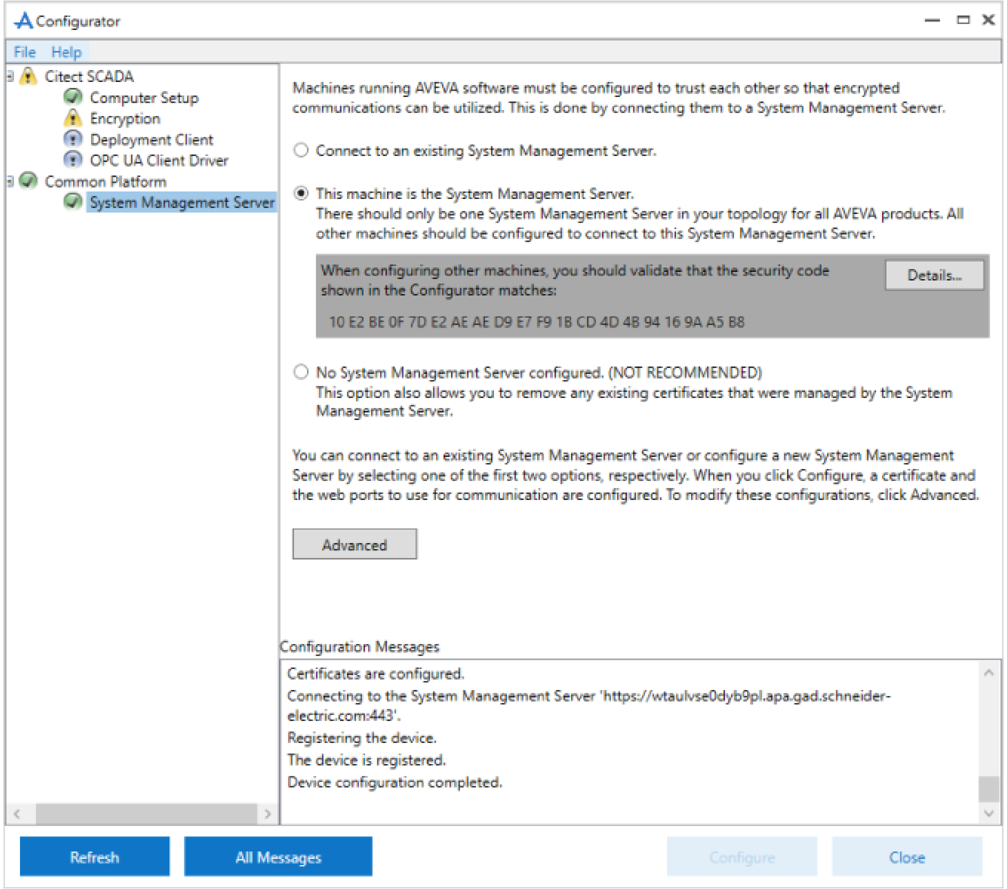
If you click **No**, the following message will be displayed in the Configuration Messages area:



Configuration Messages

User decided not to replace existing port binding.
Failed to configure the device.
Please check ArchestrA Logger for more information.

6. On successful configuration, the message “Device configuration completed” is displayed. The security code is displayed in the Configurator as shown below. To view more information about the certificate, click **Details**.



Machines running AVEVA software must be configured to trust each other so that encrypted communications can be utilized. This is done by connecting them to a System Management Server.

○ Connect to an existing System Management Server.

● This machine is the System Management Server.
There should only be one System Management Server in your topology for all AVEVA products. All other machines should be configured to connect to this System Management Server.

When configuring other machines, you should validate that the security code shown in the Configurator matches:

10 E2 BE 0F 7D E2 AE AE D9 E7 F9 1B CD 4D 4B 94 16 9A A5 B8

○ No System Management Server configured. (NOT RECOMMENDED)
This option also allows you to remove any existing certificates that were managed by the System Management Server.

You can connect to an existing System Management Server or configure a new System Management Server by selecting one of the first two options, respectively. When you click Configure, a certificate and the web ports to use for communication are configured. To modify these configurations, click Advanced.

Configuration Messages

```
Certificates are configured.
Connecting to the System Management Server 'https://wtaulvse0dyb9pl.apa.gad.schneider-
electric.com:443'.
Registering the device.
The device is registered.
Device configuration completed.
```

7. If the configuration is unsuccessful, check the ArchestrA Logger. You can access this by typing **\Program files (x86)\common files\archestra\aaLogviewer.exe** at the Windows command prompt. Alternatively, view details of the errors in the System Management Console. For more details, refer to the ArchestrA documentation.

8. Click **Close** to exit the Configurator.

Windows Active Directory

Power SCADA supports Windows Active Directory integration, including enforcement of minimal password complexity, password expiration, role based access control, etc. There are eight levels of user privileges (HMI user partitioning) and HMI user event monitoring (login/logout, shut down, control).

Power SCADA components including Servers, Control Clients, View-only Clients and mobile notifications support both user management using Windows Active Directory groups and local users.

Using Windows Active Directory, we can support various password management strategies (enable password update schedules, minimum password complexity, etc.)

NOTE: For cybersecurity purposes, we recommend that you use Windows Active Directory with strong password policy enforced 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.

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, and Power SCADA Anywhere.

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.

YubiKey 5 and FIPS models are compatible and supported with Power SCADA thick control clients.

NOTE: YubiKey NOT supported with new PSO 2020 native web client.

YubiKey models validated with “FIPS-compliant” enabled on Windows Server.

Supported YubiKey models compatible with Power SCADA Operation:

Model Number	Comments
YubiKey 5	NFC model not supported .
YubiKey 5 C FIPS	Meets AAL3 of NIST SP800-63B guidelines.

See https://www.yubico.com for more information.

Awareness and education

NOTE: Schneider Electric cyber security notifications can be found on the Schneider Electric Cybersecurity Support Portal.

Knowledge is a key step to prevent cyber intrusion.

Resources for understanding the cyber threats and how to help protect against them:

- White papers on securing critical systems. See Schneider Electric whitepaper Securing Power Systems.
- Awareness & education of social engineering techniques. See Wikipedia - Social engineering.
- Schneider Electric cybersecurity support & notifications. See Schneider Electric Cyber-security website.

Localization and translation

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

Power SCADA Component	Translatable	Comments
Power SCADA thick runtime, including Event Notification Module configuration as of 9.0	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	Can be done by translating RESX files. (Contact factory for additional detailed steps to support this.)
Power SCADA web client runtime	No	Available in the following languages: • English • French
Power SCADA engineering tools	No	Available in the following languages: • English • French (excluding Graphics Editor)
Power SCADA System Guide documentation	No	Available in the following languages: • English • French (excluding Graphics Editor) • Simplified Chinese (excluding Graphics Editor)

Power SCADA Component	Translatable	Comments
Advanced Reporting & additional Advanced Reporting modules	n/a	Available in the following languages: • English • Spanish • Simplified Chinese • French • German • Italian • Polish • Czech (excluding online help) • Traditional Chinese • Russian • Swedish • Portuguese

Licensing

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

The following license options are available:

1. Electronic software key (default licensing option when ordering)

   When running Power SCADA in virtual environments, we recommend software keys. Software keys are delivered via email and activated using the Floating License Manager software.

2. 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 89.

USB license dongle

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

- The USB license dongle is using SafeNet Sentinel HL Pro hardware.
- To learn more about the Sentinel HL Pro USB dongles refer to the vendor’s website here: https://sentinel.gemalto.com/software-monetization/sentinel-hl-pro/

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 (control or view-only)	Yes	Yes	Floating license model: On machine where Primary Server is installed Static license model: On machine where Client is installed
Redundant Client (when server redundancy is used)	Yes	Yes	On machine where Stand-by Server is installed
Event Notification Module	Yes	Yes	On machine where 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

The Advanced Reporting and Dashboards Module is included on the Power SCADA Operation 2020 with Advanced Reporting and Dashboards DVD and can be purchased as a single commercial reference with Power SCADA Operation.

The 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:

- Eight 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.
- If you prefer a USB dongle key option for development keys, you will also need to order:
  - 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 license (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 PSA104113: Event Notification Module
- 1 x PSA104112: Advanced Reporting and Dashboards Module
- 1 x PSA104114: Energy Billing Module
- 1 x PSA104115: Breaker Performance Module
- 1 x PSA104116: Energy Analysis Reports Module
- 1 x PSA104121: Capacity Management Module
- 1 x PSA104124: Power Quality Performance Module
- 1 x PSA104125: Insulation Monitoring Module
- 1 x PSA104126: Backup Power Module

NOTE: Energy Analysis Reports 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 needs to be ordered for each component whose license is being transferred.

Licensing when using virtual machines

For licensing considerations when using virtual machines, see "Licensing considerations" on page 78.

Commercial name to license key version mapping

PSO key version information displayed in the PSO software using the CiUSAFE tool for USB licenses or using the License Manager for software keys are NOT necessarily the same as the PSO commercial version names.

For those with Citect SCADA experience, this PSO key version information does not necessarily align with the respective 'native' Citect SCADA key versions. (for example, Citect 2015 used key version 7.50)

Up until v8.0, all PSO key version numbers aligned with the PSO commercial name/number and the 'native' Citect SCADA key versions. Starting with v8.0, PSO aligns commercial version names/numbers with PME instead of Citect.

PSO commercial name/number	PSO key version	Citect SCADA commercial name/number	Comments
PSO 2022	8.2	Citect SCADA 2018 R2	
PSO 9.0	8.1	Citect 2018	
PSE 8.2	8.0	Citect 2016	
PSE 8.1	8.0	Citect 2015	
PSE 8.0 SR1	8.0	Citect 2015	PSO key versions were 'shipped as' 7.5 but all licenses were re-programmed in License Portal as 8.0 in mid-2016 to enable free upgrade to PSE 8.1.
PSE 8.0	8.0	Citect 7.40	PSO key versions were 'shipped as' 7.5 but all licenses were re-programmed in License Portal as 8.0 in mid-2016 to enable free upgrade to PSE 8.1.
PSE 7.40	7.40	Citect 7.40	

Commercial references

NOTE: There are no part changes between PSO 2020 and PSO 9.0.

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 Performance Module
- PSA104125 - Insulation Monitoring Module
- PSA104126 - Backup Power Module

NOTE: The Energy Analysis Reports Module is available for purchase using the PME Sales Order portal.

Integrating with Advanced Reporting and Dashboards

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

Topic	Content
"About Advanced Reporting and Dashboards" on page 91	An overview of Advanced Reporting and Dashboards Module
"Advanced reporting customizations" on page 92	Advanced reporting customization options
"Device communication" on page 92	Device communication in Power SCADA Operation with Advanced Reporting and Dashboards

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
- Provide historical data to Power Advisor for analytics

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	ENM	Waveforms	Historical reports and dashboards	OPC DA and SMNP
Power SCADA Operation	Enabled *animated 1-line, LiveView, Power SCADA trends*	Enabled	Enabled *configured to communicate with Power SCADA*	Enabled *used for sequence of events analysis*	Basic Reports Enabled *when large data acquisition such as 1 minute logging is required by customer*	Enabled *native drivers within Power SCADA used*
Power Monitoring Expert	Disabled *Vista, PME real time tables, real time trends*	Disabled	Disabled *not configured with PME*	Enabled *used by PME PQ reports*	Enabled *PME Web Reports and Dashboards integrated into Power SCADA runtime*	Disabled *optional drivers for PME not required*

NOTE: Power SCADA Operation with Power Monitoring Expert must be the same product version to be integrated. For example, Power SCADA Operation 2020 and Power Monitoring Expert 2020 are supported. Power SCADA Operation 2020 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.
- **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.)

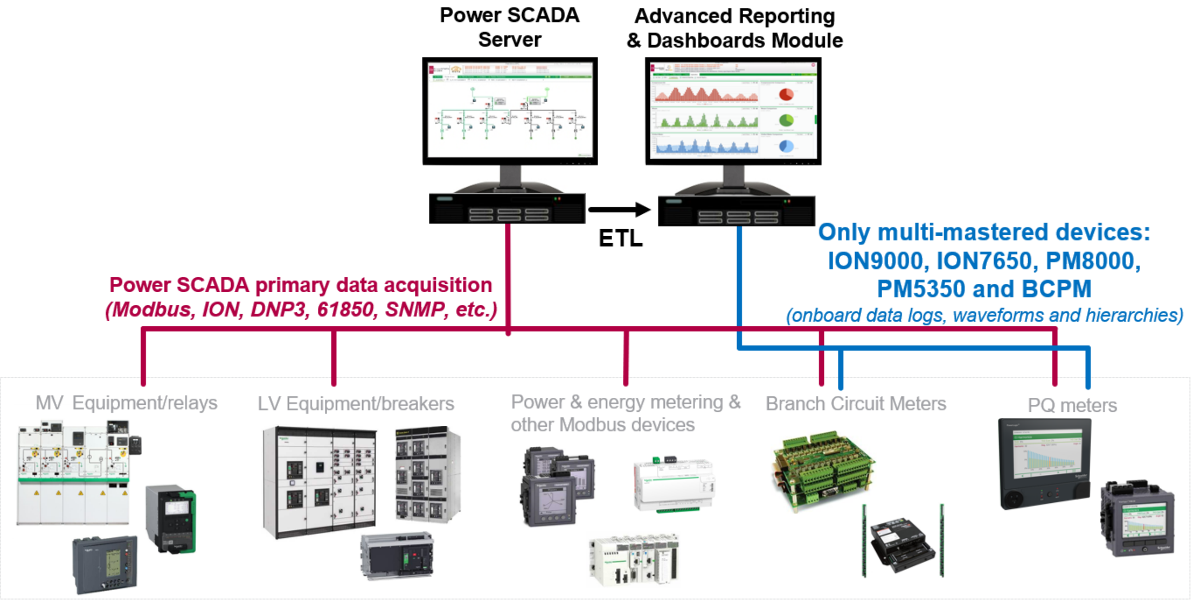
Device communication

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

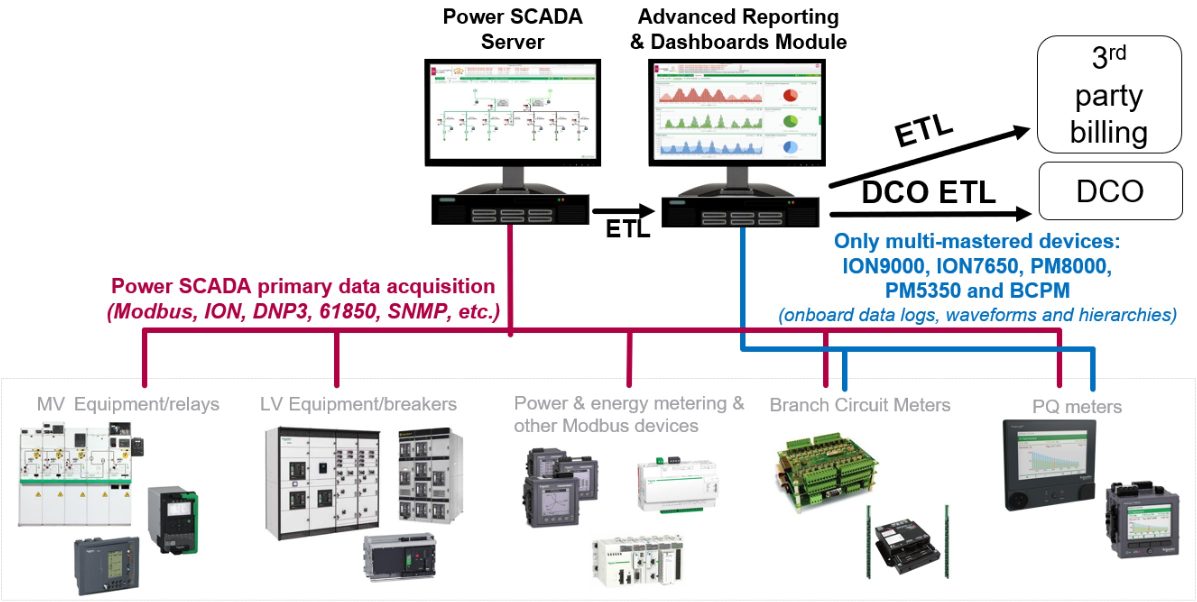
- **Single-mastering all devices** – Setting up device communications with Power SCADA Operation and then transferring device data to Power Monitoring Expert using an Extract, Transform, and Load (ETL) tool
  - Historical trending is assigned to Power SCADA for most devices
  - Having Power SCADA solely acquire data from meters provides the following benefits:
    - Improved performance: Power SCADA trend acquisition can be assigned a lower priority than real-time and alarm data thereby reducing CPU/RAM loads
    - Increased functionality: Allows PME reporting to be run on devices with protocols not natively supported by PME (e.g.: IEC-61850, DNP3, SNMP, BACnet, etc.)
    - Simplified deployments and maintenance: Devices are set up and maintained in Power SCADA only. Meaning there is no risk that device names between Power SCADA and PME are inconsistent.

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


Power SCADA Server
Advanced Reporting & Dashboards Module
ETL
Only multi-mastered devices: ION9000, ION7650, PM8000, PM5350 and BCPM
(onboard data logs, waveforms and hierarchies)
Power SCADA primary data acquisition
(Modbus, ION, DNP3, 61850, SNMP, etc.)
MV Equipment/relays
LV Equipment/breakers
Power & energy metering & other Modbus devices
Branch Circuit Meters
PQ meters

Using ETL for communication with other applications

The following image shows an example architecture for using ETL for communication with other applications, such as 3rd party billing applications or DCO.


Power SCADA Server
Advanced Reporting & Dashboards Module
ETL
ETL
DCO ETL
3rd party billing
DCO
Only multi-mastered devices: ION9000, ION7650, PM8000, PM5350 and BCPM
(onboard data logs, waveforms and hierarchies)
Power SCADA primary data acquisition
(Modbus, ION, DNP3, 61850, SNMP, etc.)
MV Equipment/relays
LV Equipment/breakers
Power & energy metering & other Modbus devices
Branch Circuit Meters
PQ meters

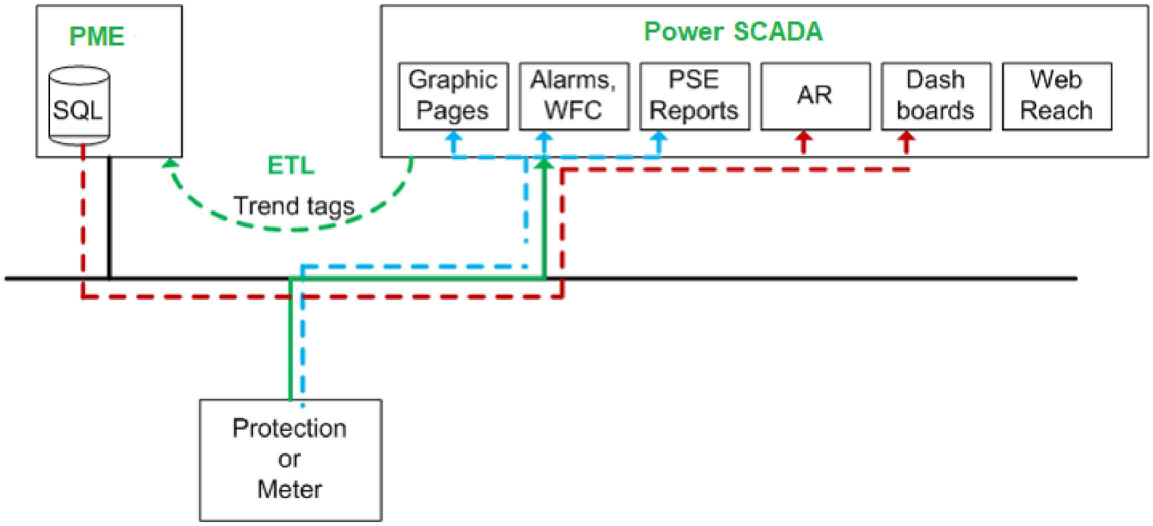
Single-mastering devices

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.

NOTE: ETL is licensed as part of Advanced Reporting Module

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 are set up 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 days' worth of data
  - 105,000 tags logged every 15 minutes: On average, the system took 95 minutes to recover the lost 1.5 days' 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 95 for details.

Multi-mastering devices

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

Devices

Power SCADA cannot single-master the following device types; 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 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.

NOTE: Disable trends in the Power SCADA profile for any branch circuit meters unless the customer would like to see real-time trending in Power SCADA.

- 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 Modules 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
Energy 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 Performance	All devices	Due to the way PQ algorithms work

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

* The Energy 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 export ETL requires data from the customer hierarchy, any devices required for the ETL should be added in Power Monitoring Expert and Power SCADA Operation.

Interoperability

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

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

Topic	Content
"EcoStruxure Building Operation" on page 97	Integration architecture, component usage, data flows, and communication design.
"EcoStruxure Web Services (EWS)" on page 100	EWS for sharing Power SCADA Operation data with EcoStruxure Building Operation (EBO) and Power Monitoring Expert (PME).
"Power SCADA OPC DA" on page 100	Standalone and redundant architectures and data flow.
"Extending Power SCADA" on page 102	Extending Power SCADA Operation using CiCode scripting and CtAPI.

EcoStruxure Building Operation

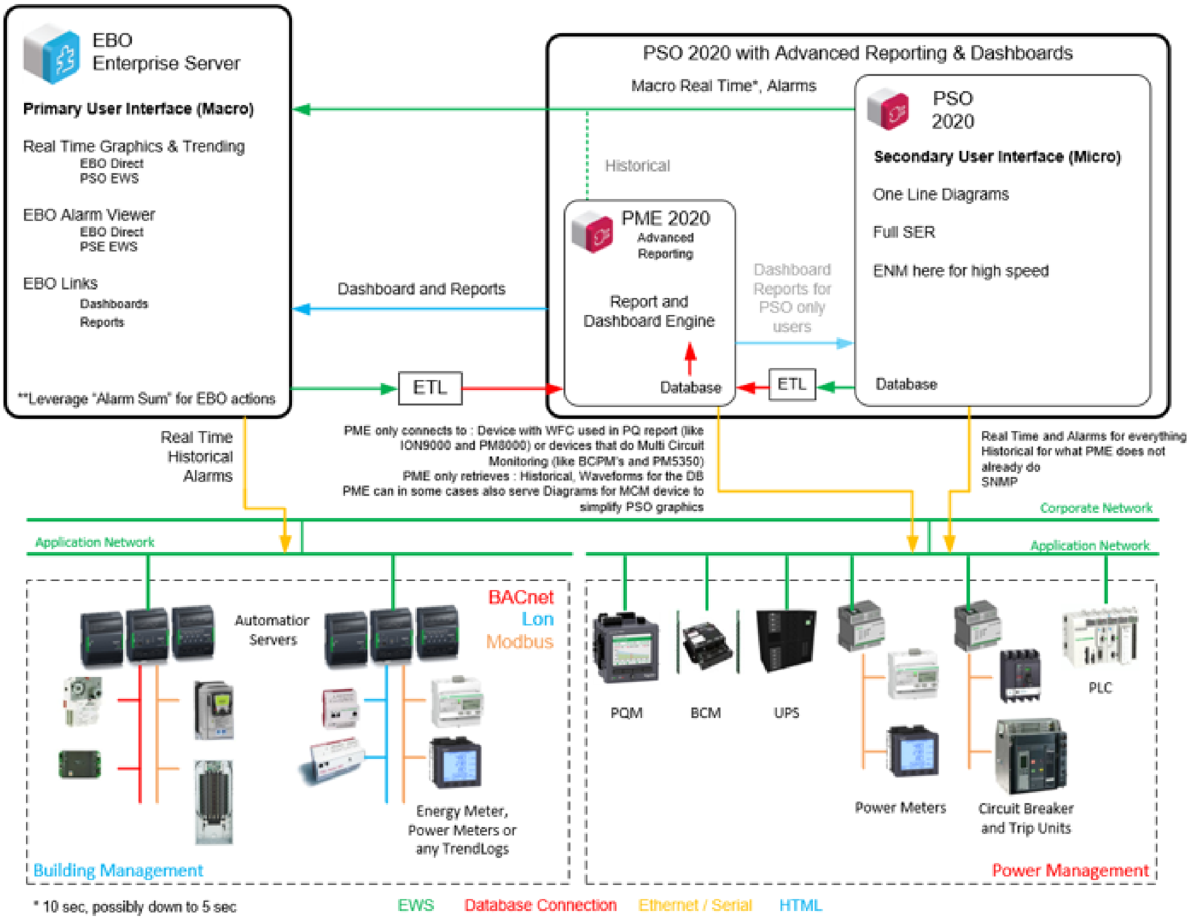
EcoStruxure Building Operation (EBO) integrated with Power SCADA Operation with Advanced Reporting and Dashboards combines electrical and mechanical systems into a single advanced solution.

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

- Power SCADA EcoStruxure Web Services (EWS) provides alarm data and high level real time data from Power SCADA Operation to EcoStruxure Building 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 Building 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 Building Operation to view electrical data



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

	EcoStruxure Building Operation	Power SCADA Operation	Power Monitoring Expert
Real time information	Enabled (graphics screens for macro level real-time data and EBO trending)	Enabled (animated one-line, LiveView)	Disabled (Vista, PME real time tables, real time trends)
Alarms	Enabled (EBO aggregates alarms from PSO and PME using EWS)	Enabled	Disabled
ENM	Not applicable	Enabled (configured to communicate with Power SCADA)	Disabled (not configured with PME)

	EcoStruxure Building Operation	Power SCADA Operation	Power Monitoring Expert
Waveforms	Not applicable	Enabled (used for Sequence of Events analysis)	Enabled (used by PME Power Quality reports)
Historical reports and dashboards	Not applicable	Basic Reports Enabled (when large data acquisition such as 1 minute logging is required by customer)	Enabled (PME Web Reports and Dashboards integrated into EBO or PSO runtime)
OPC DA and SNMP	Not applicable	Enabled (native drivers without Power SCADA are used)	Disabled (optional drivers for PME not required)

Architecture #1: Simple EcoStruxure Building Operation system without redundancy

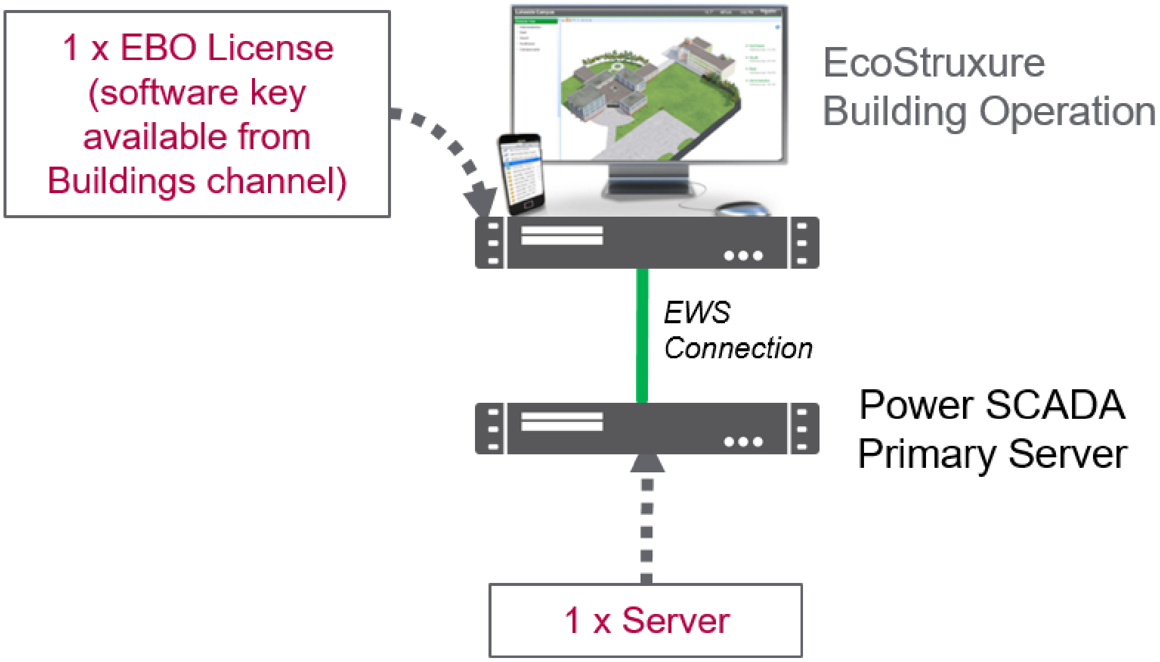
The following image represents the simplest system that can be configured for Power SCADA and EcoStruxure Building Operation. EcoStruxure Building Operation is installed on a separate machine from Power SCADA.

NOTE: We do not support EBO on the same machine as Power SCADA.

NOTE: EBO does not support redundant Power SCADA architectures.

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

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


1 x EBO License (software key available from Buildings channel)
EcoStruxure Building Operation
EWS Connection
Power SCADA Primary Server
1 x 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.

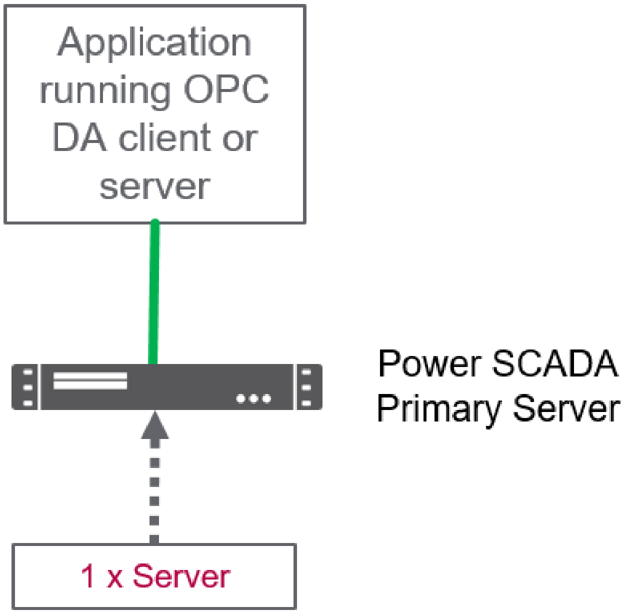
EWS is set up and configured using the Application Configuration Utility.

Power SCADA OPC DA

Architecture #1: Simple system without redundancy

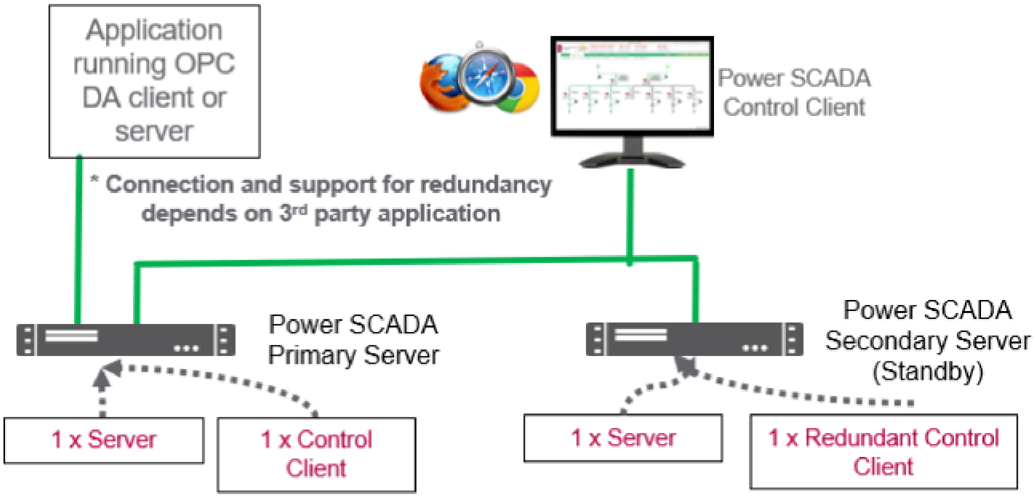
The following image illustrates the simplest system that can be configured for a 3rd party application that is consuming or sending OPC information to the Power SCADA Server.

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


Application running OPC DA client or server
Power SCADA Primary Server
1 x 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.


Application running OPC DA client or server
Power SCADA Control Client
* Connection and support for redundancy depends on 3rd party application
Power SCADA Primary Server
Power SCADA Secondary Server (Standby)
1 x Server
1 x Control Client
1 x Server
1 x Redundant Control 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.

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.

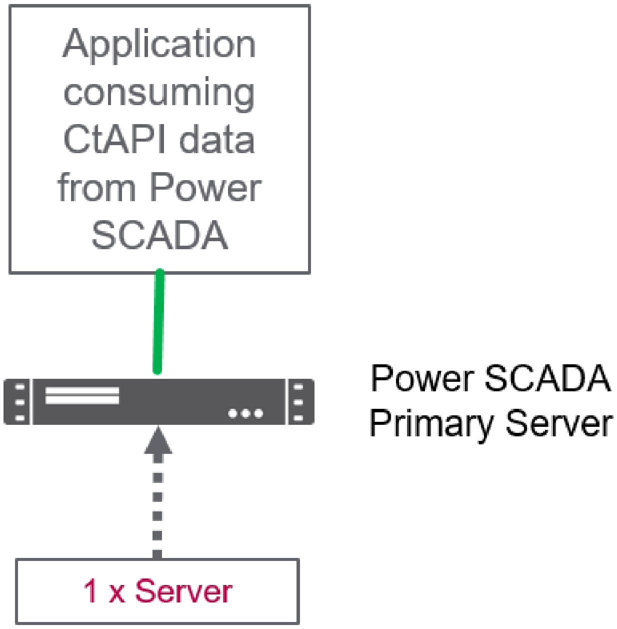
CtAPI

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

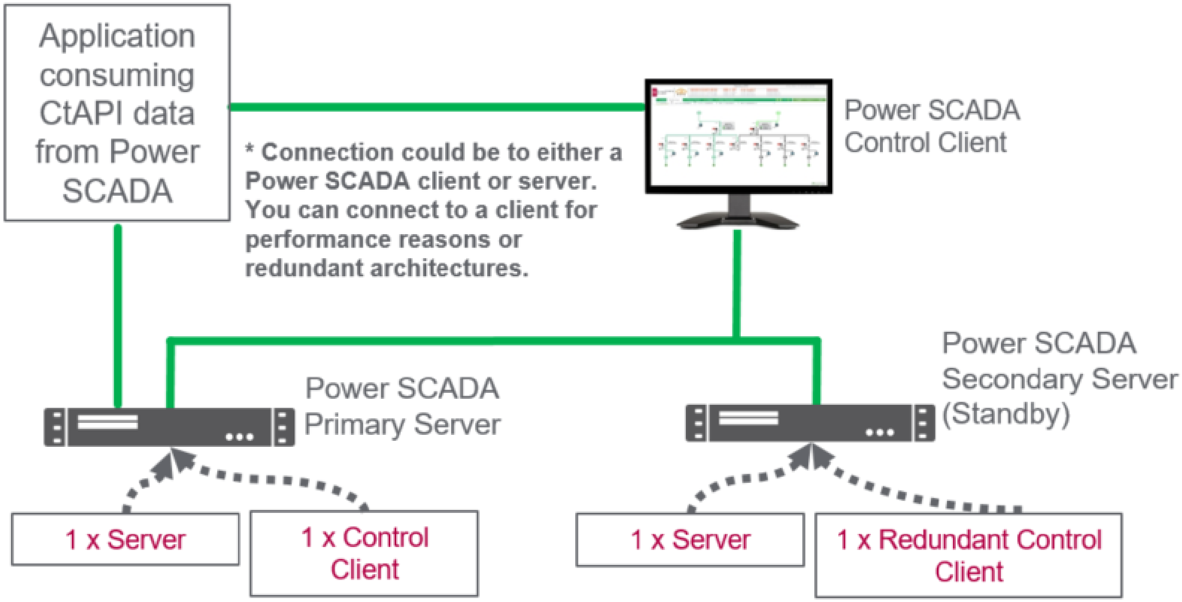
The following image illustrates the simplest system that can be configured for a 3rd party application that is consuming CtAPI data from the Power SCADA Server:


Application consuming CtAPI data from Power SCADA
Power SCADA Primary Server
1 x 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:


Application consuming CtAPI data from Power SCADA
* Connection could be to either a Power SCADA client or server. You can connect to a client for performance reasons or redundant architectures.
Power SCADA Control Client
Power SCADA Primary Server
Power SCADA Secondary Server (Standby)
1 x Server
1 x Control Client
1 x Server
1 x Redundant Control 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)

Power SCADA Anywhere

This section provides information on Power SCADA Anywhere.

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

Topic	Content
"Power SCADA Anywhere component" on page 104	A description of the Power SCADA Anywhere component, including purpose, what's new, upgrade considerations, supported operating systems, host requirements, browser requirements, licensing options, and design considerations.
"Power SCADA Anywhere architectures" on page 106	Example Power SCADA Anywhere architectures.

Topic	Content
"Power SCADA Anywhere and native web client comparison" on page 108	A comparison of features available in the thick client (Power SCADA Anywhere) and the native web client.

Power SCADA Anywhere component

NOTE: Power SCADA Anywhere is a legacy solution that is being substituted with the new native PSO 2020 web client.

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.

NOTE: View-only Clients are not supported with Power SCADA Anywhere.

What's new in Power SCADA Anywhere version 1.2

Power SCADA Anywhere version 1.2 is available with PSO 2020, including the following new features:

- Power SCADA Anywhere now runs on Windows® Server 2016 and 2019.
- You can now enable/disable automatic resizing for the Anywhere Web Client. See the topic Automatic Display Resize in the Citect Anywhere Server Installation Guide.

Upgrade considerations

Upgrade considerations from version 1.0 and 1.1 to version 1.2:

- License upgrade is not required if you are upgrading from Anywhere 1.x.
- For instructions on upgrading Power SCADA Anywhere Server, refer to the "Upgrading to a New Version" section of the Citect Anywhere Server Installation and Configuration Guide.

Supported operating systems

Power SCADA Anywhere Server 1.2 software has been tested to run on:

- Microsoft Windows® Server 2019
- Microsoft Windows® Server 2016
- Microsoft Windows® Server 2012 R2
- Microsoft Windows® Server 2012
- Microsoft Windows® Server 2008 R2 SP1

Host requirements

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

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

Browser requirements

Power SCADA Anywhere is supported in the following browsers:

- Internet Explorer 11
- Edge
- Chrome
- Safari

When using the Power SCADA Anywhere client, 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 such as Power SCADA thick client graphics pages including 1-line diagram/engine built with Citect graphics:

- PME reports, PME dashboards and PME WebReach Diagrams
- Power SCADA Basic Reports
- LiveView (Power SCADA Real Time Tables implementation)
- Event 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.

Multiple instances of Power SCADA Anywhere can be opened at the same time in a web browser.

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 86.

Design considerations

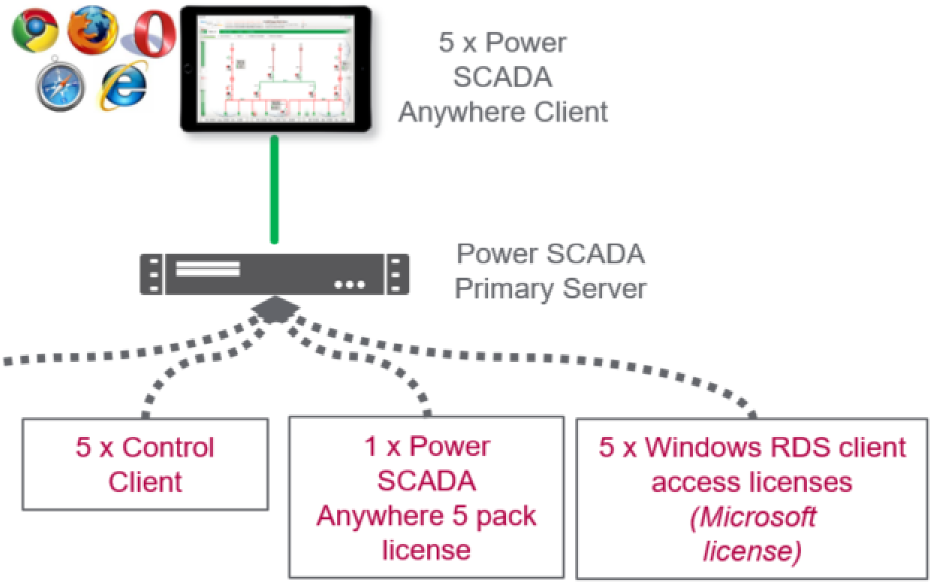
- Power SCADA Anywhere requires an equal number of Power SCADA Control Clients to be licensed.
- Power SCADA Anywhere requires a domain to use Windows Remote Desktop licenses.
  - **NOTE**: The Power SCADA Anywhere host may be a domain controller.
- Since Power SCADA Anywhere uses Windows remote desktop 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.

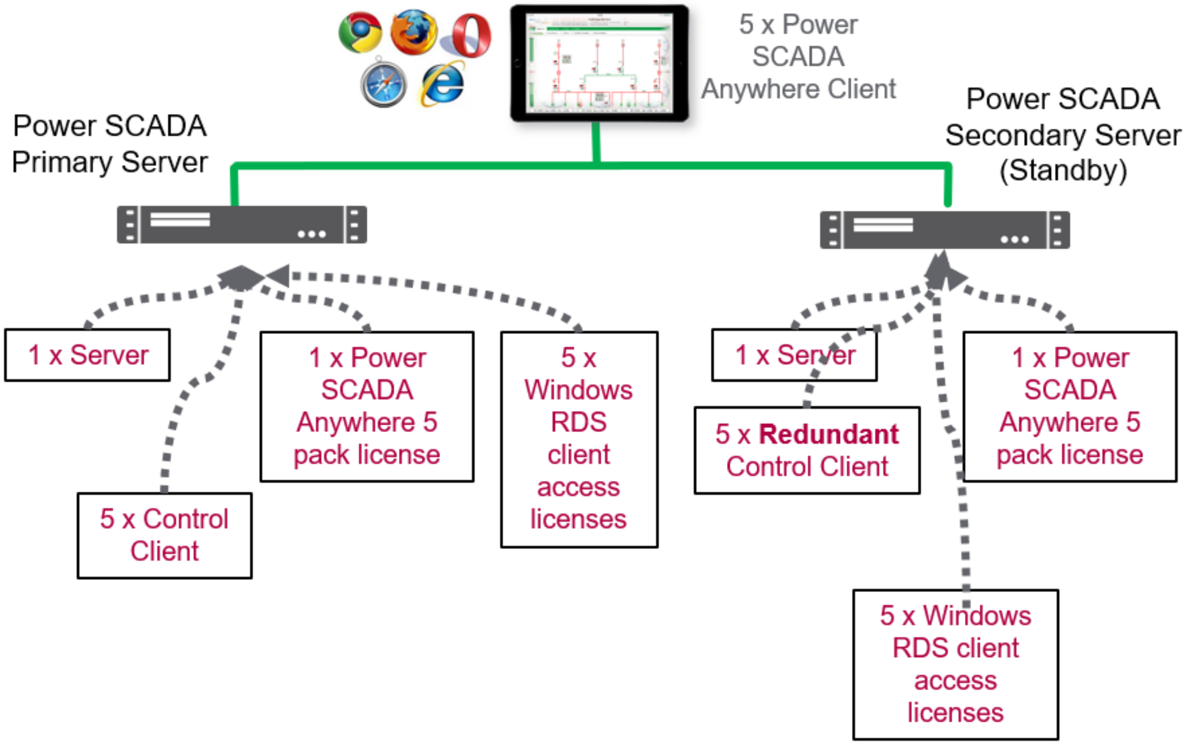
Architecture #1: Power SCADA Anywhere without redundancy

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.


5 x Power SCADA Anywhere Client
Power SCADA Primary Server
1 x Server
5 x Control Client
1 x Power SCADA Anywhere 5 pack license
5 x Windows RDS client access licenses *(Microsoft license)*

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

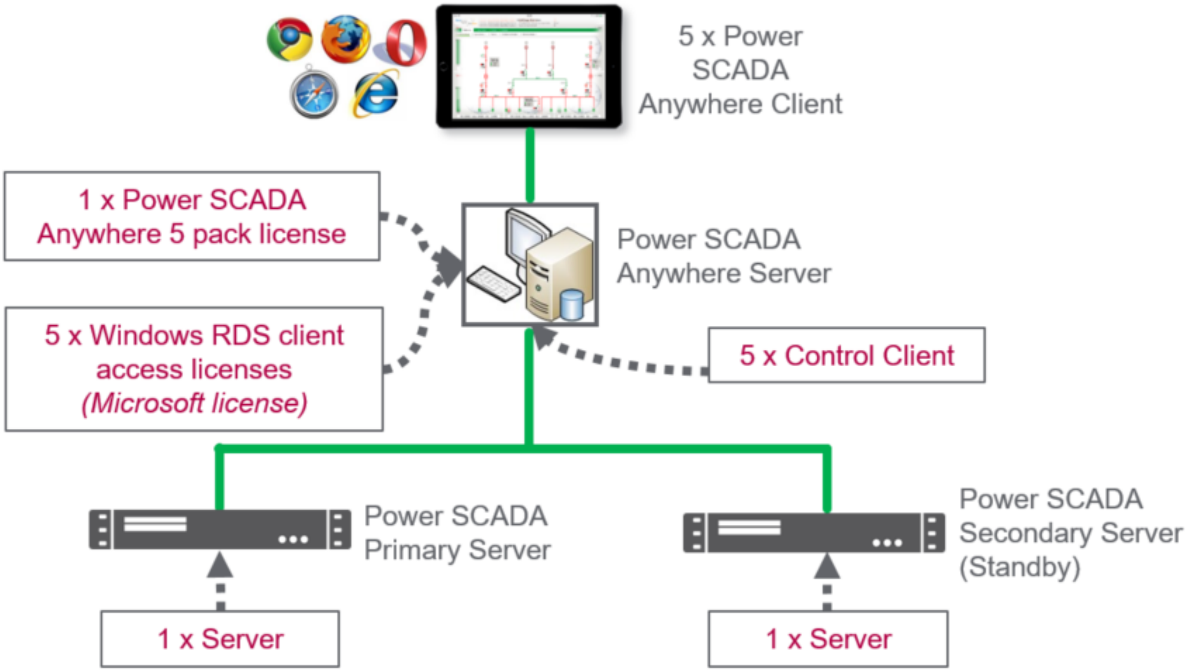
The following example architecture illustrates Power SCADA Anywhere with Power SCADA Server redundancy:


5 x Power SCADA Anywhere Client
Power SCADA Primary Server
Power SCADA Secondary Server (Standby)
1 x Server
1 x Power SCADA Anywhere 5 pack license
5 x Windows RDS client access licenses
5 x Control Client
1 x Server
5 x Redundant Control Client
1 x Power SCADA Anywhere 5 pack license
5 x Windows RDS client access licenses

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

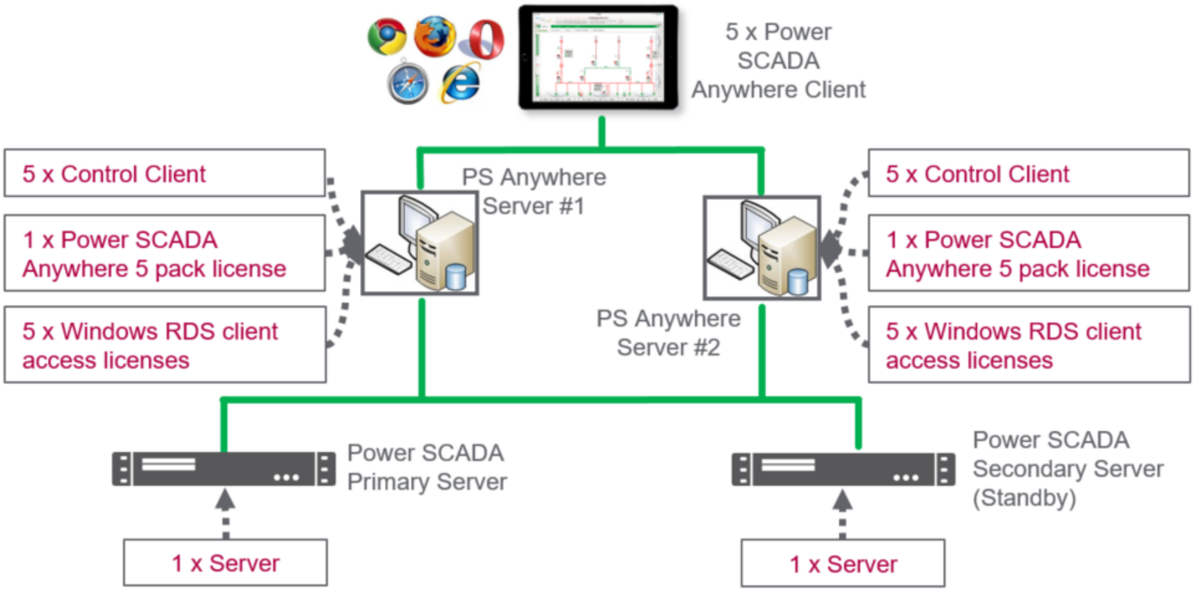
The following example architecture illustrates an isolated Power SCADA Anywhere with Server redundancy:


5 x Power SCADA Anywhere Client
1 x Power SCADA Anywhere 5 pack license
5 x Windows RDS client access licenses (Microsoft license)
Power SCADA Anywhere Server
5 x Control Client
Power SCADA Primary Server
Power SCADA Secondary Server (Standby)
1 x Server
1 x Server

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 stopped working and policies prevented client use on the Power SCADA Server machines.


5 x Power SCADA Anywhere Client
5 x Control Client
1 x Power SCADA Anywhere 5 pack license
5 x Windows RDS client access licenses
PS Anywhere Server #1
PS Anywhere Server #2
5 x Control Client
1 x Power SCADA Anywhere 5 pack license
5 x Windows RDS client access licenses
Power SCADA Primary Server
Power SCADA Secondary Server (Standby)
1 x Server
1 x Server

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

Power SCADA Anywhere and native web client comparison

The following table provides a comparison of features available in the thick client (Power SCADA Anywhere) and the native web client.

Feature	Thick client capability	Native web client capability
Graphics monitoring only	Yes	Yes (plus pan/zoom scalable graphics)
Alarms	Yes	Yes (plus SmartAlarm clustering)
Waveforms	Yes	Yes (plus waveform comparison)
Alarm banner annunciator with audible alarms	Yes	Yes (plus ability to navigate to alarm views and ability for end users to configure

Feature	Thick client capability	Native web client capability
Ability to navigate from graphic to associated alarms	Yes	Yes
Client side scripting	Yes (using CiCode)	Yes (using JavaScript)
Graphics monitoring and control	Yes	Available in PSO 2020 cumulative update
Trend viewer	Yes	Available in PSO 2020 cumulative update
Runtime container customization	Yes	Partial (plus ability for end users to configure)
Ability for local teams to translate end user runtime	Yes	No (more languages planned for future releases)
Two-factor authentication support	Yes	No (planned for future PSO release)
Ability to navigate from alarm to associated graphic (functionality introduced in PSO 9.0)	Yes	No (planned for future PSO release)
Configuration/tools: ENM configuration and PSO diagnostics tool (functionality introduced in PSO 9.0)	Yes	No (end users using web client as primary interface can use thick client runtime to use ENM configuration and diagnostics functionality)
Configuration/tools: Tag Viewer and Tag Debugger (functionality typically used by engineers or tech support for diagnostics / troubleshooting)	Yes	No (end users using web client as primary interface can use thick client runtime to use Tag Viewer / Debugger)
Configuration/tools: Scheduler configuration	Yes	No (end users using web client as primary interface can use thick client runtime to use ENM configuration and diagnostics functionality)

OFS system time stamping

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

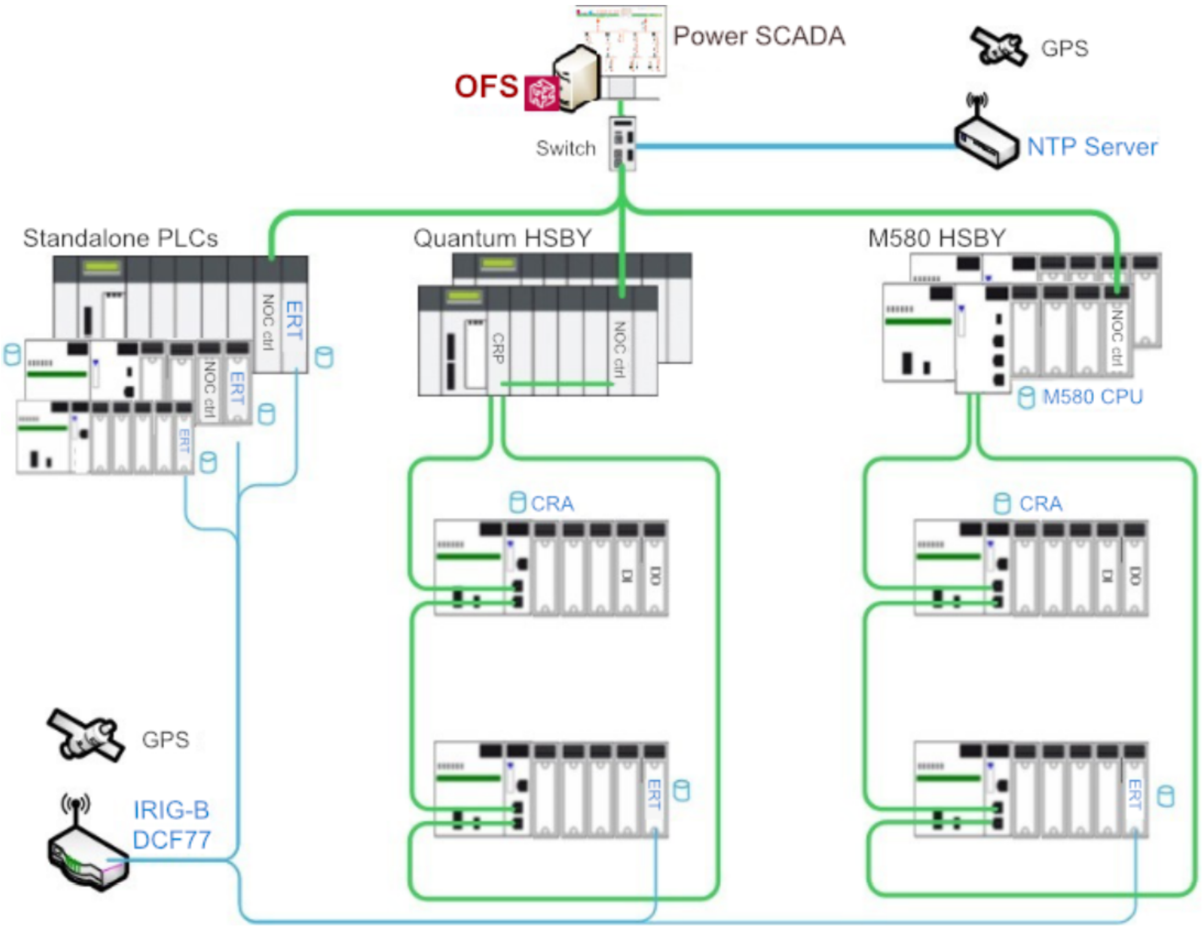
System Time Stamping helps the user analyze the source of abnormal behaviors in an automation system.

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:

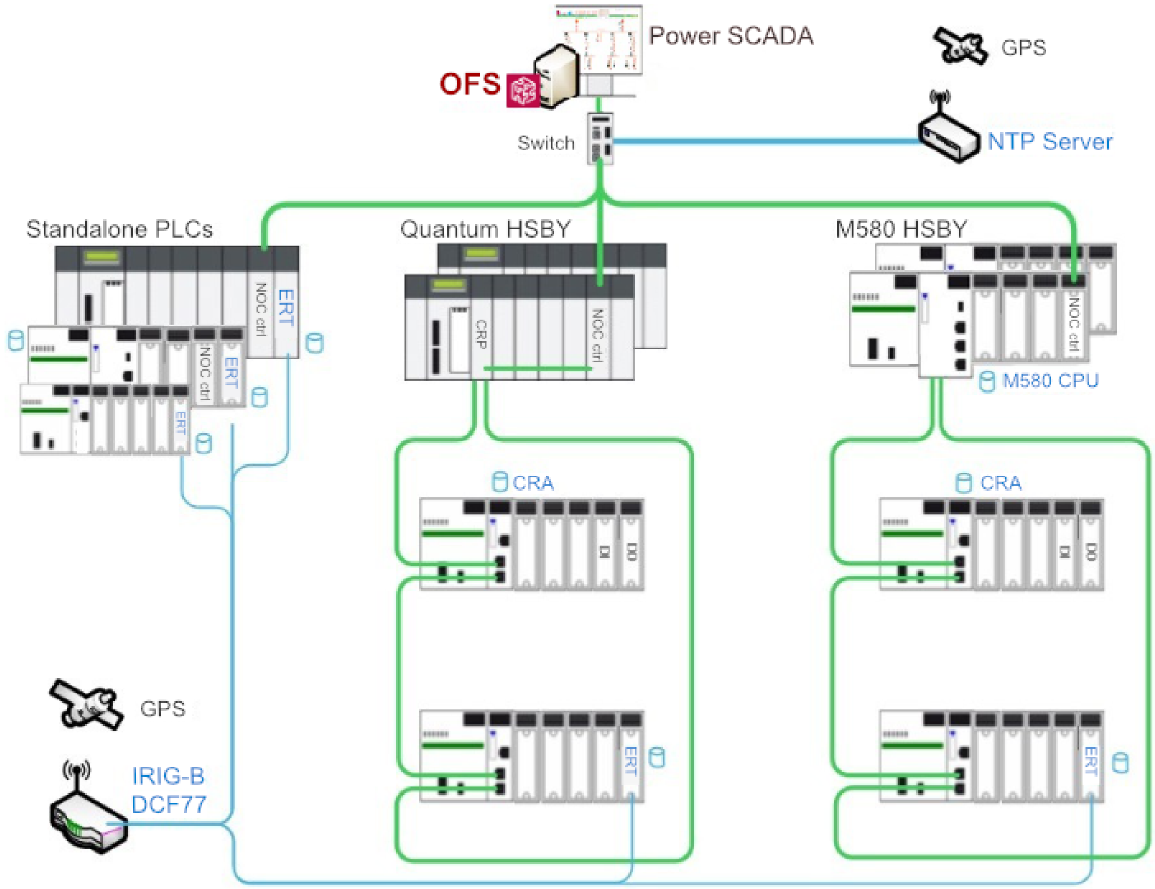
- A communication path between OFS and the time stamping sources is required, so, routing is necessary in multi-layer architectures.
- 2 OPC servers (running for HMI and SCADA) cannot simultaneously access the same time stamping source. A reservation mechanism is implemented.
- No detection of transition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



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

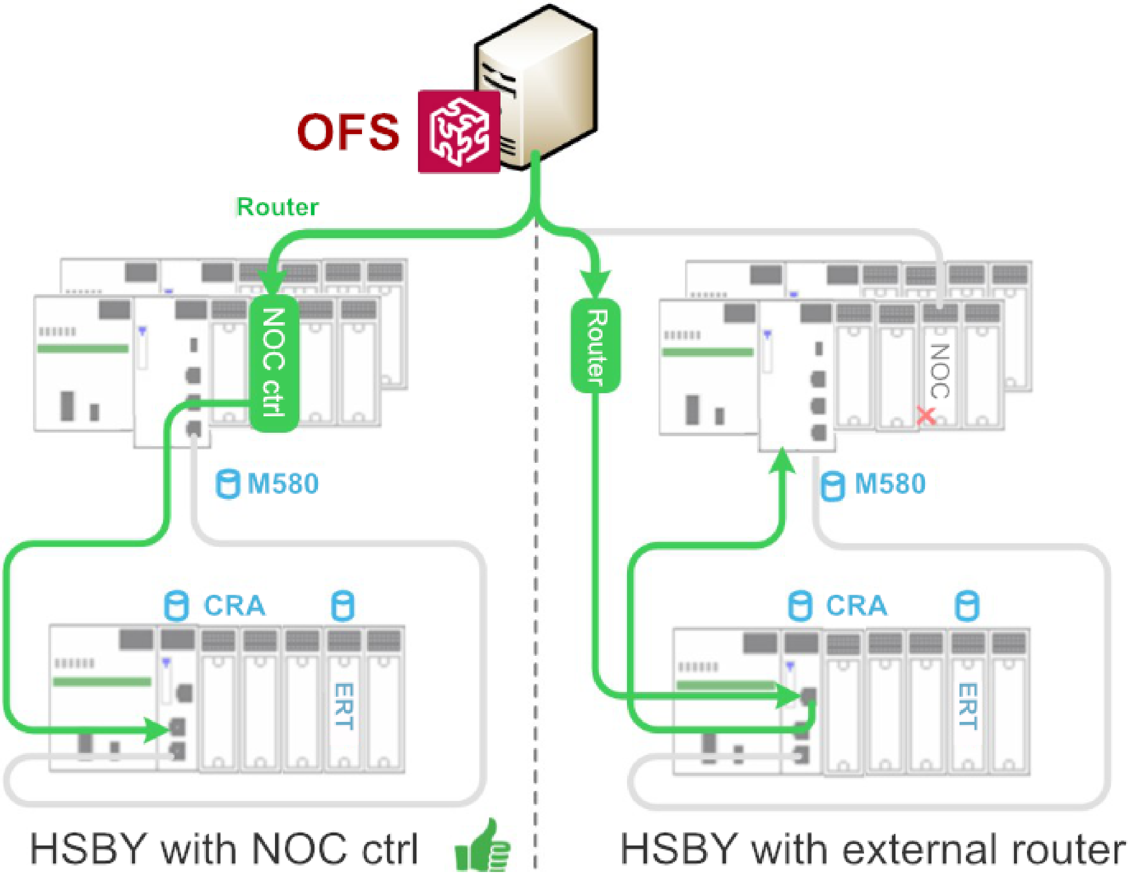
Process	System Time Stamping
1. Synchronize the time clock	ERT module is synchronized by IRIG-B/DCF77 link and x80CRA & M580 CPU are synchronized by the NTP server
2. Time stamping of events generation	I/O events are stamped by x80 ERT modules & CRA Internal variable values are stamped by the M580 CPU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

Architecture selection

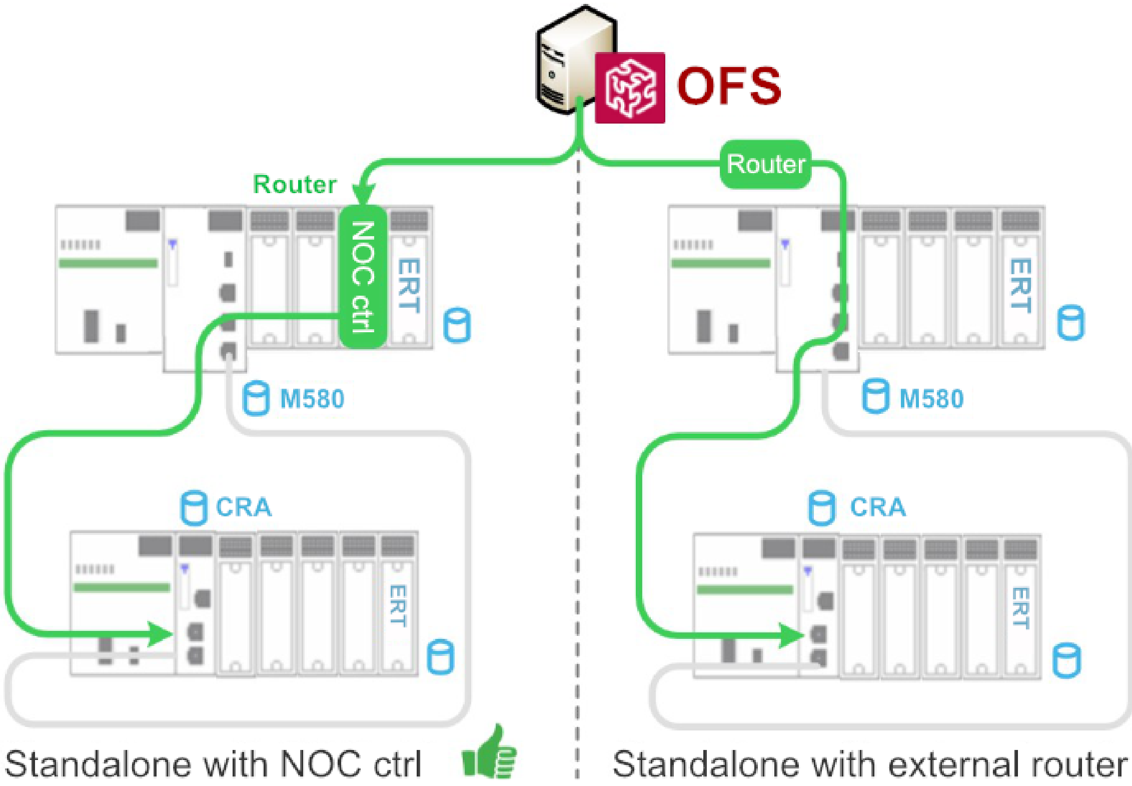


There are three types of modules which are supported by the system time stamping solution, including the M340/eX80ERT, eX80CRA, and M580 CPU. In the system time stamping architecture, OFS is used to automatically transfer the events from the time stamping module to the SCADA. As the time stamping module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

- In the standalone architecture, 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 to set up the connection between OFS and the time stamping module.

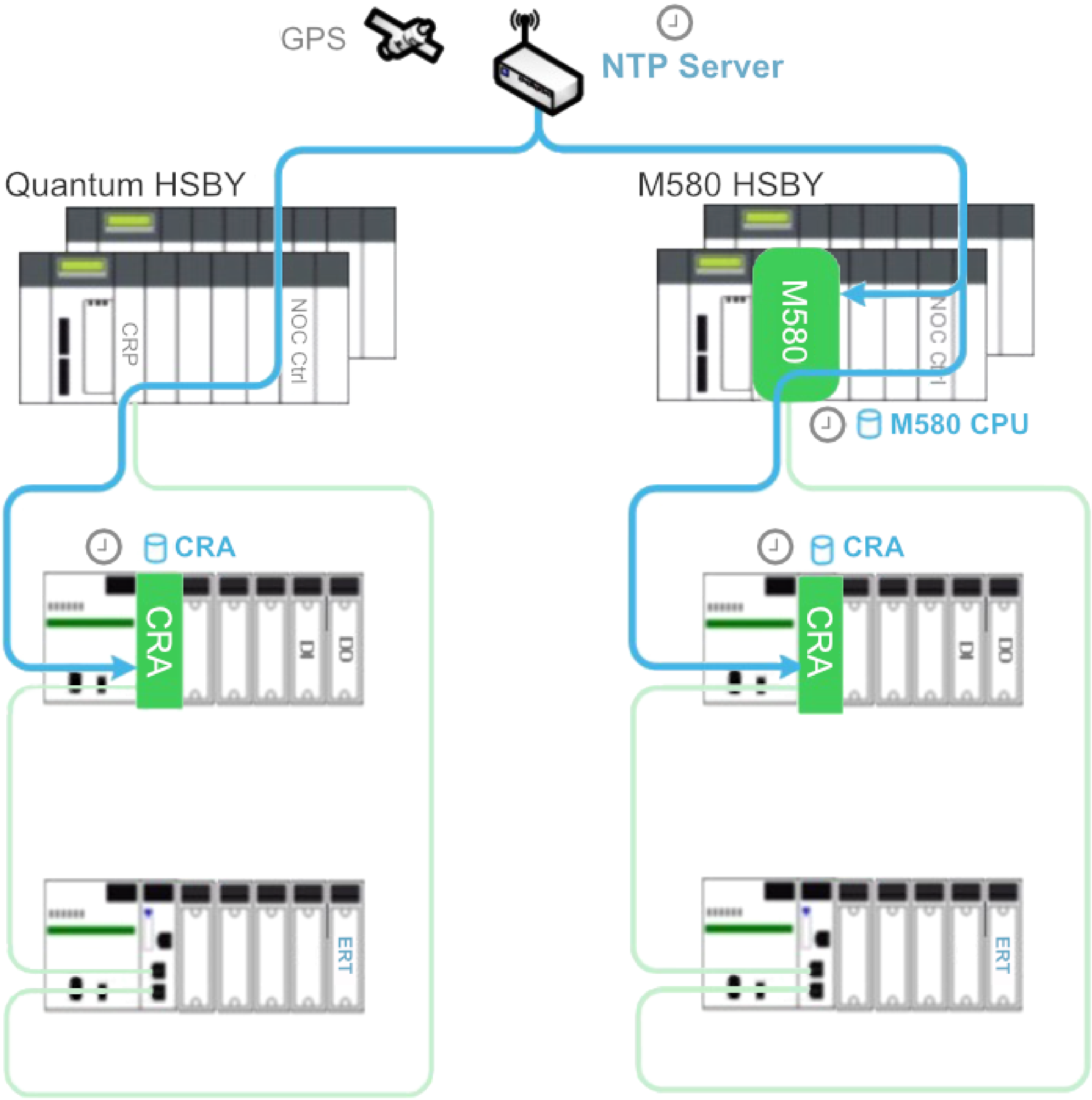


- In the HSBY architecture, either select the NOC control module as a router, or select a third-party router directly connected to the RIO network to set up the connection between OFS and the time stamping module.

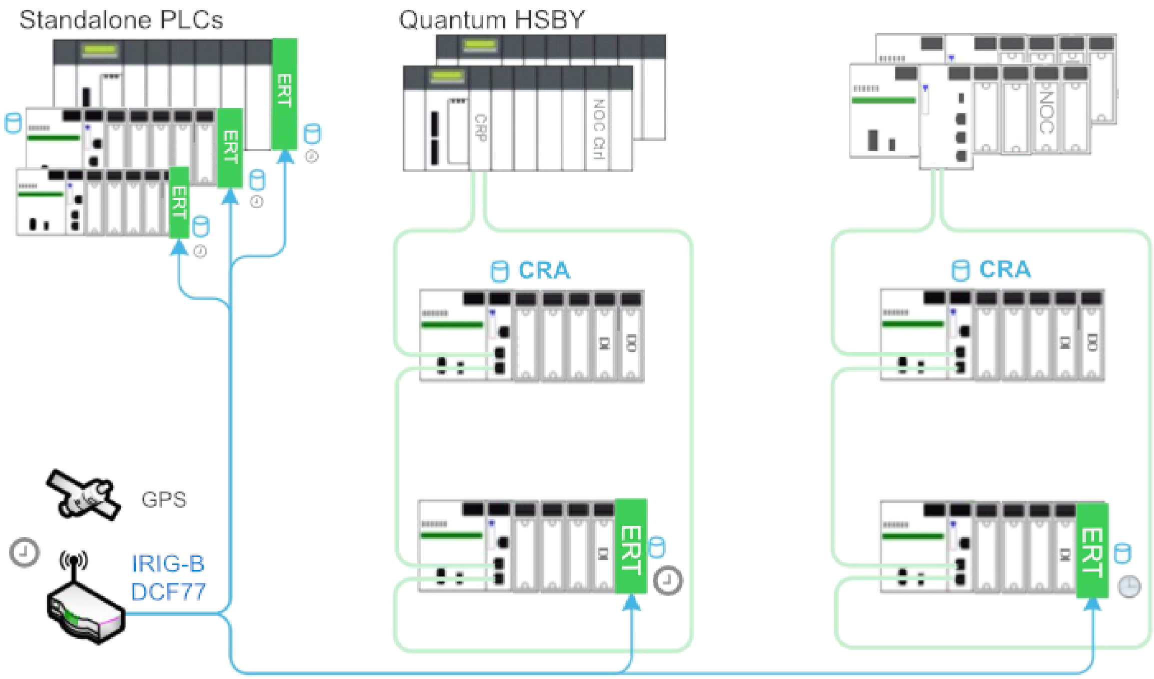


Time synchronization

- The external NTP server provides the time clock for the CPUs and CRAs. 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.

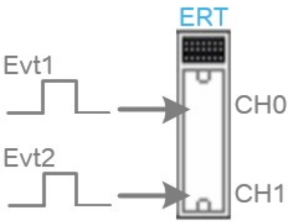
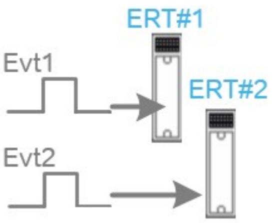
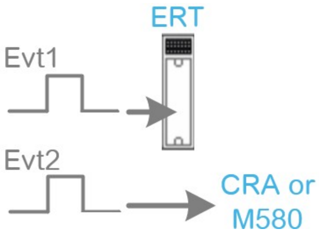
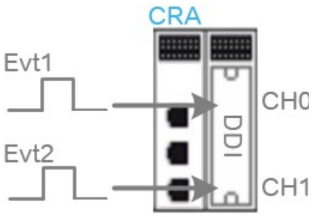
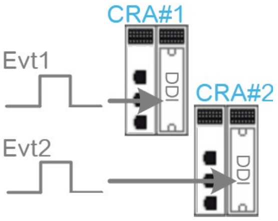
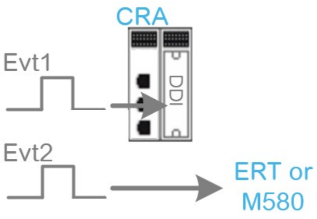
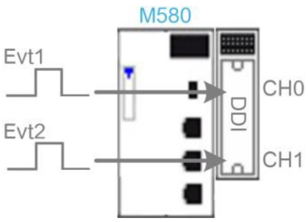
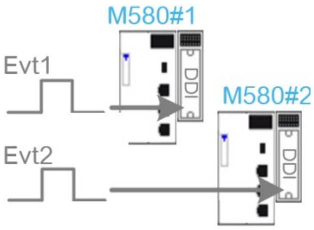
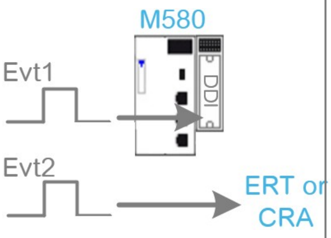


- The IRIG-B 004/5/6/7 or DCF77 signals generated by the GPS receiver are used to synchronize the ERT module's time clock.


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

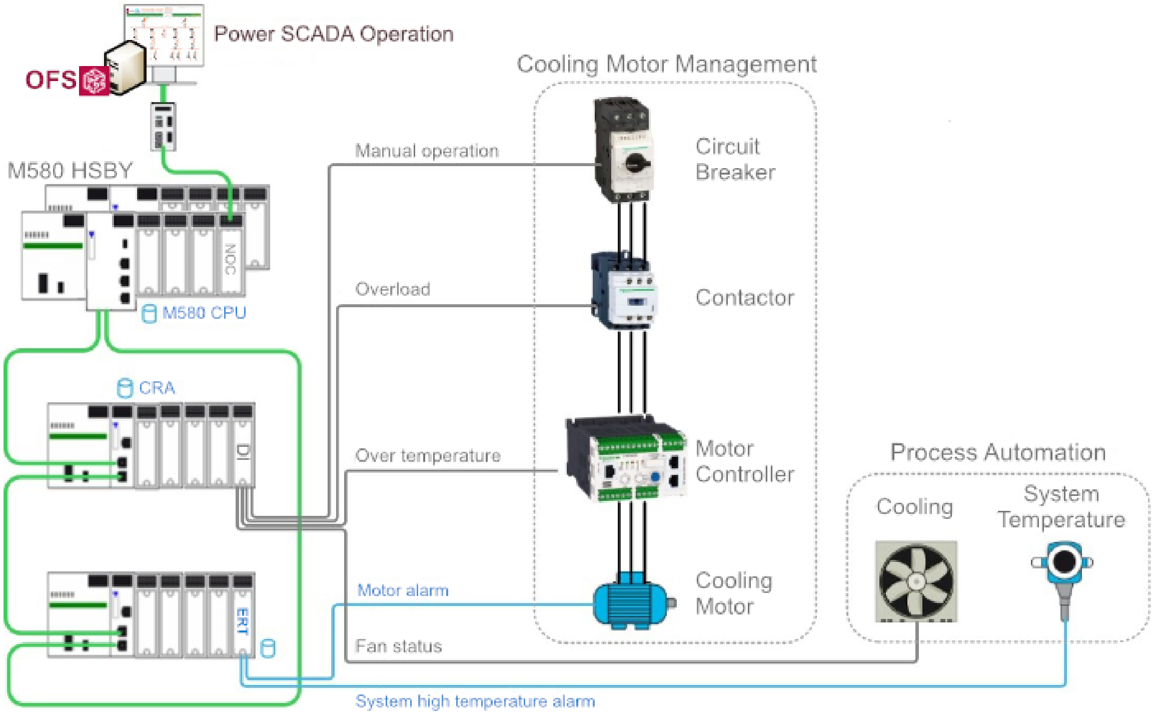
Event resolution

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.

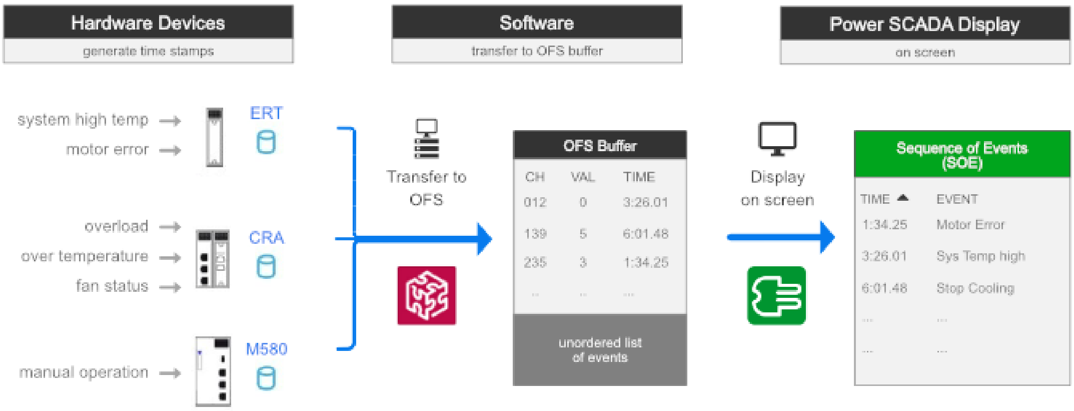


In the above diagram, a cooling control system includes a circuit breaker, a contactor, a motor controller, a motor, and a fan. The fan is used to cool down the system temperature when the temperature is higher than the pre-set value. For the process automation monitoring, some device statuses and process 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

Install and upgrade

Use the information provided in this chapter for installing, upgrading, and licensing a Power SCADA Operation 2020 system.

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

Install

Topic	Content
"Installation process" on page 120	An overview of the installation process
"Before installing the software" on page 121	Steps to prepare prior to installing Power SCADA Operation2020
"System software order of installation" on page 124	The order of installing Power SCADA Operation and its components
"Installing the software" on page 125	How to install Power SCADA Operation
"Installing the ETL Administration Tool" on page 127	How to install the PSO to PME ETL
"After installing the software" on page 129	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 130	How to uninstall and reinstall Power SCADA Operation2020

Upgrade

Topic	Content
"Upgrading" on page 130	A general overview of the steps involved in upgrading your system to 2020
"Upgrade Method" on page 133	Guidelines for choosing an upgrade method: Offline or Online.
"Upgrade Path" on page 134	A description of the number of versions to which you need to upgrade to get from your current version to Power SCADA Operation2020
"Offline Upgrade in Test Environment" on page 135	General steps for performing an upgrade in a test environment
"Offline Upgrade" on page 136	The upgrade process to perform for an Online Upgrade.
"Online Upgrade" on page 144	The upgrade process to perform for an Online Upgrade.

Topic	Content
"Migration Tools" on page 162	A description of the migration tools you must use to upgrade your pre-existing projects for use in Power SCADA Operation2020.
"Migrating from Citect SCADA" on page 172	A description of the benefits and steps to migrate a pre 7.x Citect SCADA system used for power management to Power SCADA Operation 2020.

Also refer to the "Upgrade references" on page 833 section for detailed upgrading information you need to consider, as well as Cicode functions and Citect INI settings that changed with each successive Power SCADA Operation version.

License

Topic	Content
"Licensing" on page 186	An overview of licensing a Power SCADA Operation system
"Update a Sentinel Key with CiUSAFE" on page 187	Updating USB keys
"Activating licenses using the Floating License Manager" on page 189	Using the Floating License Manager to activate licenses.
"Dynamic Point Count" on page 189	A description of how Power SCADA Operation uses a dynamic point count
"Specify the required point count for a computer" on page 190	A description of how points are specified by the computer role
"Run the software in demo mode" on page 191	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 v2020 R2.

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 121 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 vary based on the components of Power SCADA Operation with Advanced Reporting and Dashboards that you install on any computer. This section identifies the basic system software requirements, as well as requirements specific to each component. Refer to "Core components selection" on page 122 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, install the latest updates from Microsoft for your operating system and system software. See "Preparing servers" on page 121 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 "Computer requirements" on page 71 section to ensure that your hardware and system software meet the requirements for your selected installation.

Preparing servers

The software Installer performs several 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

For more information on server requirements and preparation, see the "Advanced Reporting and Dashboards component" on page 58.

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 that installs the runtime components for both a Server and Client. This installation includes 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 many 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, and the like. 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 can 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

After you select 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 is established between the Web Server and the Power SCADA Operation server, they can be on different domains if 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

Before you begin, you need the following items:

- Installation medium for Power SCADA Operation with Advanced Reporting and Dashboards and Power SCADA Operation 2020 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 4.7.2, downloaded from Microsoft.
- Installation for Microsoft SQL Server

NOTE: SQL Express is included on the Power SCADA Operation with Advanced Reporting and Dashboards ISO. Microsoft 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 2020	Power SCADA Operation 2020	Power SCADA Operation 2020 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 76 for more information.
- Install .NET 4.7.2
- 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 2020 – 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 127 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 679 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 4.7.2, 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 4.7.2:
   a. Install .NET Framework 4.7.2.
   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 122.
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 123.
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 if 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.

> **NOTE**: After installing, refer to Citect Help **Post Installation Configuration** for information on configuring a system management server, deployment server, and TLS certificate management.

Installing the ETL Administration Tool

> **NOTE**: The ETL Administration Tool is available for download on the Exchange Community. Do not install ETL on the Power SCADA server. ETL should be installed on the machine hosting Advanced Reporting and Dashboards.

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 969 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 2020 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.

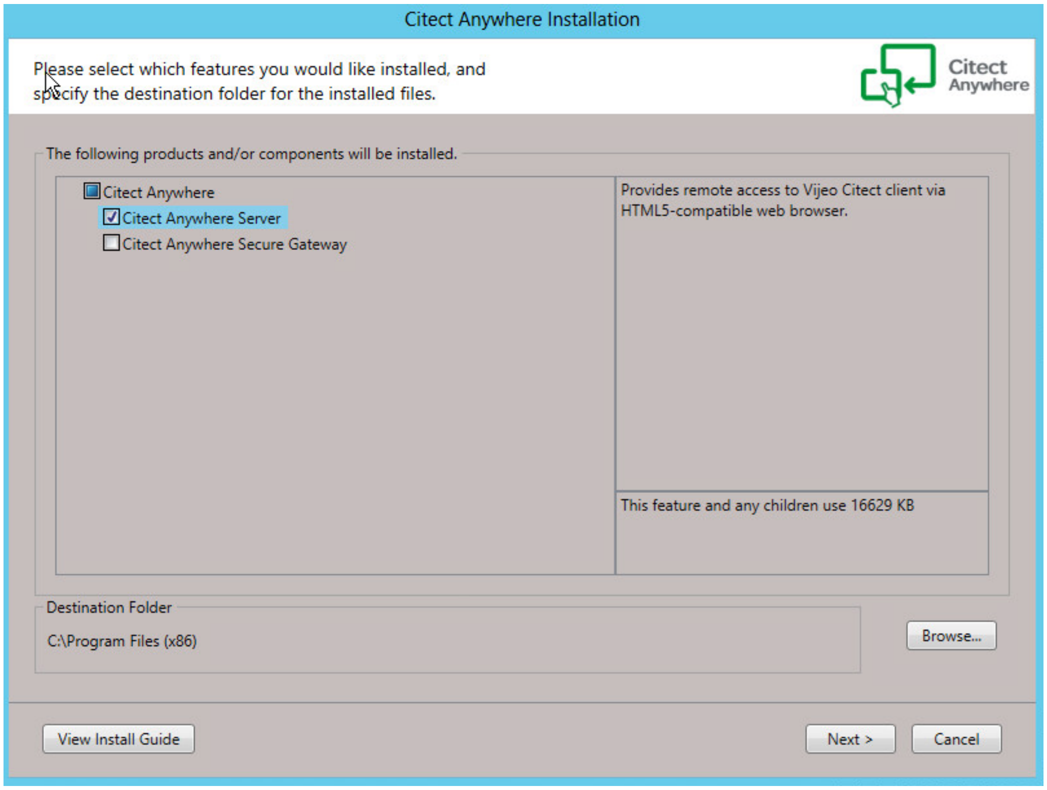
Power SCADA Anywhere is a rebranded name for Citect Anywhere. The term Power SCADA Anywhere will appear only in the end user-facing Web browser, at the login screen and the launch screen. Everything that is not end user-facing will be referred to as Citect Anywhere, including the installer, the configuration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

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 the following operating systems:
  - Windows Server 2008 R2 SP1 Standard, Enterprise (64-bit)
  - Windows Server 2012 Standard
  - Windows Server 2016
  - Windows Server 2019

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**:



Please select which features you would like installed, and specify the destination folder for the installed files.

The following products and/or components will be installed.

- ■ Citect Anywhere
  - ☑ Citect Anywhere Server
  - □ Citect Anywhere Secure Gateway

Provides remote access to Vijeo Citect client via HTML5-compatible web browser.

This feature and any children use 16629 KB

Destination Folder

C:\Program Files (x86)

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 in the Power SCADA Anywhere Installer folder.

After installing the software

Maintaining system currency

After you install and configure Power SCADA Operation 2020 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.

Uninstall and reinstall Power SCADA Operation

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

- Power SCADA Operation v2020 R2 (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\v2020 R2\User`. The first time you launch the application after you re-install it, it will locate the project data and re-link it.

Uninstall does not remove all files from the system. Decommissioning removes Power SCADA files from your system to prevent potential disclosure of sensitive, confidential, and proprietary data and software from your Power SCADA system. You risk disclosing your power system data, system configuration, user information, and passwords if you don’t decommission. We strongly recommend you decommission your system at the end of its' life. See "Decommission" on page 832 for more information.

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 131: Ensure you have the required installation files and other collateral before proceeding.
2. "Upgrade Method" on page 133: Depending on whether your system can afford downtime and loss of data, choose an upgrade method: "Offline Upgrade" on page 136 or "Online Upgrade" on page 144.
3. "Upgrade Path" on page 134: 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 v2020 R2. 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 135 to upgrade and migrate the existing project to Power SCADA Operation v2020 R2.

   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 v2020 R2, 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 136 or "Online Upgrade" on page 144:

- Power SCADA Operation 2020 ISO – Available for download from the Exchange Community
- Power SCADA Operation to Power Monitoring Expert ETL – Available for download from the Exchange Community

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 2020 outlined in the Installation Guide. Be sure to also reference the "Plan" on page 44 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.
- To choose the correct "Upgrade Path" on page 134, determine the version of Power SCADA Operation (formerly PowerSCADA Expert) currently in use at the production site. 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	
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	

Product Version	File Check	File Version	Notes
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 2020, consider whether your SCADA system can afford downtime and whether all your historical information needs to be available always. The upgrade method you choose will depend upon this.

There are 2 upgrade types:

- "Offline Upgrade" on page 136: 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 your historical information needs to be available always, 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 you perform the Offline Upgrade process in a test environment, and even before traveling to the Production site. Doing this lets you 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 144: If you need your system to be available always, 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 145. If the Offline Upgrade was earlier performed in a test environment, as noted above, the upgraded version 2020 project will be migrated to production during the Online Upgrade process as the final step in the Upgrade Path (when Power SCADA Operation 2020 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 2020. The number of necessary steps depends on whether you do an offline or online upgrade.

If you want to perform an "Offline Upgrade" on page 136, you can upgrade your project from as early as version as 7.20 SR1, directly into Power SCADA Operation 2020.

If you plan to perform an "Online Upgrade" on page 144, in which runtime and historical data are migrated and upgraded, you need to follow an upgrade path that depends on your starting version.

Advanced Reporting and Dashboards Module version 2020 must be used with Power SCADA Operation 2020, so an upgrade might 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 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 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 2020.
- **v8.0 or v8.0 SR1** - If this is your starting version, there is no intermediate version upgrade required. Upgrade directly to 2020.
- **v8.1** - If this is your starting version, there is no intermediate version upgrade required. Upgrade directly to 2020.
- **v8.2** - If this is your starting version, there is no intermediate version upgrade required. Upgrade directly to 2020.
- **v9.0** - If this is your starting version, there is no intermediate version upgrade required. Upgrade directly to 2020.

NOTE: When you perform an online upgrade to Power SCADA Operation 2020, the Accept encrypted and non-encrypted connections (mixed mode) setting on the Configurator, Encryption page is selected by default. You can clear this option prior to performing the upgrade if you want to use unencrypted communications. Alternatively, you can configure your system to use encryption after the upgrade process is complete. Information about configuring security and encryption is provided in the Configuration chapter in the Citect SCADA 2018 R2 Install Guide.

Optional Enhancements

- Integration of Diagnostics feature. See "Diagnostics Overview" on page 808 for details.
- Integration of notifications solution; in Power SCADA Operation 2020 you can open notifications from the Alarms page. See "Notifications" on page 338 for details.

- Integration of new Citect SCADA 2018 R2 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 91.
- 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 2020.
- 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 2020 and skip step 13 (Install Power SCADA Operation 9.0). This action upgrades the current version of the project directly to version 2020.

   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 2020.

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 134 from Power SCADA Operation 2020 to the project, recompile and test.

5. Backup the upgraded 2020 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 2020 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 without errors, you need to back up several files and 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 might 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 in the config folder. Gather these INI files from client machines in addition to the server machines in the system.

File	Description
Deployment configuration files	If you have deployment configured, back up the following files: • SE.Asb.Deployment.Server.WindowsService.exe.config • SE.Asb.Deployment.Node.WindowsService.exe.config. These are in the path [CtEdit]Config.
Data directory	This file is found on the path [CtEdit]Data
Deployment database	This is in the Deployment directory. For example: C:\ProgramData\AVEVA\Citect SCADA 2018 R2\Deployment.
ALMSAV.DAT and ALMINDEXSAVE.DAT (For v7.20) **OR** Alarm Database (for v7.30SR1, v7.40, v7.40SR1, v8.0, v8.0SR1, v8.1, v8.2, and v9.0)	<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 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 in the [CtEdit]User\<Project Name> folder.

File	Description
Custom ActiveX Controls (.OCX)	Power SCADA Operation includes several ActiveX controls, which will be available with the 2020 installation, but need to take a backup 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 in the Bin directory where Power SCADA Operation is installed. **NOTE**: The fixes contained in this hotfix might be included in the drivers which ship with Power SCADA Operation2020. See the Power SCADA Operation Exchange Community for additional driver downloads or Citect Driver Web for additional driver downloads.

2. Upgrade your licenses.

   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 2020, refer to "Licensing" on page 186.

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 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 2020 and any recent hotfixes available at the Power SCADA Exchange Community Downloads page.

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

4. Configure the Server Password using the Computer Setup Wizard. For more information, see Power SCADA Server password.

5. Configure the System Management Server. For more information, open the Power SCADA Studio and click **Display the Help** to open the Citect SCADA help.

   a. In the Citect SCADA help search box, type **Configure a System Management Server** and click the search icon.

   b. Follow the instructions for configuring the System Management Server.

6. Restore your project

   Restore your project. Select all included projects if available.

   NOTE: "PLS_Include" will be restored from the Power SCADA Operation 2020 install.

7. 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. After 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.

8. 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 might 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

9. Merge your .INI file

In addition to the INI settings below, identify other custom INI settings that might be required for the proper operation of the upgraded software project. The Computer Setup Editor tool is especially 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 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 "Upgrade references" on page 833.

10. 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 likely that you will encounter errors when you compile your project. One of the most common sources of errors when upgrading is Cicode functions. This is because functions changed, were deprecated, or 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.

11. 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.

12. 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.

13. 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 (Available tools)

Alarms (v7.20 SR1 and earlier)

Before you can upgrade to Power SCADA Operation 2020, 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.

14. Run your project

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

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

15. Install Power SCADA Operation 2020

    After you have completed all the steps in your "Upgrade Path" on page 134, install Power SCADA Operation 2020 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

16. Update settings in the Application Config Utility:

    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

17. Implement "Optional Enhancements" on page 134 available in Power SCADA Operation 2020.

Add optional functionality or enhancements that are now available in Power SCADA Operation 2020 that might 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 might 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, back up the configuration [INI] files for each machine as they might 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 might 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 back up 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 might be using specialty software to provide certain system functionality. These applications might be required to be updated or re-installed during the upgrade process and considered in the context of the upgrade.

Format File

The project might be using custom configuration forms in the product. This configuration is in the FRM file. For more information, see KB1579

Trend and Alarm Data

A project upgrade might 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 can encounter during your Offline Upgrade.

Not able to upgrade license key

1. Make sure you 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 stricter 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 136, you will need to follow the "Upgrade Path" on page 134, 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 2020

Validate that the server hardware running the current Power SCADA Operation project on both the primary and standby servers meets the Power SCADA Operation 2020 minimum requirements listed in "Computer requirements" on page 71. 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 "Computer requirements" on page 71.

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 131 and before traveling to the production site
2. Required Files:
   - Power SCADA Operation 2020 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 synchronizing 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 2020 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: 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 2020 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 2020.
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	Earliest Legacy Version
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
9.0	8100

   - `[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 do not 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)

- (Optional) `[Debug] Kernel = 1`: Enable this to monitor 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 or v9.0

To upgrade from v8.2 or v9.0:

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

   v8.2: [LAN]EarliestLegacyVersion = 8000.
   v9.0: [LAN]EarliestLegacyVersion = 8100.
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 message 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 136.
5. Configure the Server Password using the Computer Setup Wizard. For more information, see Power SCADA Server password.
6. Configure the System Management Server. For more information, open the Power SCADA Studio and click **Display the Help** to open the Citect SCADA help.
   a. In the Citect SCADA help search box, type **Configure a System Management Server** and click the search icon.

b. Follow the instructions for configuring the System Management Server.

> **NOTE**: Power SCADA Operation 2020 should not have encryption enabled with Accept encrypted and non-encrypted not selected, otherwise the servers will not be able to communicate. Mixed Mode should be used, or encryption should be disabled.

7. Place the backed-up Alarm database in the [CtEdit]Data directory. This will allow a quicker synchronization of alarm servers.

8. Restart the primary server. It is now upgraded.

9. Check all functionality on the new Power SCADA Operation 2020 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.

10. Power SCADA Operation 2020 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

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 172 functionality on the Primary Server.
14. The Power SCADA Operation 2020 server will synchronize its alarm database with the running PowerSCADA Expert v8.2 or Power SCADA Operation v9.0 server. You need to wait for the synchronization process to finish, and this will depend on 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.
15. Upgrade your client nodes one by one (skip this step if you are using the latest Power SCADA Operation native web client). On each client complete the steps 1 through 3 and 7 of the "Offline Upgrade" on page 136. 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.
16. Configure your System Management Server and encryption settings based on your requirements.
17. Shut down the standby server and confirm operation of the new Power SCADA Operation 2020 primary server. Validate one-lines, device communications, and event notification operation on the primary server.
18. When the newly upgraded version Power SCADA Operation 2020 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.
19. Upgrade Power SCADA Operation on the standby server according to the "Offline Upgrade" on page 136.
20. Set up the Server Password in the Computer Setup Wizard.
21. Configure your System Management Server and encryption settings based on your requirements.
22. 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.

23. 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 172 functionality on the Standby Server.

24. 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.

25. 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.

26. On both servers remove upgrade-related parameters that were set in "Prerequisites for Online Upgrade" on page 145 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 v2020 check that any pre-7.20 Alarm Save files are removed from the v2020 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.

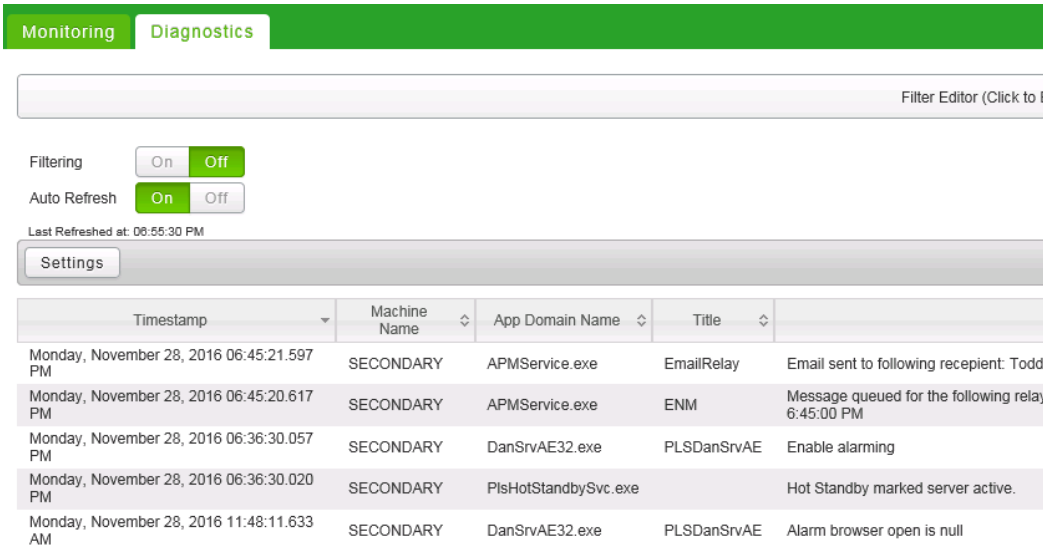
2. Restart the servers after adding the parameter for the changes to take effect.

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:



Filtering ○ On ● Off

Auto Refresh ● On ○ Off

Last Refreshed at: 06:55:30 PM

Timestamp	Machine Name	App Domain Name	Title	
Monday, November 28, 2016 06:45:21.597 PM	SECONDARY	APMService.exe	EmailRelay	Email sent to following recepient: Todd
Monday, November 28, 2016 06:45:20.617 PM	SECONDARY	APMService.exe	ENM	Message queued for the following relay 6:45:00 PM
Monday, November 28, 2016 06:36:30.057 PM	SECONDARY	DanSrvAE32.exe	PLSDanSrvAE	Enable alarming
Monday, November 28, 2016 06:36:30.020 PM	SECONDARY	PlsHotStandbySvc.exe		Hot Standby marked server active.
Monday, November 28, 2016 11:48:11.633 AM	SECONDARY	DanSrvAE32.exe	PLSDanSrvAE	Alarm browser open is null

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 136.

5. Configure the Server Password using the Computer Setup Wizard. For more information, see Power SCADA Server password.

6. Configure the System Management Server. For more information, open the Power SCADA Studio and click **Display the Help** to open the Citect SCADA help.

   a. In the Citect SCADA help search box, type **Configure a System Management Server** and click the search icon.

   b. Follow the instructions for configuring the System Management Server.

   NOTE: Power SCADA Operation 2020 should not have encryption enabled with Accept encrypted and non-encrypted not selected, otherwise the servers will not be able to communicate. Mixed Mode should be used, or encryption should be disabled.

7. Place the backed-up Alarm database in the [CtEdit]Data directory. This will allow a quicker synchronization of alarm servers.

8. Restart the primary server. It is now upgraded.

9. Check all functionality on the new Power SCADA Operation 2020 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.

10. Power SCADA Operation 2020 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

11. 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.
12. On the newly-upgraded primary server, migrate the ENM configuration to Power SCADA Operation notifications. See Migrating notifications for more information.
13. 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.
14. "Verify notifications" on page 172
15. The Power SCADA Operation 2020 server will synchronize its alarm database with the running v2015 server. You need to wait for the synchronization process to finish, and this will depend on 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.
16. Upgrade your client nodes one by one. On each client complete the steps 1 through 3 and 7 of the "Offline Upgrade" on page 136. 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.

17. Configure your System Management Server and encryption settings based on your requirements.
18. Shut down the standby server and confirm operation of the new Power SCADA Operation 2020 primary server. Validate one-lines, device communications, and event notification operation on the primary server.
19. When the newly upgraded version Power SCADA Operation 2020 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.
20. Upgrade Power SCADA Operation on the standby server according to the "Offline Upgrade" on page 136.
21. Set up the Server Password in the Computer Setup Wizard.
22. Configure your System Management Server and encryption settings based on your requirements.
23. 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.
24. 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 172
25. 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.
26. 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.
27. On both servers remove upgrade-related parameters that were set in "Prerequisites for Online Upgrade" on page 145 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 v2020 check that any pre-7.20 Alarm Save files are removed from the v2020 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:

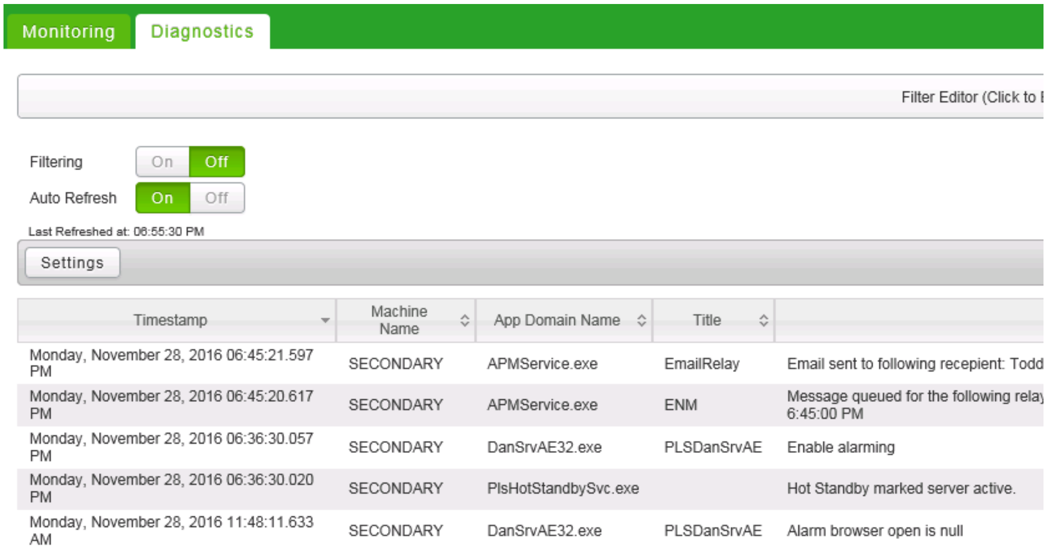
1. Check that you have added the following parameters on the .INI file to all your server nodes before you start the online upgrade:
2. Add the following parameter on the .INI file to all your server nodes before you start the online upgrade.

   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:



Filtering: On [✓ Off]

Auto Refresh: [✓ On] Off

Last Refreshed at: 06:55:30 PM

Timestamp	Machine Name	App Domain Name	Title	
Monday, November 28, 2016 06:45:21.597 PM	SECONDARY	APMService.exe	EmailRelay	Email sent to following recepient: Todd
Monday, November 28, 2016 06:45:20.617 PM	SECONDARY	APMService.exe	ENM	Message queued for the following relay 6:45:00 PM
Monday, November 28, 2016 06:36:30.057 PM	SECONDARY	DanSrvAE32.exe	PLSDanSrvAE	Enable alarming
Monday, November 28, 2016 06:36:30.020 PM	SECONDARY	PlsHotStandbySvc.exe		Hot Standby marked server active.
Monday, November 28, 2016 11:48:11.633 AM	SECONDARY	DanSrvAE32.exe	PLSDanSrvAE	Alarm browser open is null

   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 136.
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 2020
   - 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 2020 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 172

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 136. 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 v2020 clients are working correctly, shut down the standby server and confirm operation of the new Power SCADA Operation 2020 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 172

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 145 and parameters noted Troubleshooting > Remove Upgrade Parameters.

Special Considerations

Alarm Summary

The 2020 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 2020 check that any pre-7.20 Alarm Save files are removed from the 2020 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 2020 is equipped to read this information from the redundant v7.20 SR1 server that is still not upgraded.

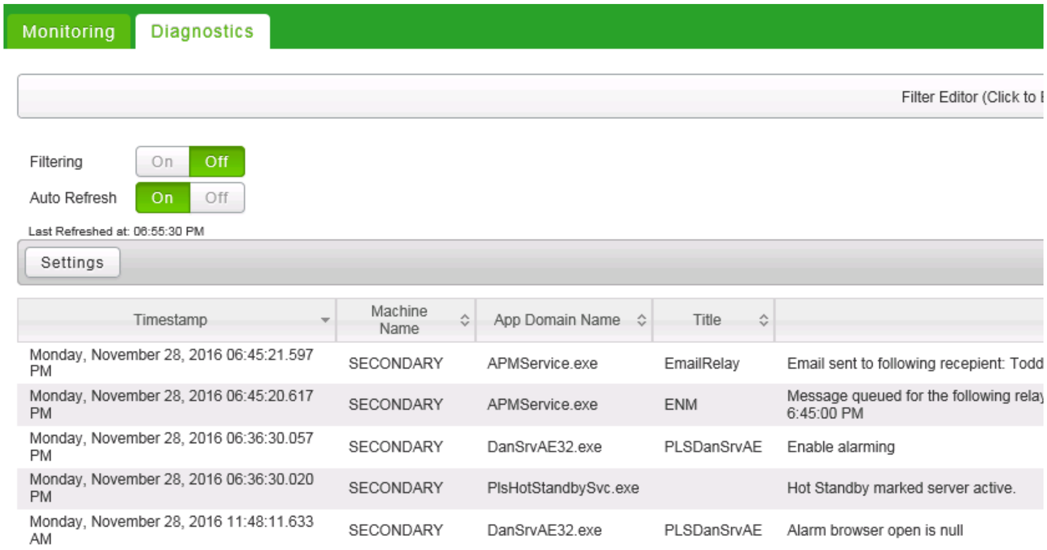
To upgrade from v7.20 or 7.20 SR1:

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:



Filtering On [✓ Off]

Auto Refresh [✓ On] Off

Last Refreshed at: 06:55:30 PM

Timestamp	Machine Name	App Domain Name	Title	
Monday, November 28, 2016 06:45:21.597 PM	SECONDARY	APMService.exe	EmailRelay	Email sent to following recepient: Todd
Monday, November 28, 2016 06:45:20.617 PM	SECONDARY	APMService.exe	ENM	Message queued for the following relay 6:45:00 PM
Monday, November 28, 2016 06:36:30.057 PM	SECONDARY	DanSrvAE32.exe	PLSDanSrvAE	Enable alarming
Monday, November 28, 2016 06:36:30.020 PM	SECONDARY	PlsHotStandbySvc.exe		Hot Standby marked server active.
Monday, November 28, 2016 11:48:11.633 AM	SECONDARY	DanSrvAE32.exe	PLSDanSrvAE	Alarm browser open is null

   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 136
5. Restart the primary server. It is now upgraded.
6. Check all functionality on the new Power SCADA Operation 2020 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 2020 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 172
14. Upgrade your client nodes one by one. On each client complete the steps 1 through 3 of the "Offline Upgrade" on page 136. 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 v2020 clients are working correctly, shut down the standby server and confirm operation of the new Power SCADA Operation 2020 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 172
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 145 and parameters noted in Troubleshooting &gt; 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 v2020 and v7.20 alarm servers, follow these steps:

1. Shutdown your 2020 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 2020 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 2020 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 2020 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 2020, 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 2020 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 could encounter during your Online Upgrade.

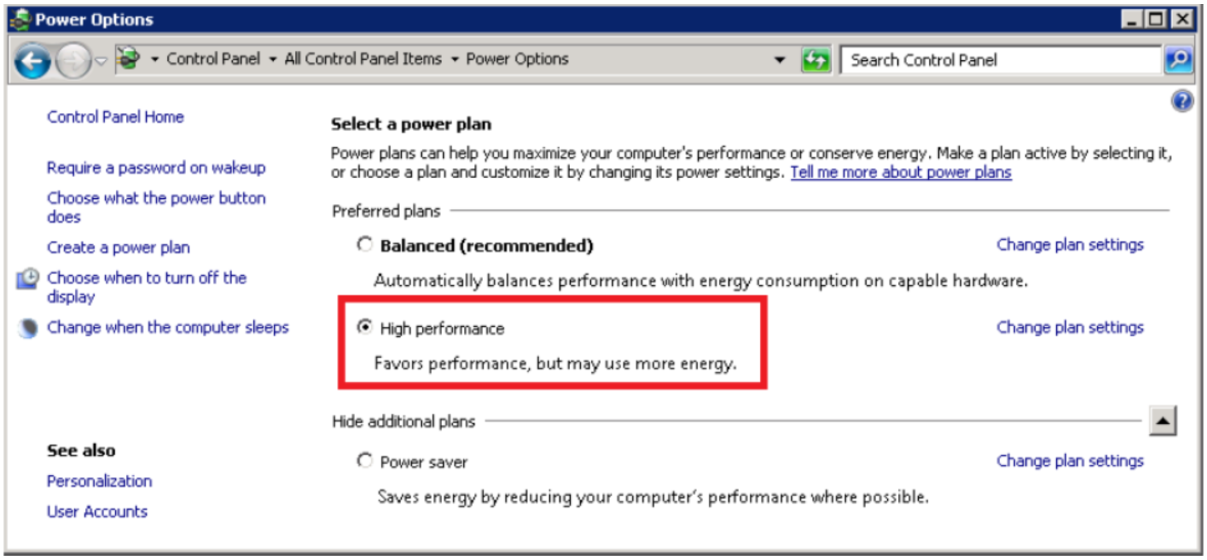
Redundant servers fail to communicate

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

1. Check the `[LAN]EarliestLegacyVersion` parameter value.
   - 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`.
   - 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.


Power Options
Control Panel ▸ All Control Panel Items ▸ Power Options
Search Control Panel
Control Panel Home
Require a password on wakeup
Choose what the power button does
Create a power plan
Choose when to turn off the display
Change when the computer sleeps
See also
Personalization
User Accounts
Change plan settings
Change plan settings
Change plan settings

Select a power plan

Power plans can help you maximize your computer's performance or conserve energy. Make a plan active by selecting it, or choose a plan and customize it by changing its power settings. Tell me more about power plans

Preferred plans

○ **Balanced (recommended)**

Automatically balances performance with energy consumption on capable hardware.

● High performance

Favors performance, but may use more energy.

Hide additional plans

○ Power saver

Saves energy by reducing your computer's performance where possible.

Remove Upgrade related parameters

After completing the upgrade process and confirming that runtime is fully functional, remove or update the following .INI parameters.

NOTE: 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 users' 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 "Upgrade references" on page 833 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 2020 does not fully upgrade your projects, and needs to be followed by using the Migration Tools. If you are 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 2020.

There are two Migration Tools: one for Citect SCADA and one for Power SCADA Operation. Both must 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.

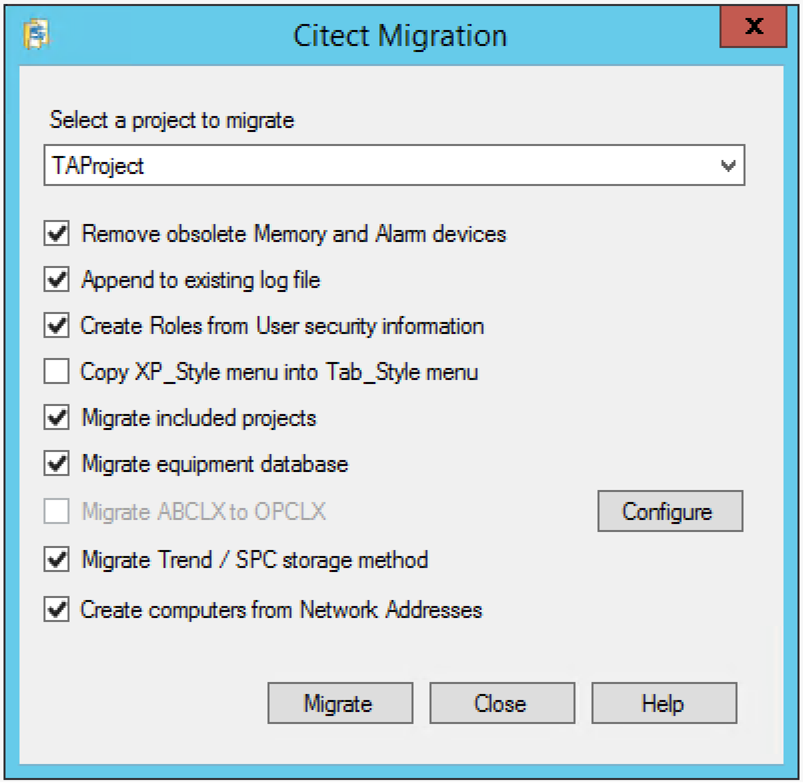
Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

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.



Select a project to migrate

TAProject

☑ Remove obsolete Memory and Alarm devices

☑ Append to existing log file

☑ Create Roles from User security information

☐ Copy XP_Style menu into Tab_Style menu

☑ Migrate included projects

☑ Migrate equipment database

☐ Migrate ABCLX to OPCLX

☑ Migrate Trend / SPC storage method

☑ Create computers from Network Addresses

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 168). **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.
Create roles from User security information	Select this option if you want to migrate the user database from an existing project (see "Creation of Roles for Existing Users" on page 170).

Option	Description
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 171).
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 Style 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.

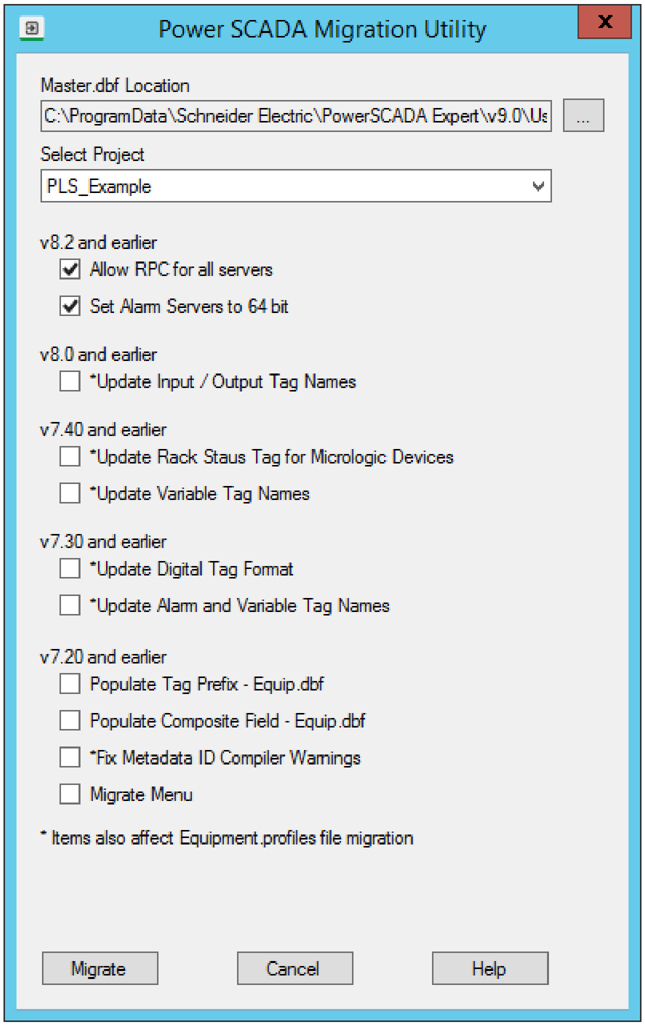
6. Click the **Close** button to close the dialog.

Using 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**.



Power SCADA Migration Utility

Master.dbf Location

C:\ProgramData\Schneider Electric\PowerSCADA Expert\v9.0\Us

Select Project

PLS_Example

v8.2 and earlier

☑ Allow RPC for all servers

☑ Set Alarm Servers to 64 bit

v8.0 and earlier

□ *Update Input / Output Tag Names

v7.40 and earlier

□ *Update Rack Staus Tag for Micrologic Devices

□ *Update Variable Tag Names

v7.30 and earlier

□ *Update Digital Tag Format

□ *Update Alarm and Variable Tag Names

v7.20 and earlier

□ Populate Tag Prefix - Equip.dbf

□ Populate Composite Field - Equip.dbf

□ *Fix Metadata ID Compiler Warnings

□ Migrate Menu

* Items also affect Equipment.profiles file migration

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 coordinated 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. If after migration you want to use the Equipment tab in the Graphics Editor, you must select a device profile **Type** in the **Add/Edit Device Profile** screen. For more information, see the Add, edit, or delete device profile section.

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 2020

   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 1235).

    e. Run the Advanced One-Line tool (see "Reviewing Genie Configurations" on page 1253).

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
v9.0 and earlier		
Update item names	Adds item names to variable tags to work with Web Applications.	Updates all item names.
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.

Element	Description	Changes
Update variable tag names	Updates edited tag names to standard tag names.	Updates tag names to the latest standard name.
v7.30 and earlier		
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 later, 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

Except for the Array Size, which was 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 after 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 have 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.

Migrating from Citect SCADA

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.**

Before the introduction of Power SCADA Operation, some customers used Citect SCADA for power management edge control applications. Customers using pre-7.x versions of Citect SCADA can migrate their systems to Power SCADA Operation to take advantage of the power management features unique to Power SCADA Operation.

Power SCADA Operation is built on Citect Studio and includes productivity tools that are designed and optimized to create the tags you need to configure power-based SCADA projects. It is important to use these productivity tools when migrating a pre-7.x Citect SCADA system to Power SCADA.

The following Power SCADA features are only supported using Power SCADA productivity tools and workflows:

- PwrModbus driver
- Profile Editor and I/O Device Manager
- Advanced One-line Configuration
- Waveforms (Comtrade)
- Native Notifications
- Power Diagnostics tools (PSO 9.0)
- Basic Reports
- Advanced Reporting and Dashboards Module
- Interoperability with EcoStruxure Building Operation
- LiveView

- Power SCADA Power Graphic Libraries
- Support for 2-factor authentication and Single-Sign-On with Advanced Reporting and Dashboards Module
- Historical data from Power SCADA Operation to Advanced Reporting and Dashboards Module database via ETL (Extract, Transform, Load) tool
- Thermal Monitoring Application

For instructions on migrating from Citect SCADA, see "Citect SCADA Migration Information" on page 881.

Backing up and restoring a Power SCADA system

A Power SCADA system can be backed up and then restored if the original Power SCADA system is no longer available.

The procedures outlined in this document describe how to backup and restore standalone and redundant Power SCADA systems, and include Power Monitoring Expert (Advanced Reporting and Dashboards).

NOTE: This procedure does not support distributed PME systems.

The Power SCADA system components and modules you will need to back up will vary, depending on your system architecture and whether your system is redundant.

NOTE: When backing up and restoring a project, ensure that "Include Subdirectories" is checked so that your graphics and advanced one-line configuration is included.

Backing up a Power SCADA system

This section includes the tasks required to create automated backups of a Power SCADA system. It also lists the installation media you need to back up, and the steps you should complete to prepare for the backup process.

NOTE: Some procedures in this section cite scripts you can use to automate the backup and restore process. Refer to www.se.com or the Exchange Extranet for the backup and restore scripts specific to your Power SCADA system version.

Before you begin

Before you backup or restore a Power SCADA system, review the steps below to prepare for the processes.

Installation media and license backup

Back up the following installation media and license files at least one time:

- Power SCADA installation media (the ISO file used for installation).
- Power SCADA License Activation IDs if using software licenses.

- If Power Monitoring Expert (PME) is present on your system:
  - PME installation media
  - PME License Activation IDs
  - PME Custom Report Pack install files

Backup directory location

Create a directory structure in a secure location on your network that can be accessed if you can no longer access the original Power SCADA system. This backup directory location will contain all the backup files required to restore your system. It must be accessible and have relevant permissions for the Administrator account to create the backup files.

System passwords

You will need to access system passwords to recover your Power SCADA system. However, it is beyond the scope of this document to provide guidance on how you should manage your passwords for backup situations. If you are uncertain how you should back up passwords, consult your IT Department.

Backing up Power SCADA

You can automate backing up Power SCADA to include most of the components that will be required to restore a system. Because Power SCADA passwords cannot be automatically backed up, you must back them up manually.

.NET 4.5.2 and WMF 5.1 are required to run the provided scripts. WMF 5.1 is available as an offline windows update installer.

1. Run PowerShell and enter the following command to determine what version of WMF is installed before proceeding:
   ```
   $PSVersionTable.PSVersion
   ```
2. Verify PS Version is 5.1 or greater. If an older version is detected, see the below table for updating to the required version. The WMF installation is available as an offline windows update (.msu) and will require a reboot after installation.

Download the WMF 5.1 package for the operating system and architecture on which you want to install it:

Operating System	Package Link
Windows Server 2019	Native
Windows Server 2016	Native
Windows Server 2012 R2	Win8.1AndW2K12R2- KB3191564-x64.msu
Windows Server 2012	W2K12-KB3191565- x64.msu

Backing up Power SCADA automatically

To automate the Power SCADA backup process, you can run the script to copy the required Power SCADA project folders to the backup directory location. You can also create a scheduled task to run the script. Refer to www.se.com or the Exchange Extranet for the backup script specific to your Power SCADA system version.

> **NOTE**: The automated Power SCADA backup does not include every Power SCADA component that needs to be backed up. The server password key must be manually backed up. After you complete the automated Power SCADA back up, see Backing up the Power SCADA Passwords and Device Profiles for details.

After you automatically back up your Power SCADA system, periodically check the backup directory drive to make sure there is sufficient space for the copied backups. See Deleting Old Backups to automate your system to delete old backups.

To automate the Power SCADA backup process:

1. In a text editor, open the **PSEvx.x_Backup.ps1** script for your version.
2. Edit the file for your system:
   a. For `<$destinationDir>`, enter your backup directory location.
   b. If you did not install Power SCADA to the default install location, edit the `<$pseBin>` and `<$pseData>` to reflect the installed location path.
3. Save the file as **PSE_Backup.ps1** in the following location:
   `C:\Program Files (x86)\Schneider Electric\BackupTasks`
4. Run PowerShell as Administrator and set the execution policy with the following cmdlet:
   `Set-ExecutionPolicy Bypass.`
5. Open Windows Task Scheduler and create a new task under Schneider Electric with a trigger to run once a week at midnight.
6. Define the new action:
   a. For Action, click **Start a program**.
   b. In Program/script, enter the following:
   `C:\Windows\syswow64\WindowsPowerShell\v1.0\powershell.exe`
   c. In Add arguments, enter the following:
   `-noninteractive -nologo -file " C:\Program Files (x86)\Schneider Electric\BackupTasks\PSEv8.1_Backup.ps1"`

> **NOTE**: Run the scheduled task to confirm that it copied the components to your backup location. This will also help you verify that you can access the backup location.

Backing up the Power SCADA passwords and device profiles

Because Power SCADA passwords and device profiles cannot be automatically backed up, you must back them up manually.

> **NOTE:** The encrypted file that stores this password cannot be transferred from one machine to another, so it is very important that you store this password somewhere secure where it can be retrieved.

Power SCADA Server password

⚠ WARNING

POTENTIAL COMPROMISE OF SYSTEM AVAILABILITY, INTEGRITY, AND CONFIDENTIALITY

Use cybersecurity best practices for password creation and management.

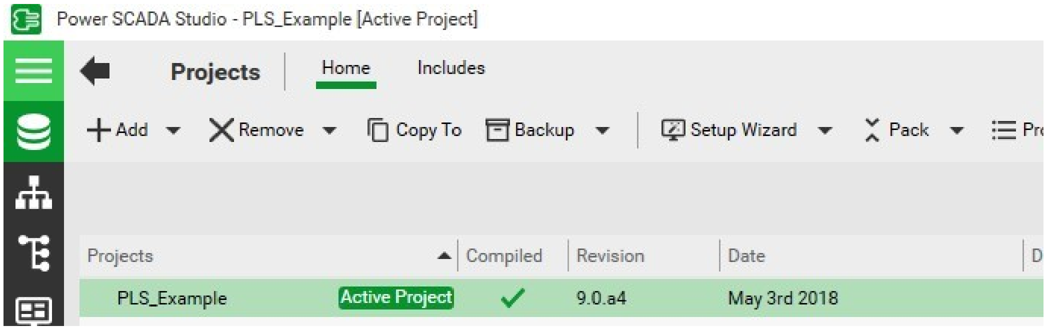
Failure to follow these instructions can result in death, serious injury, equipment damage, or permanent loss of data.

Cybersecurity policies that govern passwords vary from site to site. Work with the facility IT System Administrator to ensure that password management adheres to the site-specific cyber security policies.

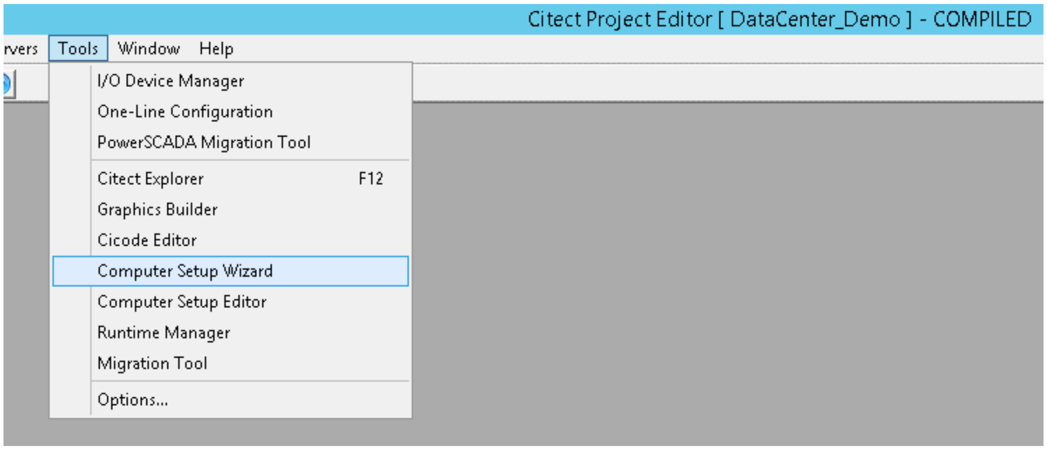
To back up the Power SCADA Server password:

1. Open the Citect Computer Setup Wizard.

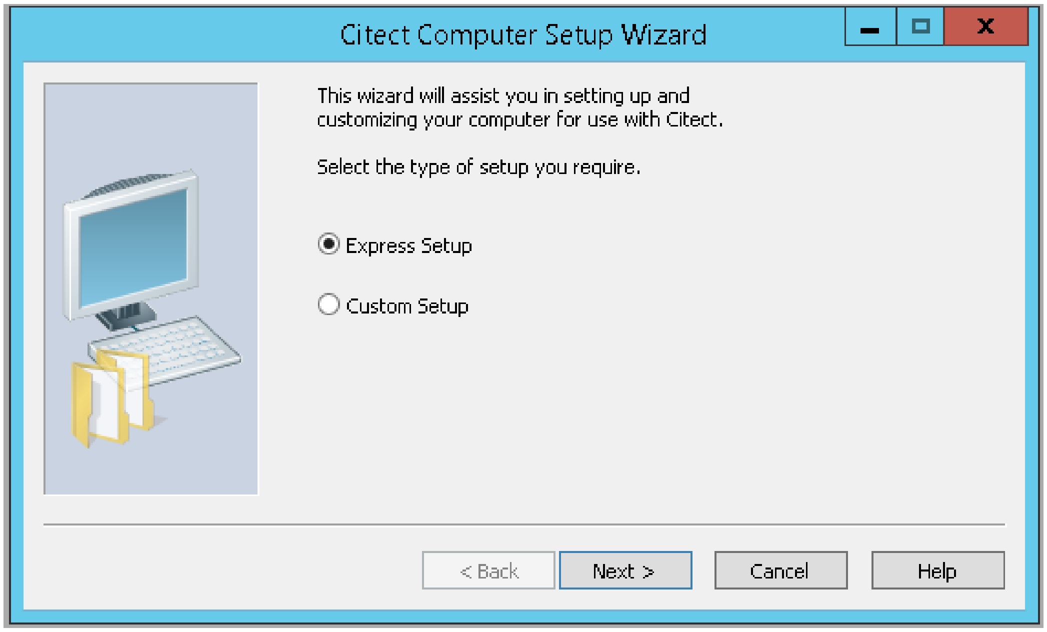
   A. Power SCADA Operation 2020, Power SCADA Operation 9.0, or PowerSCADA Expert 8.2:
   In Power SCADA Studio, in the Projects menu, click **Setup Wizard**.



   B. Power SCADA Expert 8.1:
   In the Project Editor, click **Tools** > **Computer Setup Wizard**.



2. Select **Express Setup** and click **Next**.



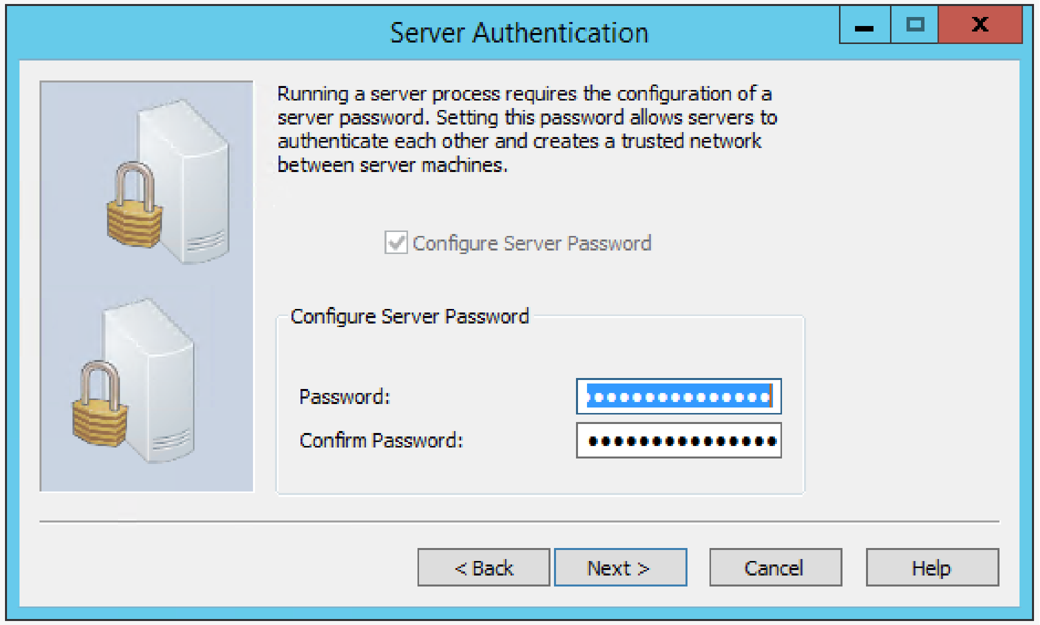
This wizard will assist you in setting up and customizing your computer for use with Citect.

Select the type of setup you require.

● Express Setup

○ Custom Setup

3. Update the password and save it to the backup directory location.



Running a server process requires the configuration of a server password. Setting this password allows servers to authenticate each other and creates a trusted network between server machines.

☑ Configure Server Password

Configure Server Password

Password: ●●●●●●●●●●●●●●●

Confirm Password: ●●●●●●●●●●●●●●●

Device Profiles

1. In Windows Explorer, navigate to the folder containing the device profiles. The default location is:
   C:\ProgramData\Schneider Electric\PowerSCADA Expert\vx.x\Applications\Profile Editor
2. Copy the entire folder and then paste it to the backup directory location.
3. To restore profiles, copy the backed-up device profiles from the backup location to the following location on the Destination Server:
   C:\ProgramData\Schneider Electric\PowerSCADA Expert\vx.x\Applications\Profile Editor

Backing up redundant Power SCADA systems

For a redundant system only, on the secondary system repeat Backing Up Power SCADA automatically.

Backing up Power Monitoring Expert

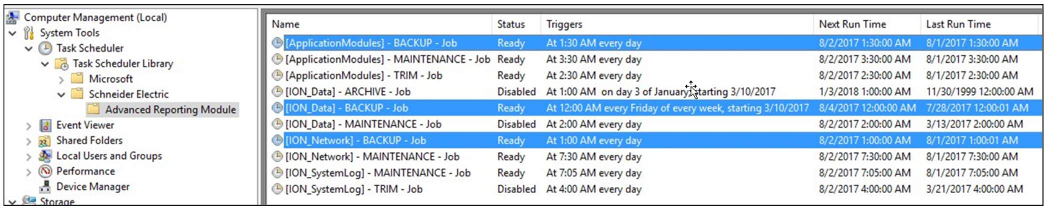
This description of backing up Power Monitoring Expert (PME):

- Applies to a standalone server implementation.
- Expects that the new PME server has the same name as the original.
- Expects that the same version of PME and SQL Server are used on both the new and old servers.
- Does not include the following items which can be configured again for a new instance:
  - Dashboard Images
  - Dashboards
  - Data Center Edition Images
  - Database archives
  - EWS config
  - Report Subscriptions
  - SQL Script Files

Power Monitoring Expert databases

The following PME databases must be backed up to a location off the PME server:

- ION_Data (PME backs up weekly)
- ION_Network (PME backs up daily)
- ApplicationModules (PME backs up daily)



Name	Status	Triggers	Next Run Time	Last Run Time
[ApplicationModules] - BACKUP - Job	Ready	At 1:30 AM every day	8/2/2017 1:30:00 AM	8/1/2017 1:30:00 AM
[ApplicationModules] - MAINTENANCE - Job	Ready	At 3:30 AM every day	8/2/2017 3:30:00 AM	8/1/2017 3:30:00 AM
[ApplicationModules] - TRIM - Job	Ready	At 2:30 AM every day	8/2/2017 2:30:00 AM	8/1/2017 2:30:00 AM
[ION_Data] - ARCHIVE - Job	Disabled	At 1:00 AM on day 3 of January, starting 3/10/2017	1/3/2018 1:00:00 AM	11/30/1999 12:00:00 AM
[ION_Data] - BACKUP - Job	Ready	At 12:00 AM every Friday of every week, starting 3/10/2017	8/4/2017 12:00:00 AM	7/28/2017 12:00:01 AM
[ION_Data] - MAINTENANCE - Job	Disabled	At 2:00 AM every day	8/2/2017 2:00:00 AM	3/13/2017 2:00:00 AM
[ION_Network] - BACKUP - Job	Ready	At 1:00 AM every day	8/2/2017 1:00:00 AM	8/1/2017 1:00:01 AM
[ION_Network] - MAINTENANCE - Job	Ready	At 7:30 AM every day	8/2/2017 7:30:00 AM	8/1/2017 7:30:00 AM
[ION_SystemLog] - MAINTENANCE - Job	Ready	At 7:05 AM every day	8/2/2017 7:05:00 AM	8/1/2017 7:05:00 AM
[ION_SystemLog] - TRIM - Job	Disabled	At 4:00 AM every day	8/2/2017 4:00:00 AM	3/21/2017 4:00:00 AM

For details, see **Scheduled jobs** in the **Windows Task Scheduler** section of the PME System Guide.

Power Monitoring Expert config folder

To reproduce the system in the case of a catastrophic failure, a copy of the \config folder should be stored in an off PME server location. This holds all the files that make your PME system unique. The copy of the \config folder only needs to be done once unless subsequent changes are made to the system.

Power Monitoring Expert diagnostics

Run a diagnostic capture using the Diagnostics Tool available in PME. Doing so stores information about the source PME system—including server specifications and OS/SQL versions—that are necessary to rebuild the system. Store the resulting .cab file in the off PME server location.

Deleting old backups

Backups can quickly fill up your backup directory drive. You can automate your system to delete old backups. The following procedure demonstrates how to delete backup files that are older than 15 days.

To delete backup files that are 15 days or older:

1. In a text editor, open **Delete_old_Backups.ps1**
2. Edit the file for your system:
   a. For Power SCADA backups: enter the path of the PSE_Backups folder. For example: C:\PSE\PSE_Backups
3. Save the file as **Delete_old_Backups.ps1** under:
   C:\Program Files (x86)\Schneider Electric\BackupTasks
4. Create a Windows scheduled task to trigger this script to run weekly once at midnight.
5. Define the new action:
   a. For **Action**, click **Start a program**.
   b. In **Program/script**, enter the following path:
   C:\Windows\syswow64\WindowsPowerShell\v1.0\powershell.exe
   c. In **Add arguments**, enter the following:
   -noninteractive -nologo -file " C:\Program Files (x86)\Schneider Electric\BackupTasks\Delete_old_Backups.ps1"

Restoring a Power SCADA system

This section describes the tasks that are required to restore a Power SCADA system on a new Destination server.

NOTE: After restoring, you may need to manually reconfigure a system management server, deployment server, and TLS certificate management. For more information, see the Citect Help **Post Installation Configuration** section.

Restoring Power SCADA

The following must be verified before restoring the backups on the Destination Power SCADA Server:

• Power SCADA is installed and working on the designated Destination Server.

• The licenses are activated on the destination system.

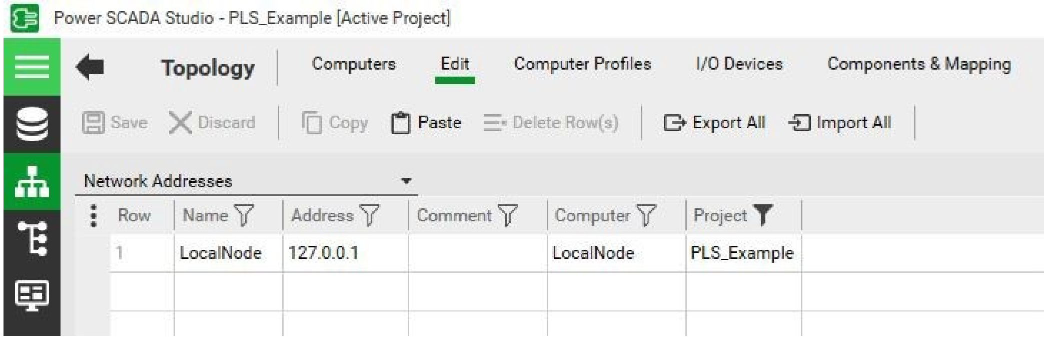
• All the relevant software and OS updates have been applied to the Destination Server.

Restoring Power SCADA from an automated backup

If you have a redundant system, you must also restore the redundant servers. See Restoring a Redundant Power SCADA system for details.

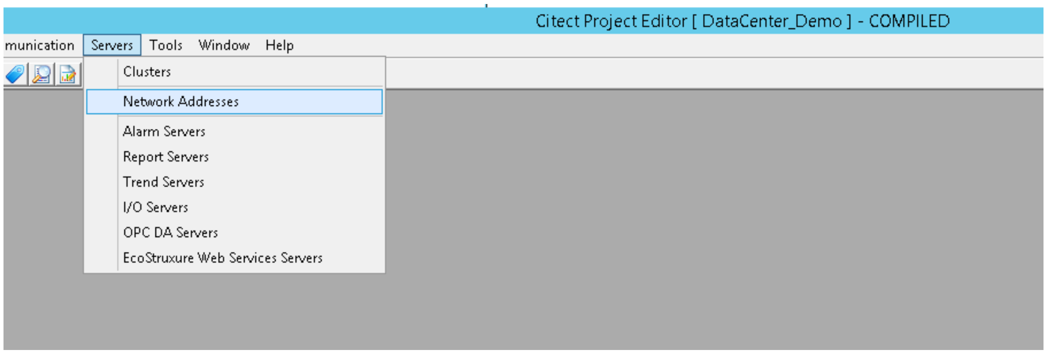
To restore Power SCADA from an automated backup on the Destination Server:

1. Copy the PSEvx.x_Restore.ps1 script on to the server.
2. Edit the file for your system:
   a. For **<$sourceDir>**, enter your backup directory location.
   b. If you did not install to the default install location, edit the **<pseBin>** and **<$pseData>** to reflect the installed location path.
3. Run PowerShell as Administrator and set the execution policy with the following cmdlet: `Set-ExecutionPolicy Bypass.`
4. Right-click on the script file and run with PowerShell.
5. If the server IP address changed, update the IP address:
   A. Power SCADA Operation 2020 or Power SCADA Expert 8.2:
   In Studio, click **Topology** > **Edit** and then select **Network Addresses** from the drop-down menu:



Row	Name	Address	Comment	Computer	Project
1	LocalNode	127.0.0.1		LocalNode	PLS_Example

   B. PowerSCADA Expert 8.1:
   In the Project Editor, click **Servers** > **Network Addresses**.



6. Update the IP address.
7. Compile the project.

Restoring Redundant Power SCADA Systems

For redundant Power SCADA systems, the following must be performed on the Destination Secondary server:

1. Copy **PSEvx.x_Restore.ps1** script and backup directory to secondary server.
2. Edit the file for your system:
   a. For **<$sourceDir>**, enter your backup directory location.
   b. If you did not install Power SCADA to the default install location, edit the **<$pseBin>** and **<$pseData>** to reflect the installed location path.
3. Run PowerShell as Administrator and set the execution policy with the following cmdlet: `Set-ExecutionPolicy Bypass`.
4. Right-click on the script file and run with PowerShell.
5. Once all the files are restored, pack and compile the project.

Restoring Power Monitoring Expert

This section includes the tasks required to restore Power Monitoring Expert (PME) backups on the Destination server.

Verify the following before restoring the backups on the Destination PME Server:

- PME is installed and working on the new PME Server.
- The new server should have the same name as the original PME Server.
- The SQL Server version of the new system needs to be the same or newer as that of the old system.
- The licenses are activated on the new system.
- All the relevant software and OS updates have been applied to the new PME Server.

Replace the Config Folder

1. Stop all ION Services:
   a. In Control Panel\Administrator Tools\Services, stop the **ION Network Router** service. This stops all ION services.
   b. Stop the **ApplicationModules CoreServicesHost** service. This stops all ApplicationsModules services.
2. Copy the backed up \config folder to the C:\Program Files (x86)\Schneider Electric\Power Monitoring Expert folder. This will update the \config folder with all the files that were unique to the original installation.
3. Leave the ION and ApplicationsModules services stopped to restore a database (Restoring the Databases from the Old System).

Connect the old databases

You need to detach, and then remove, rename, or delete the factory (new) database files before you can connect the old database files to the new system.

Detach the default databases

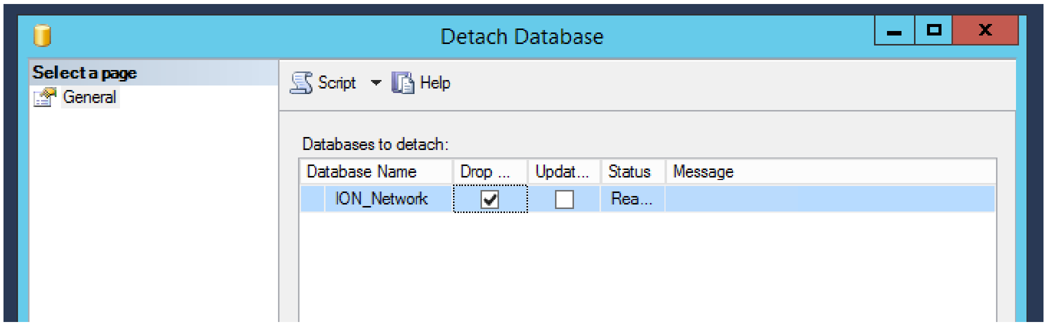
Before you can remove, rename, or delete the factory installed ION_Data, ION_Network, and ApplicationModules databases on the new PME system with the copies made from the old PME system, you must first detach the factory installed databases.

To detach the ION_Data database from the new system:

1. Go to SQL Server Management Studio and right-click ION_Data > Tasks > Detach.

   NOTE: If you cannot detach a database because of active connections, click Drop Connections in the detach dialog in SQL Server Management Studio.

   For example:



Databases to detach:

Database Name	Drop ...	Updat...	Status	Message
ION_Network	☑	□	Rea...	

2. Follow the same steps to detach the ION_Network and ApplicationModules databases.

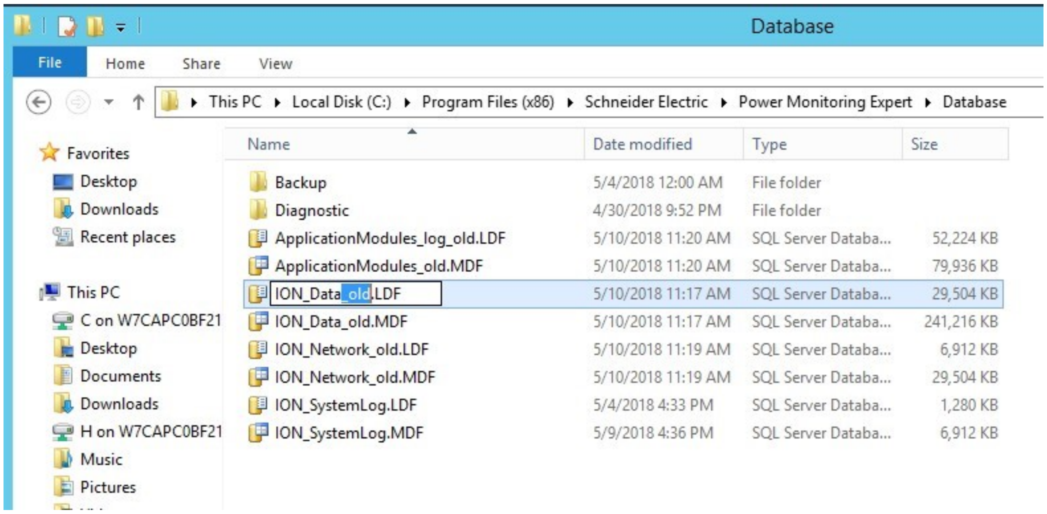
Remove, rename, or delete the factory databases

Detaching the databases will not remove the database files from the new PME folder structure. Since the old databases that you need to upgrade must be in the same location as the factory ones, you need to delete, remove, or rename the factory database files.

1. Navigate to the location where you installed the factory database files.

   NOTE: The default installation location is ...\Schneider Electric\Power Monitoring Expert\Database. However, you might have picked a different location during installation.

2. Rename the factory database files. For example:



Name	Date modified	Type	Size
Backup	5/4/2018 12:00 AM	File folder	
Diagnostic	4/30/2018 9:52 PM	File folder	
ApplicationModules_log_old.LDF	5/10/2018 11:20 AM	SQL Server Databa...	52,224 KB
ApplicationModules_old.MDF	5/10/2018 11:20 AM	SQL Server Databa...	79,936 KB
ION_Data_old.LDF	5/10/2018 11:17 AM	SQL Server Databa...	29,504 KB
ION_Data_old.MDF	5/10/2018 11:17 AM	SQL Server Databa...	241,216 KB
ION_Network_old.LDF	5/10/2018 11:19 AM	SQL Server Databa...	6,912 KB
ION_Network_old.MDF	5/10/2018 11:19 AM	SQL Server Databa...	29,504 KB
ION_SystemLog.LDF	5/4/2018 4:33 PM	SQL Server Databa...	1,280 KB
ION_SystemLog.MDF	5/9/2018 4:36 PM	SQL Server Databa...	6,912 KB

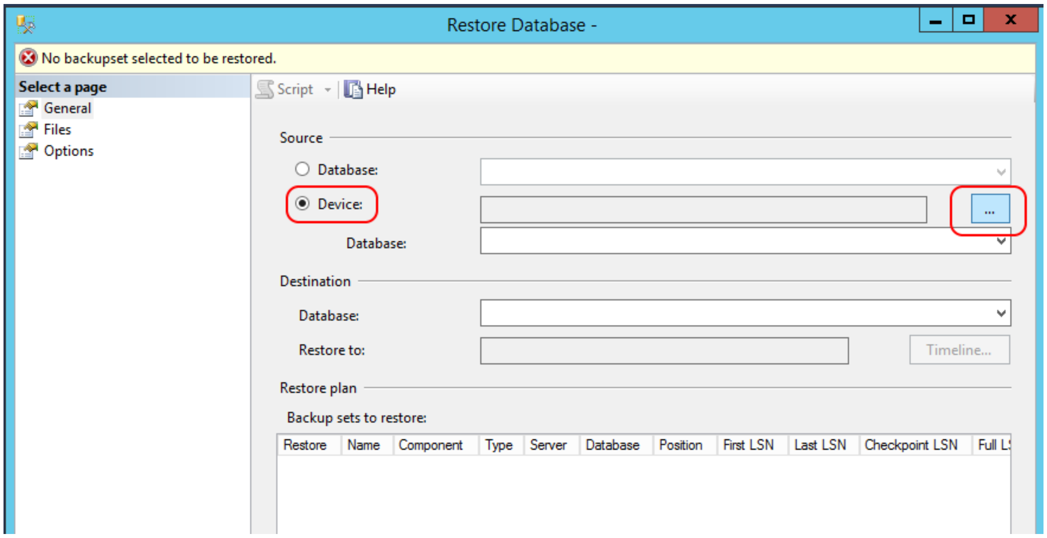
Restore the databases from the old system

Restore the backed-up databases to the new PME system at their installation location.

To restore copies of the backed up PME databases into the new PME system:

1. In SQL Server Management Studio, right-click **Databases** and then click **Restore Database**.

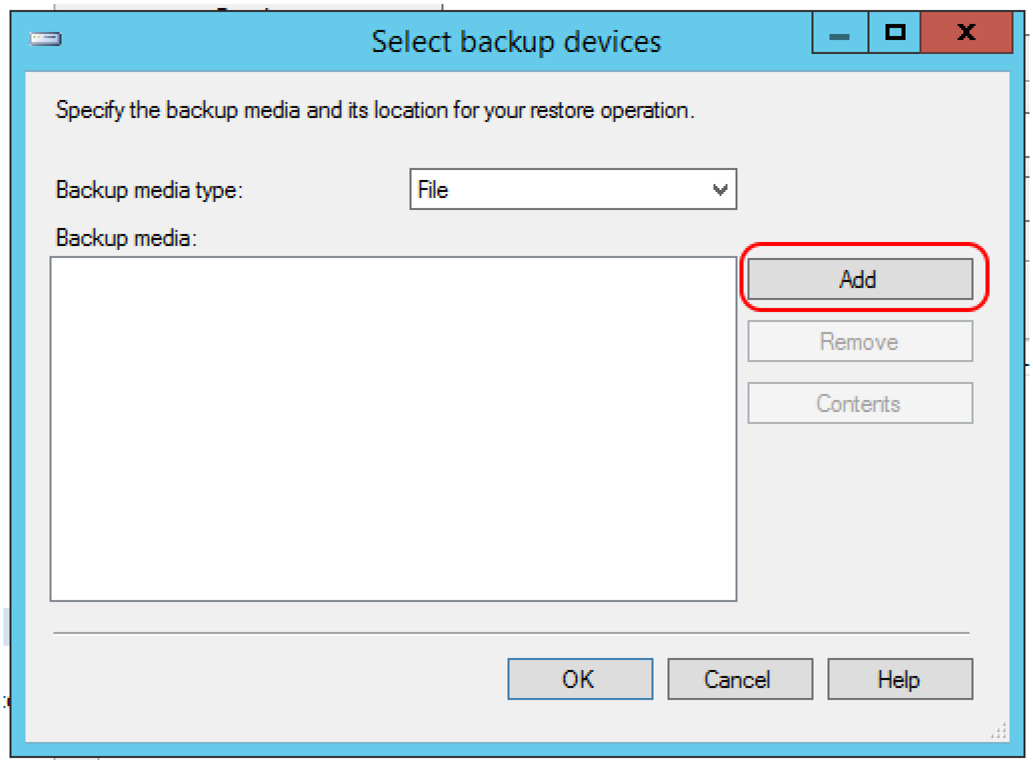
2. Click **Device** and then click the **...**(ellipsis) button.



Restore Database -

No backupset selected to be restored.

3. Click **Add**.



Select backup devices

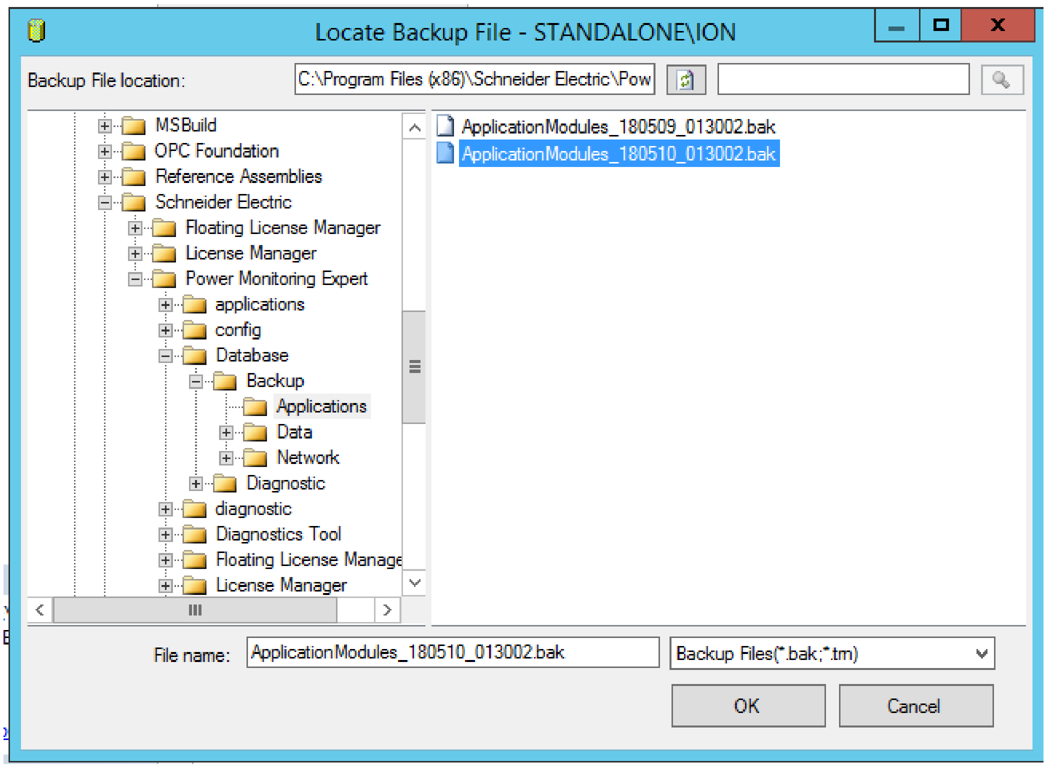
Specify the backup media and its location for your restore operation.

Backup media type: File

Backup media:

4. In the Locate Backup File dialog, navigate to the location where the backup database files are stored and enter the database name in the **File name** field.

For example:



Backup File location: C:\Program Files (x86)\Schneider Electric\Pow

ApplicationModules_180509_013002.bak

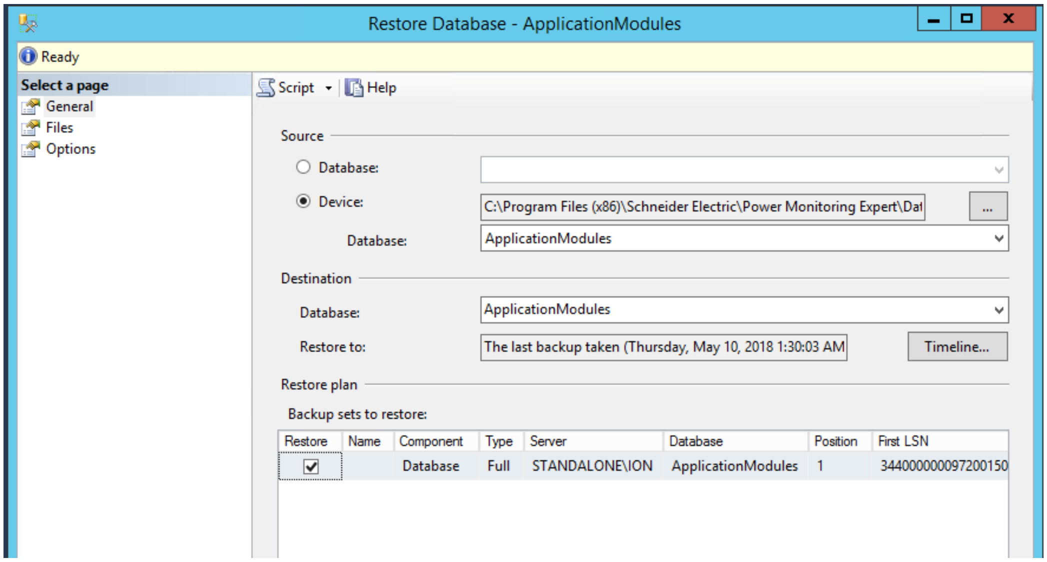
ApplicationModules_180510_013002.bak

File name: ApplicationModules_180510_013002.bak

Backup Files(*.bak;*.tm)

5. Click **OK**.

6. Make sure that **Restore** is checked:



○ Database:

● Device: C:\Program Files (x86)\Schneider Electric\Power Monitoring Expert\Dat

Database: ApplicationModules

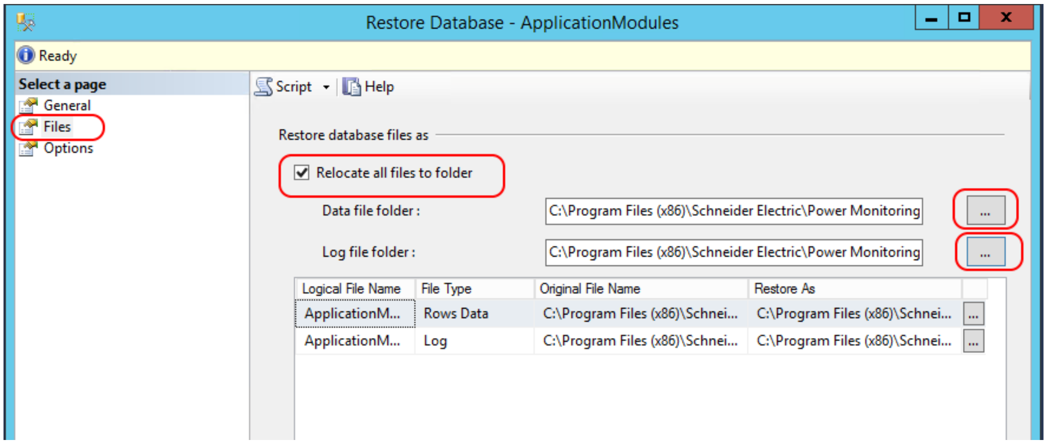
Destination Database: ApplicationModules

Restore to: The last backup taken (Thursday, May 10, 2018 1:30:03 AM

Backup sets to restore:

Restore	Name	Component	Type	Server	Database	Position	First LSN
☑		Database	Full	STANDALONE\ION	ApplicationModules	1	344000000097200150

7. Click on the **Files** tab on the upper left of the window, select **Relocate all files to folder**, and then select the location for restoring the database:



Restore database files as

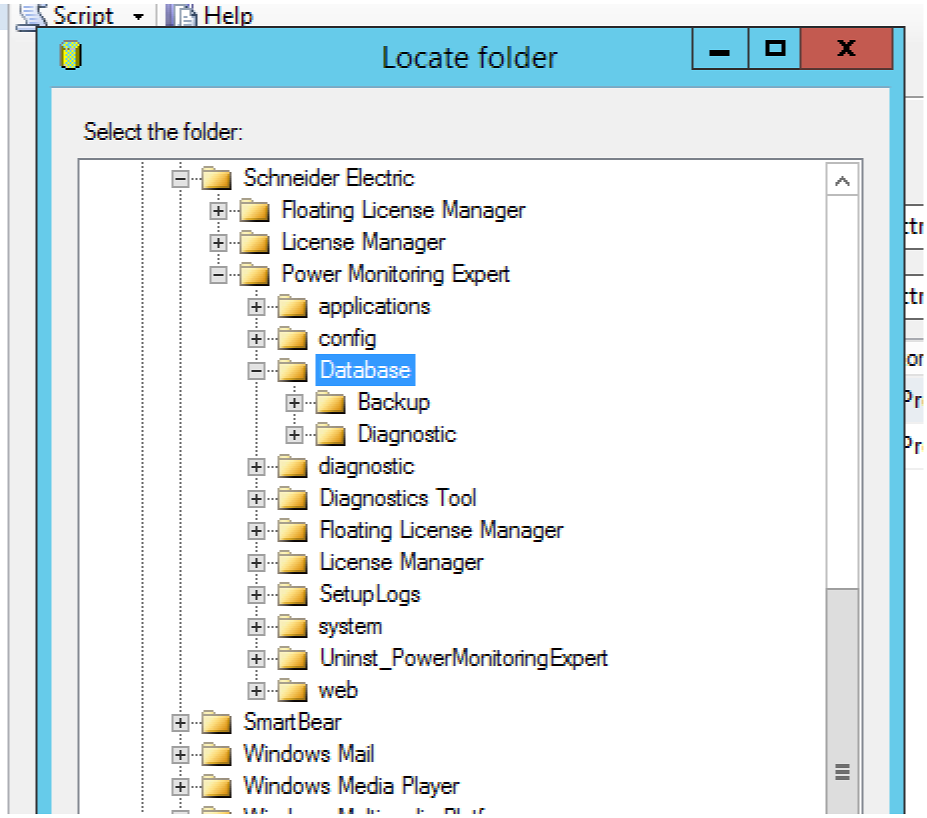
☑ Relocate all files to folder

Data file folder : C:\Program Files (x86)\Schneider Electric\Power Monitoring

Log file folder : C:\Program Files (x86)\Schneider Electric\Power Monitoring

Logical File Name	File Type	Original File Name	Restore As
ApplicationM...	Rows Data	C:\Program Files (x86)\Schnei...	C:\Program Files (x86)\Schnei...
ApplicationM...	Log	C:\Program Files (x86)\Schnei...	C:\Program Files (x86)\Schnei...

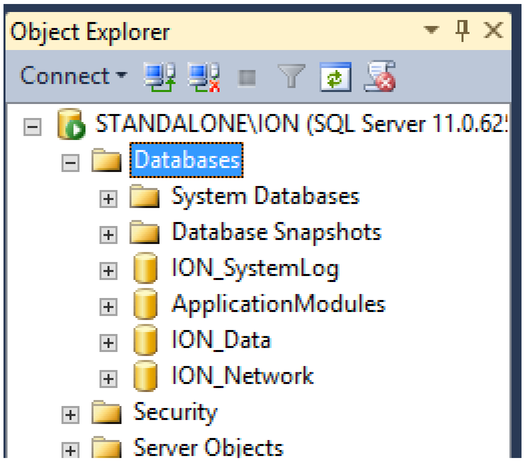
8. Select the default location for the PME databases. For example:



9. Click **OK**.

10. Repeat Steps 1 to 9 for the remaining database files (ION_Data and ION_Network).

The following image shows the restored databases:



Object Explorer

Connect

- STANDALONE\ION (SQL Server 11.0.62
  - Databases
    - System Databases
    - Database Snapshots
    - ION_SystemLog
    - ApplicationModules
    - ION_Data
    - ION_Network
  - Security
  - Server Objects

Start the Power Monitoring Expert services

This is best done by rebooting the Power Monitoring Expert server. Alternatively, you could manually restart all Automatic startup type Application Module and ION Services.

Post-restoration checks

1. Check the system log in the Management Console for errors.
2. Log in to Web Applications and ensure all applications work. For example, run the **System Configuration Report**.
3. Check the Vista diagrams and correct any Query Server or VIP links, if necessary.

Backup and restore scripts

Refer to www.se.com or the Exchange Extranet for the backup and restore scripts specific to your Power SCADA system version.

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 187.

- 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 "Activating licenses using the Floating License Manager" on page 189)

In both cases, Power SCADA Operation uses a "Dynamic Point Count" on page 189 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 computer. See "Specify the required point count for a computer" on page 190.

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 Client.

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.

Activating 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 the 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 variable tag is only counted towards your point count the first time it is requested. Even if you configured a certain tag on a 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: An 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 available. 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. If you write to more than 1 point through these interfaces, it will swap to the 15-minute demo mode.
- 8 days with unlimited tags. This is only available through special Power SCADA Operation Development keys.

Configure

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

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

Section	Description
"Configuration prerequisites" on page 193	Things to consider to help you prepare for configuring a Power SCADA project
"Configuration tools" on page 195	An introduction to the Power SCADA Operation configuration tools
"Power SCADA Projects" on page 206	Creating a Power SCADA project using Project Setup, as well as compiling, backing up, an restoring a project.
"Devices" on page 221	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 311	How to configure and work with: • Graphics pages • Animated one-lines • Menus and pages • Basic reports • LiveView • Notifications
"Web Applications" on page 368	How to configure and work with: • Alarms • Diagrams • Web Applications settings
"Assign and control user privileges" on page 686	How to configure and manage user access
"Cybersecurity" on page 592	Cybersecurity considerations including: two-factor authentication (one-time password) and McAfee white listing.
"Customize default behaviors " on page 607	How to use Cicode to customize a project, localizing a project, and running PSO as a Windows Service.

Section	Description
"System startup and validation checks" on page 617	How to validate your configured system on startup.
"Distributed systems" on page 622	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 677	How to configure a redundant server.

Configuration prerequisites

- Review the system development process provided in this document.
- Gather the supporting documents that you may need. See "Resources" on page 39 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 Plan.
- 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 built 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.

  NOTE: It is recommended that you run your system in normal mode. When possible, refrain from running applications in Administrator mode to help prevent shellcode from being successfully executed.
- **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 1238 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 808
  - "EcoStruxure Web Services setup" on page 644
- **Application Services Host:**
  - "Application Services Host—Citect Data Platform" on page 197: 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 317: Use this screen to set up the delivery mode and email address from which Power SCADA Operation 2020 basic reports will be sent.
  - "One-Line Engine configuration" on page 1236: 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 808
  - "Application Services Logging" on page 808
- **Security:**
  - "Two-Factor Authentication" on page 593
  - "Configure Single Sign-On (SSO)" on page 625

Application Services Host—Citect Data Platform

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

Use this page to link a Power SCADA 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 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 > Citect Data Platform > Connection** tab.
3. In **Citect I/O Server Address** choose the server address for the project that is running. If Citect requires encryption, this must be the computer name.
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.
7. To set up web redundancy, select the **Key Management** tab. Use the **Export Key** and **Import Key** buttons to save an encryption key and export it to another computer as an AES file. This supports the token validation key for redundant web clients. Please keep this file secured at all times.

NOTE: To provide extra security you can run Power SCADA as a service. Both Power SCADA and CoreServiceHost must be running as a service 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.
5. Click **Test Credentials**, and the test will fail. However, you can verify that the service has started by viewing the Event Log.

NOTE: If unable to successfully connect to Power SCADA, check the following:

1. In `Citect.ini`, set `[ctAPI] Remote = 1` if required.
2. In the Power SCADA Studio, ensure the Network Address 'Address' field is set to a computer name or IP Address. Localhost or 127.0.0.1 is not compatible with Citect encryption enabled.
3. In the Power SCADA Studio, ensure the Computer 'DNS Name' field is set to a computer name.
4. Run the Computer Setup wizard and ensure the 'Network Setup' is configured properly for the system. If 'Networked' is selected, ensure the 'Address Type' and 'Address Scope' is correct.
5. Changes to the project, ini settings, or Computer Setup Wizard requires the project to be restarted.

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.

Configuring service layer components

You must configure Power SCADA Operation to use the service layer components. For calls to be forwarded properly from service-to-service, configure the Web Application, PsoWebService, and PlatformServer components as described below.

Configuring the web application

If you want to change the machine name or port, you must configure the web application to know how to reach PsoWebService. The web application consists of two separate IIS services: PsoDataService and WebHmi.

To edit the web application configuration:

1. 1. Navigate to the respective web.config files:
   C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020 R2\Applications\Web\SystemDataService\Web.config
   and:
   C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020 R2\Applications\Web\WebHMI\Web.config
2. In the appSettings section of each web.config file, find the line:
   `<add key=”PsoWebService” value=”localhost:23200”/>`
   This value is the address at which the web application can contact PsoWebService, in the form
   `IPAddress:Port`.
3. Configure the IP address with the same machine name as the one used to generate the SSL certificates.

Configuring PsoWebService

You must configure PsoWebService to know its own address.

To edit the PsoWebService configuration:

1. Navigate to the appsettings.json file:
   C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020 R2

\Applications\Services\Pso Webservice

2. Configure the **Endpoint**, **ConnectionTimeoutSeconds**, and **Host** sections as follows:

- **Endpoint:** Use the same machine name and port configured in the web.config file above.
- **ConnectionTimeoutSeconds:** PsoWebService constantly receives calls from each connected PlatformServer. Set the number of seconds after which PsoWebService will consider a PlatformServer to be disconnected. If a PlatformServer doesn't call PsoWebService within that time frame, it is marked `Disconnected`.
- **Host:** Use the same machine name as Endpoint. The port numbers should not change unless there is a port conflict.

> **NOTE**: If there is a port conflict, PsoWebService will not start, and will log a message stating the application has been terminated. To resolve the issue, reconfigure PsoWebService to use a different port.

Configuring PlatformServer

Because the PlatformServer communicates with Citect to read data by default, you do not need to configure it.

To edit the PlatformServer configuration:

1. **SimulationMode:** `false`.

   > **NOTE**: Do set to anything other than `false` in a production environment.

2. **Identity / Endpoint:** Set to the `IPAddress:Port` of the machine where the PlatformServer is running.

3. **Responsibilities:** `"RealtimeData"`, `"AlarmData"`, `"DocumentData"`, `"EquipmentData"`, `"AuthenticationData"`, and `"WaveformData"`. Details what types of calls from PsoWebService this PlatformServer can handle. When you configure a responsibility, it advertises this information to its PsoWebService(s) by a ping call that PlatformServer makes up to PsoWebService. This information is recorded by PsoWebService so that it knows if it can route particular kinds of calls to the PlatformServer.

   > **NOTE**: There are currently six types of responsibilities, and their **RequestType** values must be spelled exactly as they appear above (case-sensitive). Each responsibility also has a Priority value, where 1 is the lowest priority, and a lower value indicates a higher priority. For example, if you have three PlatformServer instances on three separate machines, you could configure one to be “`Priority”: 1`, another `2`, and the other `3`. When a call comes in for that type of request to PsoWebService, if it sees multiple PlatformServer instances that service that type of request, it will send the request to the PlatformServer that has the lowest numerical value for `“Priority”`.

4. **PingInterval:** A number in units of seconds that indicates how much time passes between ping calls to the PsoWebServices.

5. **WebServices:** A json array of `IPAddress:Port` addresses of PsoWebService instances to which the PlatformServer should ping or connect. These should match the **Endpoint** values in their corresponding PsoWebService appsettings.json files.

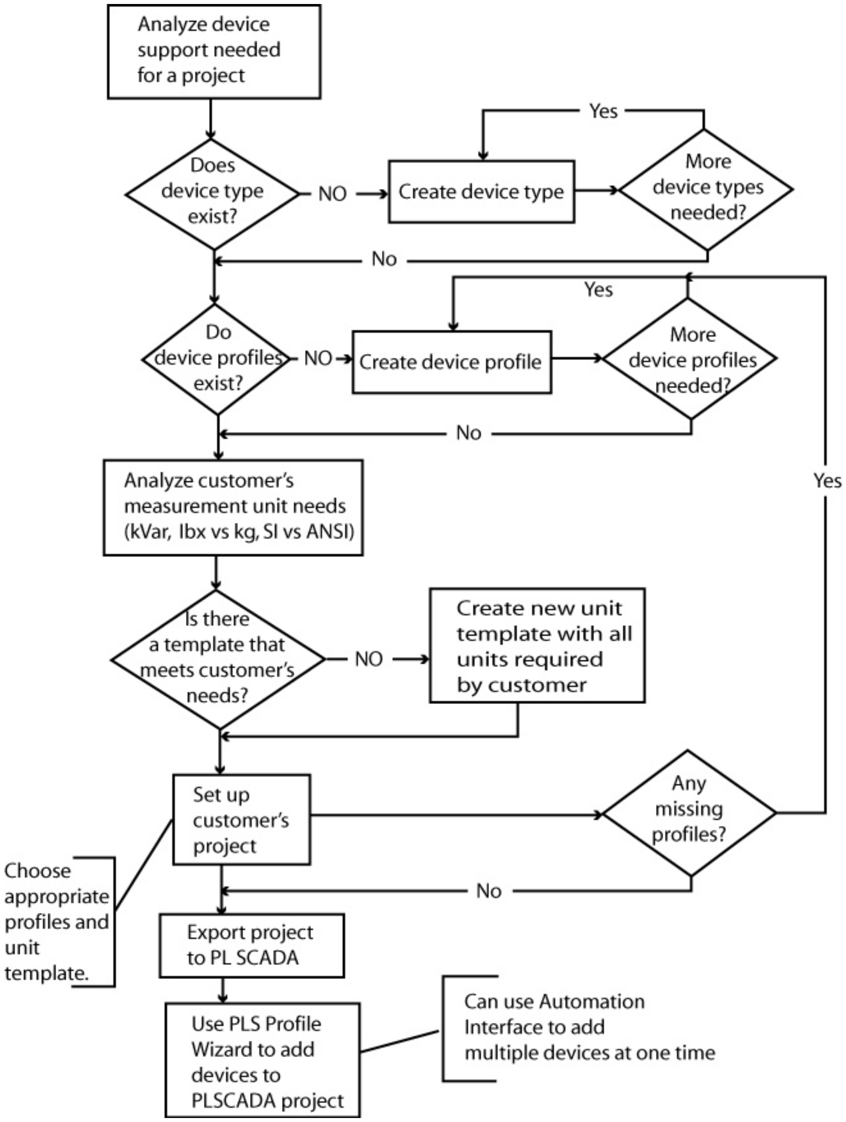
6. **Host:** Configure with the same machine name used under **Endpoint**. The port numbers should not change unless there is a port conflict.

Profile Editor typical workflows

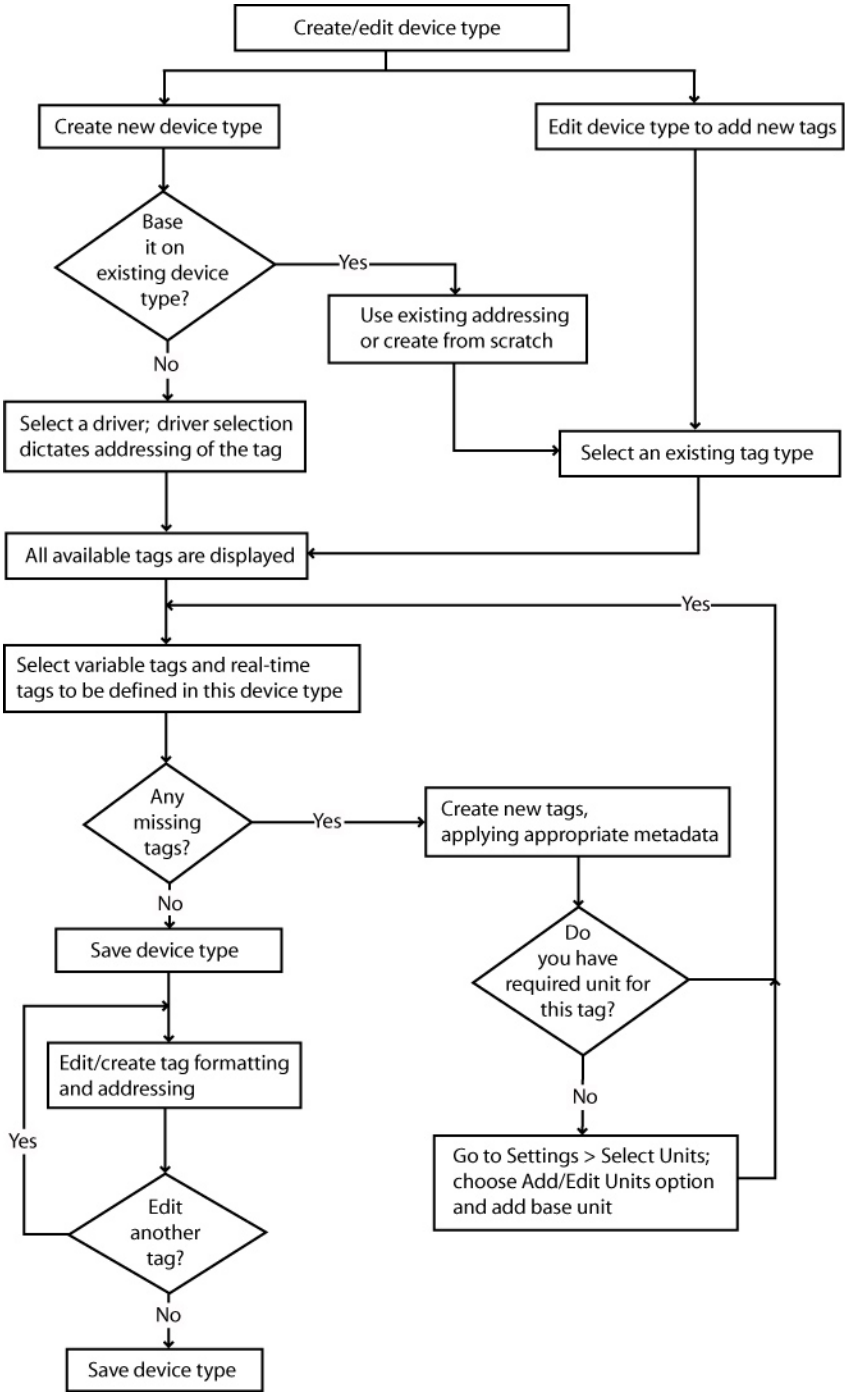
The following flow charts illustrate how to use the Profile Editor. The first illustration provides an overview, while the subsequent workflows show:

- Creating/editing a device type
- Creating/editing a device profile
- Creating/editing unit templates

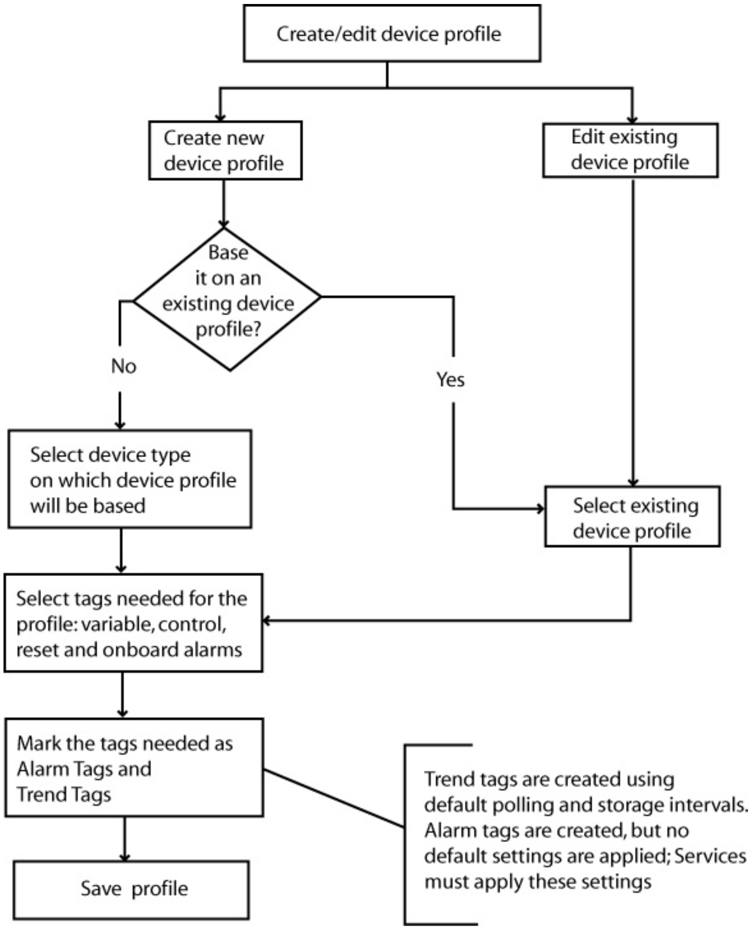
Workflow overview


Analyze device support needed for a project
Does device type exist?
NO
Create device type
More device types needed?
Yes
No
Do device profiles exist?
NO
Create device profile
More device profiles needed?
Yes
No
Analyze customer's measurement unit needs (kVar, lbx vs kg, SI vs ANSI)
Yes
Is there a template that meets customer's needs?
NO
Create new unit template with all units required by customer
Set up customer's project
Any missing profiles?
No
Choose appropriate profiles and unit template.
Export project to PL SCADA
Use PLS Profile Wizard to add devices to PLSCADA project
Can use Automation Interface to add multiple devices at one time

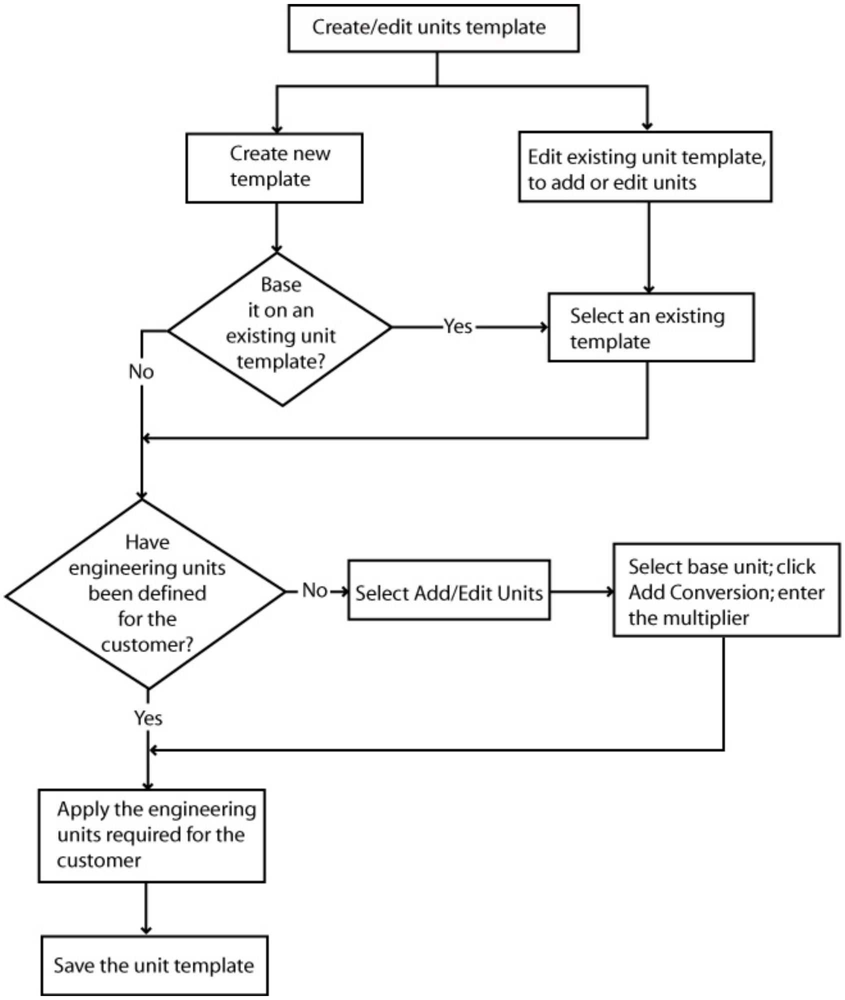
Create/edit device type


Create/edit device type
Create new device type
Edit device type to add new tags
Base it on existing device type?
Yes
Use existing addressing or create from scratch
No
Select a driver; driver selection dictates addressing of the tag
Select an existing tag type
All available tags are displayed
Yes
Select variable tags and real-time tags to be defined in this device type
Any missing tags?
Yes
Create new tags, applying appropriate metadata
No
Save device type
Do you have required unit for this tag?
Edit/create tag formatting and addressing
No
Yes
Go to Settings > Select Units; choose Add/Edit Units option and add base unit
Edit another tag?
No
Save device type

Create/edit device profile


Create/edit device profile
Create new device profile
Edit existing device profile
Base it on an existing device profile?
No
Yes
Select device type on which device profile will be based
Select existing device profile
Select tags needed for the profile: variable, control, reset and onboard alarms
Mark the tags needed as Alarm Tags and Trend Tags
Save profile
Trend tags are created using default polling and storage intervals. Alarm tags are created, but no default settings are applied; Services must apply these settings

Create/edit unit templates


Create/edit units template
Create new template
Edit existing unit template, to add or edit units
Base it on an existing unit template?
Yes
No
Select an existing template
Have engineering units been defined for the customer?
No
Select Add/Edit Units
Select base unit; click Add Conversion; enter the multiplier
Yes
Apply the engineering units required for the customer
Save the unit template

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 279.

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 279
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 288.
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 "Managing device type categories" on page 234 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 951 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:

- Restore a project
- Use the Migration Utility
- Before you add a project
- Add a project using Project Setup
- Compile a project
- Backup a project

In the Citect SCADA help file (`...\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020 R2\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

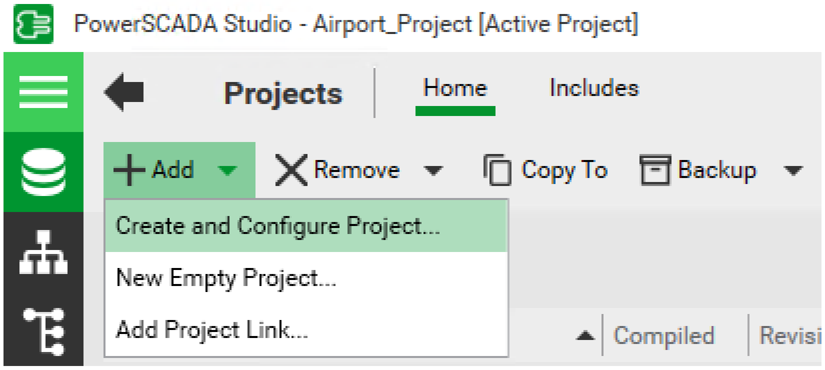
After you create the project and define its features, you can also use Project Setup to change other settings, such as devices in the CSV file, and to update your project.

For a list of project-related parameters that are created using Project Setup, see "Project Setup – Changed Parameters" on page 216

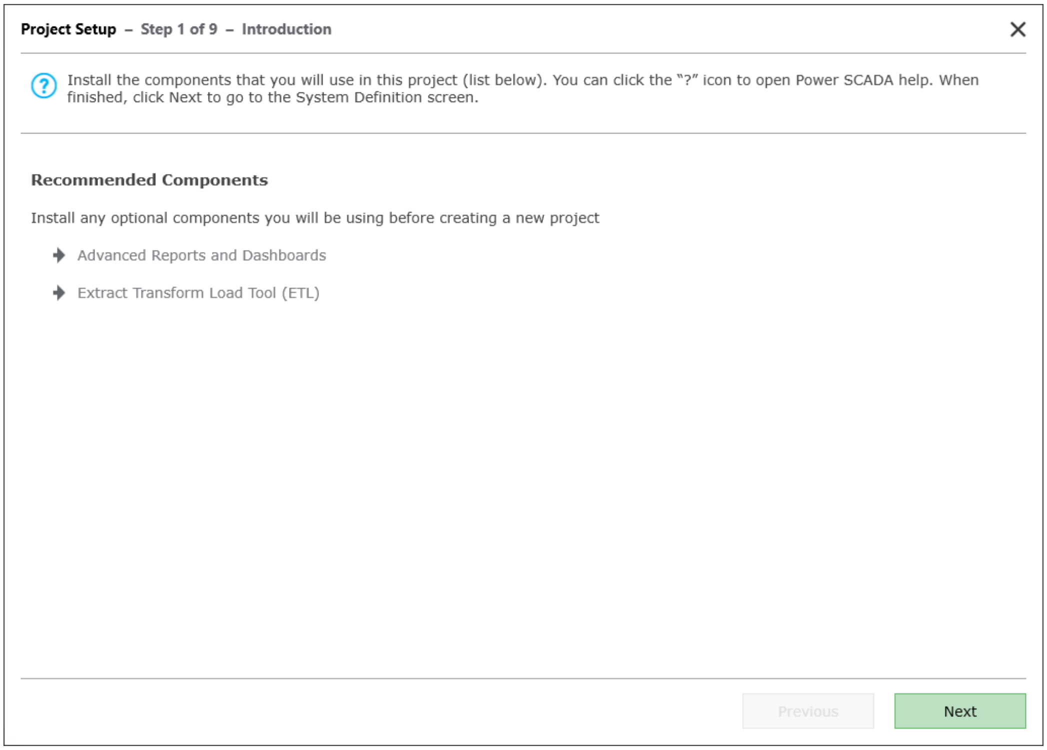
Launch Project Setup

To launch Project Setup:

1. Launch Power SCADA Studio.
2. Click **Projects**, click **Add > Create and Configure Project**.



The Introduction page appears.



Project Setup – Step 1 of 9 – Introduction

Install the components that you will use in this project (list below). You can click the "?" icon to open Power SCADA help. When finished, click Next to go to the System Definition screen.

Recommended Components

Install any optional components you will be using before creating a new project

- Advanced Reports and Dashboards
- Extract Transform Load Tool (ETL)

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 Operation2020 installation media.

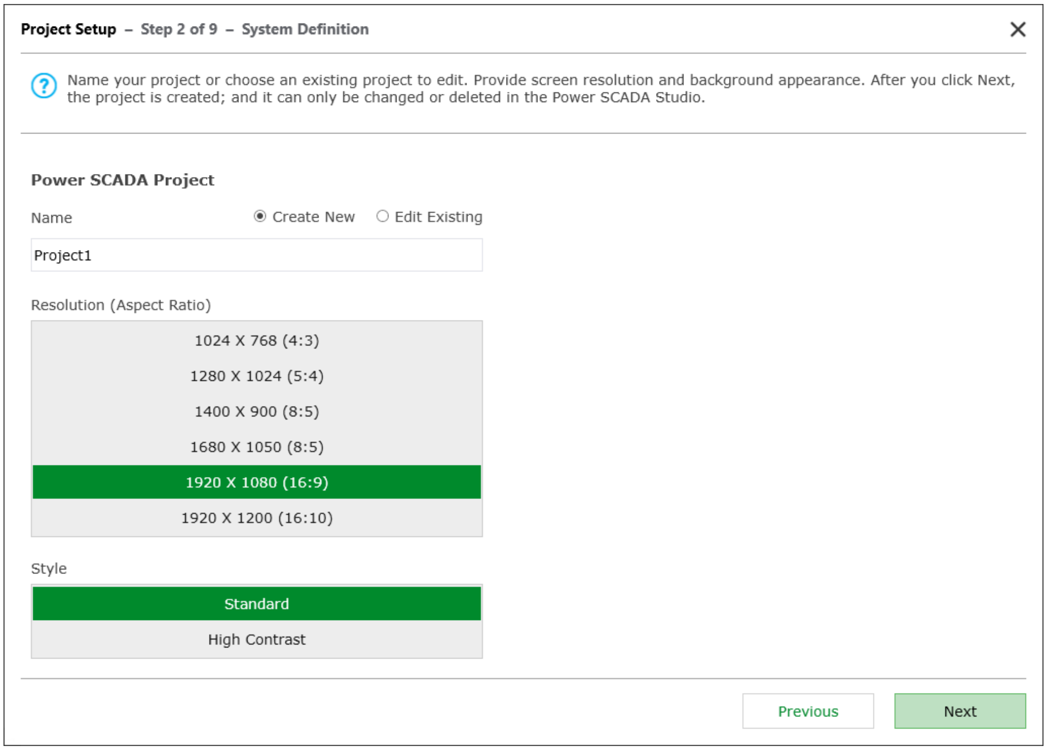
- **Advanced Reports and Dashboards** – Lets you view advanced reports and dashboards from Power Monitoring Expert. Install this component from the Power SCADA Operation installation media.
- **Extract Transform Load tool (ETL)** – Use this component to extract reporting information from Power SCADA 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.



Project Setup – Step 2 of 9 – System Definition

Name your project or choose an existing project to edit. Provide screen resolution and background appearance. After you click Next, the project is created; and it can only be changed or deleted in the Power SCADA Studio.

Power SCADA Project

Name ● Create New ○ Edit Existing

Project1

Resolution (Aspect Ratio)

- 1024 X 768 (4:3)
- 1280 X 1024 (5:4)
- 1400 X 900 (8:5)
- 1680 X 1050 (8:5)
- [✓ 1920 X 1080 (16:9)]
- 1920 X 1200 (16:10)

Style

- [✓ Standard]
- High Contrast

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.

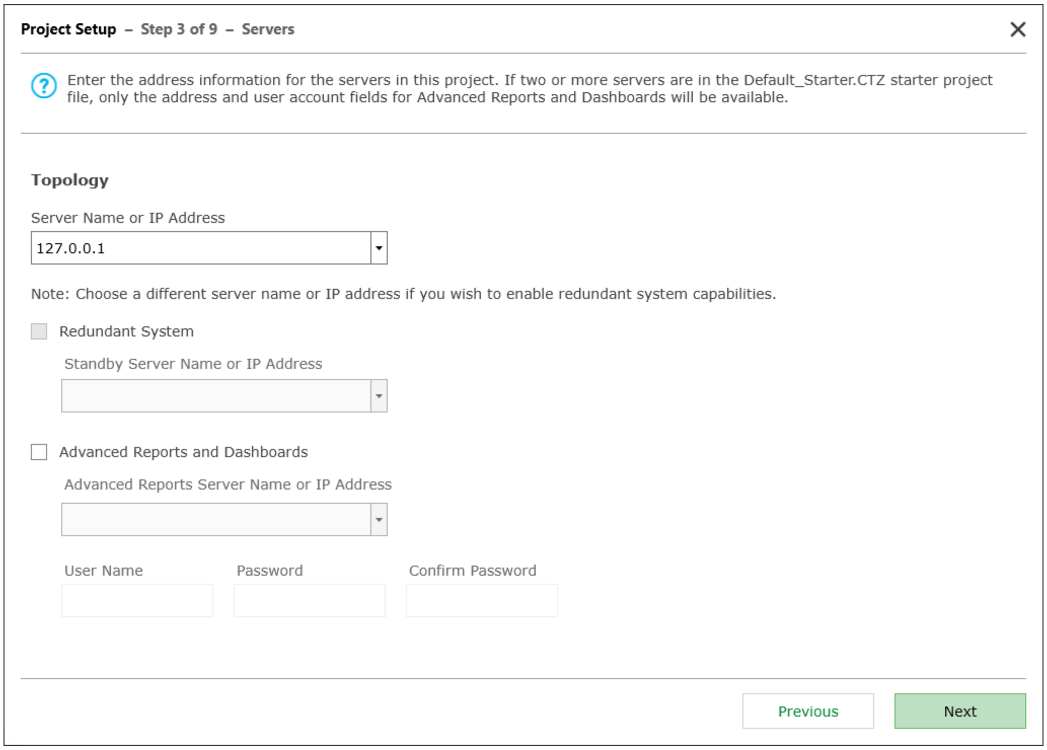
   NOTE: You can also set high contrast using the parameters in the Power SCADA Studio. Open your project in the Power SCADA Studio, then click **Settings > Parameters**. The parameter name is `IsHighContrast`. 0 = standard; 1 = high contrast.
4. Click **Next**.

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.



Project Setup – Step 3 of 9 – Servers

Enter the address information for the servers in this project. If two or more servers are in the Default_Starter.CTZ starter project file, only the address and user account fields for Advanced Reports and Dashboards will be available.

Topology

Server Name or IP Address: 127.0.0.1

Note: Choose a different server name or IP address if you wish to enable redundant system capabilities.

□ Redundant System

Standby Server Name or IP Address

□ Advanced Reports and Dashboards

Advanced Reports Server Name or IP Address

User Name

Password

Confirm Password

To define the server information:

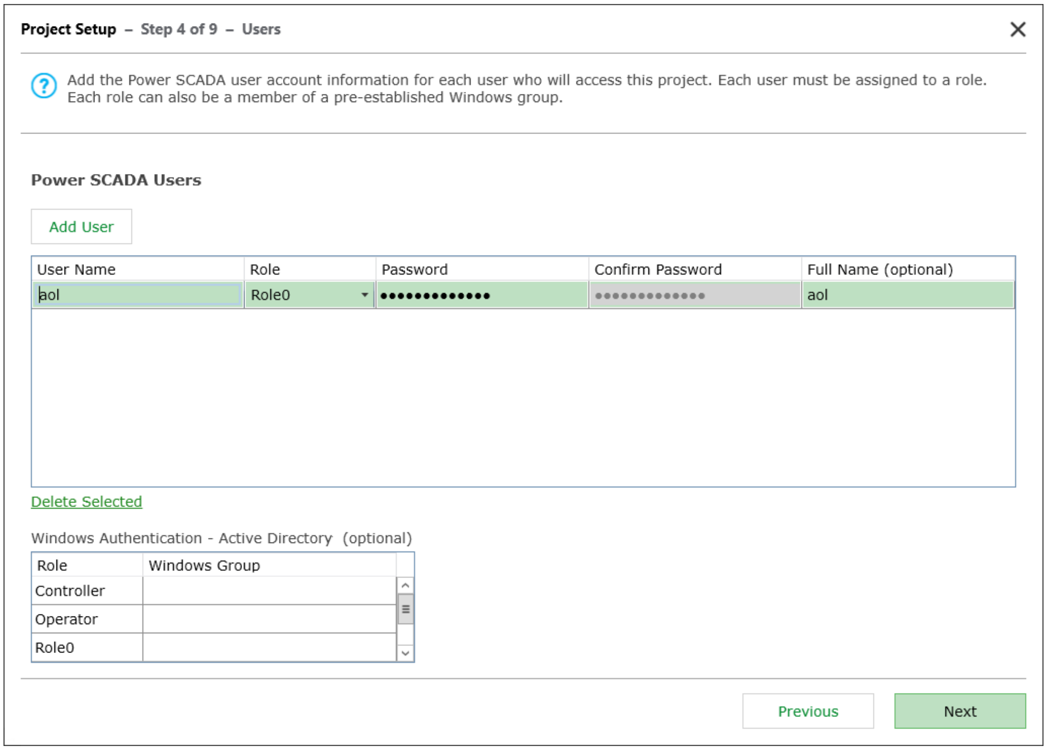
1. Enter the **Server Name or IP Address** for the project's primary server, or select it from the list.
2. (Optional) If this is a redundant system:
   a. Click **Redundant System**.
   b. Enter the server name or IP address of the standby server, or select it from the drop-down list.
3. (Optional) 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 622.

Users

Use Users to add the Power SCADA user information for each user who will access the runtime pages in this project.



Project Setup – Step 4 of 9 – Users

Add the Power SCADA user account information for each user who will access this project. Each user must be assigned to a role. Each role can also be a member of a pre-established Windows group.

Power SCADA Users

User Name	Role	Password	Confirm Password	Full Name (optional)
aol	Role0	•••••••••••••	•••••••••••••	aol

Windows Authentication - Active Directory (optional)

Role	Windows Group
Controller	
Operator	
Role0	

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 590.

6. Click **Next**.

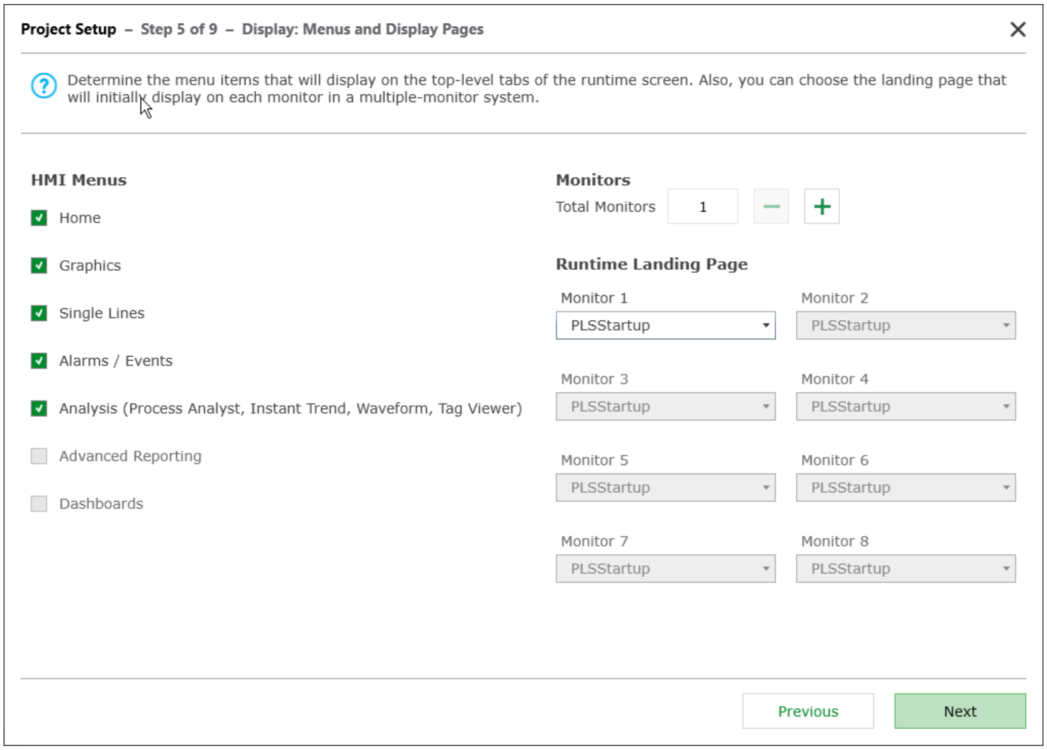
To delete a user that you previously added:

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 686.

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.



Project Setup – Step 5 of 9 – Display: Menus and Display Pages

Determine the menu items that will display on the top-level tabs of the runtime screen. Also, you can choose the landing page that will initially display on each monitor in a multiple-monitor system.

HMI Menus

☑ Home

☑ Graphics

☑ Single Lines

☑ Alarms / Events

☑ Analysis (Process Analyst, Instant Trend, Waveform, Tag Viewer)

□ Advanced Reporting

□ Dashboards

Monitors

Total Monitors: 1

Runtime Landing Page

Monitor	Landing page
Monitor 1	PLSStartup
Monitor 2	PLSStartup
Monitor 3	PLSStartup
Monitor 4	PLSStartup
Monitor 5	PLSStartup
Monitor 6	PLSStartup
Monitor 7	PLSStartup
Monitor 8	PLSStartup

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.

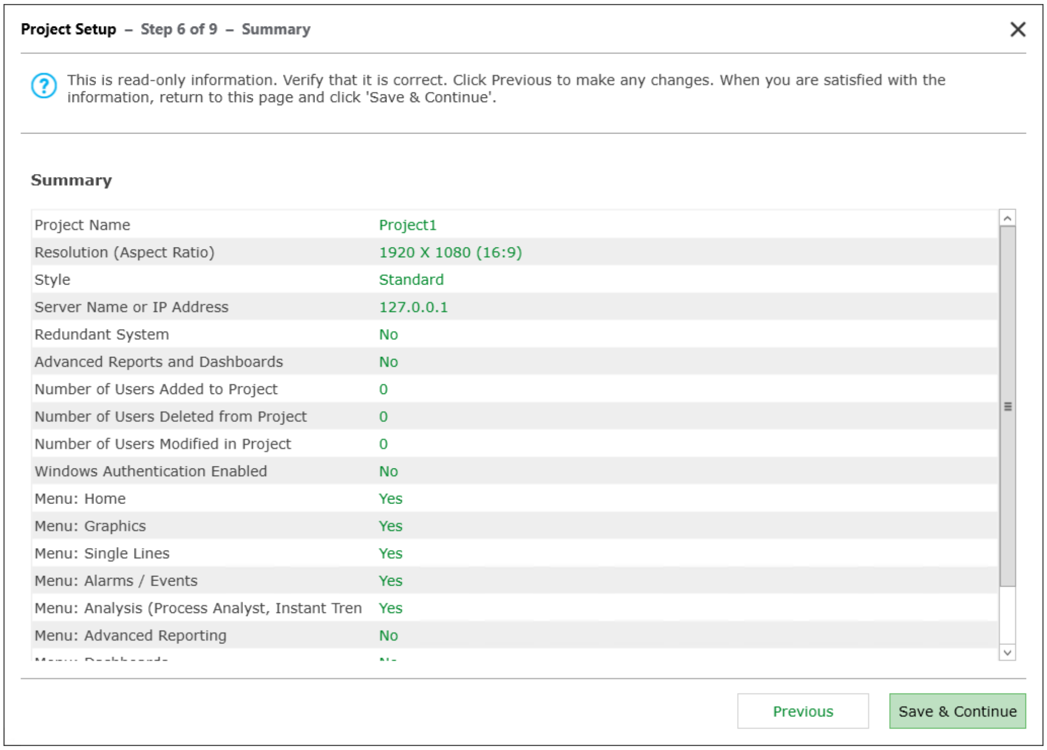
   c. For each monitor, select landing page you want to see when this monitor views Power SCADA Operation

3. Click **Next**.

For more information on Power SCADA Operation menu configuration, see "Power SCADA Runtime menus" on page 312.

Summary

Use **Summary** to verify that the project information is correct for your system.



Project Setup – Step 6 of 9 – Summary

This is read-only information. Verify that it is correct. Click Previous to make any changes. When you are satisfied with the information, return to this page and click 'Save & Continue'.

Summary

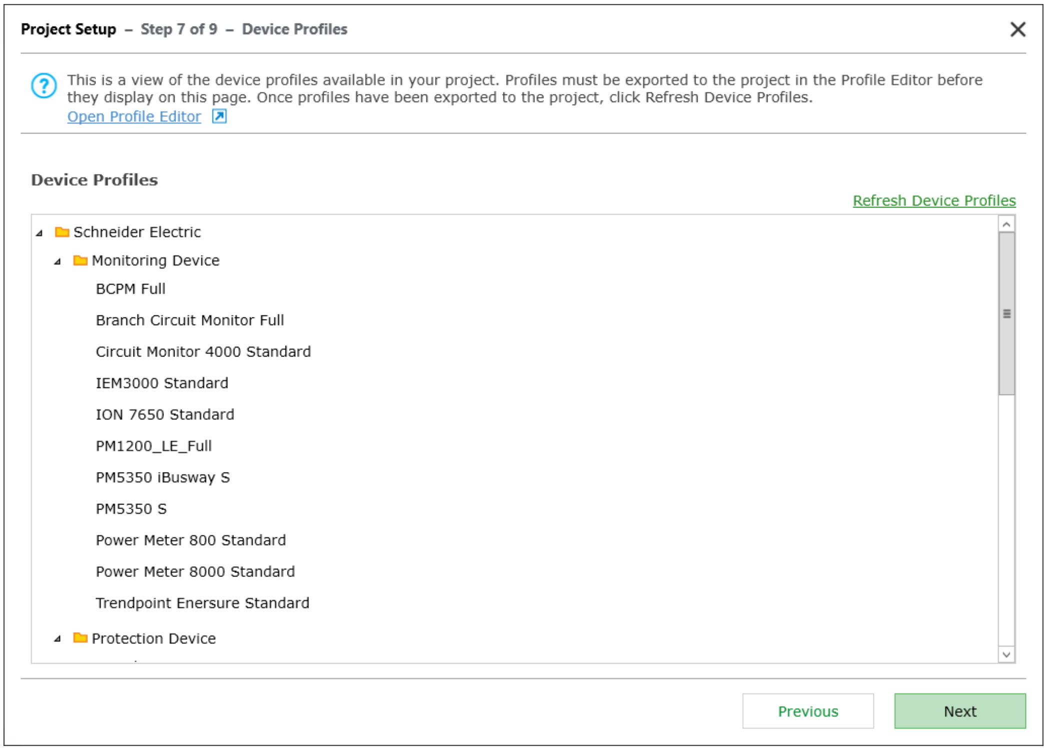
Project Name	Project1
Resolution (Aspect Ratio)	1920 X 1080 (16:9)
Style	Standard
Server Name or IP Address	127.0.0.1
Redundant System	No
Advanced Reports and Dashboards	No
Number of Users Added to Project	0
Number of Users Deleted from Project	0
Number of Users Modified in Project	0
Windows Authentication Enabled	No
Menu: Home	Yes
Menu: Graphics	Yes
Menu: Single Lines	Yes
Menu: Alarms / Events	Yes
Menu: Analysis (Process Analyst, Instant Tren	Yes
Menu: Advanced Reporting	No

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.



Project Setup – Step 7 of 9 – Device Profiles

This is a view of the device profiles available in your project. Profiles must be exported to the project in the Profile Editor before they display on this page. Once profiles have been exported to the project, click Refresh Device Profiles.

Open Profile Editor

Device Profiles

Refresh Device Profiles

- Schneider Electric
  - Monitoring Device
    - BCPM Full
    - Branch Circuit Monitor Full
    - Circuit Monitor 4000 Standard
    - IEM3000 Standard
    - ION 7650 Standard
    - PM1200_LE_Full
    - PM5350 iBusway S
    - PM5350 S
    - Power Meter 800 Standard
    - Power Meter 8000 Standard
    - Trendpoint Enersure Standard
  - Protection Device

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.

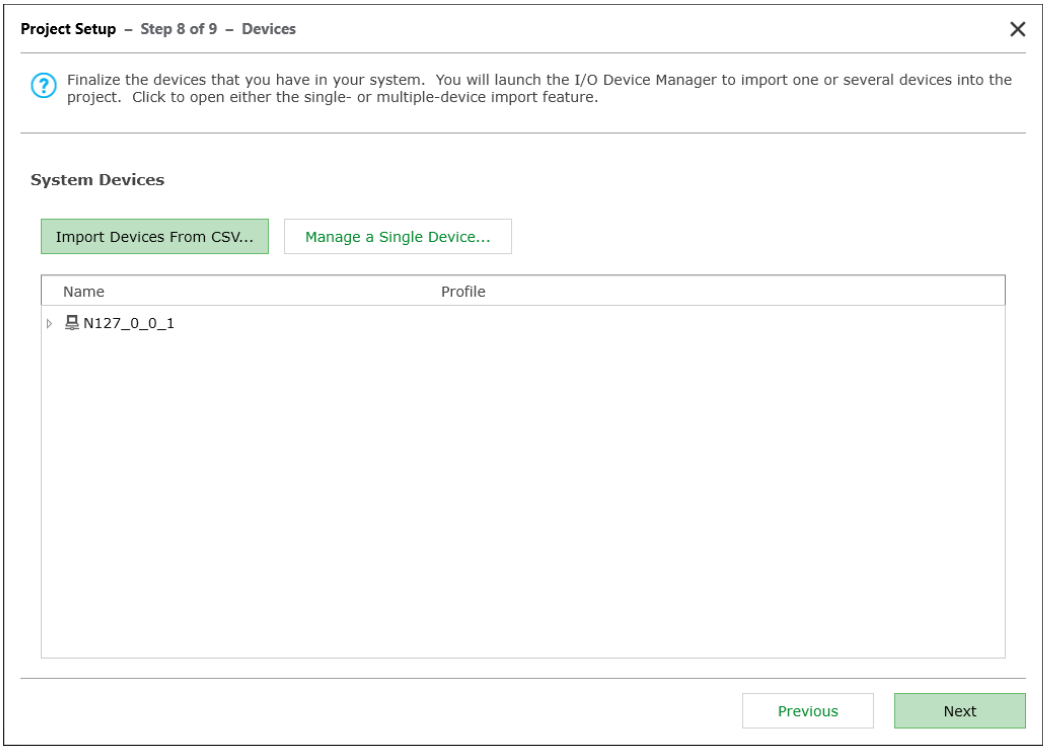
   The device profiles you added in the Profile Editor are now available to use in your project.

8. Click **Next**.

For more information on Power SCADA Operation with Advanced Reporting and Dashboards device profile configuration, see "Create Device Profiles" on page 254

Devices

Use **Devices** to add one or more devices from your system into the project.



Project Setup – Step 8 of 9 – Devices

Finalize the devices that you have in your system. You will launch the I/O Device Manager to import one or several devices into the project. Click to open either the single- or multiple-device import feature.

System Devices

Name	Profile
N127_0_0_1	

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 297

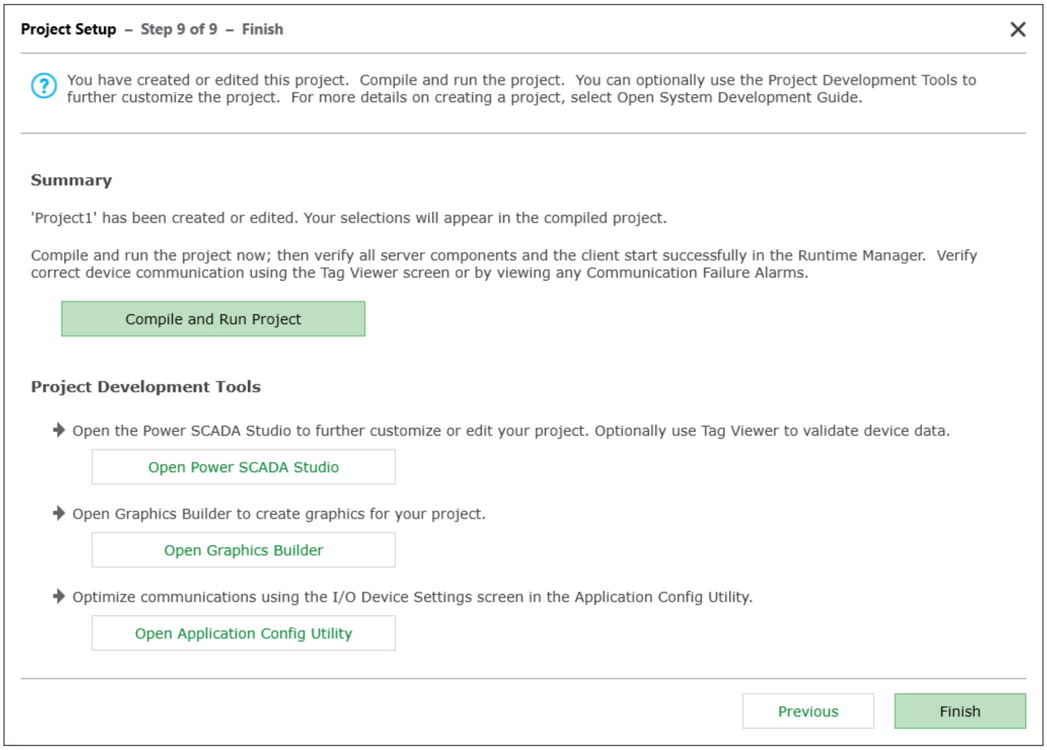
      OR

   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 291.
2. Click **Next**.

For more information on Power SCADA Operation device configuration, see "Manage I/O devices in a project" on page 289.

Finish

Use **Finish** to compile and run the project.



Project Setup – Step 9 of 9 – Finish

You have created or edited this project. Compile and run the project. You can optionally use the Project Development Tools to further customize the project. For more details on creating a project, select Open System Development Guide.

Summary

'Project1' has been created or edited. Your selections will appear in the compiled project.

Compile and run the project now; then verify all server components and the client start successfully in the Runtime Manager. Verify correct device communication using the Tag Viewer screen or by viewing any Communication Failure Alarms.

Project Development Tools

- Open the Power SCADA Studio to further customize or edit your project. Optionally use Tag Viewer to validate device data.

- Open Graphics Builder to create graphics for your project.

- Optimize communications using the I/O Device Settings screen in the Application Config Utility.

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 "Creating a Graphics page in Graphics Builder" on page 1225.
- **Open Application Config Utility** to edit or set up many project features. For more information, see "Application Configuration Utility" on page 196.

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.

Restoring 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**.

Backing up 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 279.

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 233
   - "Creating custom tags" on page 236

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 lets you create device types, device profiles, and set up projects.

NOTE: To avoid potential communication errors, use the Profile Editor to create all custom tags that will communicate with equipment.

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 225 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 247.
- **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 254 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 273 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 references:

- "Profile Editor typical workflows" on page 201
- "Profile Editor main menu options" on page 204

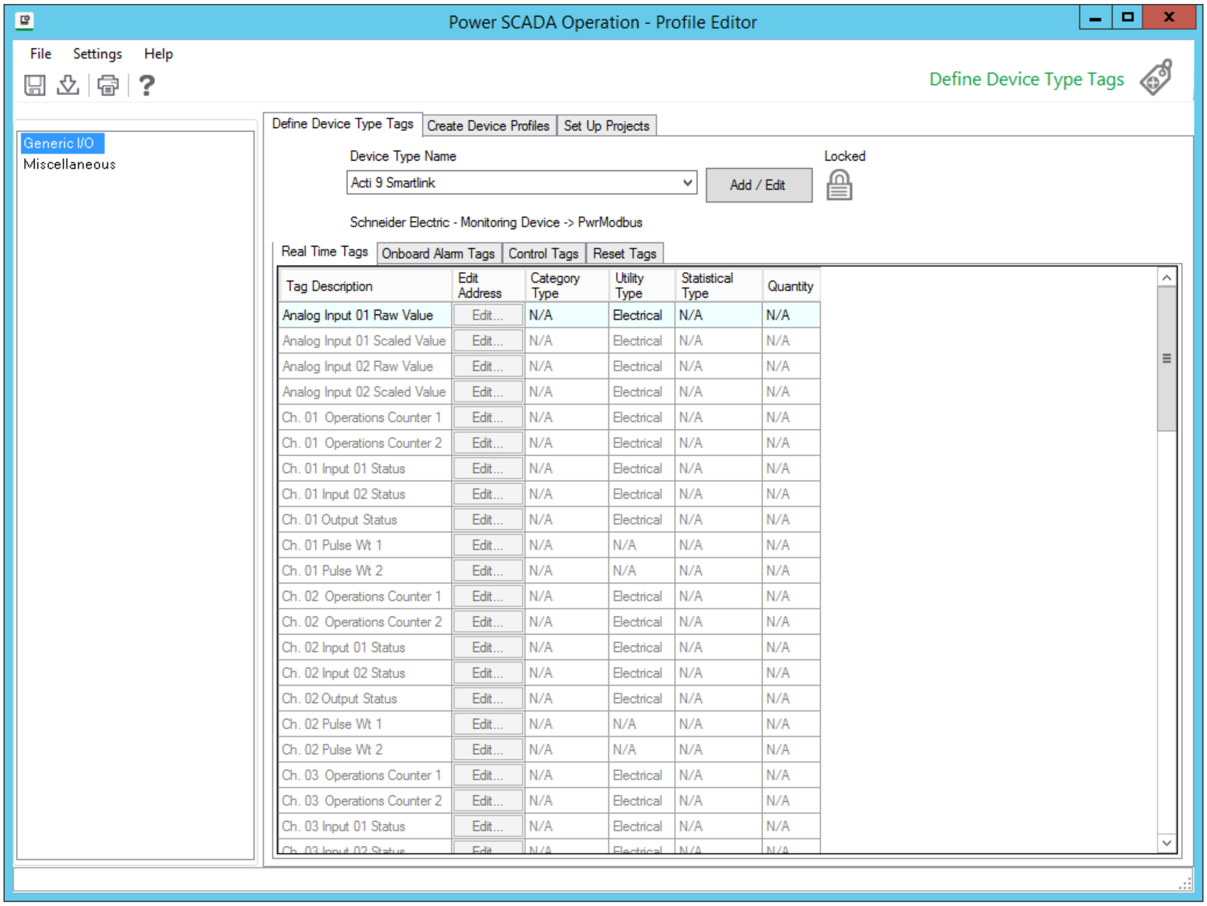
Launch the Profile Editor

> **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.



Device Type Name: Acti 9 Smartlink

Schneider Electric - Monitoring Device -> PwrModbus

Tag Description	Edit Address	Category Type	Utility Type	Statistical Type	Quantity
Analog Input 01 Raw Value	Edit...	N/A	Electrical	N/A	N/A
Analog Input 01 Scaled Value	Edit...	N/A	Electrical	N/A	N/A
Analog Input 02 Raw Value	Edit...	N/A	Electrical	N/A	N/A
Analog Input 02 Scaled Value	Edit...	N/A	Electrical	N/A	N/A
Ch. 01 Operations Counter 1	Edit...	N/A	Electrical	N/A	N/A
Ch. 01 Operations Counter 2	Edit...	N/A	Electrical	N/A	N/A
Ch. 01 Input 01 Status	Edit...	N/A	Electrical	N/A	N/A
Ch. 01 Input 02 Status	Edit...	N/A	Electrical	N/A	N/A
Ch. 01 Output Status	Edit...	N/A	Electrical	N/A	N/A
Ch. 01 Pulse Wt 1	Edit...	N/A	N/A	N/A	N/A
Ch. 01 Pulse Wt 2	Edit...	N/A	N/A	N/A	N/A
Ch. 02 Operations Counter 1	Edit...	N/A	Electrical	N/A	N/A
Ch. 02 Operations Counter 2	Edit...	N/A	Electrical	N/A	N/A
Ch. 02 Input 01 Status	Edit...	N/A	Electrical	N/A	N/A
Ch. 02 Input 02 Status	Edit...	N/A	Electrical	N/A	N/A
Ch. 02 Output Status	Edit...	N/A	Electrical	N/A	N/A
Ch. 02 Pulse Wt 1	Edit...	N/A	N/A	N/A	N/A
Ch. 02 Pulse Wt 2	Edit...	N/A	N/A	N/A	N/A
Ch. 03 Operations Counter 1	Edit...	N/A	Electrical	N/A	N/A
Ch. 03 Operations Counter 2	Edit...	N/A	Electrical	N/A	N/A
Ch. 03 Input 01 Status	Edit...	N/A	Electrical	N/A	N/A
Ch. 03 Input 02 Status	Edit...	N/A	Electrical	N/A	N/A

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.

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.

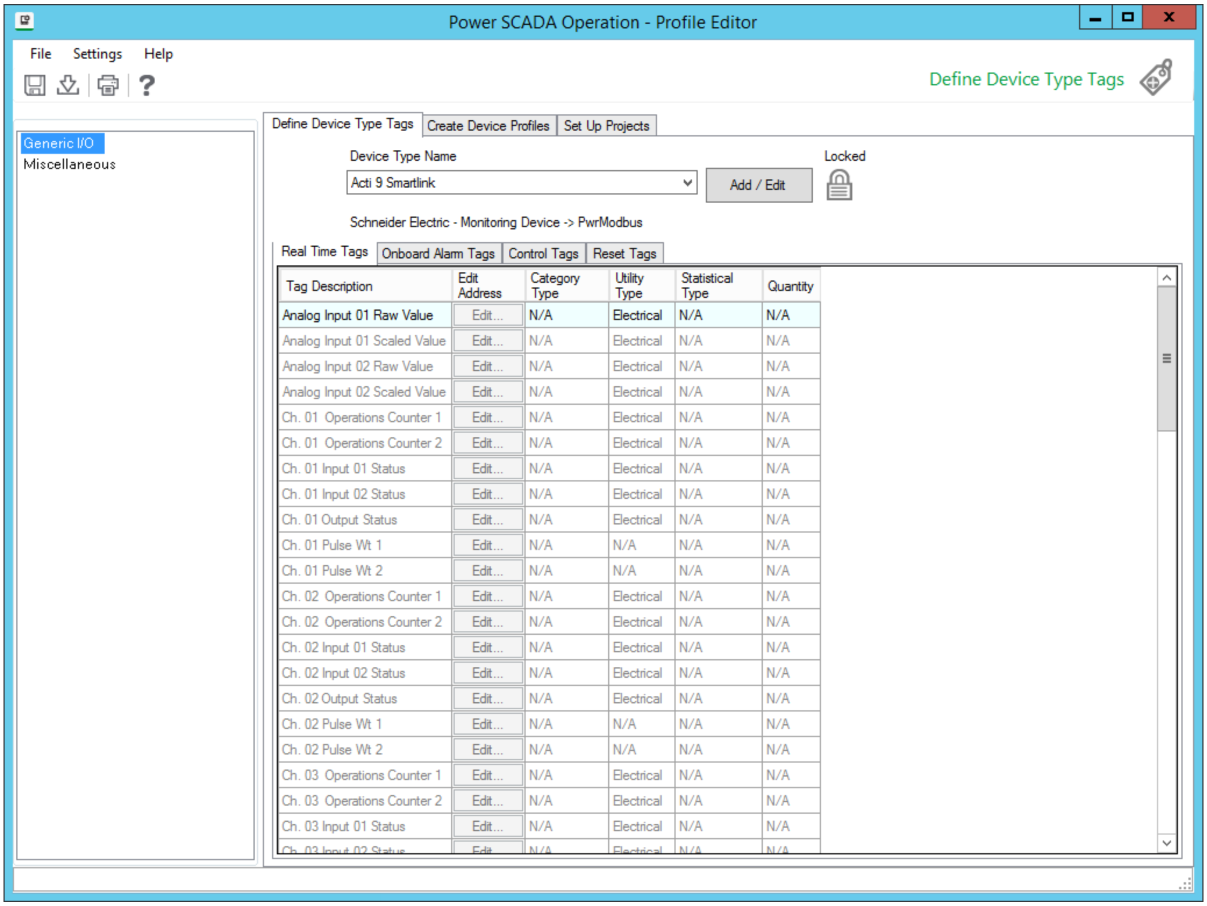
Set the screen resolution

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:



Device Type Name: Acti 9 Smartlink

Locked

Schneider Electric - Monitoring Device -> PwrModbus

Tag Description	Edit Address	Category Type	Utility Type	Statistical Type	Quantity
Analog Input 01 Raw Value	Edit...	N/A	Electrical	N/A	N/A
Analog Input 01 Scaled Value	Edit...	N/A	Electrical	N/A	N/A
Analog Input 02 Raw Value	Edit...	N/A	Electrical	N/A	N/A
Analog Input 02 Scaled Value	Edit...	N/A	Electrical	N/A	N/A
Ch. 01 Operations Counter 1	Edit...	N/A	Electrical	N/A	N/A
Ch. 01 Operations Counter 2	Edit...	N/A	Electrical	N/A	N/A
Ch. 01 Input 01 Status	Edit...	N/A	Electrical	N/A	N/A
Ch. 01 Input 02 Status	Edit...	N/A	Electrical	N/A	N/A
Ch. 01 Output Status	Edit...	N/A	Electrical	N/A	N/A
Ch. 01 Pulse Wt 1	Edit...	N/A	N/A	N/A	N/A
Ch. 01 Pulse Wt 2	Edit...	N/A	N/A	N/A	N/A
Ch. 02 Operations Counter 1	Edit...	N/A	Electrical	N/A	N/A
Ch. 02 Operations Counter 2	Edit...	N/A	Electrical	N/A	N/A
Ch. 02 Input 01 Status	Edit...	N/A	Electrical	N/A	N/A
Ch. 02 Input 02 Status	Edit...	N/A	Electrical	N/A	N/A
Ch. 02 Output Status	Edit...	N/A	Electrical	N/A	N/A
Ch. 02 Pulse Wt 1	Edit...	N/A	N/A	N/A	N/A
Ch. 02 Pulse Wt 2	Edit...	N/A	N/A	N/A	N/A
Ch. 03 Operations Counter 1	Edit...	N/A	Electrical	N/A	N/A
Ch. 03 Operations Counter 2	Edit...	N/A	Electrical	N/A	N/A
Ch. 03 Input 01 Status	Edit...	N/A	Electrical	N/A	N/A
Ch. 03 Input 02 Status	Edit...	N/A	Electrical	N/A	N/A

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 955.
2. Add and edit custom tags, see "Setting up custom tags" on page 243.
3. Add or edit device types, see "Managing device types" on page 229.
4. Establish device type categories and subcategories, used in reporting, see "Printing the .CSV file" on page 233.
5. Edit tag addresses, see “"About tags" on page 247”.

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). 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 223 for complete information.
Tag groups (left-hand pane)	
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.	

Field Name/Valid Entries	Comments
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 247 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 247 for a detailed description of address construction.
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 "Editing tag addresses" on page 236. 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.

Field Name/Valid Entries	Comments
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 "Editing tag addresses" on page 236.
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.
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 1305
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.

Field Name/Valid Entries	Comments
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 231 and "Delete a device type" on page 232 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 and any associated custom profiles.
Device Type (to Create From/to Edit/ to Delete)	select type	Select the device type that you want to create from, edit, or delete.

Field Name	Valid Entries	Comments
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.
Device Category	Choose the category for this device.	To create categories, see "Printing the .CSV file" on page 233 In addition to predefined categories, you can add custom categories. See "Managing device type categories" on page 234 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 "Printing the .CSV file" on page 233 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 shows the profiles that are associated with the selected 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 "Edit a custom tag" on page 246 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. 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 235).
   b. In **Real Time Tags** you can edit the address (see "Editing tag addresses" on page 236) 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.
   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

Standard device types do not display in this option because you cannot delete them.

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.

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 device types; you cannot delete any of the standard device types).
4. Click **Delete**. A list of associated profiles will appear in the Confirm prompt. Click **Yes** to delete the selected device type and any associated profiles.

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.

Printing the .CSV file

For each device type, device profile, or project, you can create and print 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 and print 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.

Managing 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.

Adding 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.

Editing 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.

Deleting 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 **Edit Functional Address** 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.

Creating 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 243
- "Edit generic tag addresses" on page 242

Editing 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 247 and "About logic codes" on page 253.

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 253. 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 253.
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	
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 253.
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 specify 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 253.
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.

Box Name	Field Name	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	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.
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 specify the indicator register (which becomes the first register). The second register will have the mask.
Invert Result	n/a	Not used for digital controls.

Editing 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 R2 help file: `...\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020 R2\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 246 and "Delete a custom tag" on page 246 for instructions on how to edit or delete custom tags.

NOTE: Starting in Power SCADA Operation 2020, item names have an increased importance. They drive binding in web graphics, and determine the availability of those bindings in the Graphics Editor. It is important for tags that will be used in graphics to have item names.

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.

Field Name	Valid Entries	Comments
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 246 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.
Item Name	Type the item name for the tag.	Maximum 45 characters; can only include alphanumeric characters. Must begin with a letter.
Group	Select the group.	Includes all the real-time groups (such as 100ms, controls, currents) plus onboard alarms, resets, and controls.
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 253 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.

Field Name	Valid Entries	Comments
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.	
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.

Field Name	Valid Entries	Comments
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 607). 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 "Printing the .CSV file" on page 233). 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 247.

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)

Category Name	Description
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 "Editing tag addresses" on page 236 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 940), or create a custom genie (see "Create a new genie" on page 1246).

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 253 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)	

Attributes	Modifiers	Comments

D (module—Micrologic devices)	B = BCM
	P = PM
	M = MM
	C = CCM	

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	

Attributes	Modifiers	Comments
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.
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 253

Alarm format code definitions

Attributes	Modifiers	Comments
T (type) Required	ALM = alarm	

Attributes	Modifiers	Comments

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.
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 253. 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 907.

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 225). 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 256 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.
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.

Field Name	Valid Entry	Comments
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.

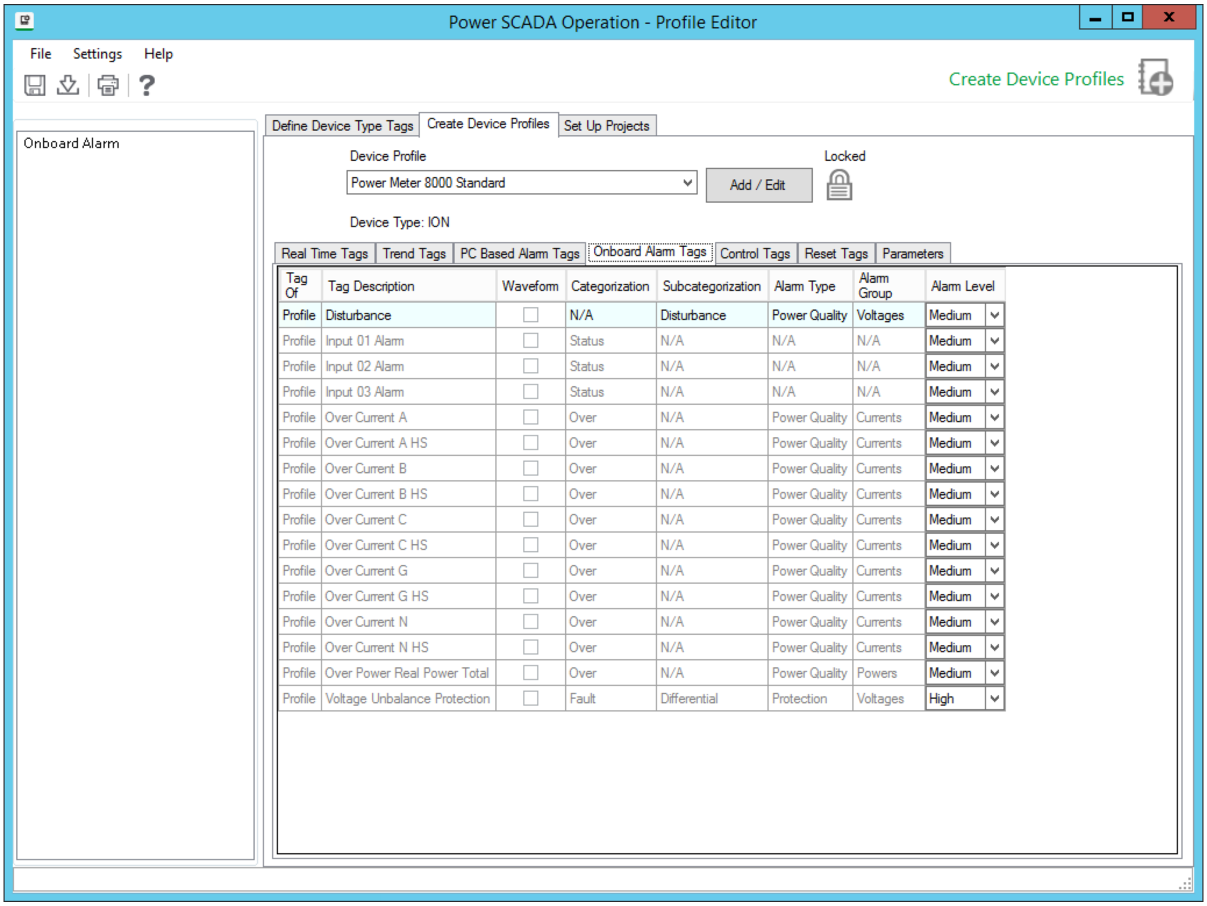
To view profile information:

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.



Device Profile: Power Meter 8000 Standard

Device Type: ION

Tag Of	Tag Description	Waveform	Categorization	Subcategorization	Alarm Type	Alarm Group	Alarm Level
Profile	Disturbance	☐	N/A	Disturbance	Power Quality	Voltages	Medium
Profile	Input 01 Alarm	☐	Status	N/A	N/A	N/A	Medium
Profile	Input 02 Alarm	☐	Status	N/A	N/A	N/A	Medium
Profile	Input 03 Alarm	☐	Status	N/A	N/A	N/A	Medium
Profile	Over Current A	☐	Over	N/A	Power Quality	Currents	Medium
Profile	Over Current A HS	☐	Over	N/A	Power Quality	Currents	Medium
Profile	Over Current B	☐	Over	N/A	Power Quality	Currents	Medium
Profile	Over Current B HS	☐	Over	N/A	Power Quality	Currents	Medium
Profile	Over Current C	☐	Over	N/A	Power Quality	Currents	Medium
Profile	Over Current C HS	☐	Over	N/A	Power Quality	Currents	Medium
Profile	Over Current G	☐	Over	N/A	Power Quality	Currents	Medium
Profile	Over Current G HS	☐	Over	N/A	Power Quality	Currents	Medium
Profile	Over Current N	☐	Over	N/A	Power Quality	Currents	Medium
Profile	Over Current N HS	☐	Over	N/A	Power Quality	Currents	Medium
Profile	Over Power Real Power Total	☐	Over	N/A	Power Quality	Powers	Medium
Profile	Voltage Unbalance Protection	☐	Fault	Differential	Protection	Voltages	High

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 696 for information on viewing waveforms in the Power SCADA Runtime.

Enabling 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. They 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 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.

Adding 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.

Managing device profiles

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.

Adding device profiles

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. (Optional) Type a device description. This will display as a tool tip in later screens.
6. Select a device profile **Type**. The default associated component will be selected and shown below in an image next to a button with the component name.

   > **NOTE**: To associate a device profile with a different component, click the button to open the Graphics Editor Component Library and select a different component.
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 277.
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 267.

    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 "Managing IEC 61850 datasets" on page 263 and "Edit IEC 61850 Report control blocks" on page 264 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.

Editing device profiles

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 "Enabling waveforms for onboard alarms" on page 308 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 266 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.

Deleting device profiles

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 "Setting up a G3200 gateway" on page 271).
2. Import the first ICD file into the Profile Editor (see "Import Filter screen" on page 283).
   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 295).
   a. If needed, add/edit datasets and report control blocks (see "Managing IEC 61850 datasets" on page 263 and "Edit IEC 61850 Report control blocks" on page 264).
   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 295.
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 "Managing IEC 61850 datasets" on page 263.

Managing 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.

Creating 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.

Creating 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**.

Copying 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 displays under the **Device Type** list in the **Device Profile DataSets**.

Editing and deleting 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.

Creating 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.

Creating 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**.

Copying 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**.

Editing and Deleting 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 "Managing IEC 61850 datasets" on page 263 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 264 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 266.

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`*`]\ProgramData\Schneider Electric\Power SCADA Operation\v2020 R2\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\v2020 R2\bin\Help\Citect SCADA`).

Create composite device profiles

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:

- "Creating a third party Modbus Device Type" on page 268
- "Creating a composite device type" on page 269
- "Creating a data concentrator" on page 271
- "Setting up a G3200 gateway" on page 271

Creating a 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 `[`*`Project Drive`*`]\ProgramData\Schneider Electric\Power SCADA Operation\v2020 R2\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.
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.

Creating 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.

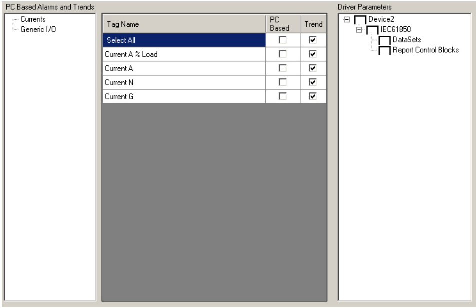
    When the profile is added to the project, PC based alarms are added to the Analog Alarms or Digital Alarms file.

    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 267.

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.



Tag Name	PC Based	Trend
Select All	□	☑
Current A % Load	□	☑
Current A	□	☑
Current N	□	☑
Current G	□	☑

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 "Managing IEC 61850 datasets" on page 263 and "Edit IEC 61850 Report control blocks" on page 264 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.

Creating a data concentrator

When you use the Profile Editor to create data concentrator or 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.

Setting up a G3200 gateway

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 294 or "Adding a serial device" on page 293 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 277.
- **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 277.

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:
   1. From the **Set Up Project** tab, click the **Customize Tag Names** sub-tab.
   2. 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.
   3. 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:
   1. From the Set Up Project tab, click the Project Parameters sub-tab.
   2. 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:

```
[Project Drive]\ProgramData\Schneider Electric\Power
SCADA Operation\v2020 R2\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 951 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 955 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 1305.)

    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 280.

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 289 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 277.
- 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 281.
- 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 279.
- 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 282.

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 288.

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 "Backing up a project" on page 219.

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 277 and "Add project parameters" on page 277 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:

[*Project Drive*]\ProgramData\Schneider Electric\Power SCADA Operation\v2020 R2\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 v2020 R2\User\[*Project*]

    Where:

    [Drive Letter]: The 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]: 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\v2020 R2\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 206.

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 288.) 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 284 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 284 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 288 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.

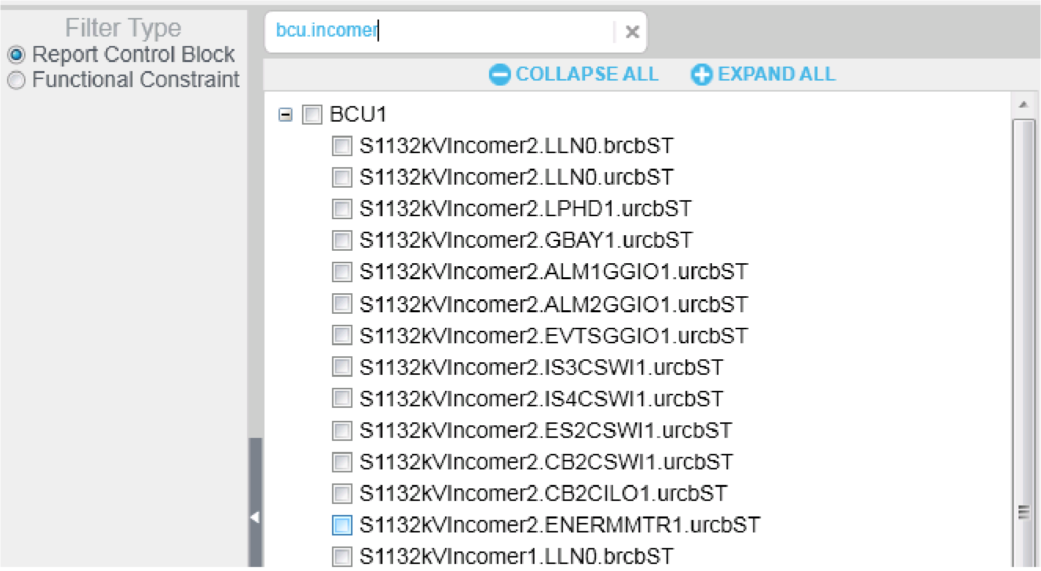
   The list of devices and their related report control blocks that are included in the import file displays in the middle column.

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*



● Report Control Block
○ Functional Constraint

bcu.incomer

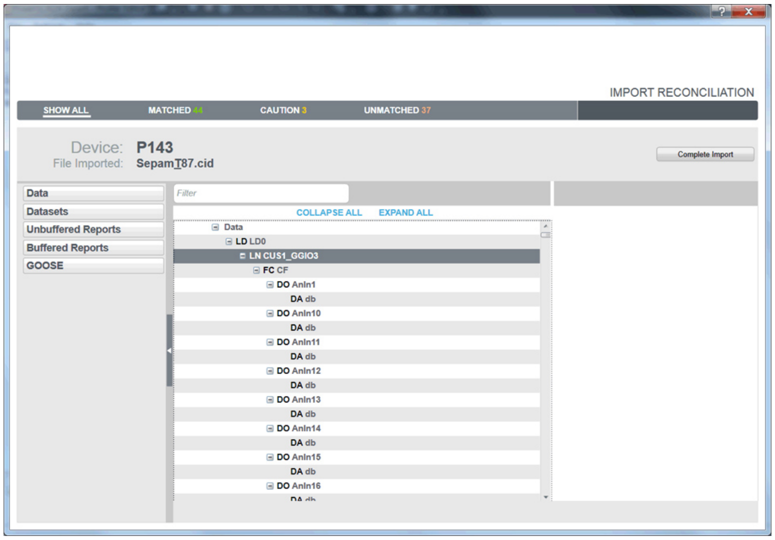
⊟ ☐ BCU1
- ☐ S1132kVIncomer2.LLN0.brcbST
- ☐ S1132kVIncomer2.LLN0.urcbST
- ☐ S1132kVIncomer2.LPHD1.urcbST
- ☐ S1132kVIncomer2.GBAY1.urcbST
- ☐ S1132kVIncomer2.ALM1GGIO1.urcbST
- ☐ S1132kVIncomer2.ALM2GGIO1.urcbST
- ☐ S1132kVIncomer2.EVTSGGIO1.urcbST
- ☐ S1132kVIncomer2.IS3CSWI1.urcbST
- ☐ S1132kVIncomer2.IS4CSWI1.urcbST
- ☐ S1132kVIncomer2.ES2CSWI1.urcbST
- ☐ S1132kVIncomer2.CB2CSWI1.urcbST
- ☐ S1132kVIncomer2.CB2CILO1.urcbST
- ☐ S1132kVIncomer2.ENERMMTR1.urcbST
- ☐ S1132kVIncomer1.LLN0.brcbST

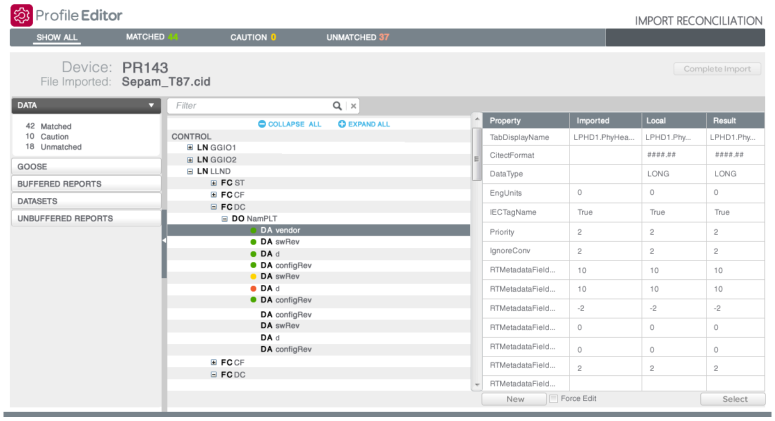
When you have selected either the functional constraints or report control blocks, click Continue. The data is filtered on the last filter option that you chose (you cannot combine filters).

The Import Reconciliation screen displays. See "Import Reconciliation screen" on page 284 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.





Property	Imported	Local	Result
TabDisplayName	LPHD1.PhyHea...	LPHD1.Phy...	LPHD1.Phy...
CitectFormat		####.##	####.##
DataType		LONG	LONG
EngUnits	0	0	0
IECTagName	True	True	True
Priority	2	2	2
IgnoreConv	2	2	2
RTMetadataField...	10	10	10
RTMetadataField...	10	10	10
RTMetadataField...	-2	-2	-2
RTMetadataField...	0	0	0
RTMetadataField...	0	0	0
RTMetadataField...	2	2	2
RTMetadataField...			

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.

Manage 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 291
- "Remove an I/O device from a project" on page 297
- "Define multiple devices using a CSV file" on page 297
- "Updating devices in a project" on page 303

For information on how to translate device information, see "Translating device information" on page 613.

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.

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 consider 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	

	Generic Serial, DNP3 via Serial	MODBUS RTU via Serial	DNP3 via TCP/IP, IEC 60870-5-104 via TCP/IP, MODBUS RTU via Gateway
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 291 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 292
- "Adding a serial device" on page 293
- "Adding a DNP3 TCP device" on page 294
- "Adding an IEC 61850 device" on page 295

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 291. 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 304. 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 304. 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 304. 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 304. 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 I/O Device Manager .
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 304. 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 "Create a CSV file to add multiple devices" on page 298 for details.

After you create the CSV file, you use it to add multiple devices to the project. See "Add multiple devices to a project using a CSV file" on page 301 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 303.

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 289 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\v2020 R2\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.

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/LLN0$BR$BRep01,CFG/LLN0$BR$BRep06
   and
   URCB: CFG/LLN0$BR$BRep01,CFG/LLN0$BR$BRep06

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.

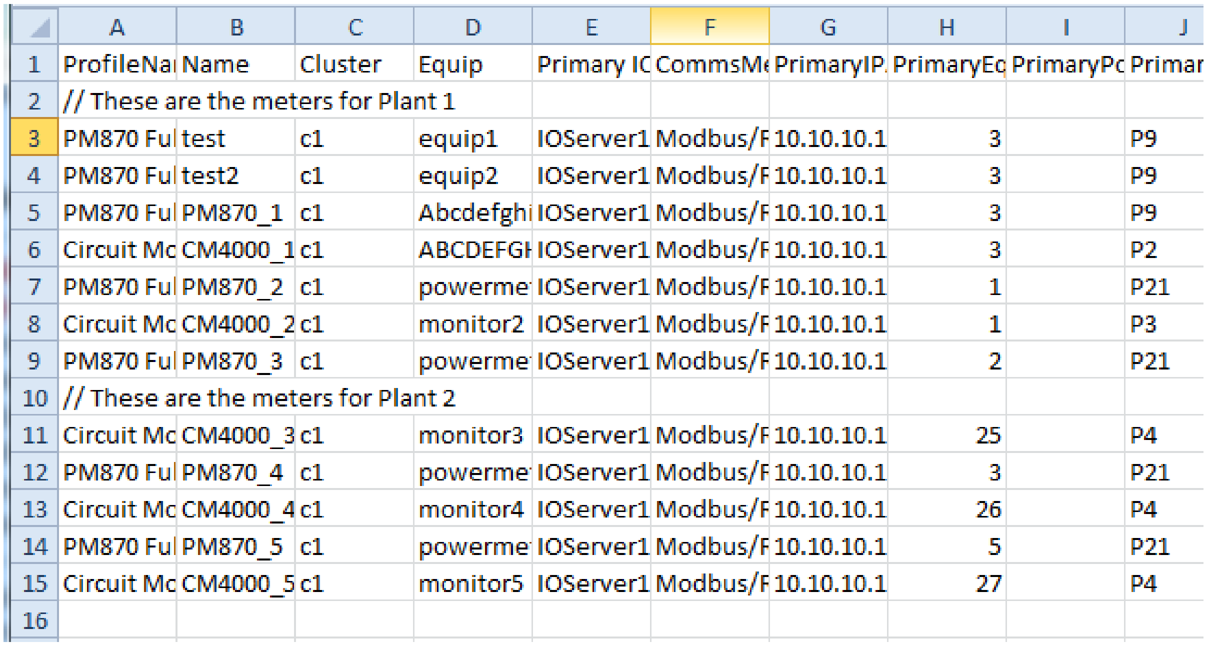
5. Comment: This is an optional description of the device; maximum 254 characters.

6. Close the example CSV file, if it is open.

See "Add multiple devices to a project using a CSV file" on page 301 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 "Create a CSV file to add multiple devices" on page 298). 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:

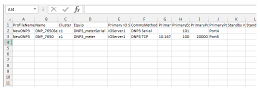
1. At the bottom left part of the window click **Export**.
2. Choose the location at which you want to store the files, and then click **OK**.

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 297.

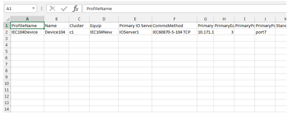
DNP3 for Serial and Ethernet



AI4

	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 I(	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		

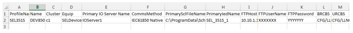
IEC104.2



A1 ProfileName

	A	B	C	D	E	F	G	H	I	J	K
1	ProfileName	Name	Cluster	Equip	Primary IO Serve	CommsMethod	Primary	PrimaryEc	PrimaryPc	PrimaryPc	Stanc
2	IEC104Device	Device104	c1	IEC104New	IOServer1	IEC60870-5-104 TCP	10.171.1	3		port7	

IEC61850



	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

Updating 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.

Editing 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" on page 305
- "Add setpoints and delays" on page 305
- "Set up an alarm based on an enumeration" on page 305
- "Change an alarm severity" on page 306

- "Enabling waveforms for onboard alarms" on page 308
- "Set parameters for event log length and historical logging of events" on page 308
- "Adding an onboard alarm tag" on page 309
- "Set up audible alarms" on page 309

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 "Managing device profiles" on page 258 for instructions. For instructions on entering setpoints and delays, see "Add setpoints and delays" on page 305.

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 "Adding an onboard alarm tag" on page 309.

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 248.

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" on page 306
- "Waveform database and special waveform tags" on page 307

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.

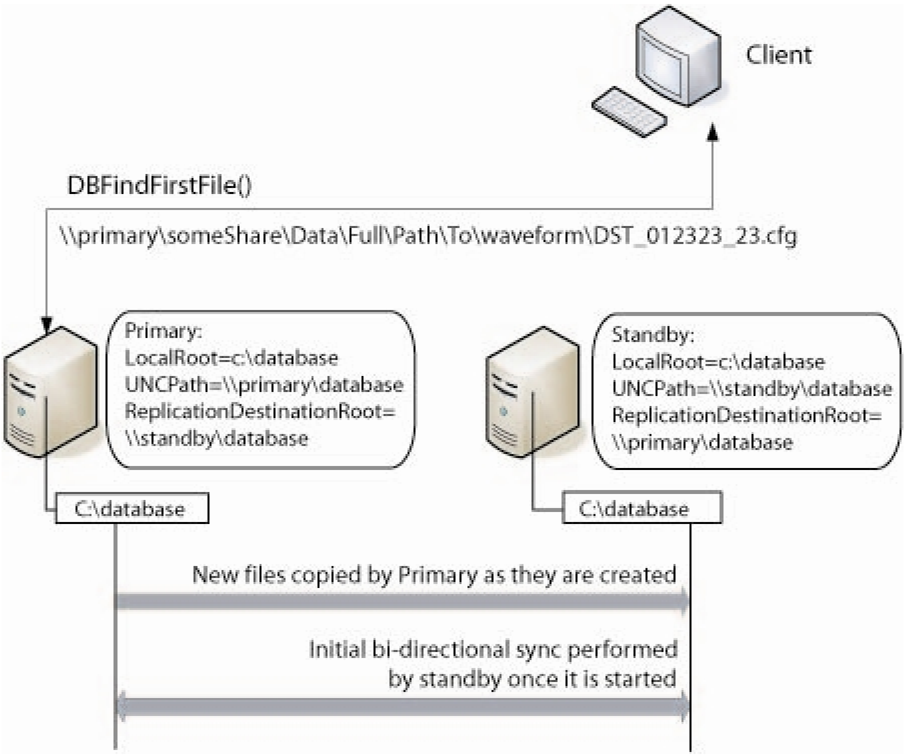
Waveform database and special waveform tags

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:


Client
DBFindFirstFile()
\\primary\someShare\Data\Full\Path\To\waveform\DST_012323_23.cfg
Primary:
LocalRoot=c:\database
UNCPath=\\primary\database
ReplicationDestinationRoot=
\\standby\database
Standby:
LocalRoot=c:\database
UNCPath=\\standby\database
ReplicationDestinationRoot=
\\primary\database
C:\database
C:\database
New files copied by Primary as they are created
Initial bi-directional sync performed
by standby once it is started

Enabling 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. They 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 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.

Adding 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 1228
- "Creating a one-line on a graphics page" on page 1241

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.

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\v2020 R2\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 313

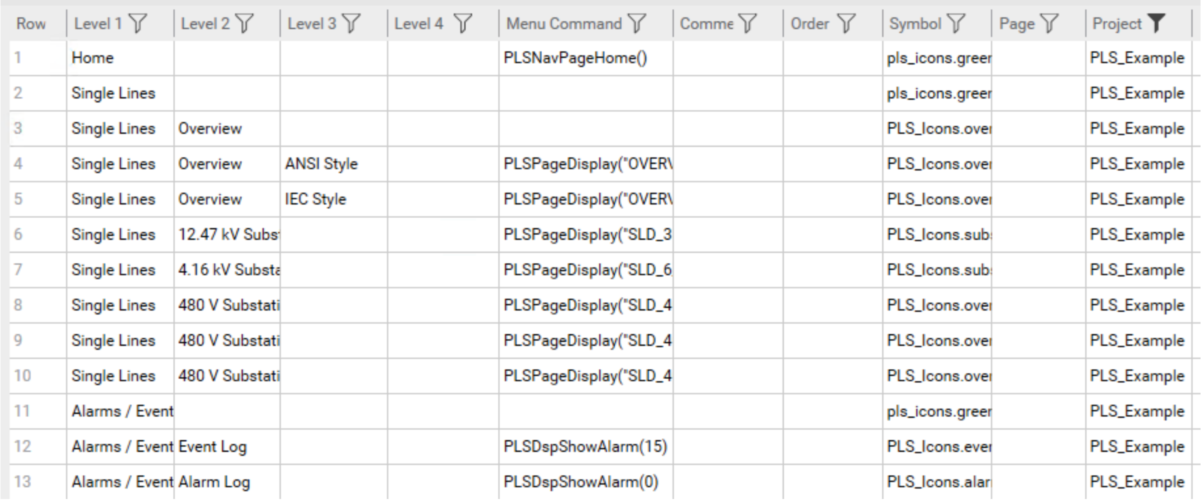
To create new genies for a project, see "Creating a one-line on a graphics page" on page 1241..

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.ovei		PLS_Example
4	Single Lines	Overview	ANSI Style		PLSPageDisplay("OVER\			PLS_Icons.ovei		PLS_Example
5	Single Lines	Overview	IEC Style		PLSPageDisplay("OVER\			PLS_Icons.ovei		PLS_Example
6	Single Lines	12.47 kV Subs			PLSPageDisplay("SLD_3			PLS_Icons.sub		PLS_Example
7	Single Lines	4.16 kV Subst			PLSPageDisplay("SLD_6			PLS_Icons.sub		PLS_Example
8	Single Lines	480 V Substati			PLSPageDisplay("SLD_4			PLS_Icons.ovei		PLS_Example
9	Single Lines	480 V Substati			PLSPageDisplay("SLD_4			PLS_Icons.ovei		PLS_Example
10	Single Lines	480 V Substati			PLSPageDisplay("SLD_4			PLS_Icons.ovei		PLS_Example
11	Alarms / Event							pls_icons.greer		PLS_Example
12	Alarms / Event	Event Log			PLSDspShowAlarm(15)			PLS_Icons.evei		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.

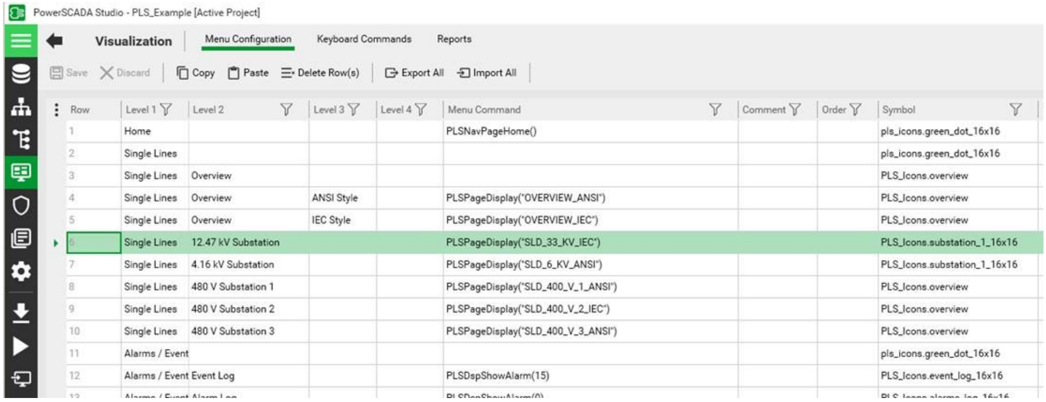
Menu Item	Description
Order	The relative position within the final graphics page. If you leave this field blank, the default value 0 is used. (64 characters maximum)
Symbol	Displays a defined image along with the description for that level. Images must already be defined in the project/include project. 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 313, 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 project Menu Configuration" on page 313 for information on this field.



Row	Level 1	Level 2	Level 3	Level 4	Menu Command	Comment	Order	Symbol
1	Home				PLSNavPageHome()			pls_icons.green_dot_16x16
2	Single Lines							pls_icons.green_dot_16x16
3	Single Lines	Overview						PLS_Icons.overview
4	Single Lines	Overview	ANSI Style		PLSPageDisplay("OVERVIEW_ANSI")			PLS_Icons.overview
5	Single Lines	Overview	IEC Style		PLSPageDisplay("OVERVIEW_IEC")			PLS_Icons.overview
6	Single Lines	12.47 kV Substation			PLSPageDisplay("SLD_33_KV_IEC")			PLS_Icons.substation_1_16x16
7	Single Lines	4.16 kV Substation			PLSPageDisplay("SLD_6_KV_ANSI")			PLS_Icons.substation_1_16x16
8	Single Lines	480 V Substation 1			PLSPageDisplay("SLD_400_V_1_ANSI")			PLS_Icons.overview
9	Single Lines	480 V Substation 2			PLSPageDisplay("SLD_400_V_2_IEC")			PLS_Icons.overview
10	Single Lines	480 V Substation 3			PLSPageDisplay("SLD_400_V_3_ANSI")			PLS_Icons.overview
11	Alarms / Event							pls_icons.green_dot_16x16
12	Alarms / Event	Event Log			PLSDspShowAlarm(15)			PLS_Icons.event_log_16x16
13	Alarms / Event	Alarm Log			PLSDspShowAlarm(0)			PLS_Icons.alarms_log_16x16

Adding 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 714.

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 325), 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 637.

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_Include 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 2020 – 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 198 and (for receiving reports via email) "Configure basic reports for email" on page 321 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 318

For descriptions of each report type, see "Basic Reports" on page 715.

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 719.

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 610

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 723 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 321.

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 2020 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 320 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 719 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\v2020 R2\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\v2020 R2\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 712 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 198 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 2020, the LiveView and reporting features will not launch. To prevent this, install Matrikon before you install Power SCADA Operation 2020. 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 330). 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 726 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 328: 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 332 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 609 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 > 2020 > 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 330.

See also:

"LiveView Formulas" on page 333

"LiveView Placeholders" on page 332

"LiveView Thresholds" on page 334

"LiveView Formatting" on page 331

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 333.
   - To add data (device names and tag names) to the real-time table, see "LiveView Placeholders" on page 332
   - 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 334.
   - To add formatting to cells, such as font and font size, see "LiveView Formatting" on page 331.

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 333
- "LiveView Thresholds" on page 334
- "LiveView Placeholders" on page 332

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 333
- "LiveView Thresholds" on page 334
- "LiveView Formatting" on page 331

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:

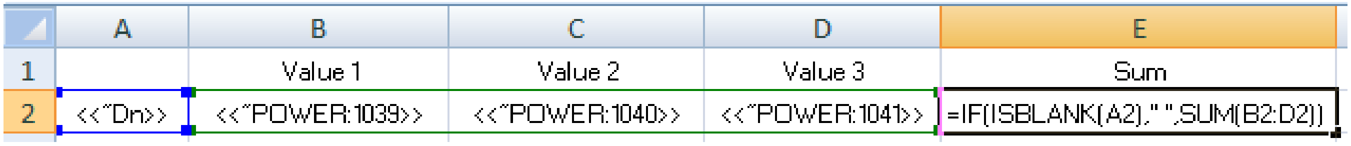
1. In LiveView Template Editor, click the **Formulas** sub-tab.
2. Choose one of the following fields:
   - **Cell:** Use this field to enable a formula for two individual cells. Then enter:
     - **Cell 1 Address:** Enter the cell address. The cell address displays in this field.
     - **Operator:** 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 for-

mulas 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 333
- "LiveView Placeholders" on page 332
- "LiveView Formatting" on page 331

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 330). 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 332
- "LiveView Formulas" on page 333
- "LiveView Thresholds" on page 334
- "LiveView Formatting" on page 331

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 330). 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 330). 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\v2020 R2\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 then 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.

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

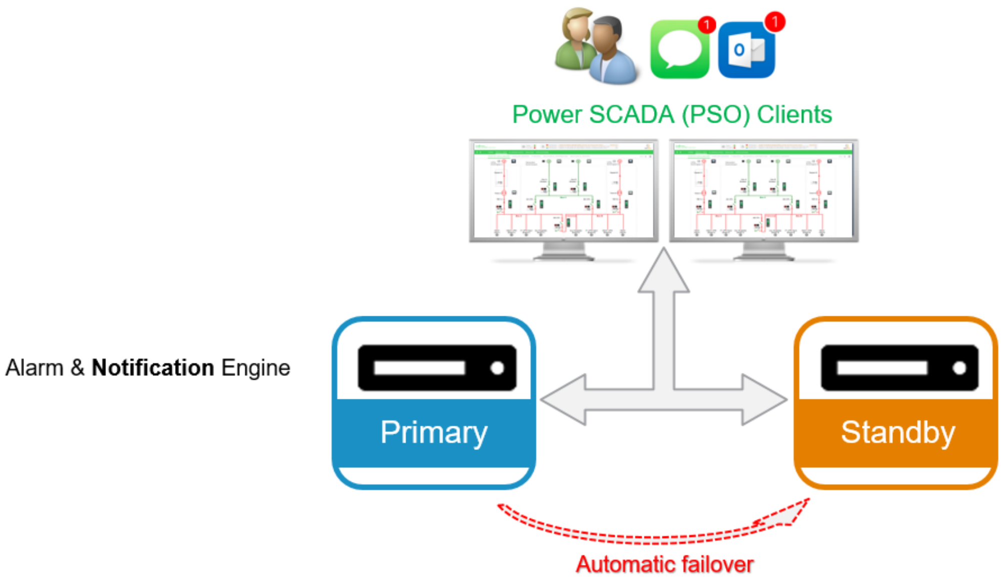
Notifications Settings accepts alarms for the Power SCADA Operation Alarm Server. Notifications Settings alerts specified recipients based on the configured notification.

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 "Migrate notifications" on page 339.

Licensing

In Power SCADA Operation 2020 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.

Migrate 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.

Migrating 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. See "7. Decommission Event Notification Module 8.x (if installed)" on page 1303 for more information.

Configure notifications

Before your system can send out notifications, you must configure the email server and the modem COM port to send SMS messages.

Configuring the Email Server

To send notifications using email, you must configure the email server.

To configure the email server:

1. "Opening Notifications Settings" on page 345
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.

Configuring 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. "Opening Notifications Settings" on page 345.
2. Click **Settings and Diagnostics** and then click **SMS Setup**.

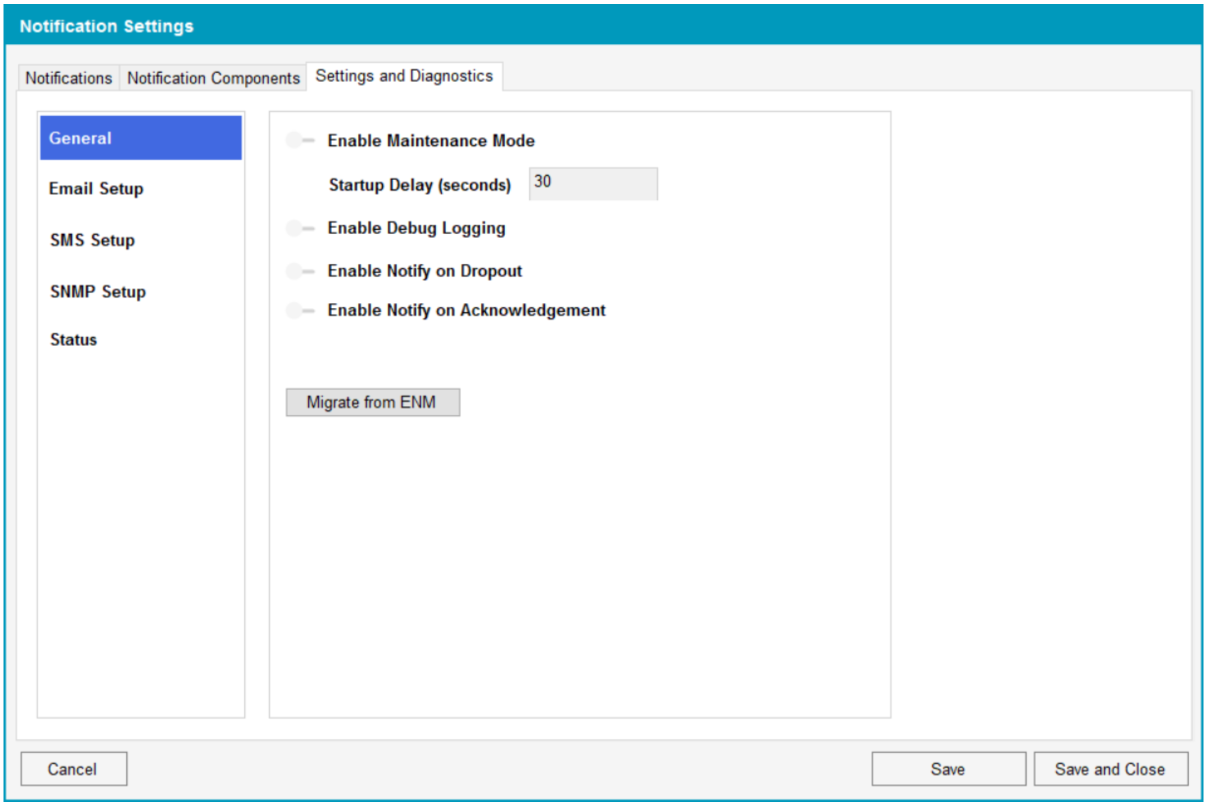
3. In COM Port, enter the modem COM port.
4. (Optional) Set the other SMS values.

   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.



Enable Maintenance Mode

Startup Delay (seconds) 30

Enable Debug Logging

Enable Notify on Dropout

Enable Notify on Acknowledgement

You can define the following settings:

- **Enable Maintenance Mode** – Disables notifications. See "Using Maintenance Mode" on page 343 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.)

To use Maintenance Mode:

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, clear **Send Configuration Announcements**.

   Typically, you would only clear 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 **Maintenance Mode** to disable it, and then click **Save**.

Create notifications

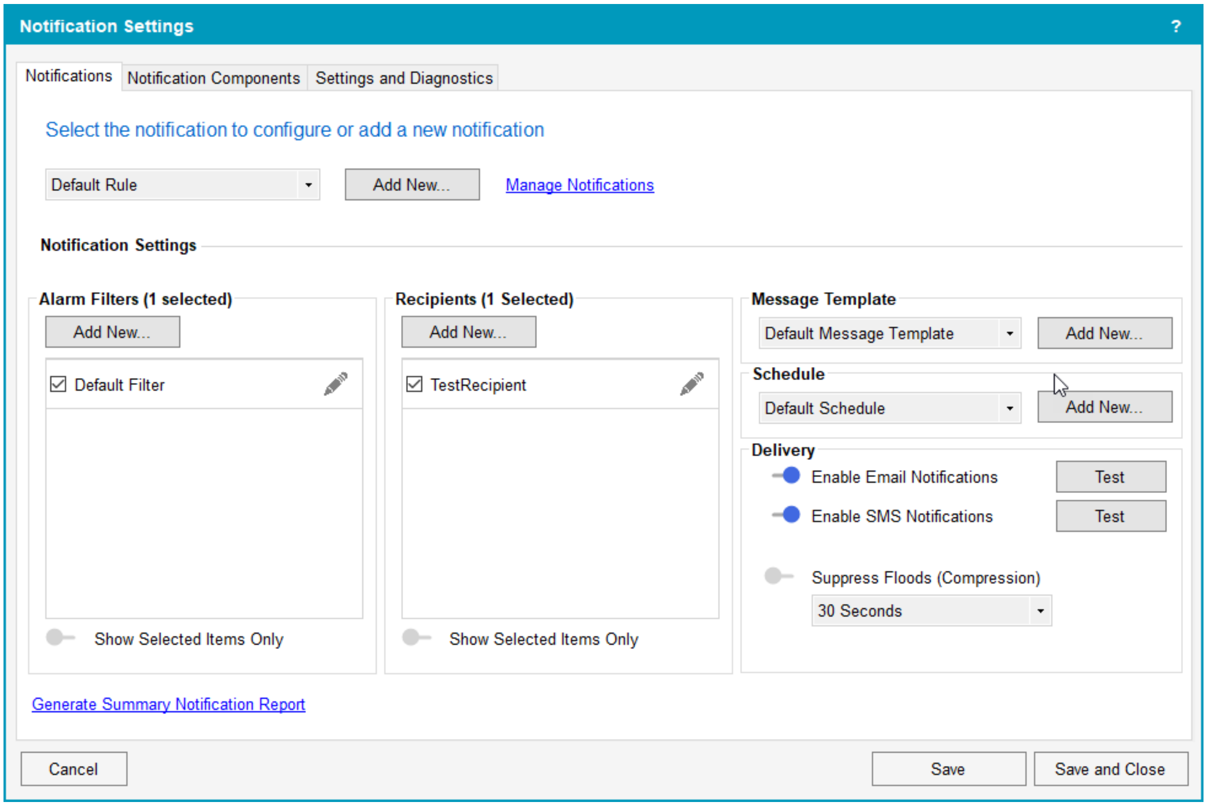
⚠ WARNING

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:



Select the notification to configure or add a new notification

Default Rule

Notification Settings

Alarm Filters (1 selected)

☑ Default Filter

Recipients (1 Selected)

☑ TestRecipient

Message Template: Default Message Template

Schedule: Default Schedule

Delivery

● Enable Email Notifications

● Enable SMS Notifications

○ Suppress Floods (Compression)

30 Seconds

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.

Creating a notification workflow

Create your system notifications either by editing and duplicating the default notification, or by adding a new one.

Creating a notification involves the following tasks:

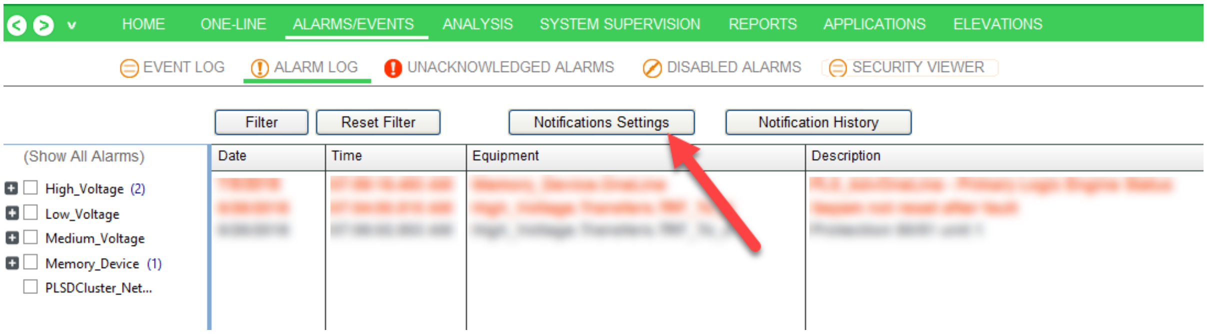
1. Add a new notification or duplicate an existing notification
2. Filter the alarms to be included in the notification.
3. Add recipients to the notification.
4. Define the schedule when recipients can receive the notification.
5. Set the notification relay.
6. Test the notification.

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.

Opening Notifications Settings

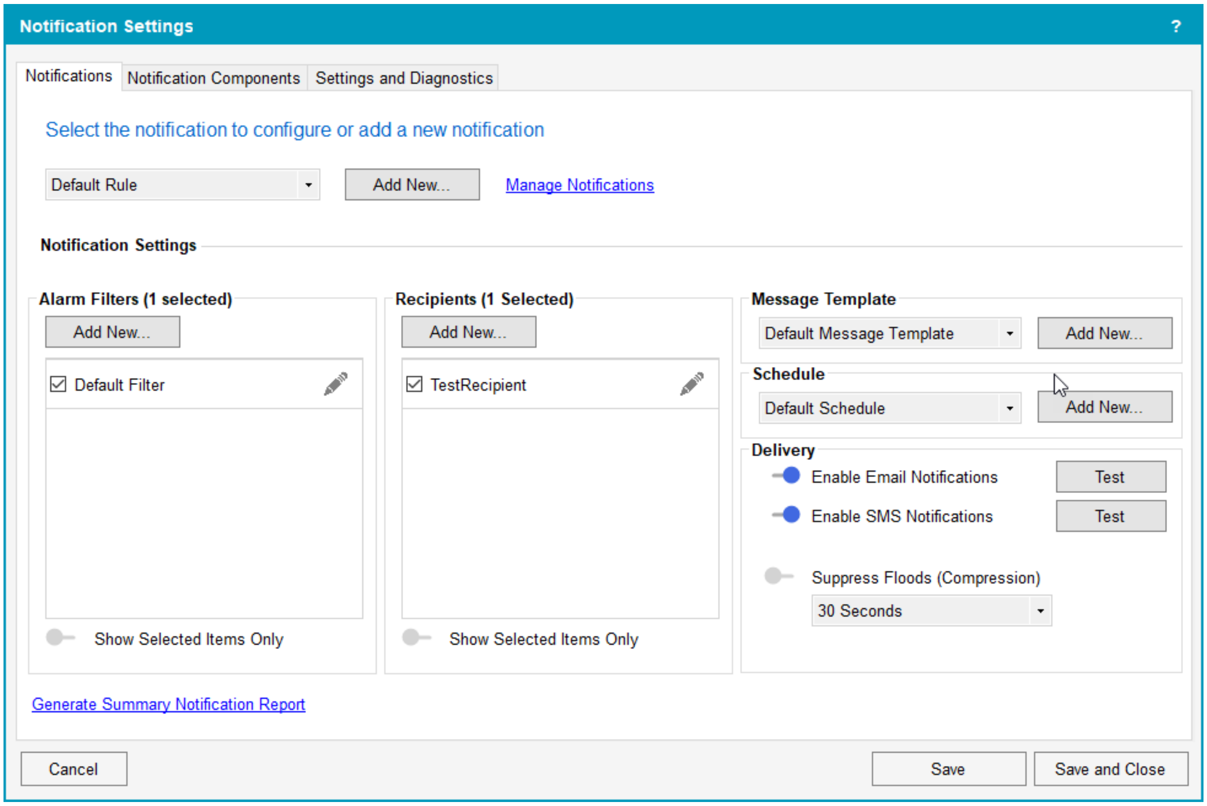
Open Notifications Settings from the Power SCADA Runtime.



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 appears:


Notification Settings

Select the notification to configure or add a new notification

Default Rule

Notification Settings

Alarm Filters (1 selected)

☑ Default Filter

Recipients (1 Selected)

☑ TestRecipient

Message Template: Default Message Template

Schedule: Default Schedule

Delivery

- Enable Email Notifications
- Enable SMS Notifications
- Suppress Floods (Compression): 30 Seconds

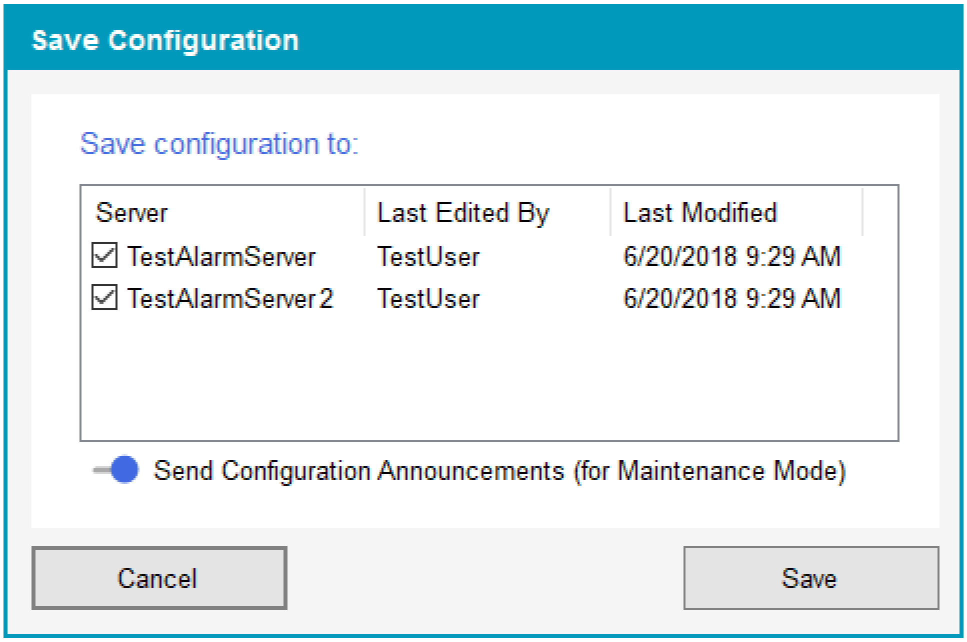
Assuming notifications from Event Notification Module (ENM) were not migrated, when you first open Notifications Settings, a notification is included by default. This default notification includes an alarm filter that includes all alarms in the system, a fictitious recipient, a message template, and a schedule.

Create your system notifications either by editing and duplicating the default notification, or adding a new one.

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:



Save configuration to:

Server	Last Edited By	Last Modified
☑ TestAlarmServer	TestUser	6/20/2018 9:29 AM
☑ TestAlarmServer2	TestUser	6/20/2018 9:29 AM

● Send Configuration Announcements (for Maintenance Mode)

Click the server or servers to which you want to apply the changes and then click **Save**.

Creating 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.

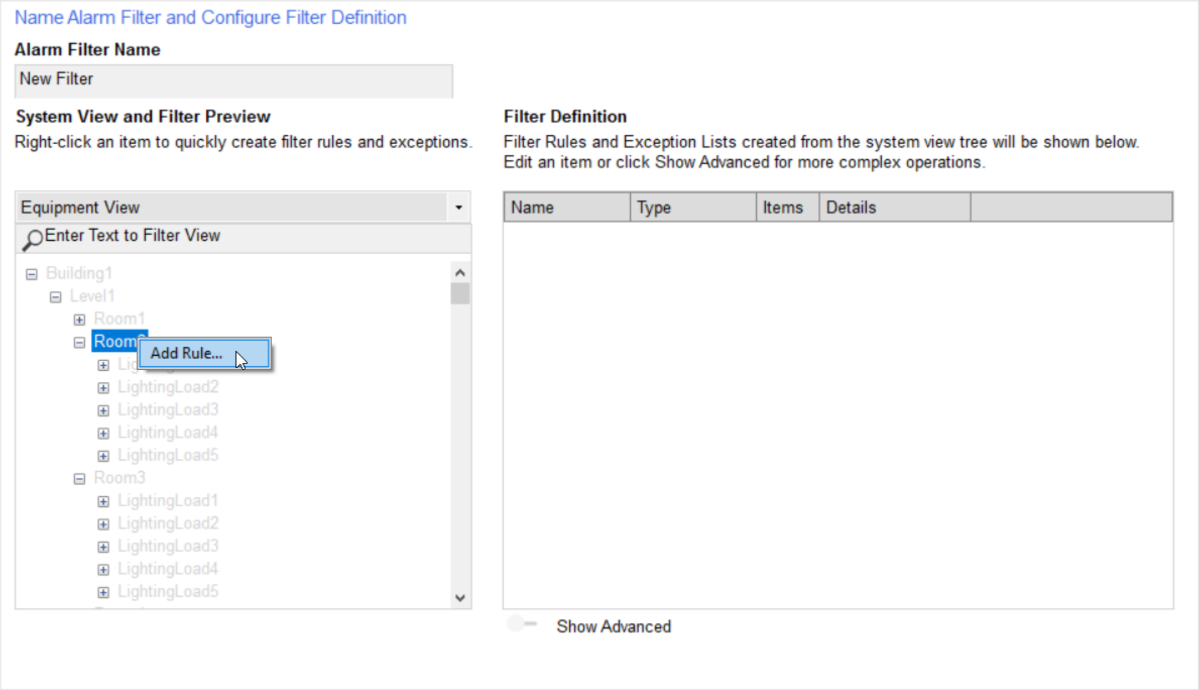
Rules

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:



Name Alarm Filter and Configure Filter Definition

Alarm Filter Name

New Filter

System View and Filter Preview

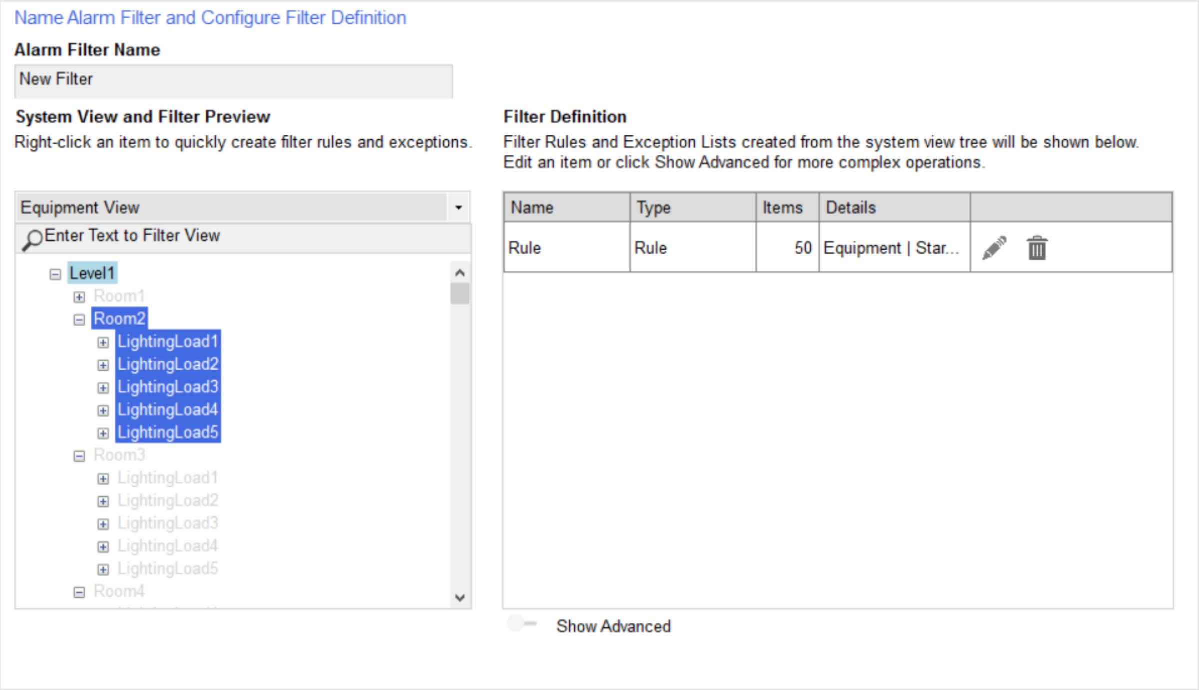
Right-click an item to quickly create filter rules and exceptions.

Filter Definition

Filter Rules and Exception Lists created from the system view tree will be shown below. Edit an item or click Show Advanced for more complex operations.

Name	Type	Items	Details	

Result: All the tags in the node and child nodes are added to the filter definition.



Name Alarm Filter and Configure Filter Definition

Alarm Filter Name

New Filter

System View and Filter Preview

Right-click an item to quickly create filter rules and exceptions.

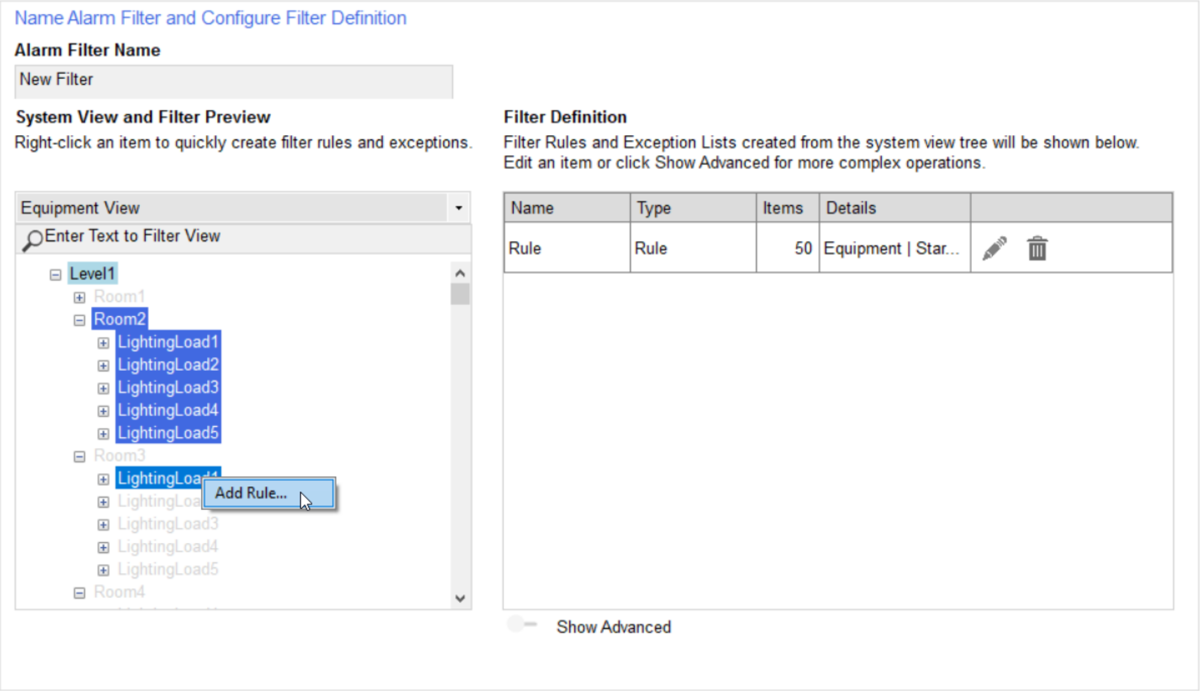
Filter Definition

Filter Rules and Exception Lists created from the system view tree will be shown below. Edit an item or click Show Advanced for more complex operations.

Name	Type	Items	Details	
Rule	Rule	50	Equipment | Star...	✎ 🗑

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:



Name Alarm Filter and Configure Filter Definition

Alarm Filter Name

New Filter

System View and Filter Preview

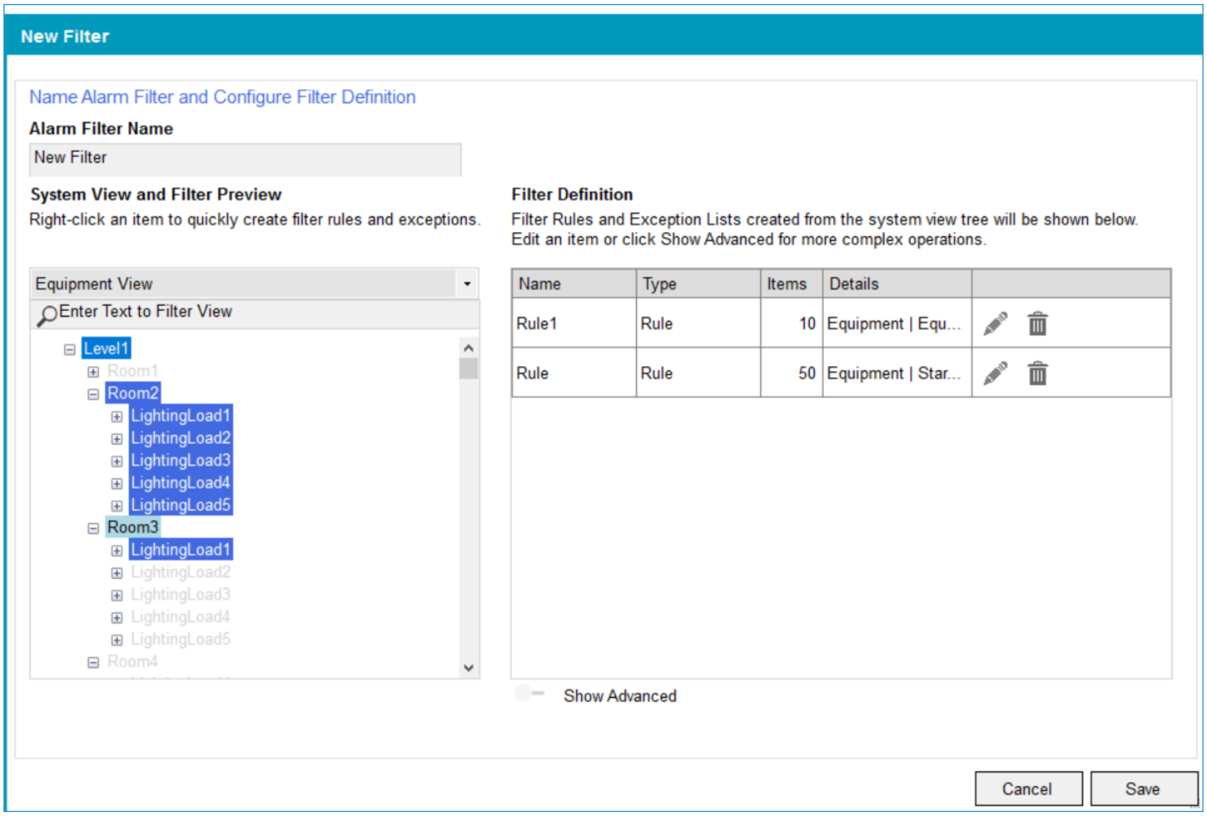
Right-click an item to quickly create filter rules and exceptions.

Filter Definition

Filter Rules and Exception Lists created from the system view tree will be shown below. Edit an item or click Show Advanced for more complex operations.

Name	Type	Items	Details
Rule	Rule	50	Equipment | Star...

Result: All the tags in LightingLoad1 are added to the filter definition.



New Filter

Name Alarm Filter and Configure Filter Definition

Alarm Filter Name

New Filter

System View and Filter Preview

Right-click an item to quickly create filter rules and exceptions.

Filter Definition

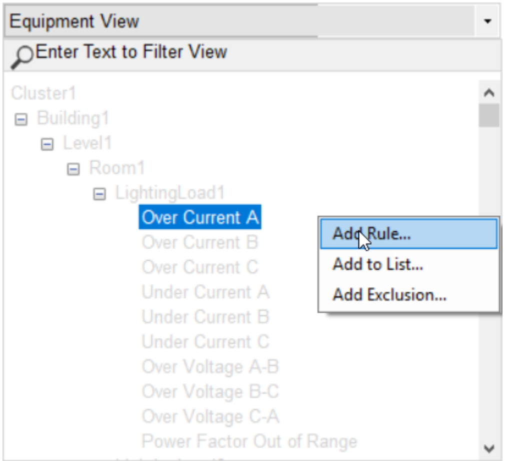
Filter Rules and Exception Lists created from the system view tree will be shown below. Edit an item or click Show Advanced for more complex operations.

Name	Type	Items	Details
Rule1	Rule	10	Equipment | Equ...
Rule	Rule	50	Equipment | Star...

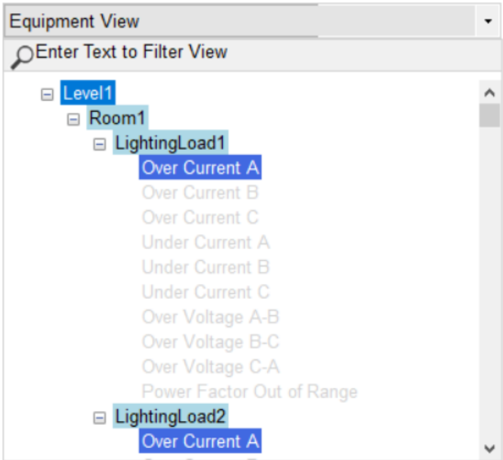
TIP: Notice the shading in the **Filter Preview**: Room2 and all its child nodes are highlighted in blue because all of their tags are part of the filter definition. Room3 is highlighted in light blue to indicate that some of its child node tags have been added to the filter definition.

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:



Result:

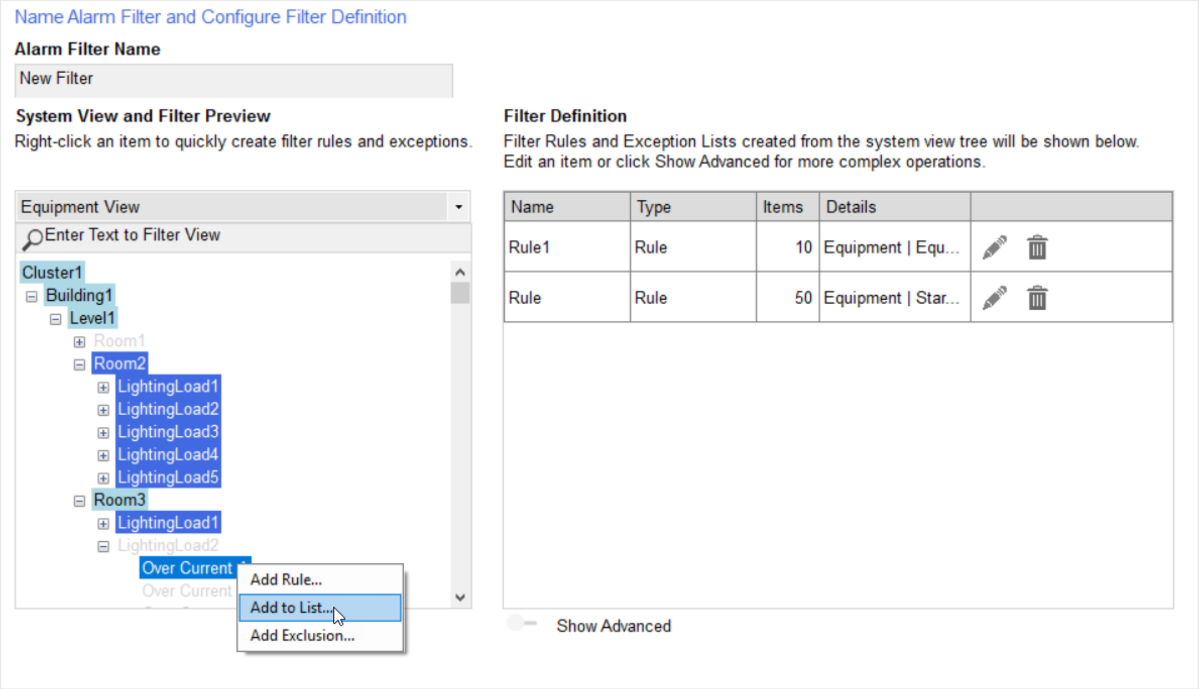


Lists

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:



Name Alarm Filter and Configure Filter Definition

Alarm Filter Name

New Filter

System View and Filter Preview

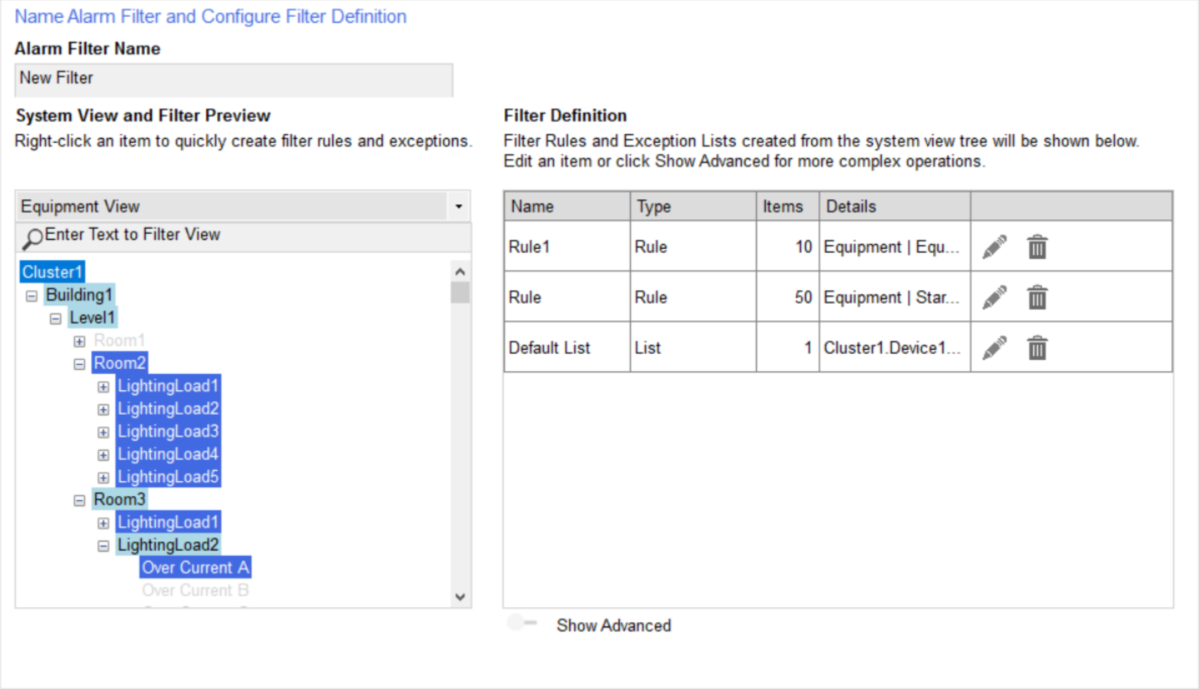
Right-click an item to quickly create filter rules and exceptions.

Filter Definition

Filter Rules and Exception Lists created from the system view tree will be shown below. Edit an item or click Show Advanced for more complex operations.

Name	Type	Items	Details
Rule1	Rule	10	Equipment | Equ...
Rule	Rule	50	Equipment | Star...

Result: The tag is added to the filter definition.



Name Alarm Filter and Configure Filter Definition

Alarm Filter Name

New Filter

System View and Filter Preview

Right-click an item to quickly create filter rules and exceptions.

Filter Definition

Filter Rules and Exception Lists created from the system view tree will be shown below. Edit an item or click Show Advanced for more complex operations.

Name	Type	Items	Details
Rule1	Rule	10	Equipment | Equ...
Rule	Rule	50	Equipment | Star...
Default List	List	1	Cluster1.Device1...

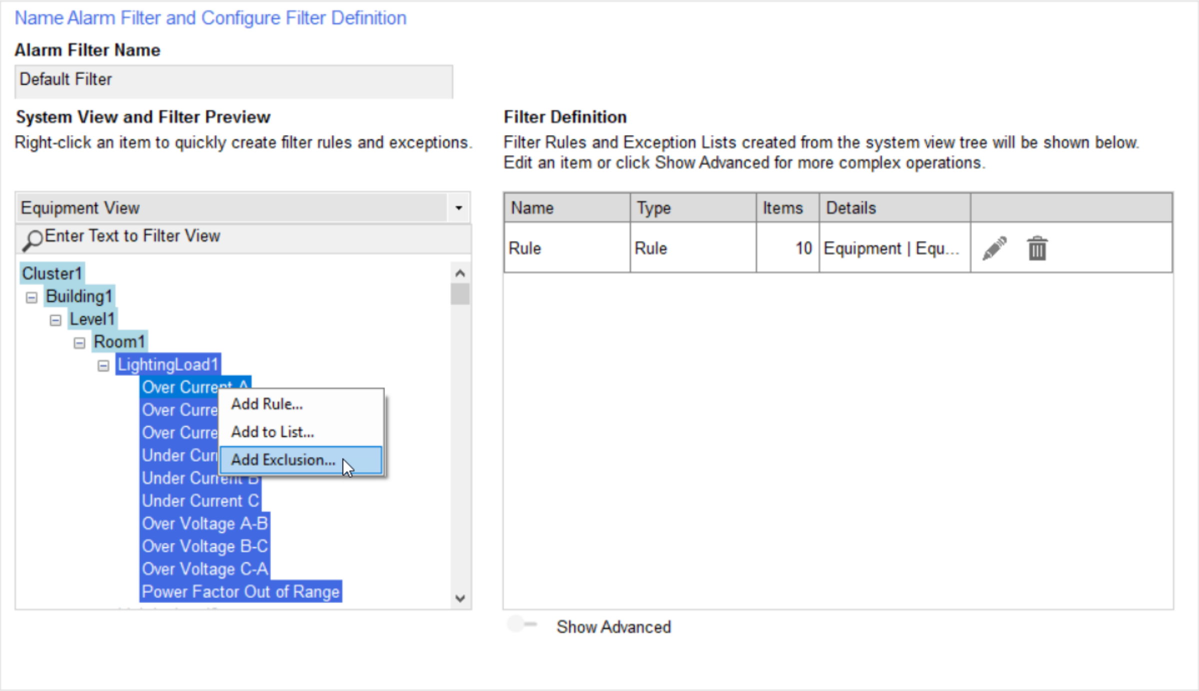
Exclusions

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.



Name Alarm Filter and Configure Filter Definition

Alarm Filter Name

Default Filter

System View and Filter Preview

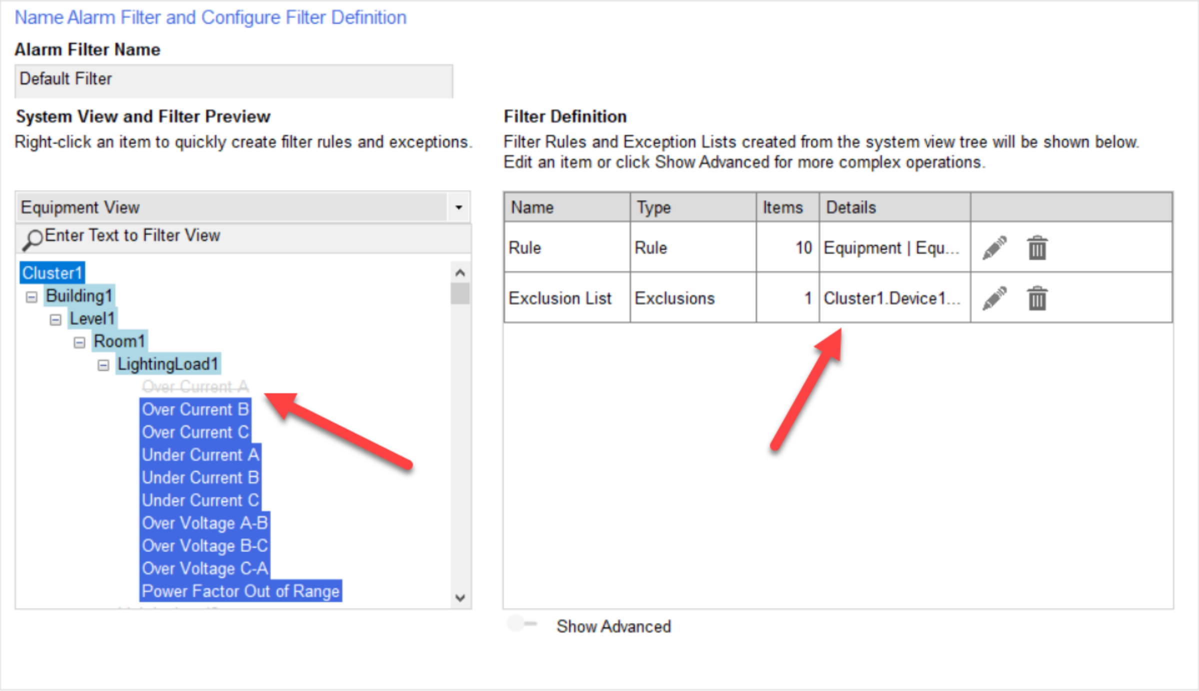
Right-click an item to quickly create filter rules and exceptions.

Filter Definition

Filter Rules and Exception Lists created from the system view tree will be shown below. Edit an item or click Show Advanced for more complex operations.

Name	Type	Items	Details
Rule	Rule	10	Equipment | Equ...

Result: The tag is removed from the rule.



Name Alarm Filter and Configure Filter Definition

Alarm Filter Name

Default Filter

System View and Filter Preview

Right-click an item to quickly create filter rules and exceptions.

Filter Definition

Filter Rules and Exception Lists created from the system view tree will be shown below. Edit an item or click Show Advanced for more complex operations.

Name	Type	Items	Details
Rule	Rule	10	Equipment | Equ...
Exclusion List	Exclusions	1	Cluster1.Device1...

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 "Creating basic alarm filters" on page 353 for more information. Create advanced alarm filters using the dedicated rule, list and exclusion filter windows. See "Creating advanced alarm filters" on page 354 for more information.

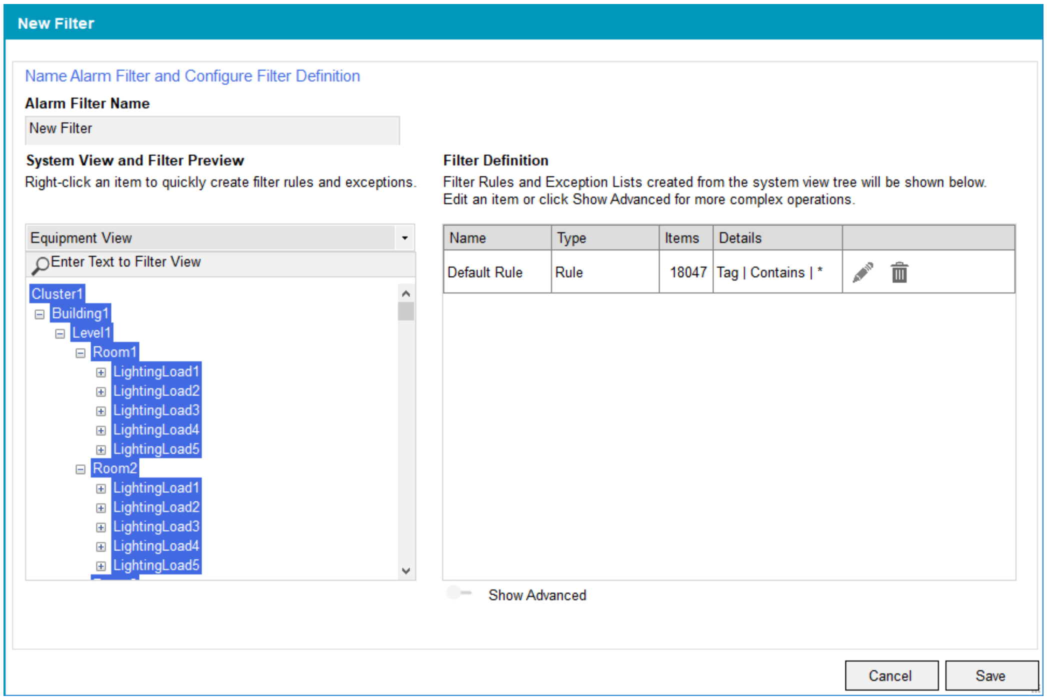
Creating basic alarm filters

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 "About Alarm Filters" on page 347 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**.



New Filter

Name Alarm Filter and Configure Filter Definition

Alarm Filter Name

New Filter

System View and Filter Preview

Right-click an item to quickly create filter rules and exceptions.

Filter Definition

Filter Rules and Exception Lists created from the system view tree will be shown below.
Edit an item or click Show Advanced for more complex operations.

Name	Type	Items	Details
Default Rule	Rule	18047	Tag | Contains | *

   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**:

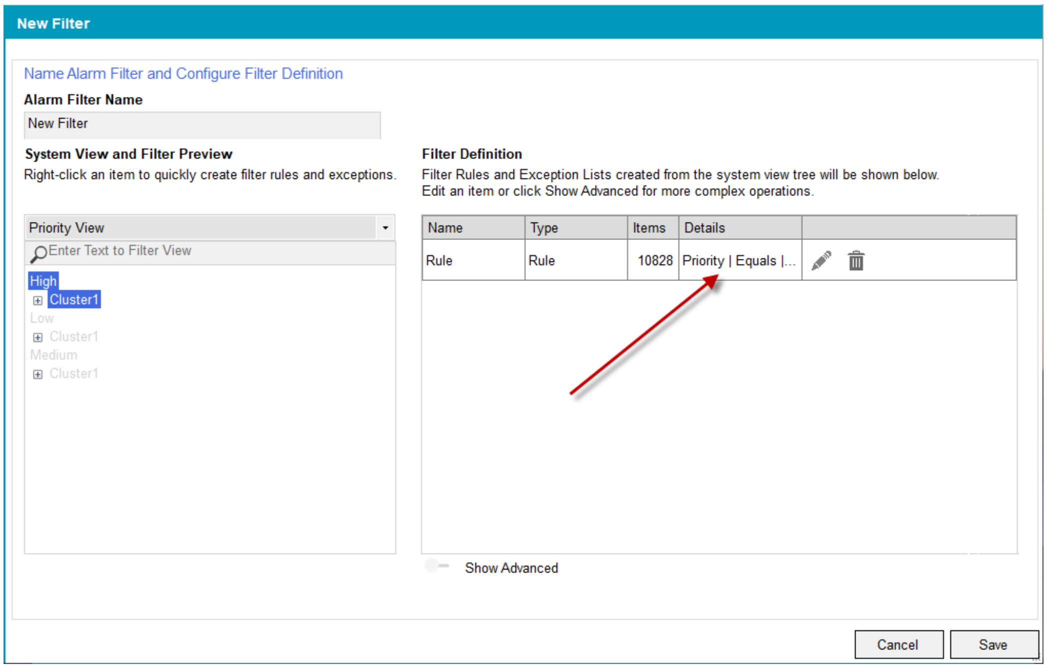
   a. Select the system view that sorts the alarms for your needs.

      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:



New Filter

Name Alarm Filter and Configure Filter Definition

Alarm Filter Name

New Filter

System View and Filter Preview

Right-click an item to quickly create filter rules and exceptions.

Priority View

Enter Text to Filter View

- High
  - ⊞ Cluster1
- Low
  - ⊞ Cluster1
- Medium
  - ⊞ Cluster1

Filter Definition

Filter Rules and Exception Lists created from the system view tree will be shown below. Edit an item or click Show Advanced for more complex operations.

Name	Type	Items	Details
Rule	Rule	10828	Priority | Equals |...

> **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 "Creating advanced alarm filters" on page 354.

Creating 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.

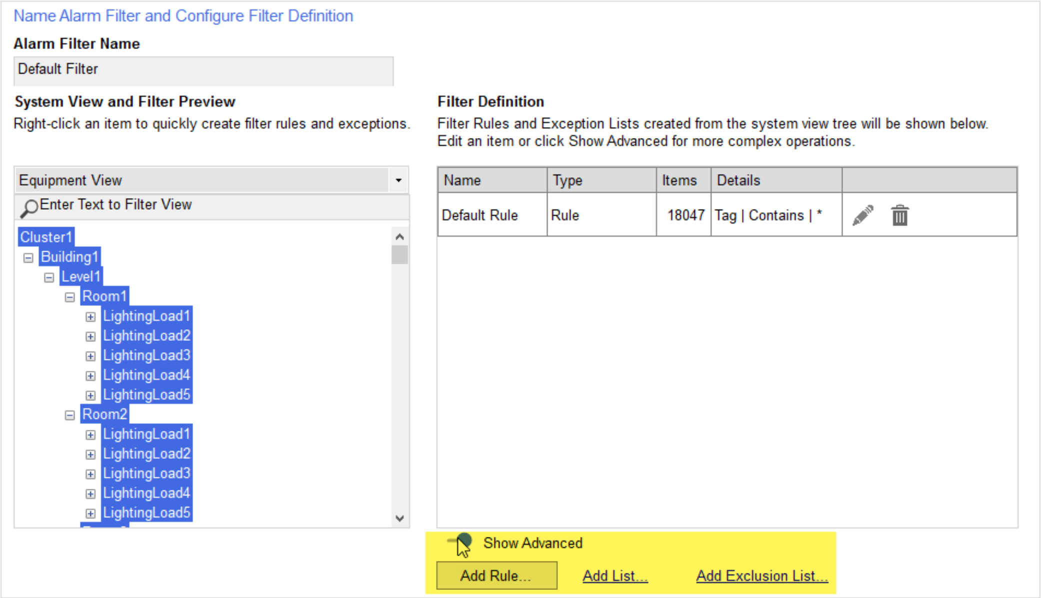
> **NOTE**: Before creating advanced alarm filters, you should have a good understand of alarm filter rules, lists, and exclusions. See "About Alarm Filters" on page 347 for details. Also note the following:
> – If an alarm filter contains an exclusion that is met, the notification will not be sent. Therefore, use exclusions with care.
> – Thoroughly test your alarm notifications before deploying them on a live system.

Viewing the 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.



Name Alarm Filter and Configure Filter Definition

Alarm Filter Name

Default Filter

System View and Filter Preview

Right-click an item to quickly create filter rules and exceptions.

Filter Definition

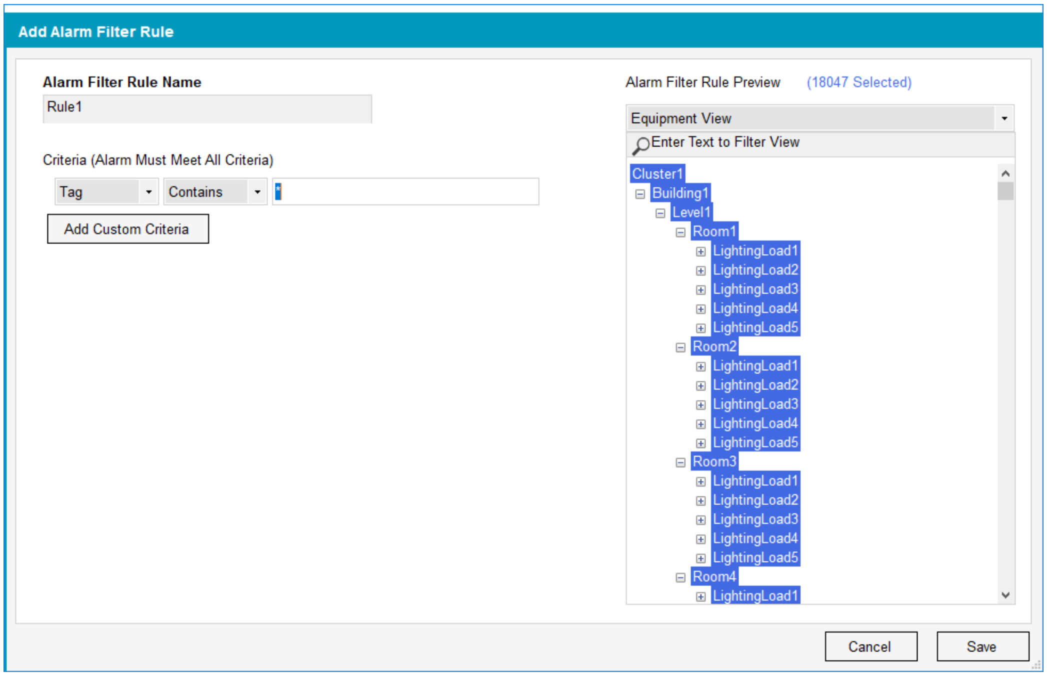
Filter Rules and Exception Lists created from the system view tree will be shown below. Edit an item or click Show Advanced for more complex operations.

Name	Type	Items	Details
Default Rule	Rule	18047	Tag | Contains | *

Adding a custom rule

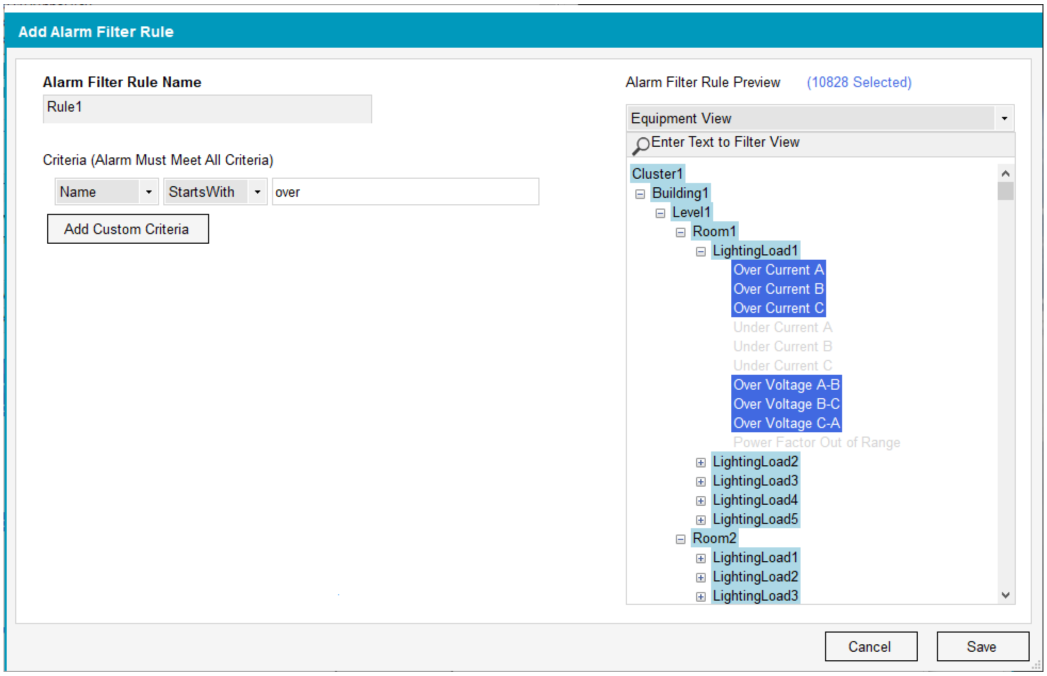
1. Click **Add Rule**.

   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**.

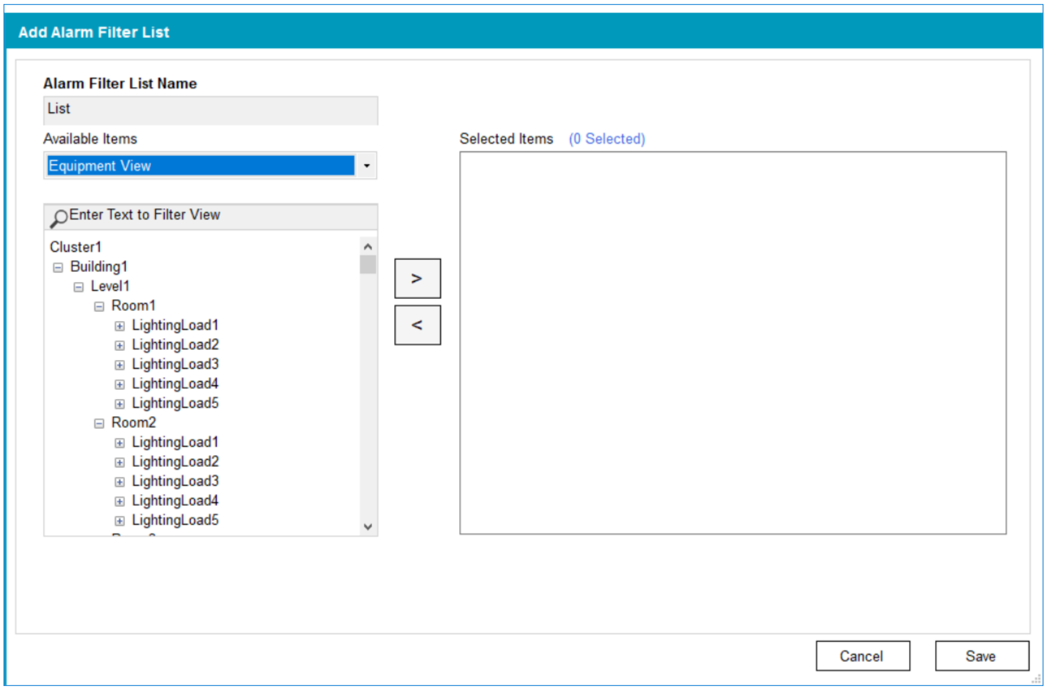
Adding a custom list or exclusion list

NOTE: The procedure for adding lists and exclusion lists is the same. This following procedure adds an alarm filter list.

To add a custom list or exclusion list:

1. Click **Add List**.

   The Add Alarm Filter List window appears.



Add Alarm Filter List

Alarm Filter List Name: List

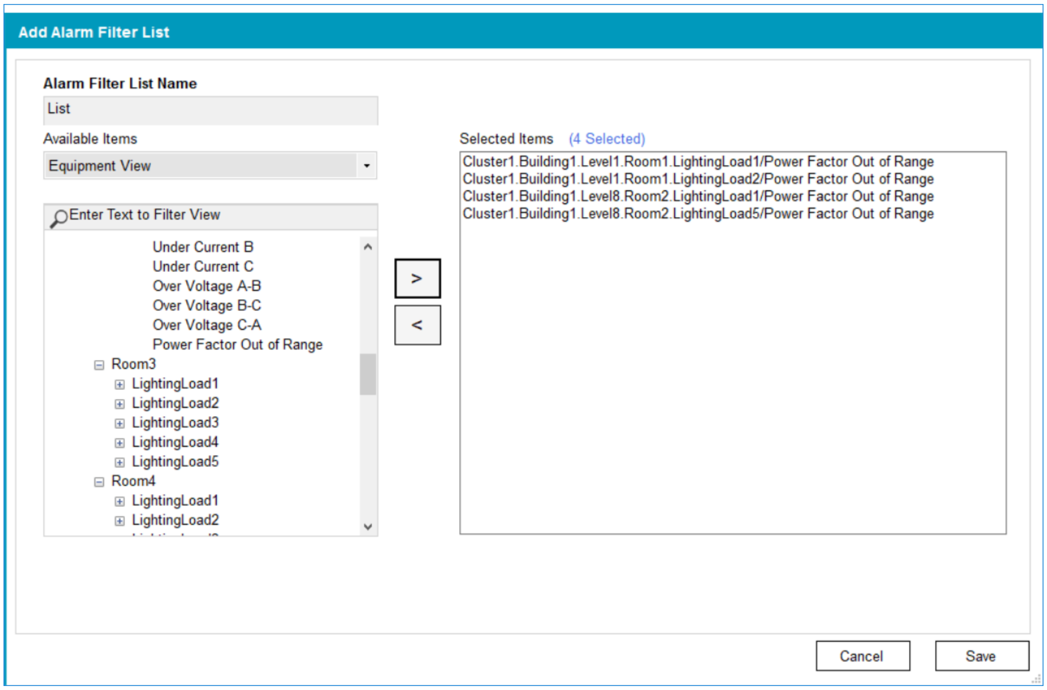
Available Items: Equipment View

Selected Items (0 Selected)

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.



Alarm Filter List Name: List

Available Items: Equipment View

Enter Text to Filter View

Under Current B
Under Current C
Over Voltage A-B
Over Voltage B-C
Over Voltage C-A
Power Factor Out of Range
Room3
LightingLoad1
LightingLoad2
LightingLoad3
LightingLoad4
LightingLoad5
Room4
LightingLoad1
LightingLoad2

Selected Items (4 Selected)

Cluster1.Building1.Level1.Room1.LightingLoad1/Power Factor Out of Range
Cluster1.Building1.Level1.Room1.LightingLoad2/Power Factor Out of Range
Cluster1.Building1.Level8.Room2.LightingLoad1/Power Factor Out of Range
Cluster1.Building1.Level8.Room2.LightingLoad5/Power Factor Out of Range

7. When you are finished adding tags, click **Save**.

Adding alarm filters to a notification

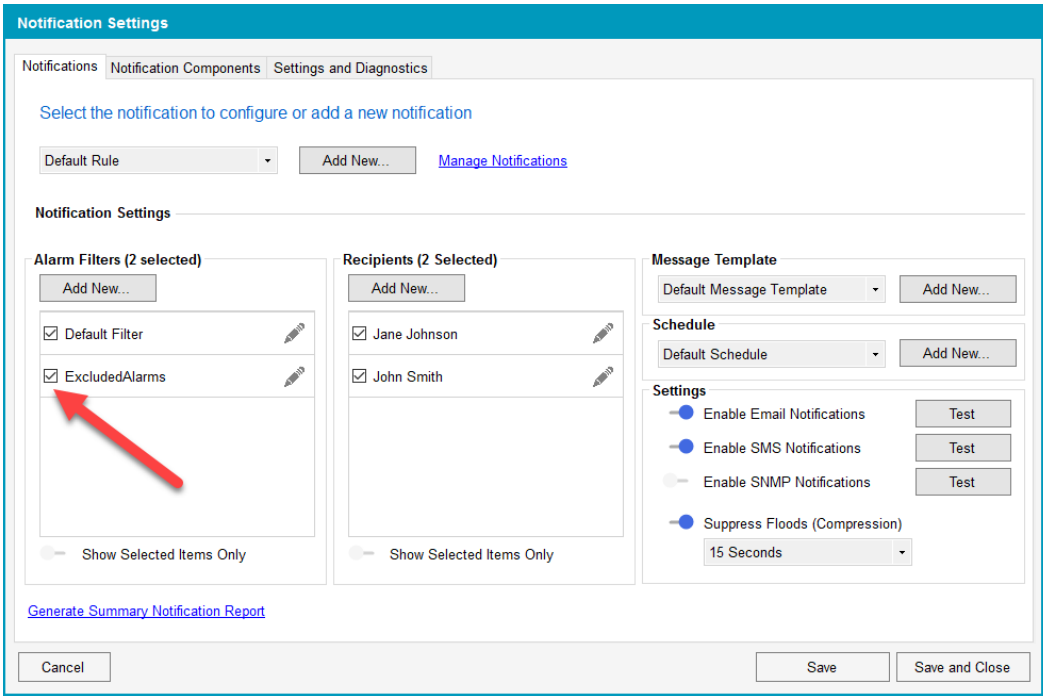
After you create an alarm filter, you need to add it to the notification.

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:



Notification Settings

Notifications | Notification Components | Settings and Diagnostics

Select the notification to configure or add a new notification

Default Rule ▾ | Add New... | Manage Notifications

Notification Settings

Alarm Filters (2 selected)
Add New...
☑ Default Filter
☑ ExcludedAlarms
Show Selected Items Only

Recipients (2 Selected)
Add New...
☑ Jane Johnson
☑ John Smith
Show Selected Items Only

Message Template
Default Message Template ▾ | Add New...

Schedule
Default Schedule ▾ | Add New...

Settings
● Enable Email Notifications | Test
● Enable SMS Notifications | Test
○ Enable SNMP Notifications | Test
● Suppress Floods (Compression)
15 Seconds ▾

Generate Summary Notification Report

Cancel | Save | Save and Close

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**.

Adding 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.

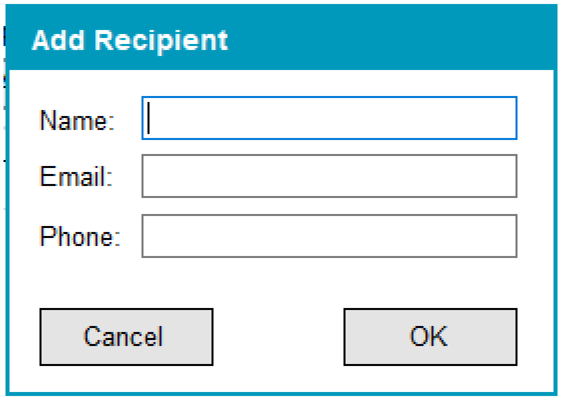
For more information on recipients, see "Managing recipients" on page 360.

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.



Add Recipient

Name:

Email:

Phone:

Cancel | OK

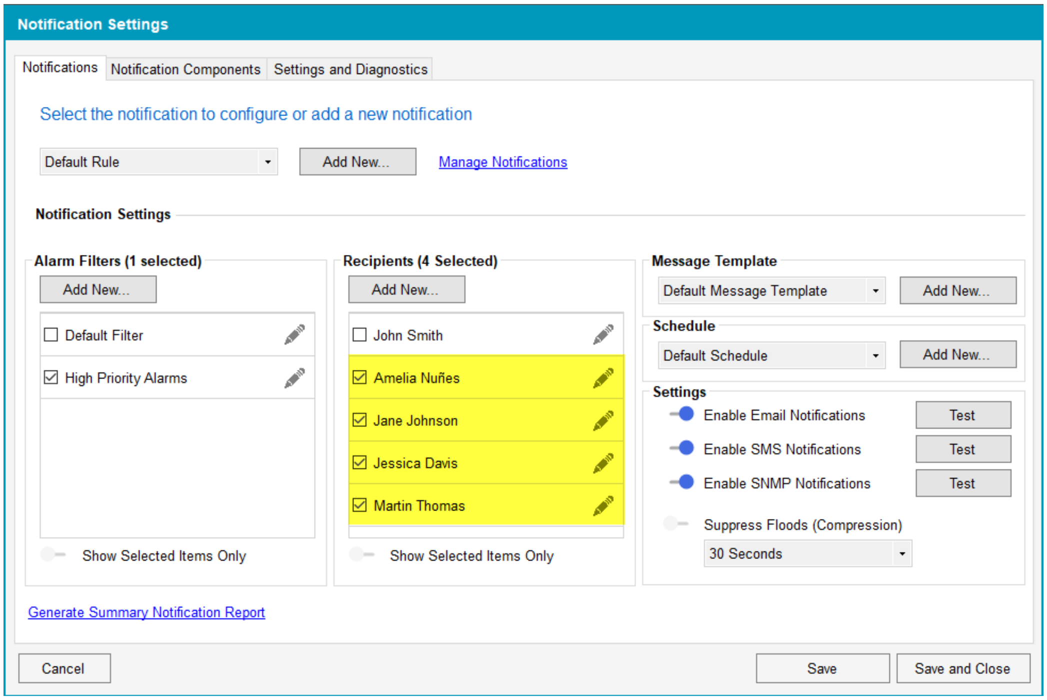
   b. Enter the recipient details.

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:



Notification Settings

Notifications | Notification Components | Settings and Diagnostics

Select the notification to configure or add a new notification

Default Rule ▾ | Add New... | Manage Notifications

Notification Settings

Alarm Filters (1 selected)

Add New...

☐ Default Filter

☑ High Priority Alarms

Show Selected Items Only

Recipients (4 Selected)

Add New...

☐ John Smith

☑ Amelia Nuñes

☑ Jane Johnson

☑ Jessica Davis

☑ Martin Thomas

Show Selected Items Only

Message Template

Default Message Template ▾ | Add New...

Schedule

Default Schedule ▾ | Add New...

Settings

Enable Email Notifications | Test

Enable SMS Notifications | Test

Enable SNMP Notifications | Test

Suppress Floods (Compression)

30 Seconds ▾

Generate Summary Notification Report

Cancel | Save | Save and Close

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**.

Managing recipients

From the **Notification Components** tab, you can add, edit, and delete recipients.

Adding 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.

Editing a recipient

1. From the recipient list, click **Edit** ✎ .
2. Edit the recipient details and then click **Save**.

Deleting 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 "Managing schedules" on page 361.

Managing schedules

Add, edit, and delete schedules in the **Notification Components** pane.

> **NOTE**: For **Phone**, enter numbers only. Do not enter parentheses or hyphens.

Adding 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.

Editing a schedule

1. From the schedule list, click ✎ (**Edit**).
2. Edit the schedule details and then click **OK**.

Deleting 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.

Adding 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.

Managing message templates

Renaming 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**.

Deleting 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.

Enabling and testing notification delivery

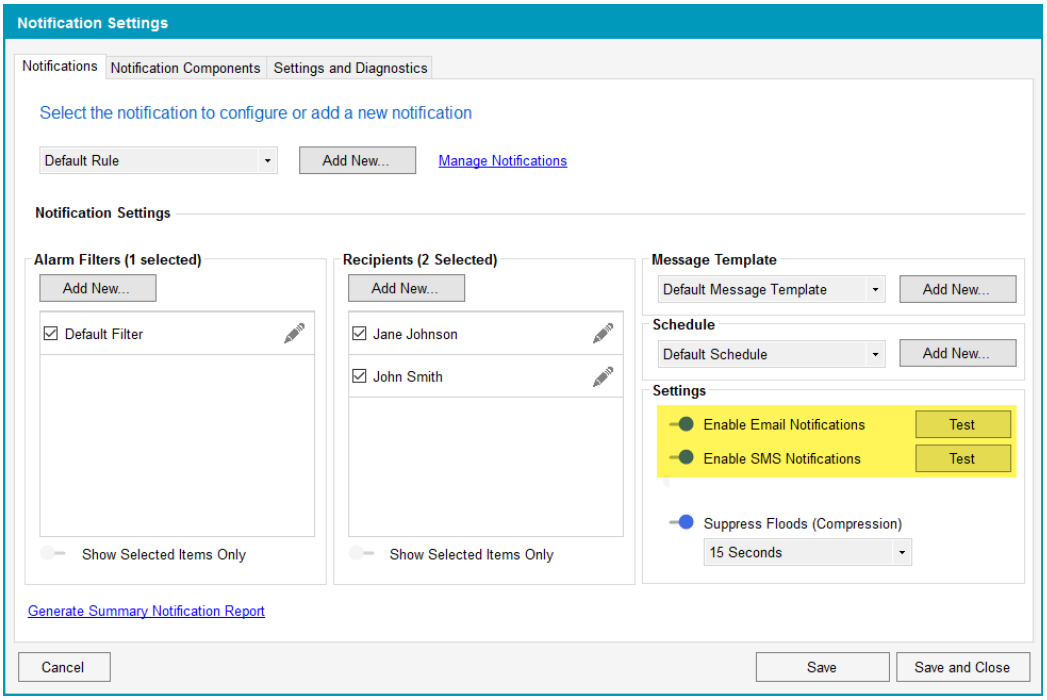
After you have configured all the notification components, choose the delivery methods that Notifications Settings will use to notify people if an alarm occurs.

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.



Notification Settings

Select the notification to configure or add a new notification

Default Rule

Notification Settings

Alarm Filters (1 selected)

☑ Default Filter

Recipients (2 Selected)

☑ Jane Johnson

☑ John Smith

Message Template: Default Message Template

Schedule: Default Schedule

Settings

Enable Email Notifications

Enable SMS Notifications

Suppress Floods (Compression): 15 Seconds

3. Click **Save**.

Managing 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 343 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.

Renaming a notification

1. In the **Notifications** pane, click **Manage Notifications**.
2. In Manage Notifications, click **Edit Name**.
3. Edit the name then click **OK**.

Duplicating 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.

Deleting a notification

1. In the **Notifications** pane, click **Manage Notifications**.
2. In Manage Notifications, click 🗑.
3. Click **Yes** to confirm the deletion.

Suppressing 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:

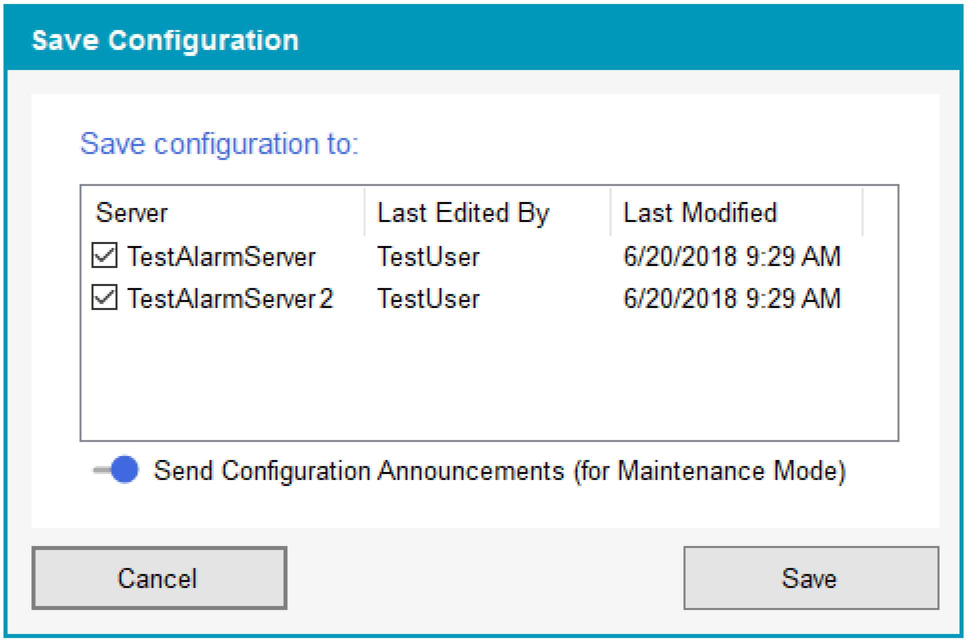
You enable suppress floods and set the time period to 30 seconds. If 500 alarms occur during that time period, Notifications sends out 2 messages:

- 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:



Save Configuration

Save configuration to:

Server	Last Edited By	Last Modified
☑ TestAlarmServer	TestUser	6/20/2018 9:29 AM
☑ TestAlarmServer2	TestUser	6/20/2018 9:29 AM

● Send Configuration Announcements (for Maintenance Mode)

Creating 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 367.

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:

   ```
   [Project Drive]\ProgramData\Schneider Electric\Power
   SCADA Operation\v2020 R2\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.

Web Applications

The Web Applications component provides access to web-based Power SCADA Operation applications.

Refer to the following topics to configure the Web Applications:

- "Alarms configuration" on page 368
- "Diagrams configuration" on page 389
- "Trends configuration" on page 569
- "Web Applications settings" on page 574

Alarms configuration

Use Alarms to view incidents, alarms and events. You access the information in Alarms through views which are saved in the View Library. Power SCADA Operation comes with a number of pre-configured System Views. These system views cannot be deleted or modified, but you can create additional views and customize them to meet your needs.

⚠ **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

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, or equipment damage.

TIP: Open Alarms from the **ALARMS** link in the Web Applications banner.

Define number of alarms to be recorded, batch processing time intervals, and session timeout

Use this procedure to customize the number of records to retain, length of time for retention, batch processing intervals, and session timeout due to inactivity. This can help lessen the time needed to load alarms on your platform server on startup.

1. Open the Power SCADA Operation **appsettings.json** file.

   Example location: C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020\Applications\Platform Server

2. Change values for:

   - ArchiveCheckHours: archive interval for alarms past maximums and age ranges.
   - ArchiveMaxRecords: maximum number of records to retain.
   - ArchiveMaxMonths: maximum of age of retained records.
   - AlarmQueryRangeWeeks: time range of retained records in weeks.
   - AlarmQueryWindowHours: number of hours to batch process at one time in hours.
   - AlarmQueryTimeoutMinutes: session inactivity timeout in minutes.

3. **Save** the appsettings.json file.

Power SCADA Operation ships with default Alarm and Incident categories. To customize these categories, you must create a classifications JSON file. When you are editing a JSON file, use a JSON editing tool to ensure the JSON will parse. For an example JSON file, see:

C:\ProgramData\Schneider Electric\Power SCADA Operation\v2020 R2\Examples

Save the JSON file to the project directory.

For information on how to configure the Alarms application, see:

- "Adding a new Alarms view" on page 370
- "Copying an Alarms view" on page 372
- "Editing an Alarms view" on page 372
- "Moving an Alarms view" on page 373

- "Deleting an Alarms view" on page 374
- "Setting a default Alarms view" on page 375
- "Creating alarm menus" on page 755
- "Exporting alarm menus" on page 382
- "Importing alarm menus" on page 384
- "Alarm Views" on page 575

For reference information see:

- "Alarms UI" on page 1263

For information on how to use Alarms, see "Alarms" on page 730.

Adding a new Alarms view

Add new Alarms views to access certain types of alarms, incidents, or events. For example, create views to see unacknowledged alarms, high priority alarms, or power quality incidents. You can also create views that only include certain sources, eliminating information you are not interested in seeing.

To add a new Alarms view:

1. In the Alarm application, open the view library and navigate to the folder where you want to create the view.

   NOTE: The System Views folder is read-only. You cannot add folders or alarm views to the System Views folder.

2. (Optional) Add a new folder by clicking **Add Folder** 🗀+ at the bottom of the library panel, or by clicking **Add Folder** in the **Options** menu ☰ at the top of the library.

3. In the view library, at the bottom of the panel, click **Add View** +, or click **Add View** in the **Options** menu ☰ at the top of the library. This creates a new view and opens the view settings.

4. In View Settings, enter a view name, select a location where to save the view in the library, set access permissions, and select the view type.

   NOTE: A public item is visible to all users in your user group. A private item is visible to you and any user in your user group with Edit permissions on this item type. See "Assign and control user privileges" on page 686 for details.

5. Adjust the filter settings for Priority, State, Sources, and Categories to customize the view if necessary.

   NOTE: Not all of these filters are available for all view types.

6. **Save** the view.

To copy an existing Alarms view:

1. In the Alarm application, open the view library and navigate to the view you want to copy.
2. Right-click the view name or click **Options** ⋮ for this view, and select **Duplicate** to create a copy in the same folder. Select **Copy To** to create a copy in a different folder.
3. (Optional) In the view library, select the new view, right-click the view name or click **Options** ⋮ for this view, and select **Edit** to open View Settings. You can also open View Settings by double-clicking the view name. Change the view name, location, access permissions and view type, and adjust the filter settings for Priority, State, Sources, and Categories to customize the view if necessary.

   NOTE: A public item is visible to all users in your user group. A private item is visible to you and any user in your user group with Edit permissions on this item type. See "Assign and control user privileges" on page 686 for details.

   NOTE: Not all of these filters are available for all view types.

4. **Save** the view.

NOTE: To copy a system view, use **Copy To** to create a copy in a different location. You can also open the System View for Edit and then click **Save as New** in the view settings to create a copy in View Library > Home. You cannot use **Duplicate** because the System Views folder is read-only.

Related topics:

- Adding a new Alarms view
- "Copying an Alarms view" on page 372
- "Editing an Alarms view" on page 372
- "Moving an Alarms view" on page 373
- "Deleting an Alarms view" on page 374
- "Setting a default Alarms view" on page 375
- "Creating alarm menus" on page 755
- "Exporting alarm menus" on page 382
- "Importing alarm menus" on page 384
- "Alarm Views" on page 575

For reference information see:

- "Alarms UI" on page 1263

For information on how to use Alarms, see "Alarms" on page 730.

Copying an Alarms view

Copy Alarms views to quickly create new views that are the same as, or similar to existing views. For example, create a copy of a view to experiment with the view settings without affecting the original view. You can also use a copy of a view as a starting point for a new view that shares many of the settings of the original view.

To copy an Alarms view:

1. In the alarm viewer, open the view library and navigate to the view you want to copy.
2. Right-click the view name or click **Options** ⋮ for this view, and select **Duplicate** to create a copy in the same folder. Select **Copy To** to create a copy in a different folder.
3. (Optional) In the view library, select the new view, right-click the view name or click **Options** ⋮ for this view, and select **Edit** to open View Settings. You can also open View Settings by double-clicking the view name. Change the view name, and adjust the filter settings for Priority, State, Sources, and Categories to customize the view if necessary.

   NOTE: Not all of these filters are available for all view types.

4. **Save** the View.

NOTE: You cannot **Duplicate** a system view because the System Views folder is read-only. Use **Copy To** instead to create a copy in a different location.

Related topics:

- "Adding a new Alarms view" on page 370
- Copying an Alarms view
- "Editing an Alarms view" on page 372
- "Moving an Alarms view" on page 373
- "Deleting an Alarms view" on page 374
- "Setting a default Alarms view" on page 375
- "Creating alarm menus" on page 755
- "Exporting alarm menus" on page 382
- "Importing alarm menus" on page 384
- "Alarm Views" on page 575

For reference information see:

- "Alarms UI" on page 1263

For information on how to use Alarms, see "Alarms" on page 730.

Editing an Alarms view

Edit Alarms views to update the view name, the filter settings, or the location of the view in the view library.

> **NOTE**: You cannot overwrite system views. If you edit the settings of a system view and click **Save as New**, a copy of the view is created in View Library > Home.

To edit an Alarms view:

1. In the alarm viewer, open the view library and navigate to the view you want to edit.
2. Right-click the view name or click **Options** ⋮ for this view, and select **Edit** to open View Settings. You can also open View Settings by double-clicking the view name. Change the view name, location, access permissions and view type, and adjust the filter settings for Priority, State, Sources, and Categories to customize the view as necessary.
3. > **NOTE**: A public item is visible to all users in your user group. A private item is visible to you and any user in your user group with Edit permissions on this item type. See "Assign and control user privileges" on page 686 for details.

   > **NOTE**: Not all of these filters are available for all view types.
4. **Save** the view.

Related topics:

- "Adding a new Alarms view" on page 370
- "Copying an Alarms view" on page 372
- Editing an Alarms view
- "Moving an Alarms view" on page 373
- "Deleting an Alarms view" on page 374
- "Setting a default Alarms view" on page 375
- "Creating alarm menus" on page 755
- "Exporting alarm menus" on page 382
- "Importing alarm menus" on page 384
- "Alarm Views" on page 575

For reference information see:

- "Alarms UI" on page 1263

For information on how to use Alarms, see "Alarms" on page 730.

Moving an Alarms view

Move Alarms views to a different location in the view library to make them easier to find or easier to manage.

To move an Alarms view:

1. In the alarm viewer, open the view library and navigate to the view you want to move.
2. Right-click the view name or click **Options** ⋮ for this view, and select **Move To**. This opens the Select Location window.
3. In Select Location, select the location you want to move this view to.
4. Click **OK** to move the view .

NOTE: You cannot move system views or the System Views folder.

Related topics:

- "Adding a new Alarms view" on page 370
- "Copying an Alarms view" on page 372
- "Editing an Alarms view" on page 372
- Moving an Alarms view
- "Deleting an Alarms view" on page 374
- "Setting a default Alarms view" on page 375
- "Creating alarm menus" on page 755
- "Exporting alarm menus" on page 382
- "Importing alarm menus" on page 384
- "Alarm Views" on page 575

For reference information see:

- "Alarms UI" on page 1263

For information on how to use Alarms, see "Alarms" on page 730.

Deleting an Alarms view

Delete Alarms views that are no longer needed.

To delete an Alarms view:

1. In the alarm viewer, open the view library and navigate to the view you want to delete.
2. Right-click the view name or click **Options** ⋮ for this view, and select **Delete**
3. In Delete Content, click **Yes**, to delete the view from the view library.

NOTE: You cannot delete system views or the System Views folder.

Related topics:

- "Adding a new Alarms view" on page 370
- "Copying an Alarms view" on page 372
- "Editing an Alarms view" on page 372

- "Moving an Alarms view" on page 373
- Deleting an Alarms view
- "Setting a default Alarms view" on page 375
- "Creating alarm menus" on page 755
- "Exporting alarm menus" on page 382
- "Importing alarm menus" on page 384
- "Alarm Views" on page 575

For reference information see:

- "Alarms UI" on page 1263

For information on how to use Alarms, see "Alarms" on page 730.

Setting a default Alarms view

The default Alarms view is the view that opens when you first open the Alarms application. You can set a default for your own workspace or the entire system.

NOTE: Access to this application or function is controlled by user privileges. See "Assign and control user privileges" on page 686 for details.

To set a default Alarms view:

1. In the alarm viewer, open the view library and navigate to the view you want to set as default.
2. Right-click the view name or click **Options** ⋮ for this view, and select **Set as default**. This opens the Configure Default Item dialog.
3. In Configure Default Item, enable **Set as my default** or **Set as system default**.
4. Click **OK** to save the default settings.

Related topics:

- "Adding a new Alarms view" on page 370
- "Copying an Alarms view" on page 372
- "Editing an Alarms view" on page 372
- "Moving an Alarms view" on page 373
- "Deleting an Alarms view" on page 374
- Setting a default Alarms view
- "Creating alarm menus" on page 755
- "Exporting alarm menus" on page 382
- "Importing alarm menus" on page 384
- "Alarm Views" on page 575

For reference information see:

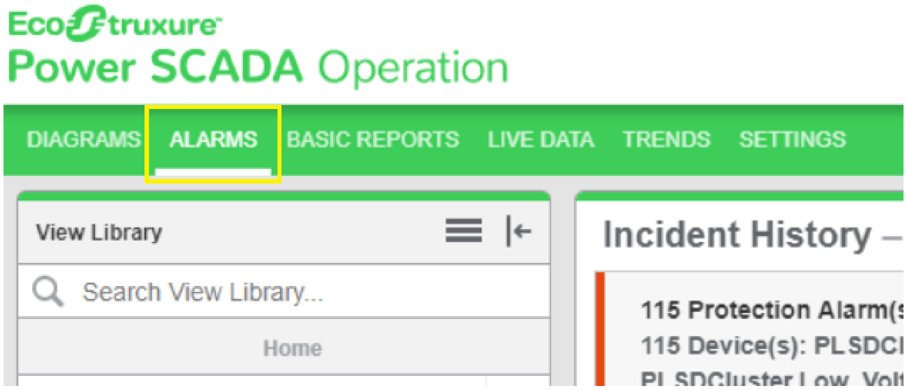
- "Alarms UI" on page 1263

For information on how to use Alarms, see "Alarms" on page 730.

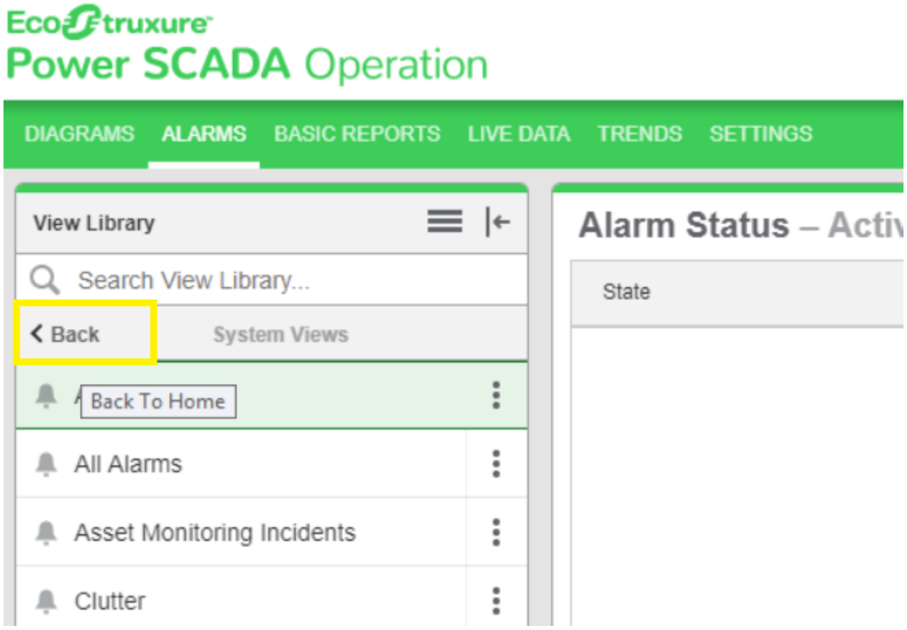
Creating alarm menus

To create alarm menus:

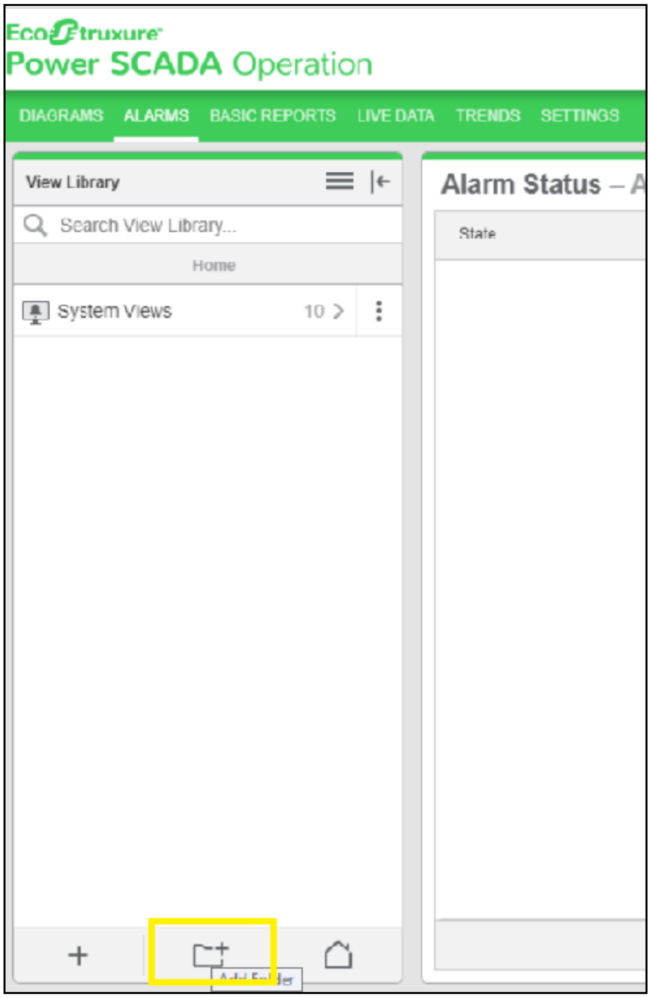
1. Log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)
2. Click **ALARMS**.



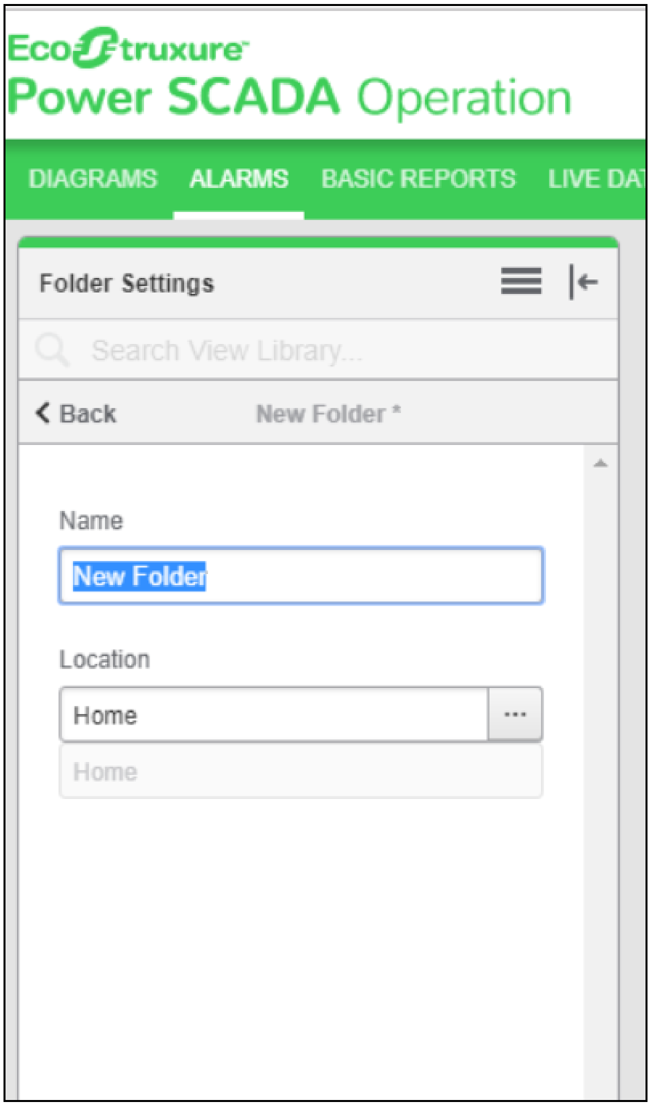
3. Click on **Back** to go back to the Home page.



4. At the bottom of the **View Library**, click **Add Folder**:


EcoStruxure
Power SCADA Operation
DIAGRAMS
ALARMS
BASIC REPORTS
LIVE DATA
TRENDS
SETTINGS
View Library
Search View Library...
Home
System Views
10
Alarm Status – A
State
Add Folder

5. Enter the folder **Name**:

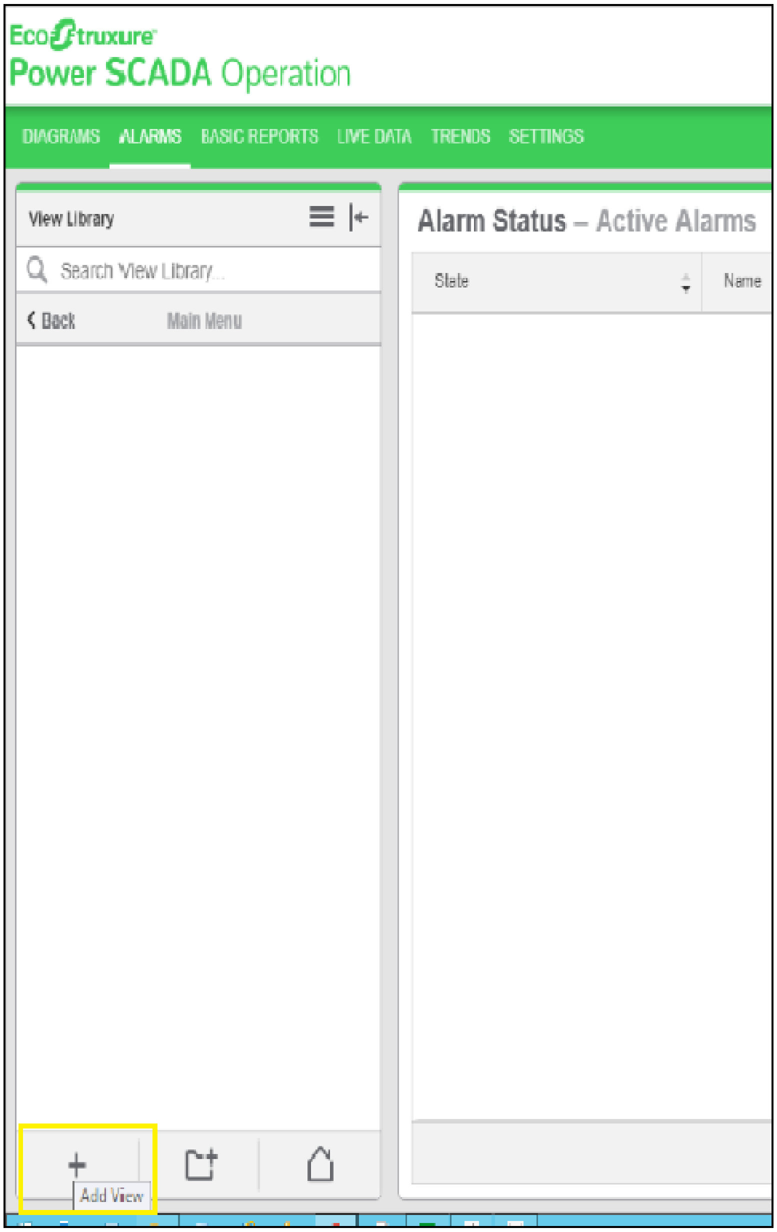


Name
New Folder

Location
Home

Home

6. At the bottom of the **View Library**, click **Add View**:



7. Configure the **View** by setting the following values based on your requirements:

    a. **View Name**: Type the view name.

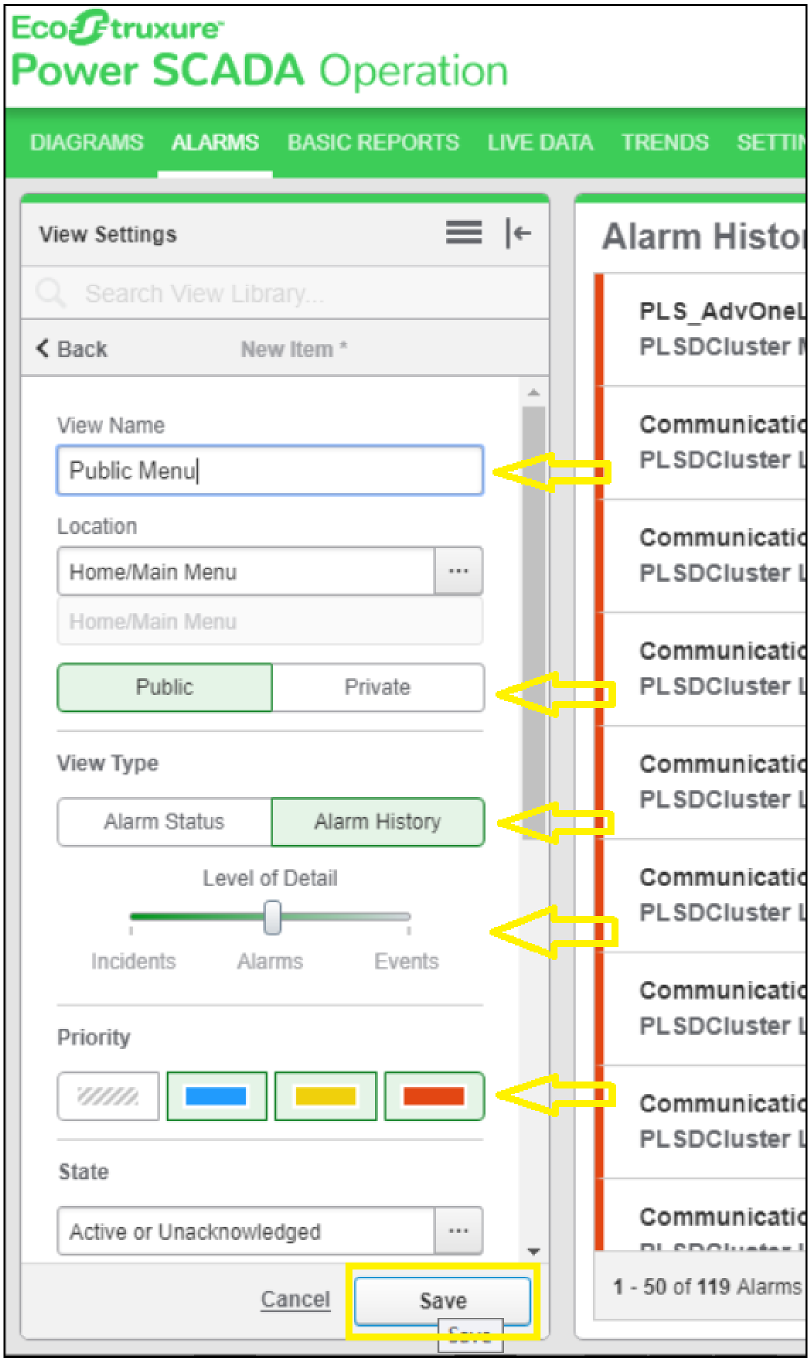
    b. **Location**: Select the location to display.

    c. Select **Public** or **Private**.

    d. **View Type**

    e. **Priority**

f. **State**



View Name
Public Menu

Location
Home/Main Menu
Home/Main Menu

[✓ Public] Private

View Type
Alarm Status [✓ Alarm History]

Level of Detail
Incidents [✓ Alarms] Events

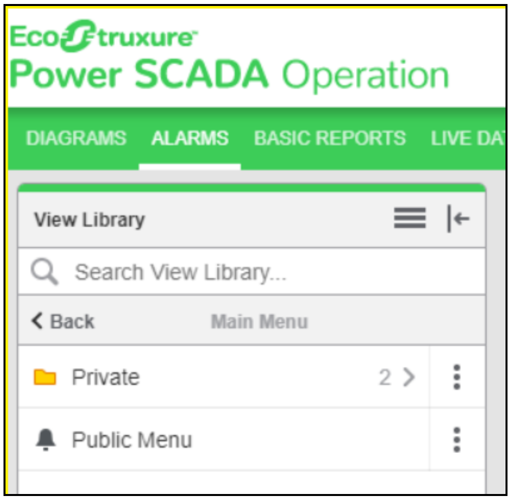
Priority
▨ 🟦 🟨 🟥

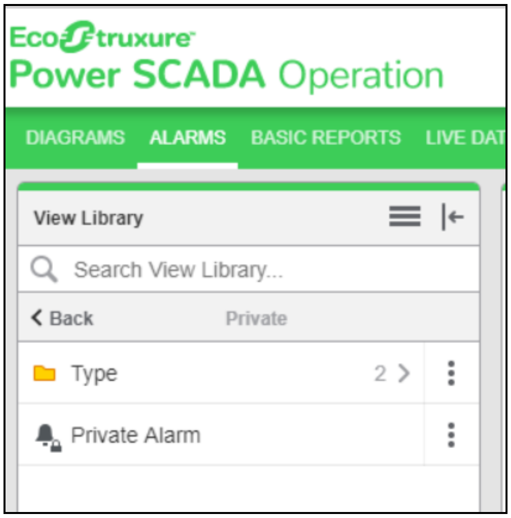
State
Active or Unacknowledged

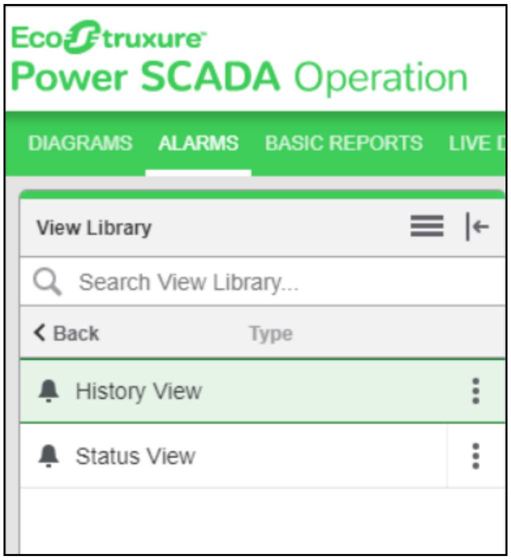
g. Click **Save**.

8. (Optional) Repeat steps from 4 through 6 to add more sub-folders or views inside the folder.

For reference, see the following images to add sub-folders or views:







Related topics:

- "Adding a new Alarms view" on page 370
- "Copying an Alarms view" on page 372
- "Editing an Alarms view" on page 372
- "Moving an Alarms view" on page 373
- "Deleting an Alarms view" on page 374
- "Setting a default Alarms view" on page 375
- Creating alarm menus
- "Exporting alarm menus" on page 382
- "Importing alarm menus" on page 384
- "Alarm Views" on page 575

For reference information see:

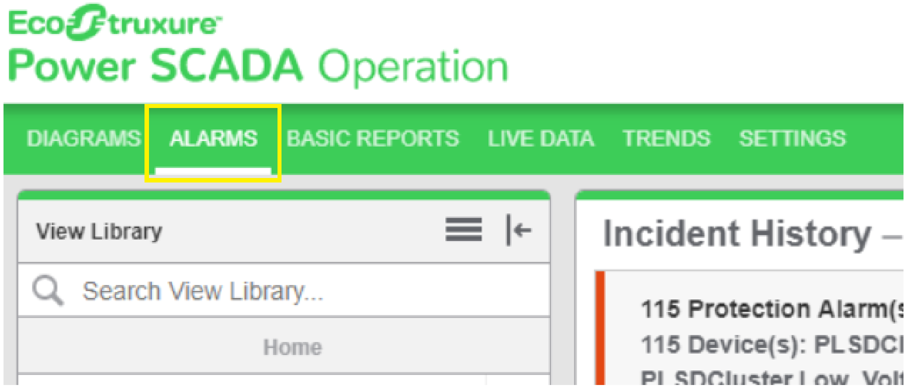
- "Alarms UI" on page 1263

For information on how to use Alarms, see "Alarms" on page 730.

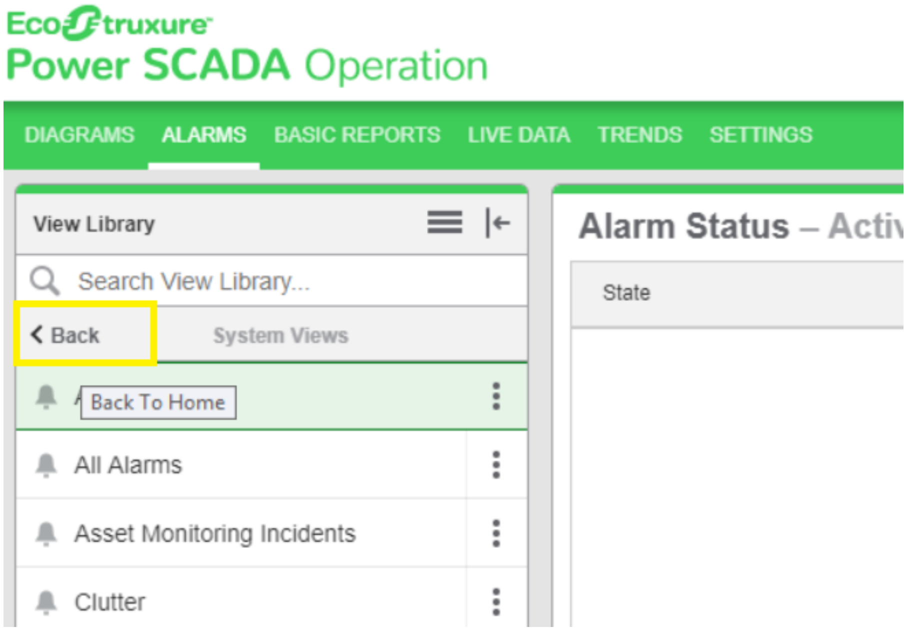
Exporting alarm menus

To export an alarm menu:

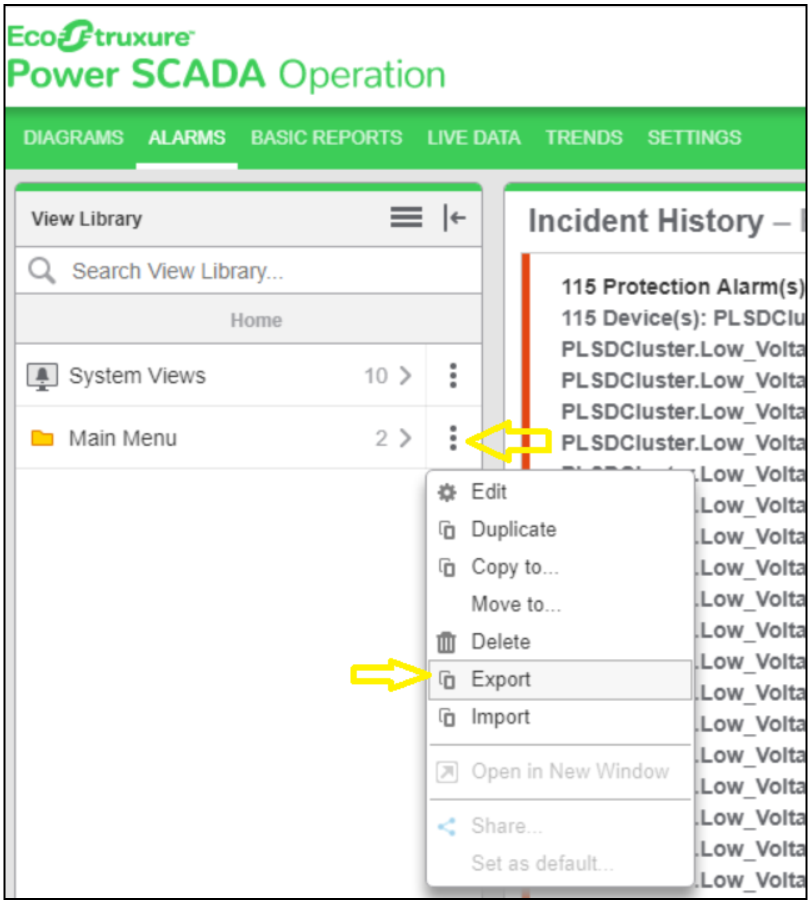
1. Log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)
2. Click **ALARMS**:



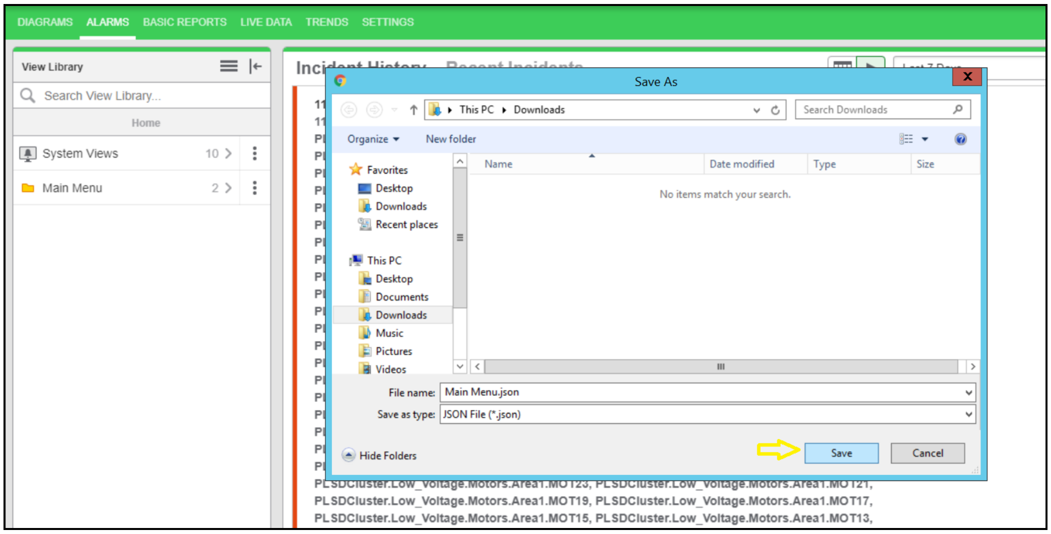
The following screen is displayed.



3. On the **Home** view, select the 3 dots, and then click **Export**:



4. Select the JSON file export location, and then click **Save**.



File name: Main Menu.json

Save as type: JSON File (*.json)

PLSDCluster.Low_Voltage.Motors.Area1.MOT19, PLSDCluster.Low_Voltage.Motors.Area1.MOT17,
PLSDCluster.Low_Voltage.Motors.Area1.MOT15, PLSDCluster.Low_Voltage.Motors.Area1.MOT13,

5. Check the file in the folder to confirm that the Alarm Menus were exported.

Related topics:

- "Adding a new Alarms view" on page 370
- "Copying an Alarms view" on page 372
- "Editing an Alarms view" on page 372
- "Moving an Alarms view" on page 373
- "Deleting an Alarms view" on page 374
- "Setting a default Alarms view" on page 375
- "Creating alarm menus" on page 755
- Exporting alarm menus
- "Importing alarm menus" on page 384
- "Alarm Views" on page 575

For reference information, see:

- "Alarms UI" on page 1263

For information on how to use Alarms, see "Alarms" on page 730.

Importing alarm menus

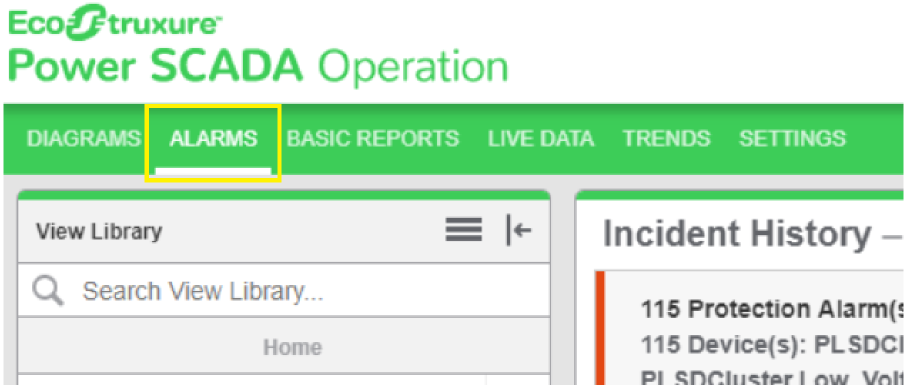
Prerequisites:

An alarms menu that was previously exported.

To import an alarm menu:

1. Log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)

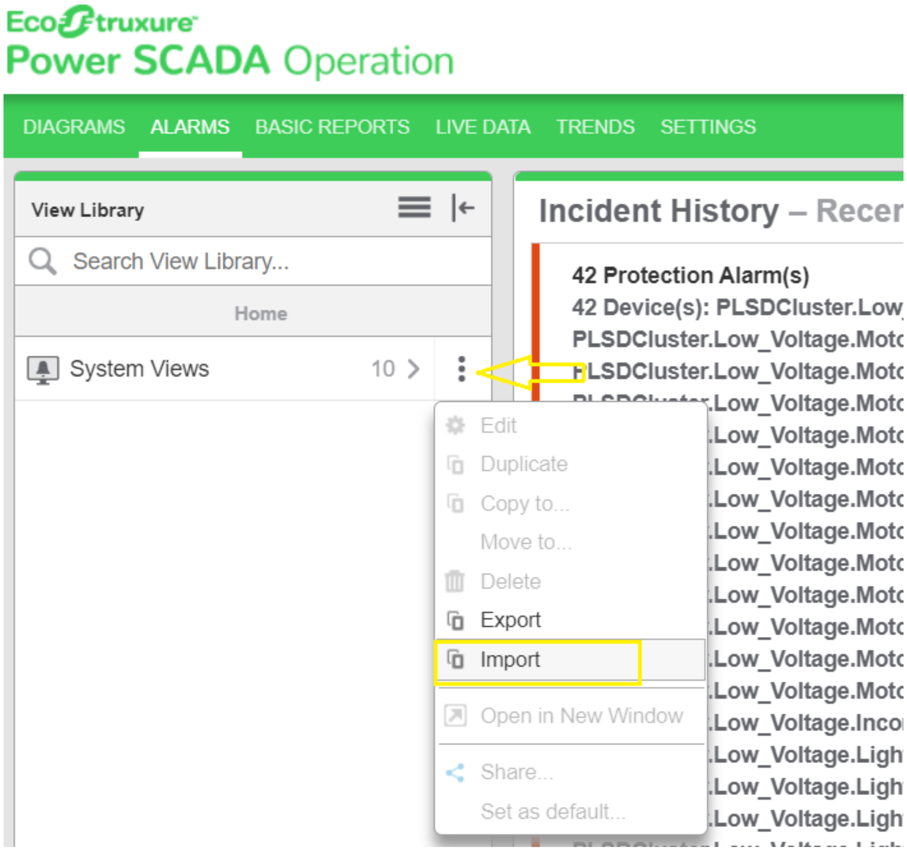
2. Click on **ALARMS** tab on header menu as per the image below.

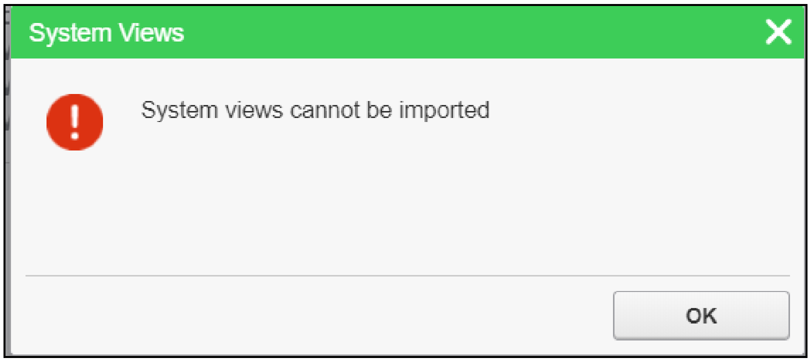


3. (Optional) Add a folder to which you want to import the exported Alarm Menus.

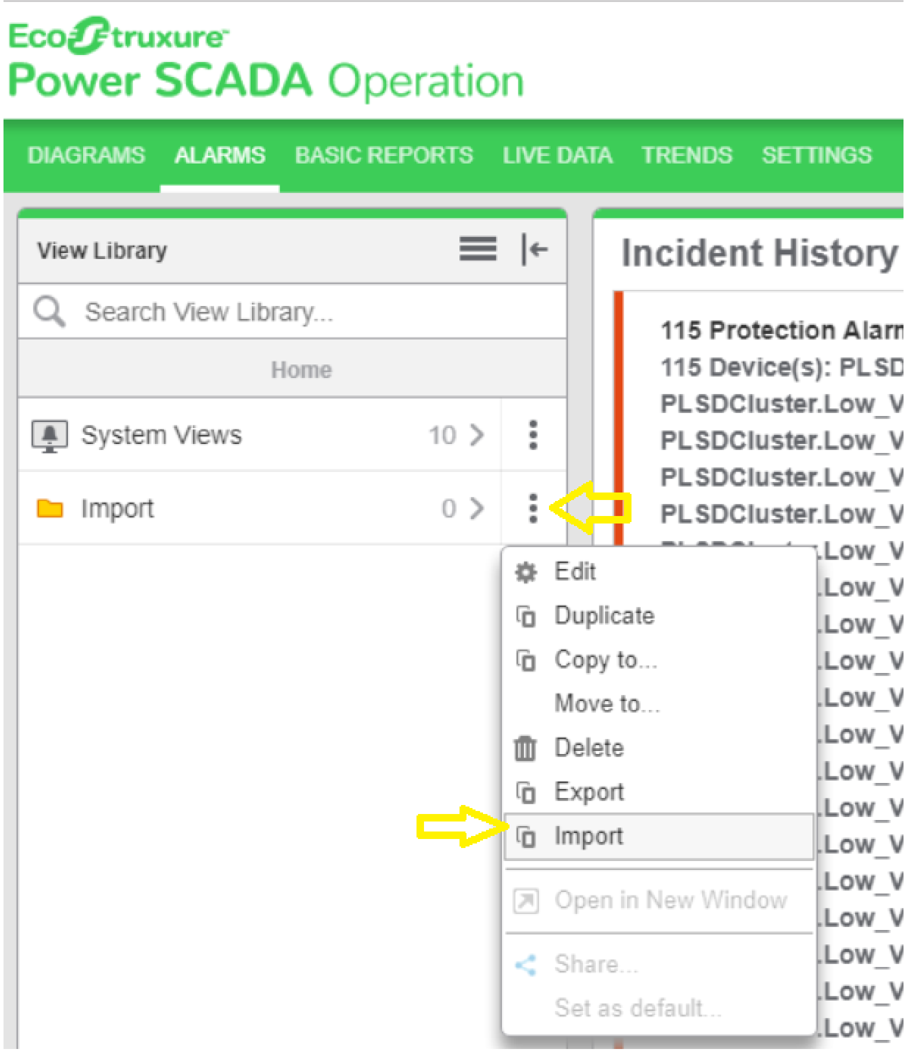
4. If an import folder does not already exist, create a folder to which you want to import the exported Alarm Menus.

   You cannot import a menu into **System View**.

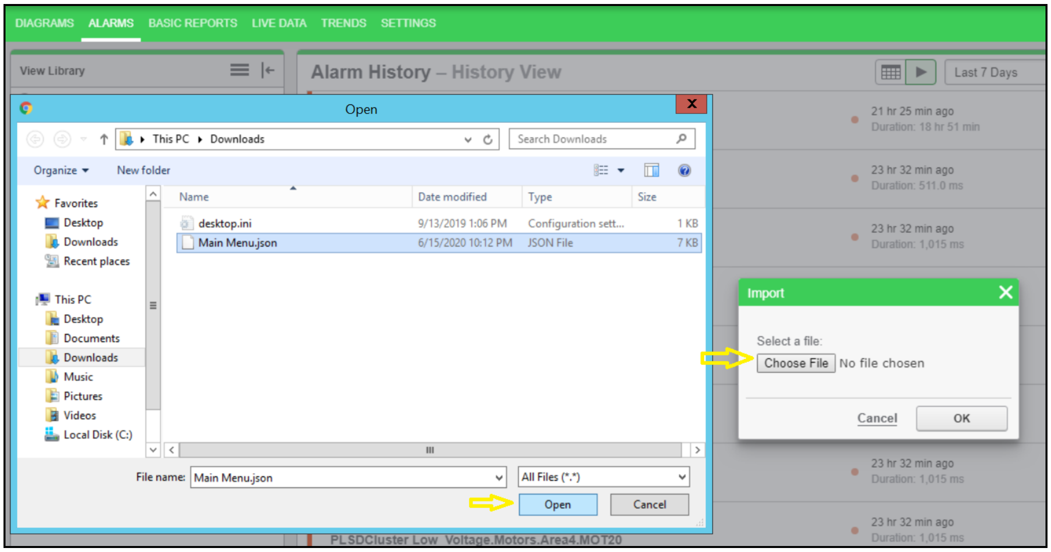




5. Select the 3 dots on the folder, and then click **Import**:



6. Click **Choose File**, navigate to the specific folder, and then select the file to be imported:

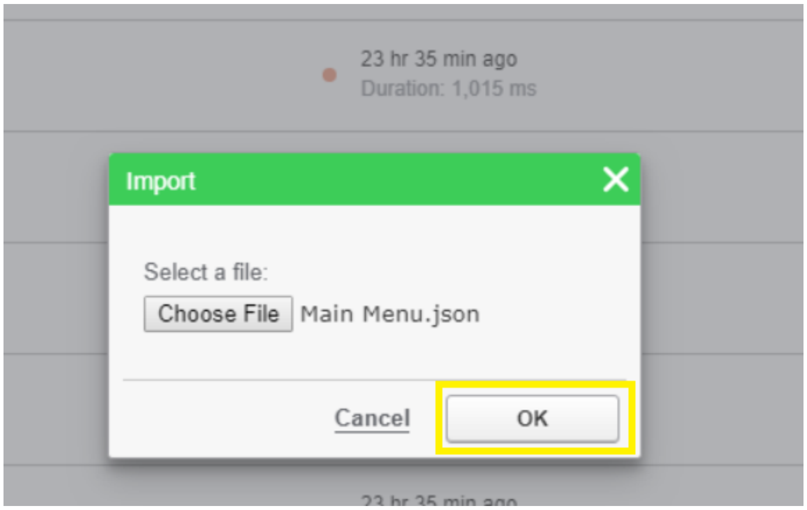


Name	Date modified	Type	Size
desktop.ini	9/13/2019 1:06 PM	Configuration sett...	1 KB
Main Menu.json	6/15/2020 10:12 PM	JSON File	7 KB

File name: Main Menu.json

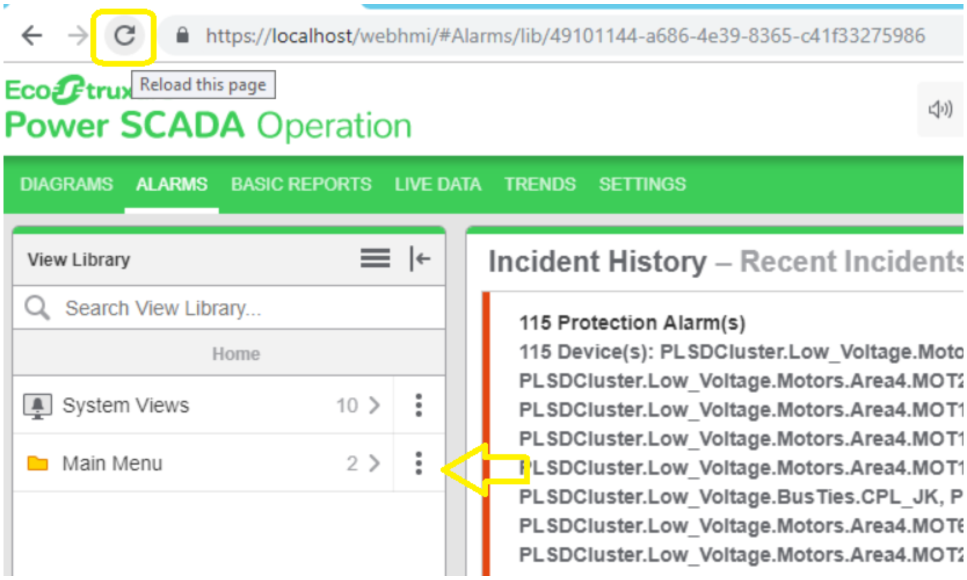
All Files (*.*)

7. Click **OK** to Import the **Alarm Menu**.



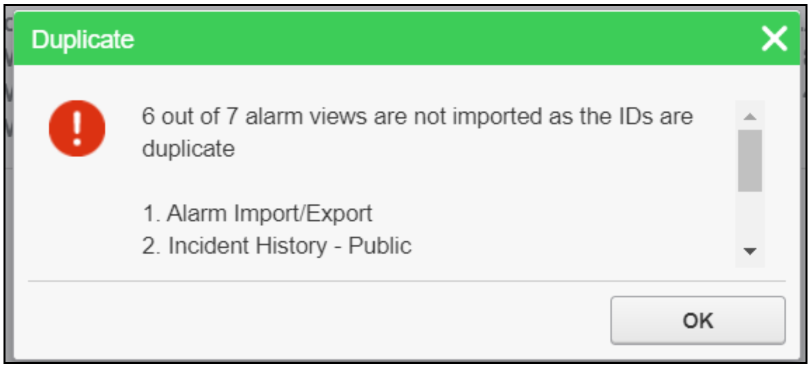
Select a file: Main Menu.json

8. Refresh your browser to see the imported alarm menus:



115 Protection Alarm(s)
115 Device(s): PLSDCluster.Low_Voltage.Moto
PLSDCluster.Low_Voltage.Motors.Area4.MOT2
PLSDCluster.Low_Voltage.Motors.Area4.MOT1
PLSDCluster.Low_Voltage.Motors.Area4.MOT1
PLSDCluster.Low_Voltage.Motors.Area4.MOT1
PLSDCluster.Low_Voltage.BusTies.CPL_JK, P
PLSDCluster.Low_Voltage.Motors.Area4.MOT6
PLSDCluster.Low_Voltage.Motors.Area4.MOT2

9. If the views found duplicate alarm views, then the following message appears indicating that duplicate views were not imported:



6 out of 7 alarm views are not imported as the IDs are duplicate

1. Alarm Import/Export
2. Incident History - Public

10. Check and confirm that all the menus and views were imported into **Library** file.

> **NOTE**: If a view that contains device details is imported into a system that does not contain that device, its details will not be displayed because that device is not mapped to an equipment.

Related topics:

- "Adding a new Alarms view" on page 370
- "Copying an Alarms view" on page 372
- "Editing an Alarms view" on page 372
- "Moving an Alarms view" on page 373
- "Deleting an Alarms view" on page 374
- "Setting a default Alarms view" on page 375
- "Creating alarm menus" on page 755

- "Exporting alarm menus" on page 382
- Importing alarm menus
- "Alarm Views" on page 575

For reference information, see:

- "Alarms UI" on page 1263

For information on how to use Alarms, see "Alarms" on page 730.

Diagrams configuration

Use Diagrams to view historical and real-time data in one-line and graphics diagrams.

Open Diagrams from the Diagrams link in the Web Applications banner.

For information on how to configure Diagrams, see:

- Graphics pages
- Advanced one-line diagrams
- Alarm Integration

Graphics pages

Use the Graphics Editor to create graphics pages for viewing in Diagrams.

For information on how to configure graphics pages, see:

- Adding a graphics page in the Graphics Editor
- Adding a custom component in the Graphics Editor
- Defining the Diagrams menu structure

For more information on the Graphics Editor, TAC Graphics Markup Language (TGML), or the legacy Graphics Builder, see the Graphics Appendices section.

NOTE: When backing up and restoring a project, ensure that "Include Subdirectories" is checked so that your graphics and advanced one-line configuration are included.

Graphics pages prerequisites

Before you create a graphics page make sure you:

- Create a project in the Profile Editor
- Add a project with the same name to Power SCADA Operation; add at least one cluster, network address, and server
- Ensure the project is set to Active Project
- Export the project from the Profile Editor
- Use the I/O Device Manager to add devices to the project
- Compile the project

Adding graphics pages

To add a new graphics page in the Graphics Editor:

1. In the Power SCADA Operation folder, open the Graphics Editor.
2. On the Graphics Editor screen, click **File** > **New** > **Graphic**.

   NOTE: For details on adding a new or existing Component, see the Adding a custom component in the Graphics Editor section.

3. From the left pane, drag and drop equipment and components to create the graphics page.
4. To automatically connect equipment or components, select **Auto Connect** on the toolbar. The button will turn green when selected. You can turn off Auto Connect if you want to place objects close together without connecting them.

   NOTE: If you do not use Auto Connect for connecting equipment or components, you must manually enter the BusName properties.

5. There are 2 ways to connect equipment or components with Auto Connect: drag and drop, and click and select. When you drag or click an object to connect it, the connector(s) will turn green. To connect:
   a. Drag the equipment over the component connector to which you want to connect, and when the connector turns green, drop the equipment.
   b. Click the equipment connector so it turns green, then click the component connector to which you want to connect it. The connector selected second will snap to the connector selected first.

      In the Properties pane, under the Custom section, the connector component's corresponding BusName property will auto-populate with the busbar that is automatically created.

      NOTE: Property busbar names are dependent on the component type.

6. To disconnect equipment and components, hover over the connector until it turns yellow, then right-click and select **Disconnect**. The component will move away from the busbar to show that it is disconnected.

   NOTE: You must disconnect each component individually. When the last connection to a busbar is disconnected, the busbar will automatically be removed from the page.

7. To animate a component in the Graphics Viewer, set up the component-specific Custom properties in the Properties pane.
8. To edit the Graphics Viewer menus, see the Defining the Graphics Viewer menu structure section.
9. Click **Save**.

   NOTE: You can only Open or Save a file in the project's TGML folder.

10. If a custom property is incorrectly configured on any graphics page in the project, after clicking **Save**, a red link displaying the number of binding issues will replace the green 0 Binding Issue(s) link in the toolbar, and the Connection Debugger window will appear. In the Connection Debugger window, you can view any binding issue details by component Type or Page. Scroll to correct any fields with missing information.
11. To re-open the Connection Debugger window, click the binding issues link in the toolbar.

For more information on TGML, components and snippets, binds and links, and other Graphics Editor functionality, see the Graphics Appendices section.

Adding custom components

You can add a custom component in two ways: duplicating an existing component, or creating a new component.

To add a custom component by duplicating an existing component in the Graphics Editor:

1. In the Power SCADA Operation folder, open the Graphics Editor.
2. On the Graphics Editor screen, in the Components pane, find a standard component that is the same type as the custom component you want to create:
   a. ATS
   b. Breaker
   c. Meter
   d. Motor
   e. Source
   f. Switch
   g. Transformer
3. Right-click the component, select **Duplicate**, and enter a unique name for the new custom component.
4. Open the My Components category, right-click on the new custom component, and then select **Edit**.

   A new instance of the Graphics Editor opens with the new custom component selected.
5. In the Objects pane, open the Group element and delete any figures (Line, Ellipse, etc.) that you do not want in the custom component. To maintain the existing advanced one-line animations, keep the Script and all Binds.

   NOTE: You must keep the following:
   - Rectangle Background
   - Conditions
   - The component ID prefix for the type you selected (Example: **Breaker.**MyCustomBreaker)
   - All existing custom properties

6. Add or modify figures and connector points for the custom component.

> **NOTE**: For more information, see the Graphics Editor Figures Overview section.

To add a new custom component, see the following sections:

- Creating a New Component
- Creating a one-line on a graphics page

For more information on TGML, components and snippets, binds and links, and other Graphics Editor functionality, see the Graphics Appendices section.

Changing the background color of a graphics page

To change the background color of an individual graphics page in your project:

1. In the Objects pane, select the **Tgml** node.
2. In the Properties pane under Appearance, set the Background property.

Changing the background color of a component

You can change a component background color at the component level or by an individual instance of a component.

To change the component background color at the component level:

1. Right-click the component that you want to edit, and then select **Edit**.

   > **NOTE**: You cannot edit standard components; however, you can edit a standard component which has been duplicated. See Adding a custom component in the Graphics Editor for details.

2. Create a background rectangle that is the same size as the component, then right-click and select **Arrange** > **Send to Back** so it is one of the first objects in the Objects pane.
3. In the Properties pane under Appearance, edit the Fill property. You can select **None** to make the background transparent or set a Custom Color.
4. Set the Stroke property to **None**.

To change an individual instance of a component:

1. Select the component, then in the Objects pane, select **Rectangle Background**.
2. In the Properties pane under Appearance, edit the Fill property. You can select **None** to make the background transparent or set a Custom Color.

Defining the Diagrams menu structure

To define the Diagrams viewer menu structure:

1. In Windows Explorer, navigate to the project TGML directory in the project folder: `C:\ProgramData\Schneider Electric\Power SCADA Operation\v2020`

`R2\User\[Project Name]\TGML`

NOTE: By default, new TGML files are saved in the TGML folder from the active project TGML path.

2. Create folders and sub-folders with your desired menu item names and structure, then place the TGML files in the respective folders.

   The TGML folder and sub-folder names will become menu items appearing in alphabetical order. The TGML files in each folder will also be listed alphabetically as menu items within the parent menu item (folder).

   NOTE: Any TGML files in a folder will appear first, before any sub-folder menu items. The `AolConfig.json` file in the TGML folder is a special system file and should not be moved. This file will not affect the menu configuration.

   NOTE: If a TGML graphic file is saved starting with an ! (exclamation point), it will not display in the Diagrams viewer. Use this naming format for pop-ups, templates, and other TGML files to which you do not want the user to navigate. For more information, see the Operating chapter Diagrams Overview.

Interactive TGML graphics

You can create TGML graphics that can accept user input and read values from the PSO.

⚠ **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, or equipment damage.

This section includes examples that configure the following TGML graphic components to create user interactive TGML graphics:

Topic	Component description
"TGML Sample to User Interactive Write" on page 395	Write perform write operations, such as open or close a breaker.
"Conditional Write" on page 405	Operate a breaker or digital output based on a condition

Topic	Component description
"On Demand Read" on page 411	Read the values from PSO without binding the components in the TGML file
"Single or Multiple DataPoint Write" on page 415	Perform generic write operations for a single or block write
"Write and Confirm" on page 422	Perform breaker operations, On/Off digital outputs, and set CT and PT ratios
"Write and Confirm User Interactive" on page 429	Perform breaker operations, On/Off digital outputs, set CT and PT ratios, and alarm set-points. This component prompts the user for an input (DataPoint) value, and then checks and confirms whether the value is written properly in the respective register
"User Input Write Operation" on page 436	Write set-point register values by prompting the user to input the value
"Analog Write Operation" on page 445	Update or configure set-point registers for alarm, set up temperature, scaling values, and CT PT ratios.

Turning off credential requirements for control components

You can turn off credential requirements for individual control components, if required. For instance, providing credential confirmation may not be needed in cases such as changing alarm settings, temperature control, modifying fan speeds, etc.

Prerequisites:

Control components with properties. See Read and Write Alarm Properties for detailed information on how to write properties to control components.

⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

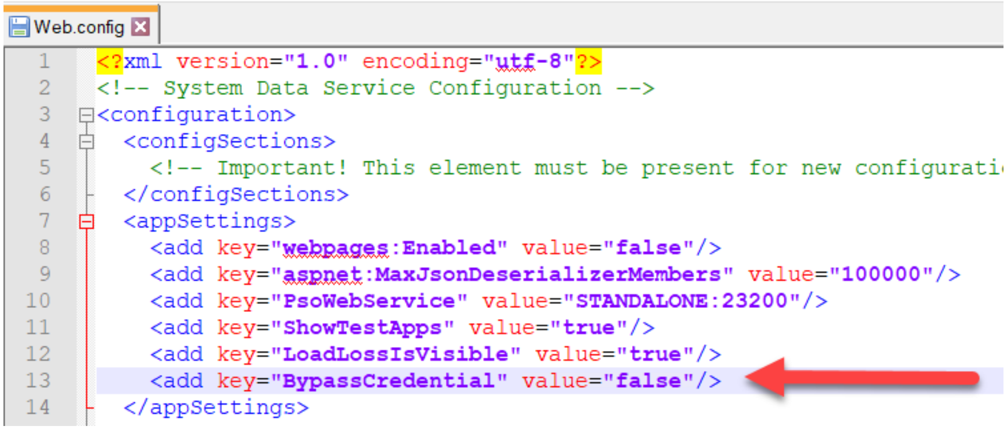
Do not turn off credential requirements for critical control or protection applications where human or equipment safety relies on credential confirmation.

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

To turn off credential requirements for individual control components:

1. Open **Control Panel** > **Administrative Tools** > **Internet Information Services (IIS) Manager**.
2. In the Connections pane, under Sites, expand Default Web Site, and right-click **PsoDataService**.
3. From the drop-down menu, select **Explore**.
4. From the Windows Explorer location, open Web.config in a text editor.
5. In the `configuration > appSettings` section, edit the value for BypassCredential to `true`. If the Web.config file does not have this property, you can enter it manually. In the

`configuration > appSettings` section, enter `<add key="BypassCredential" value="true"/>`.



```
<?xml version="1.0" encoding="utf-8"?>
<!-- System Data Service Configuration -->
<configuration>
  <configSections>
    <!-- Important! This element must be present for new configurati
  </configSections>
  <appSettings>
    <add key="webpages:Enabled" value="false"/>
    <add key="aspnet:MaxJsonDeserializerMembers" value="100000"/>
    <add key="PsoWebService" value="STANDALONE:23200"/>
    <add key="ShowTestApps" value="true"/>
    <add key="LoadLossIsVisible" value="true"/>
    <add key="BypassCredential" value="false"/>
  </appSettings>
```

6. Save the file and restart IIS.
7. In Graphics Editor, select your desired control component.
8. In the Properties pane, in the field, IsCredentialConfirmationRequired, edit the value to **False**.

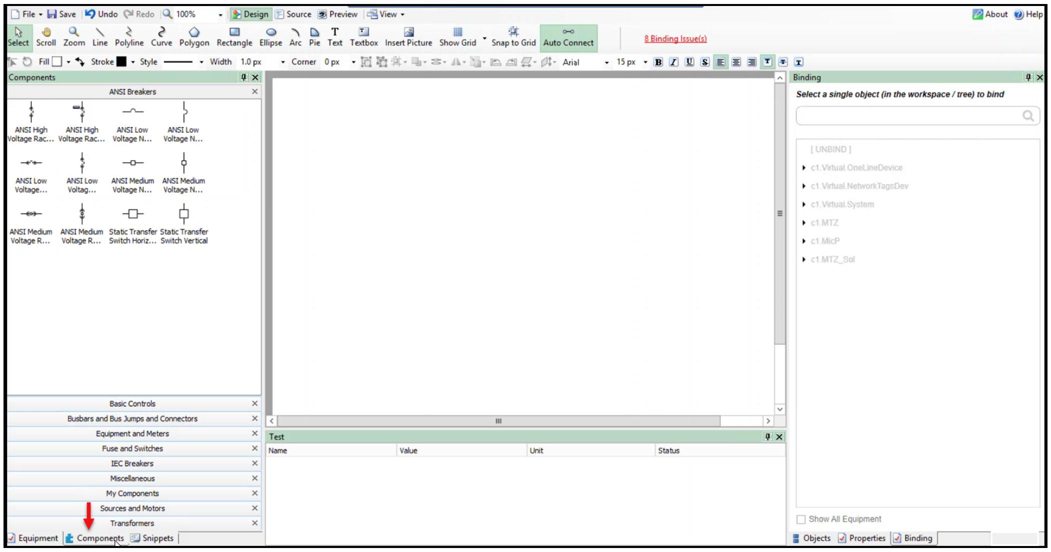
In WebHMI, on the DIAGRAMS tab, you can observe your control component will no longer require credentials for control operations.

TGML Sample to User Interactive Write

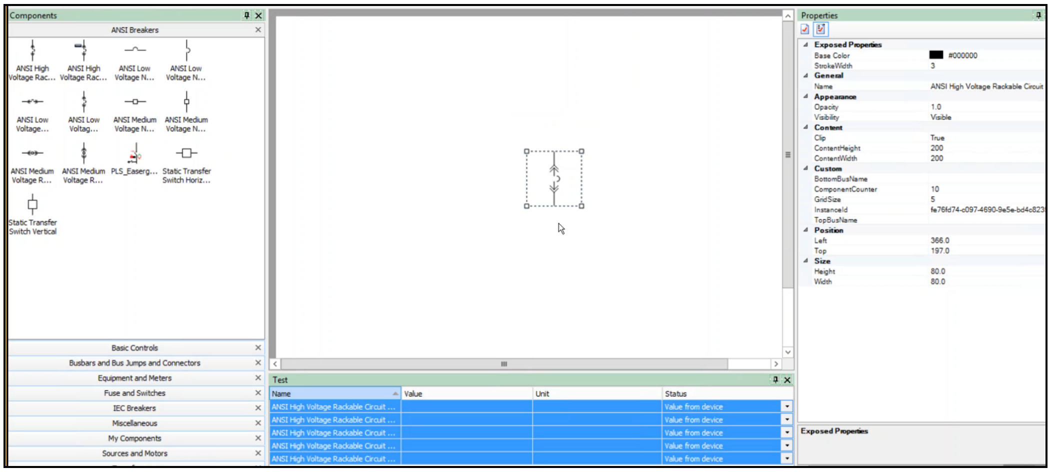
You can use TGML graphics to perform write operations. This topic uses an example to illustrate how to do this.

To create the TGML graphics page sample with a write operation:

1. Open the Graphics Editor: Go to **Start > Power SCADA Operation > Graphics Editor**.
2. At the bottom left, click **Components**.

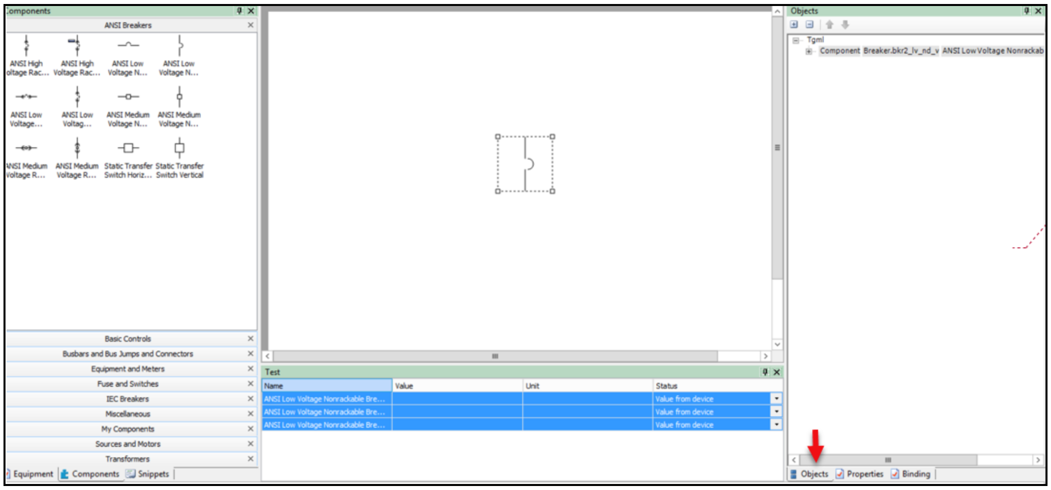


3. Drag and drop the **Breaker** components to the workspace based on your requirement.



Exposed Properties
Base Color #000000
StrokeWidth 3
General
Name ANSI High Voltage Rackable Circuit
Appearance
Opacity 1.0
Visibility Visible
Content
Clip True
ContentHeight 200
ContentWidth 200
Custom
BottomBusName
ComponentCounter 10
GridSize 5
InstanceId fe76fd74-c097-4690-9e5e-bd4c823
TopBusName
Position
Left 366.0
Top 197.0
Size
Height 80.0
Width 80.0

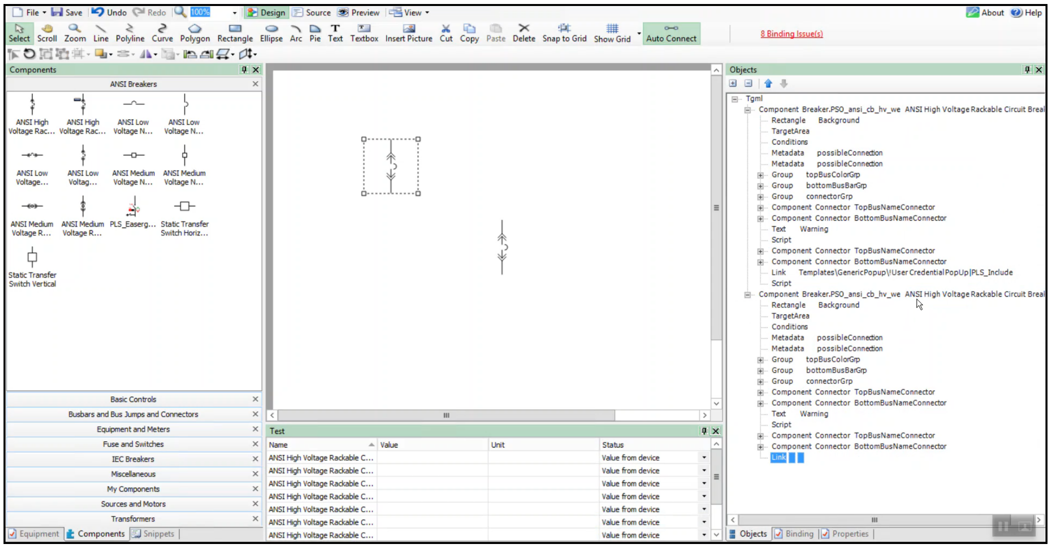
4. At the bottom right corner, click **Objects**.



5. Expand the **Tgml** node.

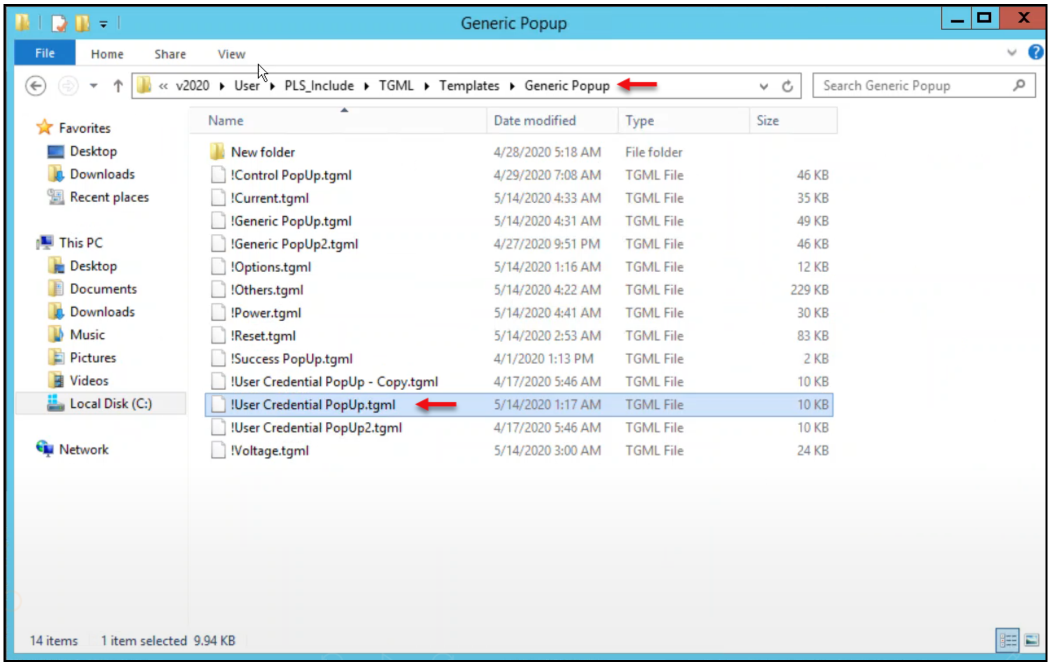
6. Right-click the **Component**, and then click **New > Link**.

The following screen is displayed:



```
Tgml
  Component Breaker.PSO_ansi_cb_hv_we ANSI High Voltage Rackable Circuit Breal
    Rectangle Background
    TargetArea
    Conditions
    Metadata possibleConnection
    Metadata possibleConnection
    Group topBusColorGrp
    Group bottomBusBarGrp
    Group connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Text Warning
    Script
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Link Templates\GenericPopup\!User CredentialPopUp|PLS_Include
    Script
  Component Breaker.PSO_ansi_cb_hv_we ANSI High Voltage Rackable Circuit Breal
    Rectangle Background
    TargetArea
    Conditions
    Metadata possibleConnection
    Metadata possibleConnection
    Group topBusColorGrp
    Group bottomBusBarGrp
    Group connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Text Warning
    Script
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Link
```

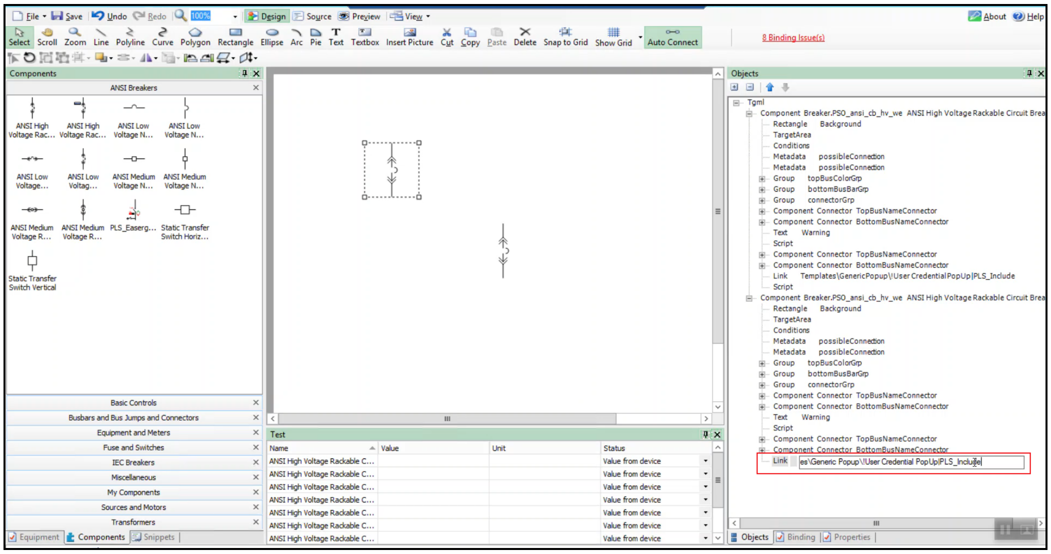
7. In Windows Explorer, navigate to the TGML Template gallery and then copy the file location of User Credential PopUp:



Name	Date modified	Type	Size
New folder	4/28/2020 5:18 AM	File folder	
!Control PopUp.tgml	4/29/2020 7:08 AM	TGML File	46 KB
!Current.tgml	5/14/2020 4:33 AM	TGML File	35 KB
!Generic PopUp.tgml	5/14/2020 4:31 AM	TGML File	49 KB
!Generic PopUp2.tgml	4/27/2020 9:51 PM	TGML File	46 KB
!Options.tgml	5/14/2020 1:16 AM	TGML File	12 KB
!Others.tgml	5/14/2020 4:22 AM	TGML File	229 KB
!Power.tgml	5/14/2020 4:41 AM	TGML File	30 KB
!Reset.tgml	5/14/2020 2:53 AM	TGML File	83 KB
!Success PopUp.tgml	4/1/2020 1:13 PM	TGML File	2 KB
!User Credential PopUp - Copy.tgml	4/17/2020 5:46 AM	TGML File	10 KB
!User Credential PopUp.tgml	5/14/2020 1:17 AM	TGML File	10 KB
!User Credential PopUp2.tgml	4/17/2020 5:46 AM	TGML File	10 KB
!Voltage.tgml	5/14/2020 3:00 AM	TGML File	24 KB

14 items 1 item selected 9.94 KB

8. Enter the file location in the Link as **Templates\Generic Popup\!User Credential Pop Up|PLS_Include**.

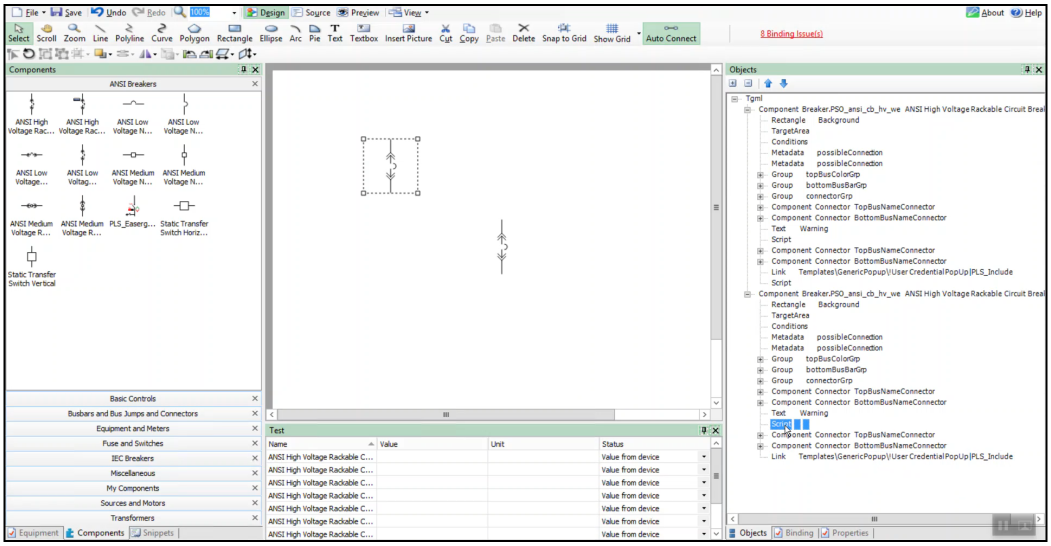


```
Tgml
  Component Breaker.PSO_ansi_cb_hv_we  ANSI High Voltage Rackable Circuit Brea
    Rectangle   Background
    TargetArea
    Conditions
    Metadata   possibleConnection
    Metadata   possibleConnection
    Group   topBusColorGrp
    Group   bottomBusBarGrp
    Group   connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Text   Warning
    Script
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Link   Templates\GenericPopup\!User CredentialPopUp|PLS_Include
    Script
  Component Breaker.PSO_ansi_cb_hv_we  ANSI High Voltage Rackable Circuit Brea
    Rectangle   Background
    TargetArea
    Conditions
    Metadata   possibleConnection
    Metadata   possibleConnection
    Group   topBusColorGrp
    Group   bottomBusBarGrp
    Group   connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Text   Warning
    Script
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Link   es\Generic Popup\!User Credential PopUp|PLS_Include
```

Test

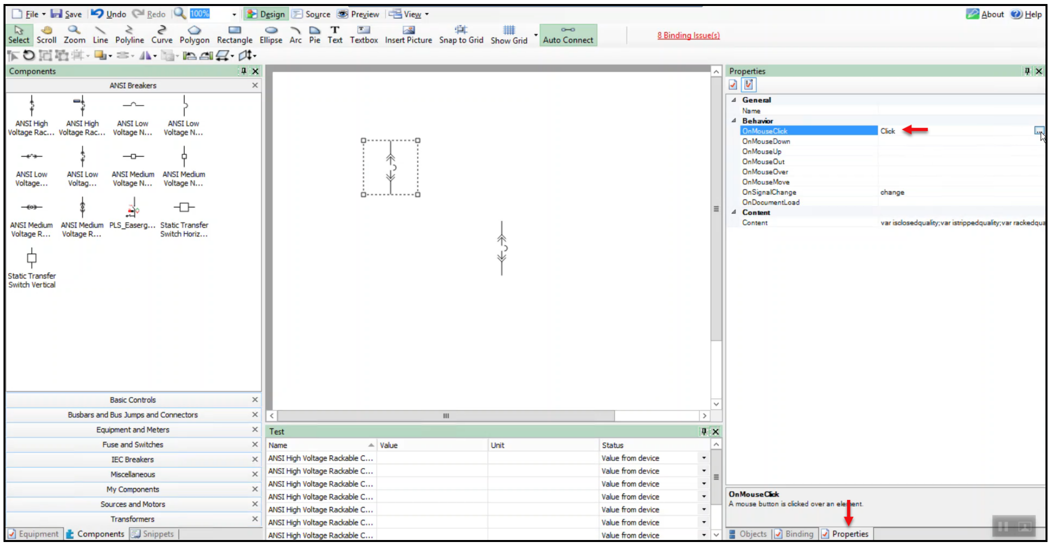
Name	Value	Unit	Status
ANSI High Voltage Rackable C...			Value from device
ANSI High Voltage Rackable C...			Value from device
ANSI High Voltage Rackable C...			Value from device
ANSI High Voltage Rackable C...			Value from device
ANSI High Voltage Rackable C...			Value from device
ANSI High Voltage Rackable C...			Value from device
ANSI High Voltage Rackable C...			Value from device

9. Click **Script**.

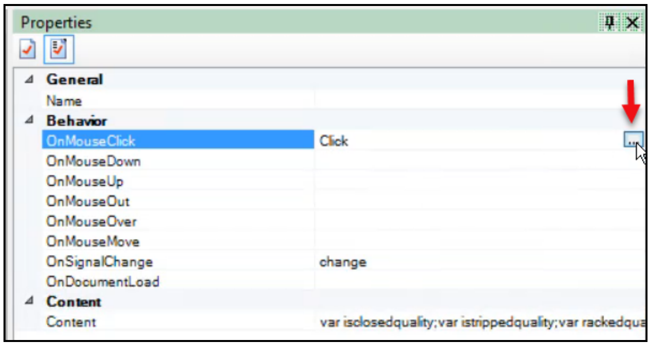


10. In the right bottom, click **Properties**.

11. In **Behavior > OnMouseClick**, enter **Click**.



12. Click the ellipsis button next to the Click value.


Properties

General
Name

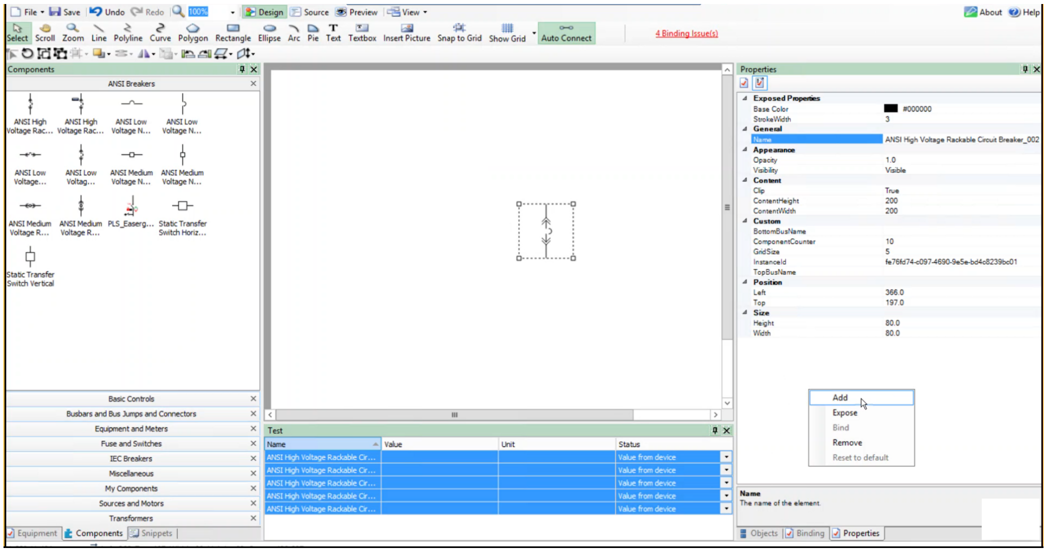
Behavior
OnMouseClick Click
OnMouseDown
OnMouseUp
OnMouseOut
OnMouseOver
OnMouseMove
OnSignalChange change
OnDocumentLoad

Content
Content var isclosedquality;var istrippedquality;var rackedqua

13. In the code editor, enter the following code, and then click **OK**.

```
functionclick(evt)
{
    var DataPointVal = prompt("Please enter the
operation","");sessionStorage.setItem("DataPointVal",DataPointVal);
    var componentName = evt.getCurrentTarget().getAttribute("Name");
    var connector = evt.getCurrentTarget().getElementsByTagName("Link");
    var instanceId = evt.getCurrentTarget().getAttribute("InstanceId");
    var dataPoint = componentName+"."+evt.getCurrentTarget().getAttribute
("DataPoint");
    var title = componentName;
    var customExpose = evt.getCurrentTarget().getAttribute("Sub-
stituteNames");
    //Height & width can be configured by the user
    var width = 500;
    var height = 385;
    var show TitleBar = "Yes";
    var showUnamePwd = "Yes";
    var usercredbottom = 30;
    var usercredleft = 15;
    var usercredbackcolor = "white";
    var usercredwidth = 70;
    var usercredheight = 33;

    var unamepwdwidth = 95;
    var unamepwdcolor = "#9FA0A4";

    for(var i=0;i<connector.length;i++){
        var connectorName=connector.item (i).getAttribute("Name");
        invoke(connectorName,"Type=PopUp | ComponentName =" + componentName
+ " | InstanceID =" + instanceId + " | DataPoint = "+dataPoint +"| Title=" +
title + " | Width=" + width + " | Height=" + height + " | ShowTitleBar =" +
showTitleBar + " | ShowUnamePwd = " + showUnamePwd +" | UserCredBottom =
"+usercredbottom +" | UserCredLeft = "+ usercredleft+" | UserCredWidth =
"+usercredwidth +" | UserCredHeight = "+usercredheight +" |
UserCredBackColor = "+usercredbackcolor+" | UnamePwdWidth =
"+unamepwdwidth+" | UnamePwdColor = "+unamepwdcolor);
    }
}
```

14. In the bottom right, click **Properties**, right-click and then click **Add**.

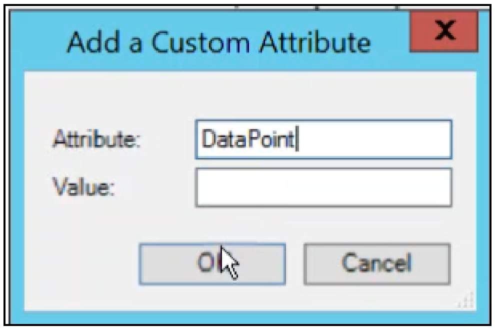


Name	Value	Unit	Status
ANSI High Voltage Rackable Cir...			Value from device
ANSI High Voltage Rackable Cir...			Value from device
ANSI High Voltage Rackable Cir...			Value from device
ANSI High Voltage Rackable Cir...			Value from device
ANSI High Voltage Rackable Cir...			Value from device

Exposed Properties	
Base Color	#000000
StrokeWidth	3
General	
Name	ANSI High Voltage Rackable Circuit Breaker_002
Appearance	
Opacity	1.0
Visibility	Visible
Content	
Clip	True
ContentHeight	200
ContentWidth	200
Custom	
BottomBusName	
ComponentCounter	10
GridSize	5
InstanceId	fe76fd74-c097-4690-9e5e-bd4c8239bc01
TopBusName	
Position	
Left	366.0
Top	197.0
Size	
Height	80.0
Width	80.0

Name
The name of the element.

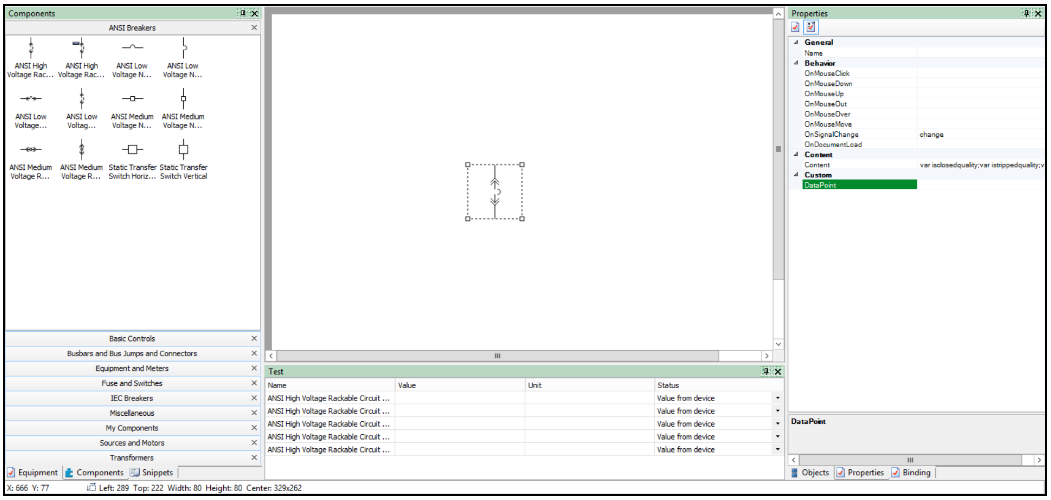
15. For **Attribute**, enter `DataPoint`, and then click **OK**. (Leave **Value** blank.)



Attribute: DataPoint

Value:

The following screen is displayed:



X: 666 Y: 77 Left: 289 Top: 222 Width: 80 Height: 80 Center: 329x262

Name	Value	Unit	Status
ANSI High Voltage Rackable Circuit …			Value from device
ANSI High Voltage Rackable Circuit …			Value from device
ANSI High Voltage Rackable Circuit …			Value from device
ANSI High Voltage Rackable Circuit …			Value from device
ANSI High Voltage Rackable Circuit …			Value from device

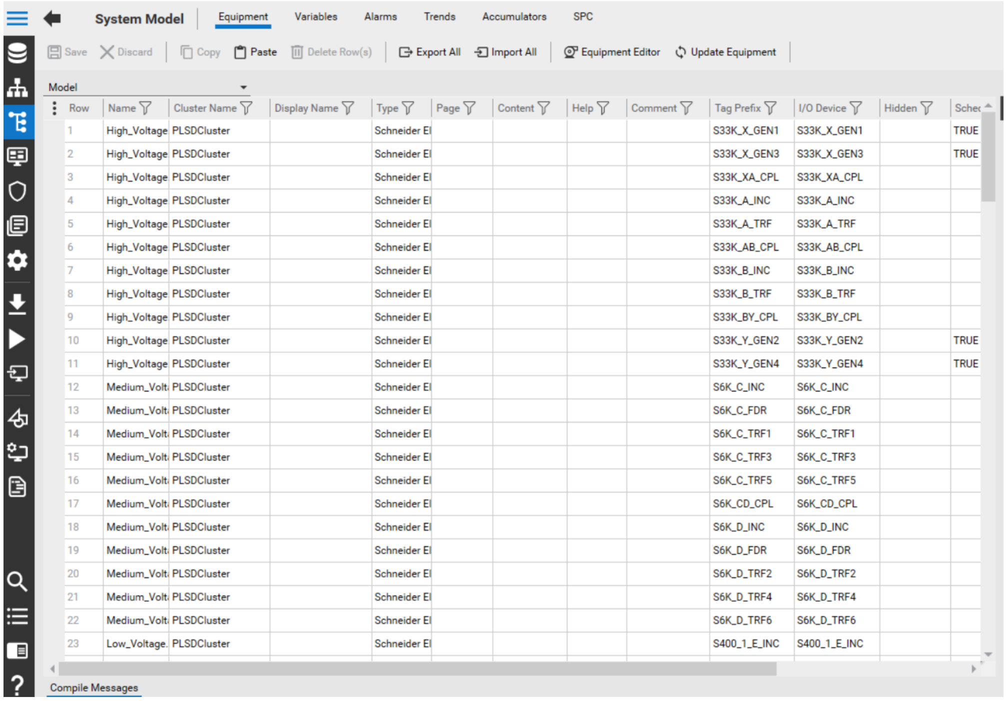
General	
Name	
Behavior	
OnMouseClick	
OnMouseDown	
OnMouseUp	
OnMouseOut	
OnMouseOver	
OnMouseMove	
OnSignalChange	change
OnDocumentLoad	
Content	
Content	var isclosedquality;var istrippedquality;v
Custom	
DataPoint	

DataPoint

16. Get the DataPoint value:

    a. Open Power SCADA Studio.

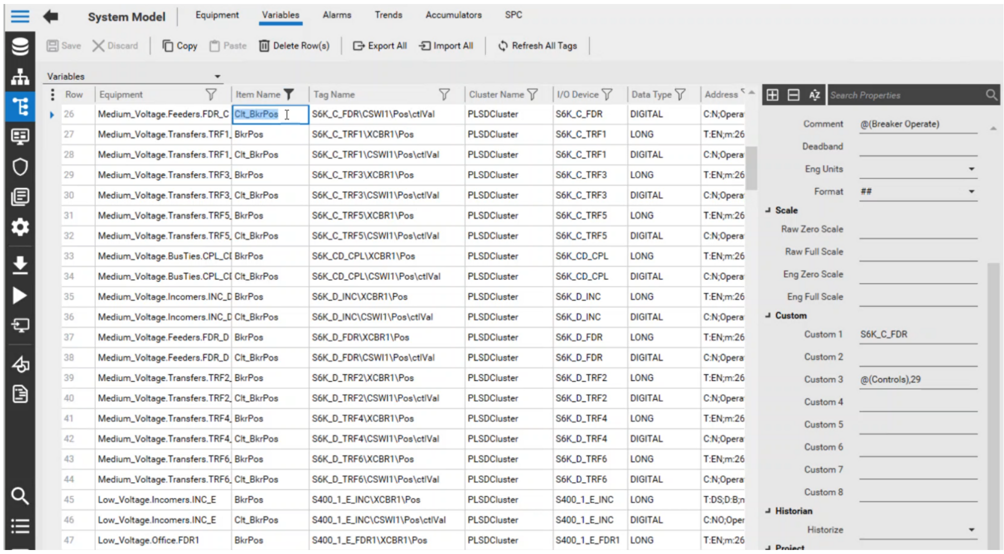
    b. From the left icon pane, click **System Model**.



Model

Row	Name	Cluster Name	Display Name	Type	Page	Content	Help	Comment	Tag Prefix	I/O Device	Hidden	Schec
1	High_Voltage	PLSDCluster		Schneider El					S33K_X_GEN1	S33K_X_GEN1		TRUE
2	High_Voltage	PLSDCluster		Schneider El					S33K_X_GEN3	S33K_X_GEN3		TRUE
3	High_Voltage	PLSDCluster		Schneider El					S33K_XA_CPL	S33K_XA_CPL		
4	High_Voltage	PLSDCluster		Schneider El					S33K_A_INC	S33K_A_INC		
5	High_Voltage	PLSDCluster		Schneider El					S33K_A_TRF	S33K_A_TRF		
6	High_Voltage	PLSDCluster		Schneider El					S33K_AB_CPL	S33K_AB_CPL		
7	High_Voltage	PLSDCluster		Schneider El					S33K_B_INC	S33K_B_INC		
8	High_Voltage	PLSDCluster		Schneider El					S33K_B_TRF	S33K_B_TRF		
9	High_Voltage	PLSDCluster		Schneider El					S33K_BY_CPL	S33K_BY_CPL		
10	High_Voltage	PLSDCluster		Schneider El					S33K_Y_GEN2	S33K_Y_GEN2		TRUE
11	High_Voltage	PLSDCluster		Schneider El					S33K_Y_GEN4	S33K_Y_GEN4		TRUE
12	Medium_Volt	PLSDCluster		Schneider El					S6K_C_INC	S6K_C_INC		
13	Medium_Volt	PLSDCluster		Schneider El					S6K_C_FDR	S6K_C_FDR		
14	Medium_Volt	PLSDCluster		Schneider El					S6K_C_TRF1	S6K_C_TRF1		
15	Medium_Volt	PLSDCluster		Schneider El					S6K_C_TRF3	S6K_C_TRF3		
16	Medium_Volt	PLSDCluster		Schneider El					S6K_C_TRF5	S6K_C_TRF5		
17	Medium_Volt	PLSDCluster		Schneider El					S6K_CD_CPL	S6K_CD_CPL		
18	Medium_Volt	PLSDCluster		Schneider El					S6K_D_INC	S6K_D_INC		
19	Medium_Volt	PLSDCluster		Schneider El					S6K_D_FDR	S6K_D_FDR		
20	Medium_Volt	PLSDCluster		Schneider El					S6K_D_TRF2	S6K_D_TRF2		
21	Medium_Volt	PLSDCluster		Schneider El					S6K_D_TRF4	S6K_D_TRF4		
22	Medium_Volt	PLSDCluster		Schneider El					S6K_D_TRF6	S6K_D_TRF6		
23	Low_Voltage.	PLSDCluster		Schneider El					S400_1_E_INC	S400_1_E_INC		

c. Click **Variables**, and then copy the **Item Name** of the breaker on which you want to add the write operation.



Row	Equipment	Item Name	Tag Name	Cluster Name	I/O Device	Data Type	Address
26	Medium_Voltage.Feeders.FDR_C	Clt_BkrPos	S6K_C_FDR\CSWI1\Pos\ctlVal	PLSDCluster	S6K_C_FDR	DIGITAL	C:N;Opera
27	Medium_Voltage.Transfers.TRF1_	BkrPos	S6K_C_TRF1\XCBR1\Pos	PLSDCluster	S6K_C_TRF1	LONG	T:EN;m:26
28	Medium_Voltage.Transfers.TRF1_	Clt_BkrPos	S6K_C_TRF1\CSWI1\Pos\ctlVal	PLSDCluster	S6K_C_TRF1	DIGITAL	C:N;Opera
29	Medium_Voltage.Transfers.TRF3_	BkrPos	S6K_C_TRF3\XCBR1\Pos	PLSDCluster	S6K_C_TRF3	LONG	T:EN;m:26
30	Medium_Voltage.Transfers.TRF3_	Clt_BkrPos	S6K_C_TRF3\CSWI1\Pos\ctlVal	PLSDCluster	S6K_C_TRF3	DIGITAL	C:N;Opera
31	Medium_Voltage.Transfers.TRF5_	BkrPos	S6K_C_TRF5\XCBR1\Pos	PLSDCluster	S6K_C_TRF5	LONG	T:EN;m:26
32	Medium_Voltage.Transfers.TRF5_	Clt_BkrPos	S6K_C_TRF5\CSWI1\Pos\ctlVal	PLSDCluster	S6K_C_TRF5	DIGITAL	C:N;Opera
33	Medium_Voltage.BusTies.CPL_CI	BkrPos	S6K_CD_CPL\XCBR1\Pos	PLSDCluster	S6K_CD_CPL	LONG	T:EN;m:26
34	Medium_Voltage.BusTies.CPL_CI	Clt_BkrPos	S6K_CD_CPL\CSWI1\Pos\ctlVal	PLSDCluster	S6K_CD_CPL	DIGITAL	C:N;Opera
35	Medium_Voltage.Incomers.INC_D	BkrPos	S6K_D_INC\XCBR1\Pos	PLSDCluster	S6K_D_INC	LONG	T:EN;m:26
36	Medium_Voltage.Incomers.INC_D	Clt_BkrPos	S6K_D_INC\CSWI1\Pos\ctlVal	PLSDCluster	S6K_D_INC	DIGITAL	C:N;Opera
37	Medium_Voltage.Feeders.FDR_D	BkrPos	S6K_D_FDR\XCBR1\Pos	PLSDCluster	S6K_D_FDR	LONG	T:EN;m:26
38	Medium_Voltage.Feeders.FDR_D	Clt_BkrPos	S6K_D_FDR\CSWI1\Pos\ctlVal	PLSDCluster	S6K_D_FDR	DIGITAL	C:N;Opera
39	Medium_Voltage.Transfers.TRF2_	BkrPos	S6K_D_TRF2\XCBR1\Pos	PLSDCluster	S6K_D_TRF2	LONG	T:EN;m:26
40	Medium_Voltage.Transfers.TRF2_	Clt_BkrPos	S6K_D_TRF2\CSWI1\Pos\ctlVal	PLSDCluster	S6K_D_TRF2	DIGITAL	C:N;Opera
41	Medium_Voltage.Transfers.TRF4_	BkrPos	S6K_D_TRF4\XCBR1\Pos	PLSDCluster	S6K_D_TRF4	LONG	T:EN;m:26
42	Medium_Voltage.Transfers.TRF4_	Clt_BkrPos	S6K_D_TRF4\CSWI1\Pos\ctlVal	PLSDCluster	S6K_D_TRF4	DIGITAL	C:N;Opera
43	Medium_Voltage.Transfers.TRF6_	BkrPos	S6K_D_TRF6\XCBR1\Pos	PLSDCluster	S6K_D_TRF6	LONG	T:EN;m:26
44	Medium_Voltage.Transfers.TRF6_	Clt_BkrPos	S6K_D_TRF6\CSWI1\Pos\ctlVal	PLSDCluster	S6K_D_TRF6	DIGITAL	C:N;Opera
45	Low_Voltage.Incomers.INC_E	BkrPos	S400_1_E_INC\XCBR1\Pos	PLSDCluster	S400_1_E_INC	LONG	T:DS;D:B;n
46	Low_Voltage.Incomers.INC_E	Clt_BkrPos	S400_1_E_INC\CSWI1\Pos\ctlVal	PLSDCluster	S400_1_E_INC	DIGITAL	C:NO;Oper
47	Low_Voltage.Office.FDR1	BkrPos	S400_1_E_FDR1\XCBR1\Pos	PLSDCluster	S400_1_E_FDR1	LONG	T:EN;m:26

Comment: @(Breaker Operate)
Deadband:
Eng Units:
Format: ##

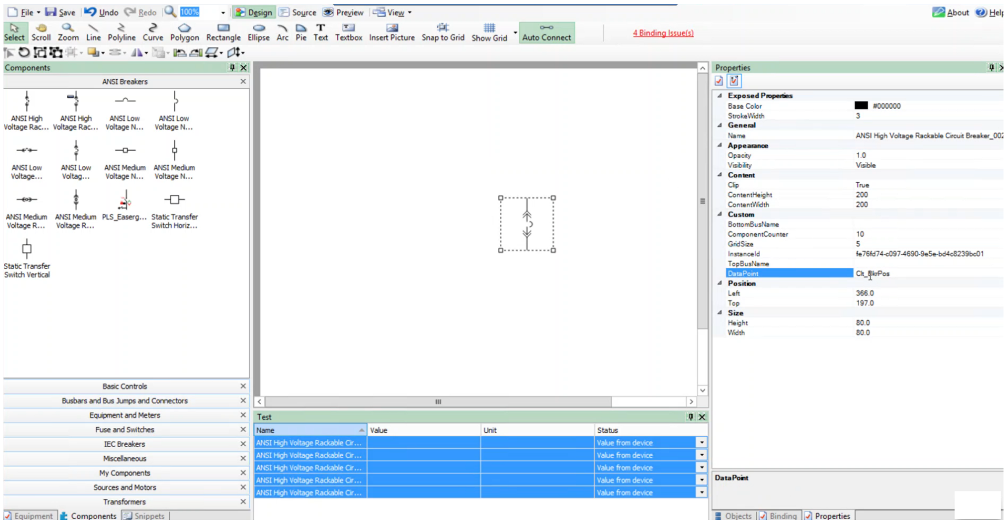
Scale
Raw Zero Scale:
Raw Full Scale:
Eng Zero Scale:
Eng Full Scale:

Custom
Custom 1: S6K_C_FDR
Custom 2:
Custom 3: @(Controls),29
Custom 4:
Custom 5:
Custom 6:
Custom 7:
Custom 8:

Historian
Historize:

Project

d. Enter the Item Name (in this example: **Clt_BkrPos**) in the **DataPoint** field of breaker component custom property:



Exposed Properties	
Base Color	#000000
StrokeWidth	3
General	
Name	ANSI High Voltage Rackable Circuit Breaker_002
Appearance	
Opacity	1.0
Visibility	Visible
Content	
Clip	True
ContentHeight	200
ContentWidth	200
Custom	
BottomBusName	
ComponentCounter	10
GridSize	5
InstanceId	fe76fd74-c097-4690-9e5e-bd4c8239bc01
TopBusName	
DataPoint	Clt_BkrPos
Position	
Left	366.0
Top	197.0
Size	
Height	80.0
Width	80.0

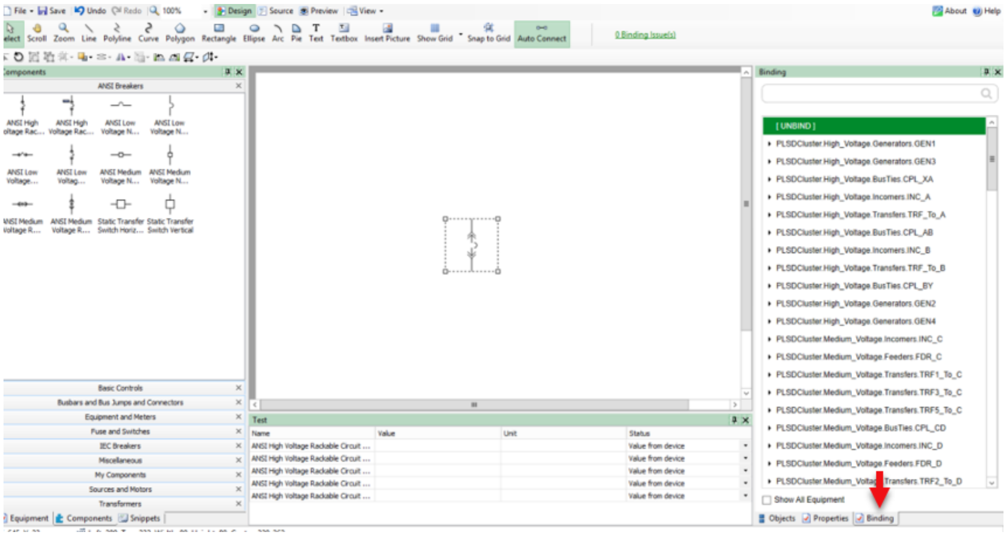
DataPoint

Test

Name	Value	Unit	Status
ANSI High Voltage Rackable Cir...			Value from device
ANSI High Voltage Rackable Cir...			Value from device
ANSI High Voltage Rackable Cir...			Value from device
ANSI High Voltage Rackable Cir...			Value from device
ANSI High Voltage Rackable Cir...			Value from device

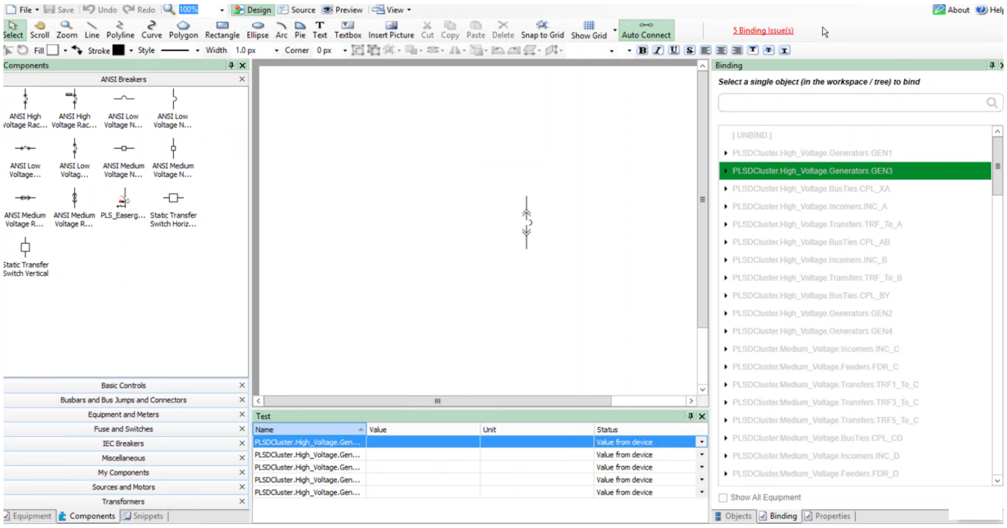
17. Bind the TGML graphic:

    a. At the bottom right corner, click **Binding**:


File
Save
Undo
Redo
100%
Design
Source
Preview
View
About
Help
Select
Scroll
Zoom
Line
Polyline
Curve
Polygon
Rectangle
Ellipse
Arc
Pie
Text
Textbox
Insert Picture
Show Grid
Snap to Grid
Auto Connect
0 Binding Issue(s)
Components
ANSI Breakers
ANSI High Voltage Rac...
ANSI High Voltage Rac...
ANSI Low Voltage N...
ANSI Low Voltage N...
ANSI Low Voltage...
ANSI Low Voltag...
ANSI Medium Voltage N...
ANSI Medium Voltage N...
ANSI Medium Voltage R...
ANSI Medium Voltage R...
Static Transfer Switch Horiz...
Static Transfer Switch Vertical
Basic Controls
Busbars and Bus Jumps and Connectors
Equipment and Meters
Fuse and Switches
IEC Breakers
Miscellaneous
My Components
Sources and Motors
Transformers
Equipment
Components
Snippets
Binding
[UNBIND]
PLSDCluster.High_Voltage.Generators.GEN1
PLSDCluster.High_Voltage.Generators.GEN3
PLSDCluster.High_Voltage.BusTies.CPL_XA
PLSDCluster.High_Voltage.Incomers.INC_A
PLSDCluster.High_Voltage.Transfers.TRF_To_A
PLSDCluster.High_Voltage.BusTies.CPL_AB
PLSDCluster.High_Voltage.Incomers.INC_B
PLSDCluster.High_Voltage.Transfers.TRF_To_B
PLSDCluster.High_Voltage.BusTies.CPL_BY
PLSDCluster.High_Voltage.Generators.GEN2
PLSDCluster.High_Voltage.Generators.GEN4
PLSDCluster.Medium_Voltage.Incomers.INC_C
PLSDCluster.Medium_Voltage.Feeders.FDR_C
PLSDCluster.Medium_Voltage.Transfers.TRF1_To_C
PLSDCluster.Medium_Voltage.Transfers.TRF3_To_C
PLSDCluster.Medium_Voltage.Transfers.TRF5_To_C
PLSDCluster.Medium_Voltage.BusTies.CPL_CD
PLSDCluster.Medium_Voltage.Incomers.INC_D
PLSDCluster.Medium_Voltage.Feeders.FDR_D
PLSDCluster.Medium_Voltage.Transfers.TRF2_To_D
Show All Equipment
Objects
Properties
Binding
Test

Name	Value	Unit	Status
ANSI High Voltage Rackable Circuit ...			Value from device
ANSI High Voltage Rackable Circuit ...			Value from device
ANSI High Voltage Rackable Circuit ...			Value from device
ANSI High Voltage Rackable Circuit ...			Value from device
ANSI High Voltage Rackable Circuit ...			Value from device

    b. From the drop down, select the breaker to bind the breaker component on which you want to do the write operation:


File
Save
Undo
Redo
100%
Design
Source
Preview
View
About
Help
Select
Scroll
Zoom
Line
Polyline
Curve
Polygon
Rectangle
Ellipse
Arc
Pie
Text
Textbox
Insert Picture
Cut
Copy
Paste
Delete
Snap to Grid
Show Grid
Auto Connect
5 Binding Issue(s)
Fill
Stroke
Style
Width
1.0 px
Corner
0 px
Components
ANSI Breakers
ANSI High Voltage Rac...
ANSI High Voltage Rac...
ANSI Low Voltage N...
ANSI Low Voltage N...
ANSI Low Voltage...
ANSI Low Voltag...
ANSI Medium Voltage N...
ANSI Medium Voltage N...
ANSI Medium Voltage R...
ANSI Medium Voltage R...
PLS_Easerg...
Static Transfer Switch Horiz...
Static Transfer Switch Vertical
Basic Controls
Busbars and Bus Jumps and Connectors
Equipment and Meters
Fuse and Switches
IEC Breakers
Miscellaneous
My Components
Sources and Motors
Transformers
Equipment
Components
Snippets
Binding
[UNBIND]
PLSDCluster.High_Voltage.Generators.GEN1
PLSDCluster.High_Voltage.Generators.GEN3
PLSDCluster.High_Voltage.BusTies.CPL_XA
PLSDCluster.High_Voltage.Incomers.INC_A
PLSDCluster.High_Voltage.Transfers.TRF_To_A
PLSDCluster.High_Voltage.BusTies.CPL_AB
PLSDCluster.High_Voltage.Incomers.INC_B
PLSDCluster.High_Voltage.Transfers.TRF_To_B
PLSDCluster.High_Voltage.BusTies.CPL_BY
PLSDCluster.High_Voltage.Generators.GEN2
PLSDCluster.High_Voltage.Generators.GEN4
PLSDCluster.Medium_Voltage.Incomers.INC_C
PLSDCluster.Medium_Voltage.Feeders.FDR_C
PLSDCluster.Medium_Voltage.Transfers.TRF1_To_C
PLSDCluster.Medium_Voltage.Transfers.TRF3_To_C
PLSDCluster.Medium_Voltage.Transfers.TRF5_To_C
PLSDCluster.Medium_Voltage.BusTies.CPL_CD
PLSDCluster.Medium_Voltage.Incomers.INC_D
PLSDCluster.Medium_Voltage.Feeders.FDR_D
Show All Equipment
Objects
Binding
Properties
Test

Select a single object (in the workspace / tree) to bind

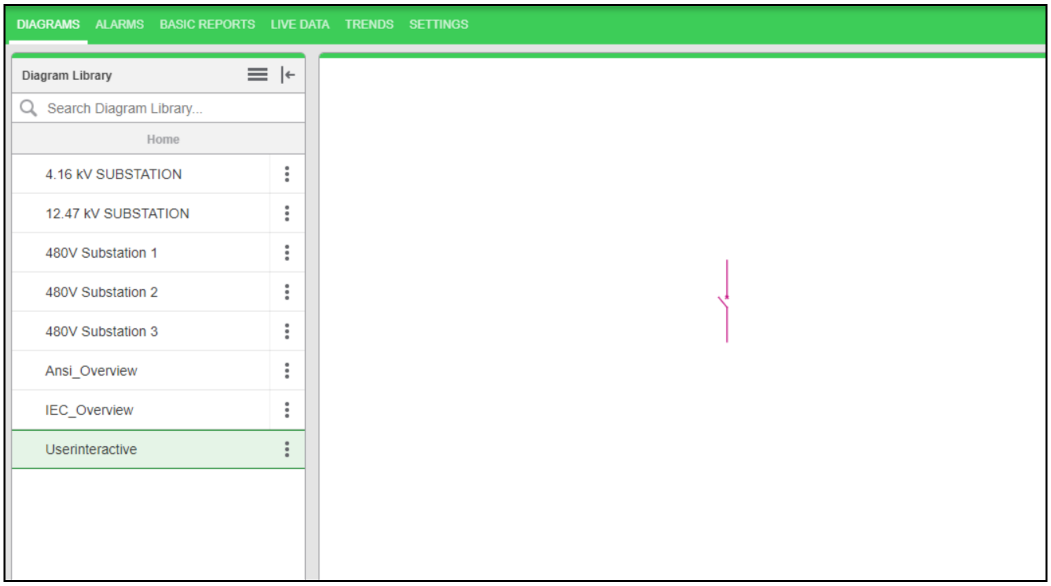
Name	Value	Unit	Status
PLSDCluster.High_Voltage.Gen...			Value from device
PLSDCluster.High_Voltage.Gen...			Value from device
PLSDCluster.High_Voltage.Gen...			Value from device
PLSDCluster.High_Voltage.Gen...			Value from device
PLSDCluster.High_Voltage.Gen...			Value from device

18. Click **Save**.

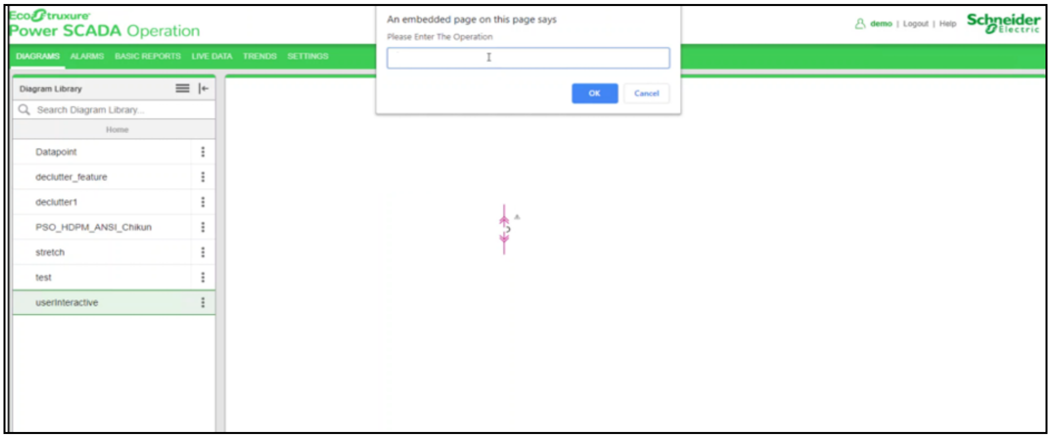
Test the changes:

1. Log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)

2. Click **Userinteractive** tab on the left panel:



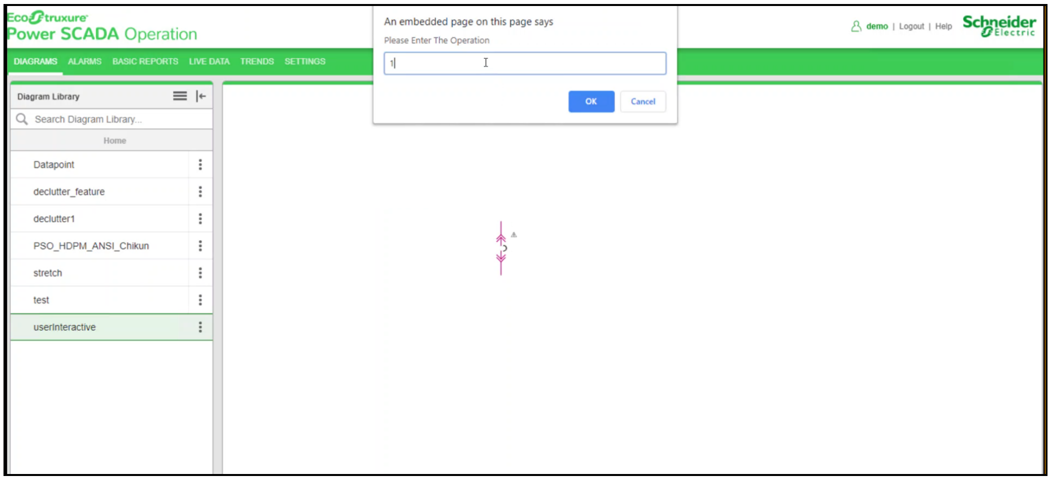
3. Click on the **Breaker** component in the workspace. A pop-up is displayed.



An embedded page on this page says

Please Enter The Operation

4. Type the value as **1** to close the breaker.



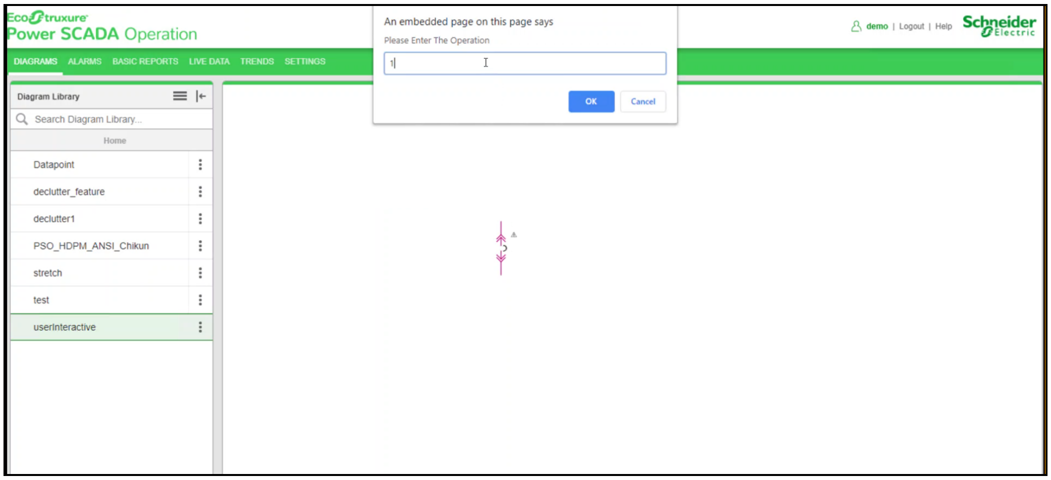
An embedded page on this page says

Please Enter The Operation

1

> **NOTE**: The Breaker can be controlled by giving required operation values: **0** for Open, or **1** for Close.

5. Enter **1** to close breaker, and then click **OK**.

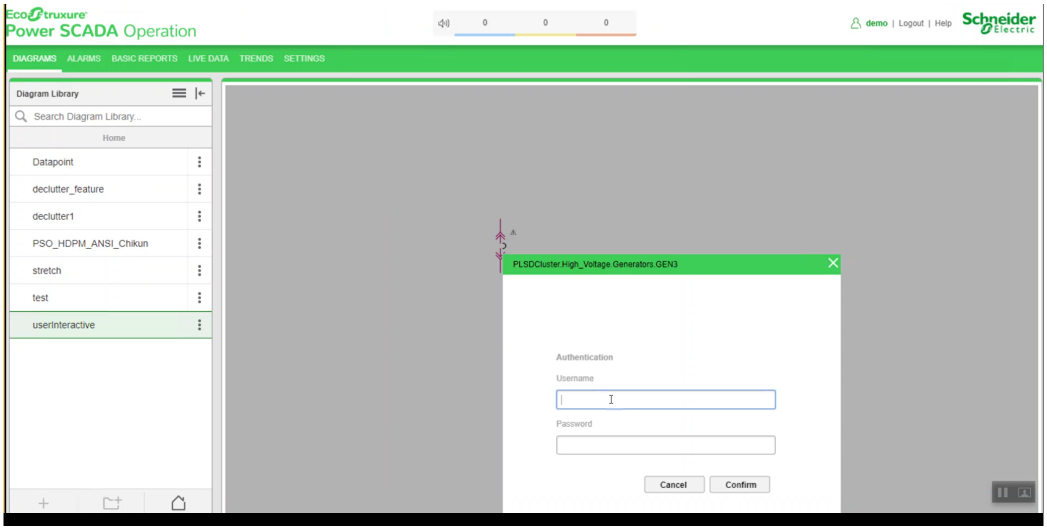


An embedded page on this page says

Please Enter The Operation

1

6. Enter your **Username** and **Password**, and then click **Confirm**.



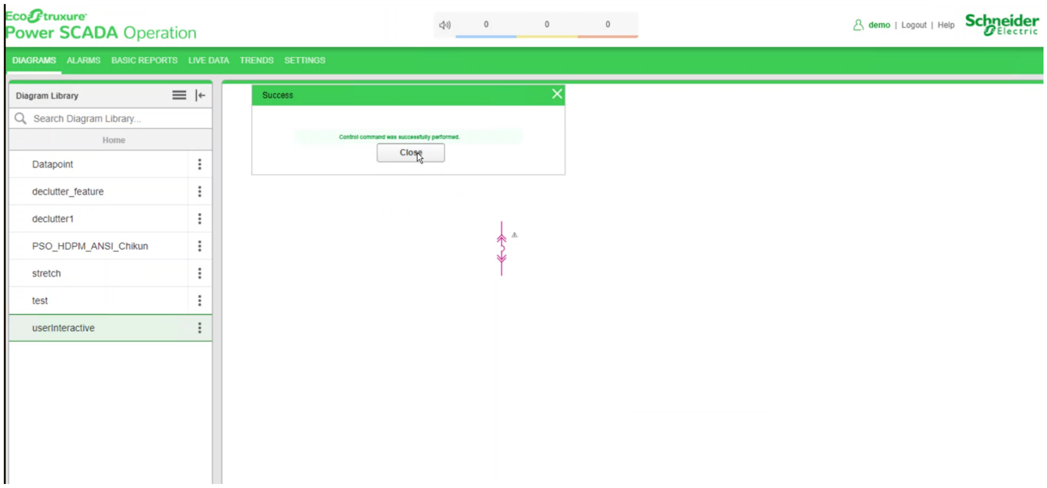
PLSDCluster.High_Voltage Generators.GEN3

Authentication

Username

Password

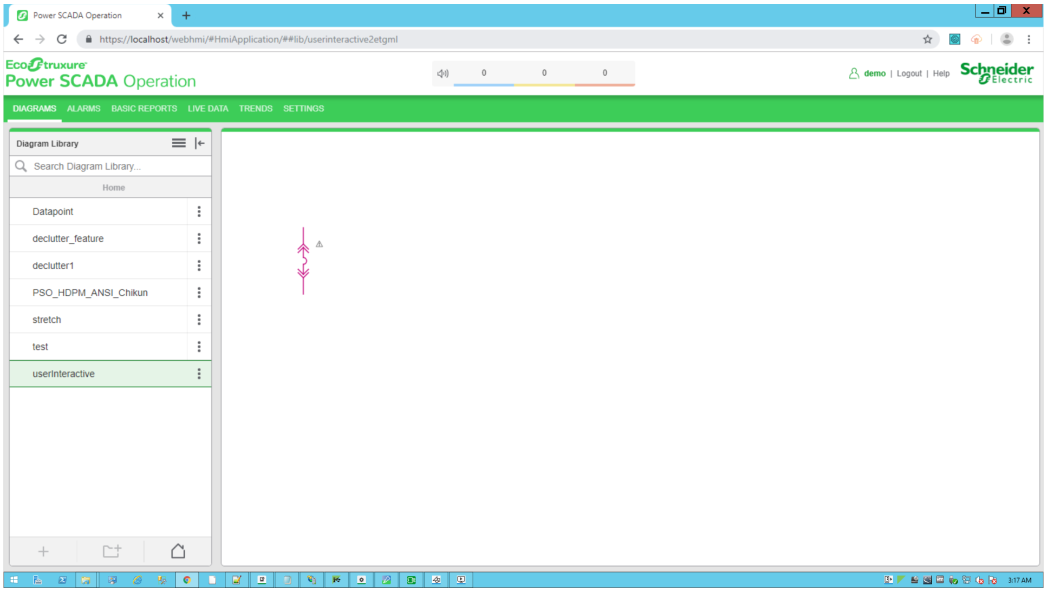
7. Click **Close**.



Success

Control command was successfully performed.

The breaker is closed:

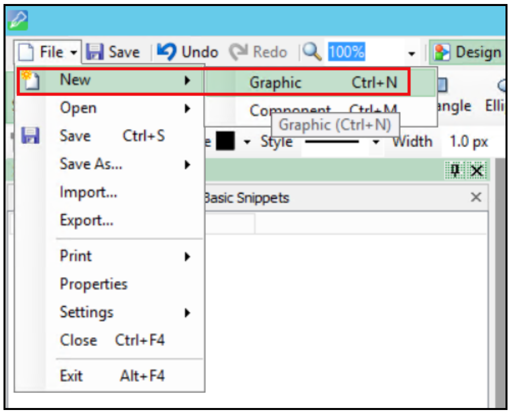

Power SCADA Operation
https://localhost/webhmi/#HmiApplication/##lib/userinteractive2etgml
EcoStruxure Power SCADA Operation
demo | Logout | Help
Schneider Electric
DIAGRAMS ALARMS BASIC REPORTS LIVE DATA TRENDS SETTINGS
Diagram Library
Search Diagram Library...
Home
Datapoint
declutter_feature
declutter1
PSO_HDPM_ANSI_Chikun
stretch
test
userInteractive
3:17 AM

Conditional Write

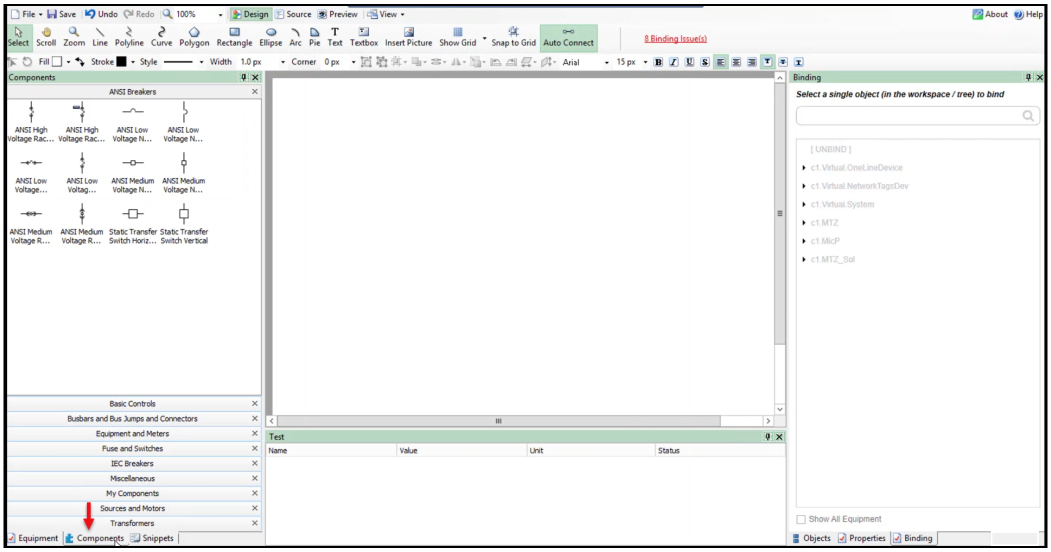
Use the Conditional Write component to operate the breaker or digital output based on a condition. For example: to ensure a breaker should open only if it is closed. Conditional Write checks whether the correct DataPoint values are written. If they are wrong, the write operation is not be performed.

To use Conditional Write:

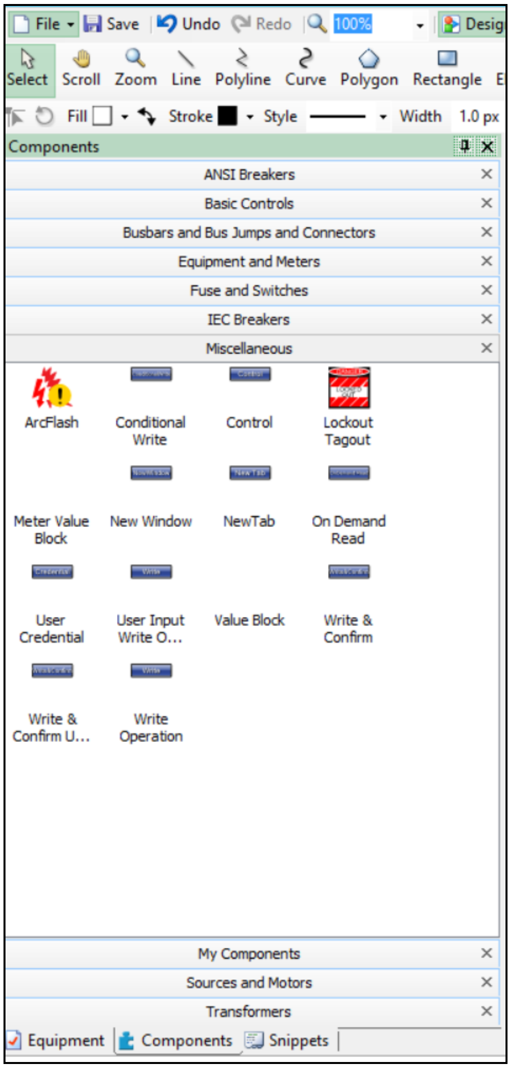
1. In the Graphics Editor, create a new graphic file:



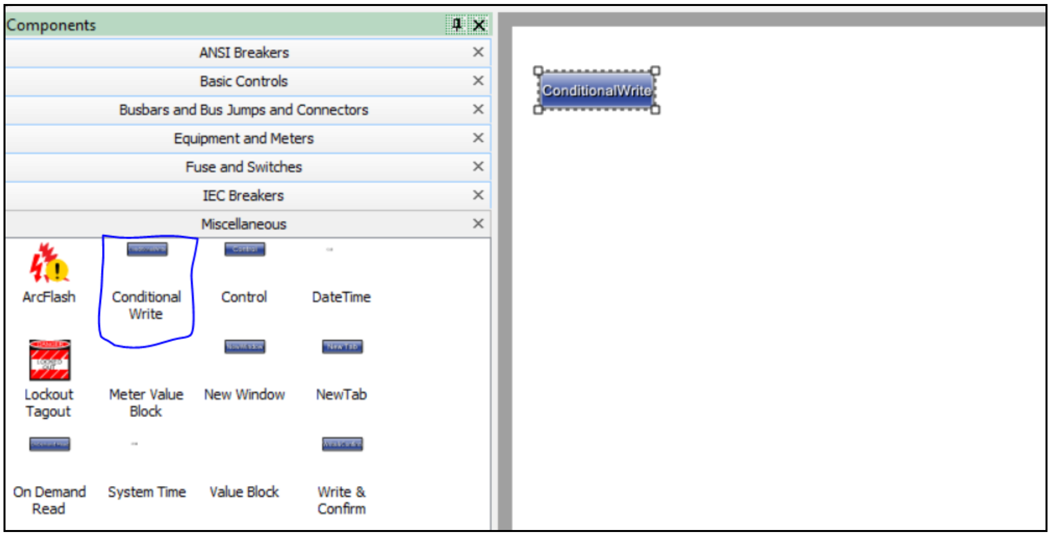
2. Go to **Components** pane in the bottom left corner of the screen.



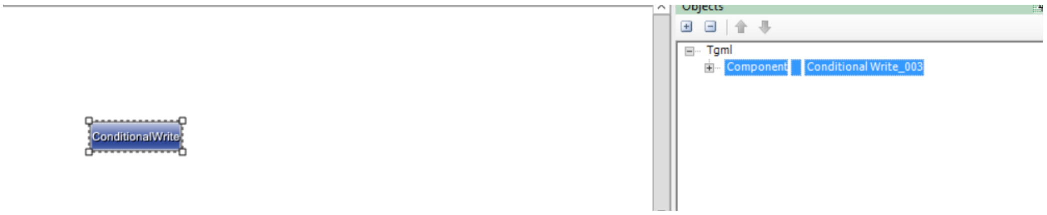
3. In the **Components** pane, expand **Miscellaneous**.



4. Drag and drop the **Conditional Write** component from **Miscellaneous** section to the editor as folows:



5. Go to the **Object** pane in the bottom right corner and click the **Component** which is binded under the TGML as folows:



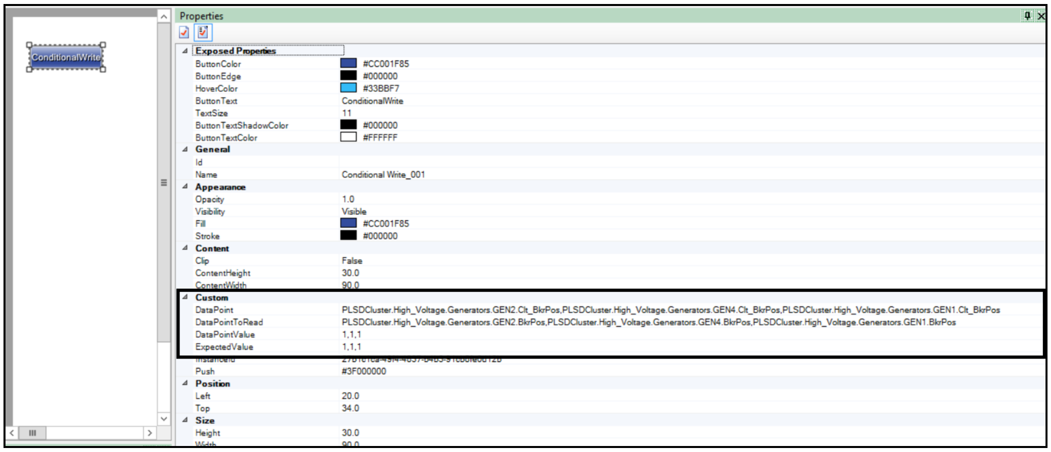
6. Go to the **Properties** pane (beside the **Object** pane of the component) and type the below properties as per your requirement as shown below.

   a. **DataPoint**: Specify the fully qualified item names to do the write operation. Commas can be used as a delimiter to do the write operation. If only one DataPoint is needed, then comma is not required.

   b. **DataPointToRead**: Specify the fully qualified item names to read and verify the item names are written correclty.

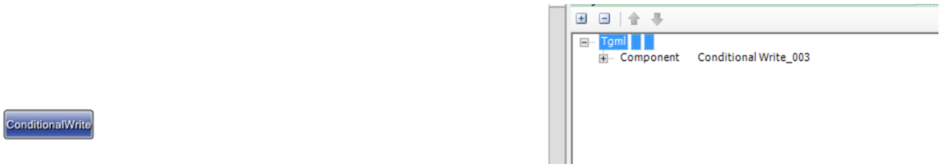
   c. **DatapointValue**: Specify the value to write which was specified in **DataPoint** field.

d. **Expected Value**: Specify the expected value to verify the final value.

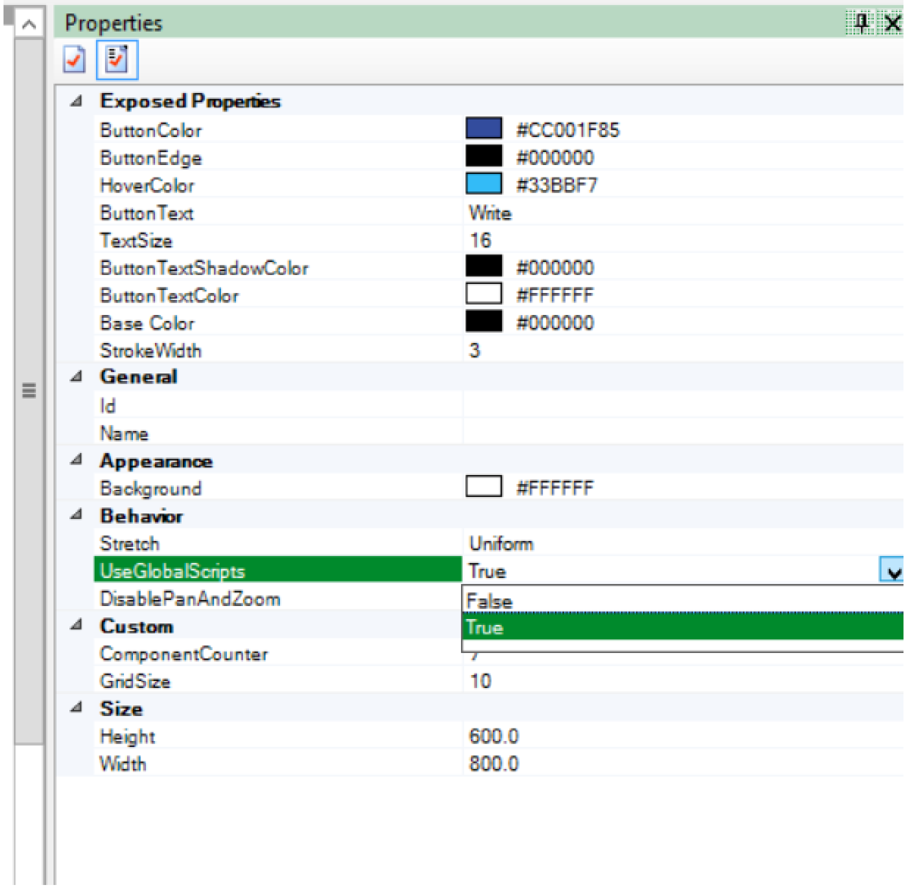


Exposed Properties	
ButtonColor	#CC001F85
ButtonEdge	#000000
HoverColor	#33BBF7
ButtonText	ConditionalWrite
TextSize	11
ButtonTextShadowColor	#000000
ButtonTextColor	#FFFFFF
General	
Id	
Name	Conditional Write_001
Appearance	
Opacity	1.0
Visibility	Visible
Fill	#CC001F85
Stroke	#000000
Content	
Clip	False
ContentHeight	30.0
ContentWidth	90.0
Custom	
DataPoint	PLSDCluster.High_Voltage.Generators.GEN2.Clt_BkrPos,PLSDCluster.High_Voltage.Generators.GEN4.Clt_BkrPos,PLSDCluster.High_Voltage.Generators.GEN1.Clt_BkrPos
DataPointToRead	PLSDCluster.High_Voltage.Generators.GEN2.BkrPos,PLSDCluster.High_Voltage.Generators.GEN4.BkrPos,PLSDCluster.High_Voltage.Generators.GEN1.BkrPos
DataPointValue	1,1,1
ExpectedValue	1,1,1
InstanceId	[illegible]
Push	#3F000000
Position	
Left	20.0
Top	34.0
Size	
Height	30.0
Width	90.0

7. Go to **Object** pane and click on **Tgml** as shown below.



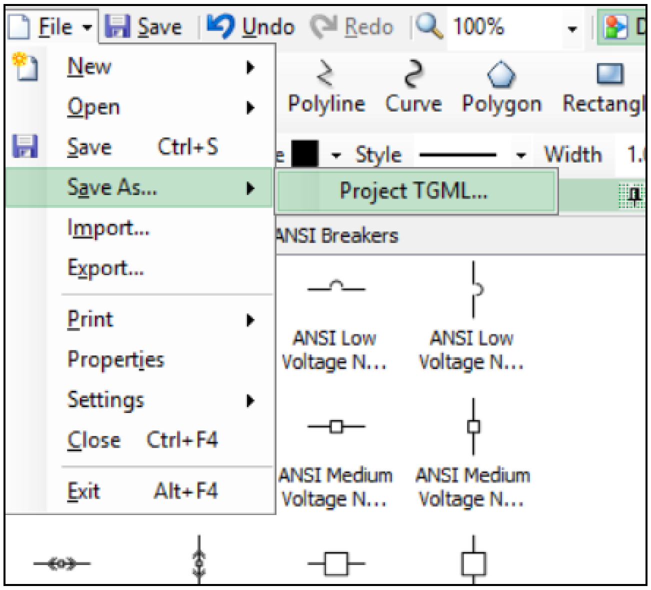
8. Go to **Properties** pane again, select true from the drop-down in the **UseGlobalScripts** attribute section as shown below.



Properties

Exposed Properties	
ButtonColor	#CC001F85
ButtonEdge	#000000
HoverColor	#33BBF7
ButtonText	Write
TextSize	16
ButtonTextShadowColor	#000000
ButtonTextColor	#FFFFFF
Base Color	#000000
StrokeWidth	3
General	
Id	
Name	
Appearance	
Background	#FFFFFF
Behavior	
Stretch	Uniform
UseGlobalScripts	True
DisablePanAndZoom	False True
Custom	
ComponentCounter	
GridSize	10
Size	
Height	600.0
Width	800.0

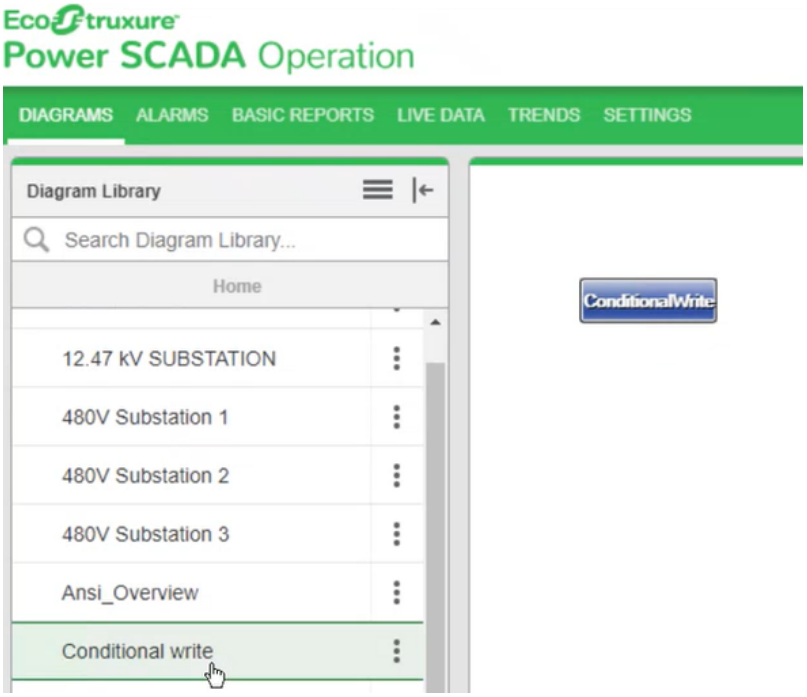
9. Go to **File > Save As > Project TGML**.



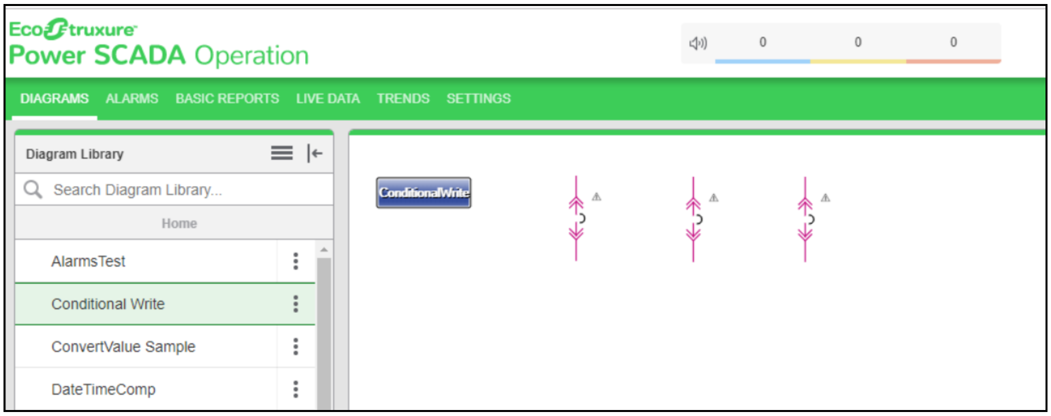
Test the changes:

1. Log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)

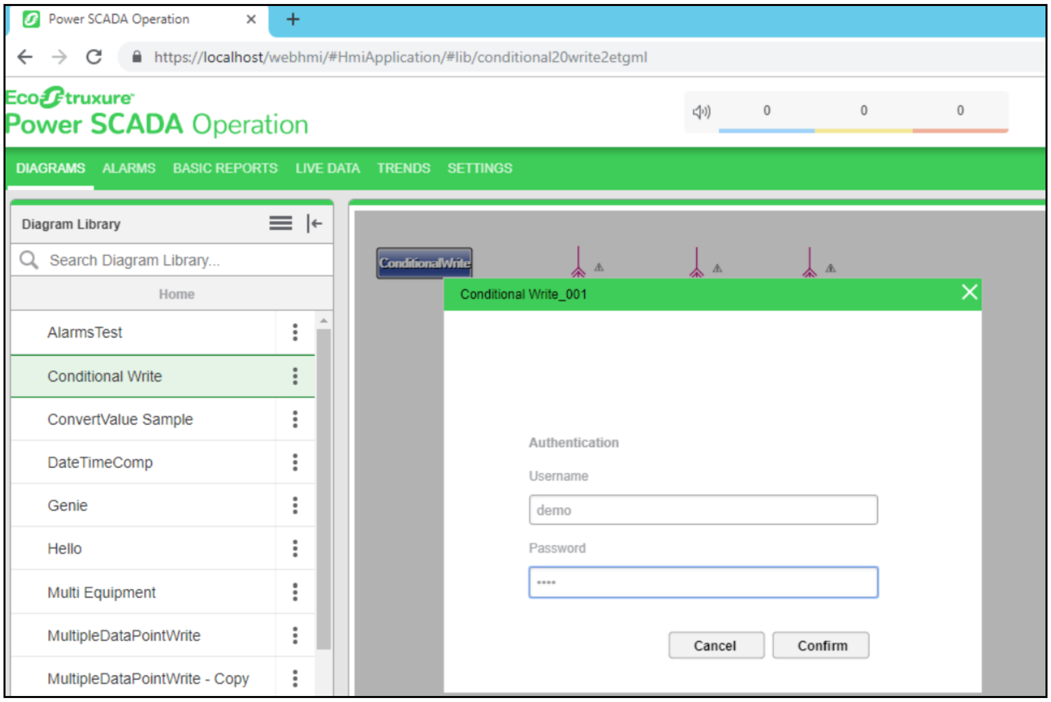
2. In the **Diagram Library**, click the **ConditionalWrite** component you created:



3. For example, verify if the breakers operation is working properly in this feature.



4. Click **ConditionalWrite** and it will prompt for **user credential popup**:



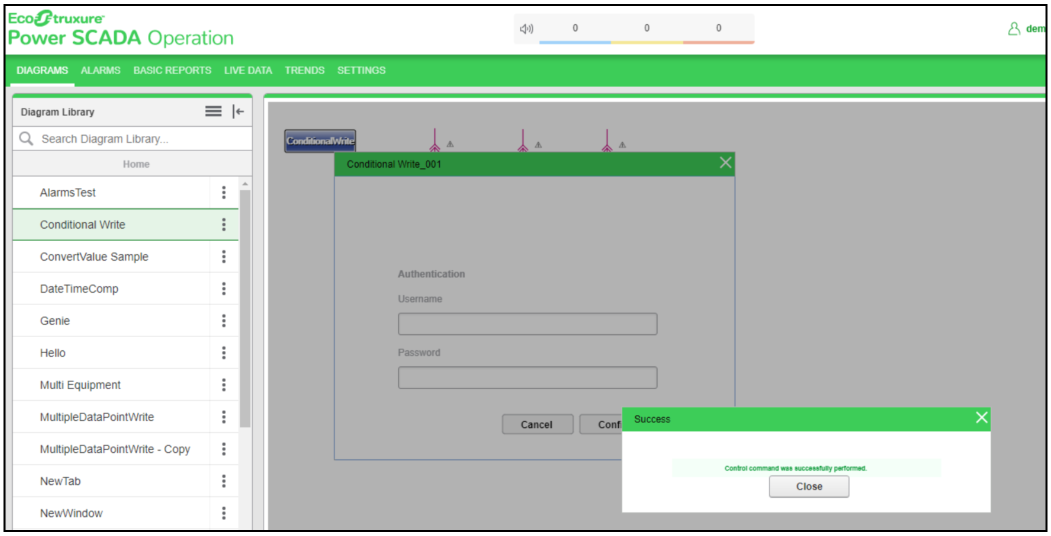
Username: demo

Password: ••••

If the read value does not match with the expected value, then the user credential popup message is not displayed.

5. Enter your username and password, and then click **Confirm**. To control which components require authentication, see Turning off credential requirements for control components.

   Upon successful operation, a success popup window is displayed:



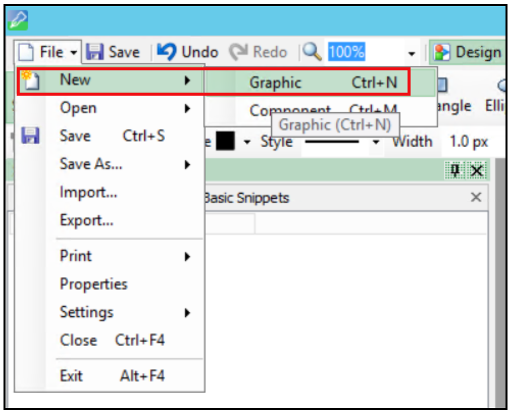
Control command was successfully performed.

On Demand Read

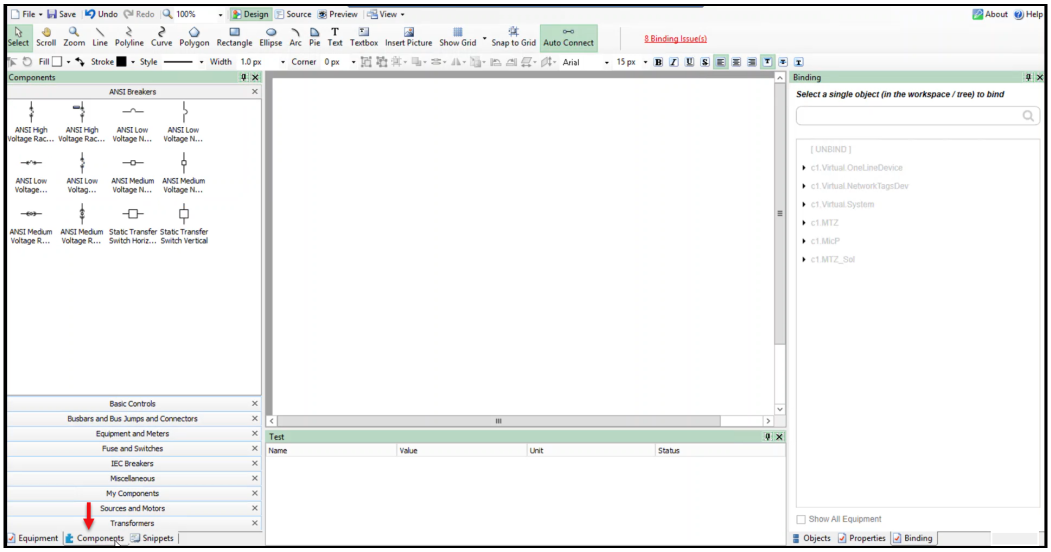
Use the On Demand Read component to read the values from PSO without binding the components in the TGML file. The advantage of this feature is that it reads the value from PSO on an as-needed basis, instead of polling PSO on a certain time interval.

To use On Demand Read:

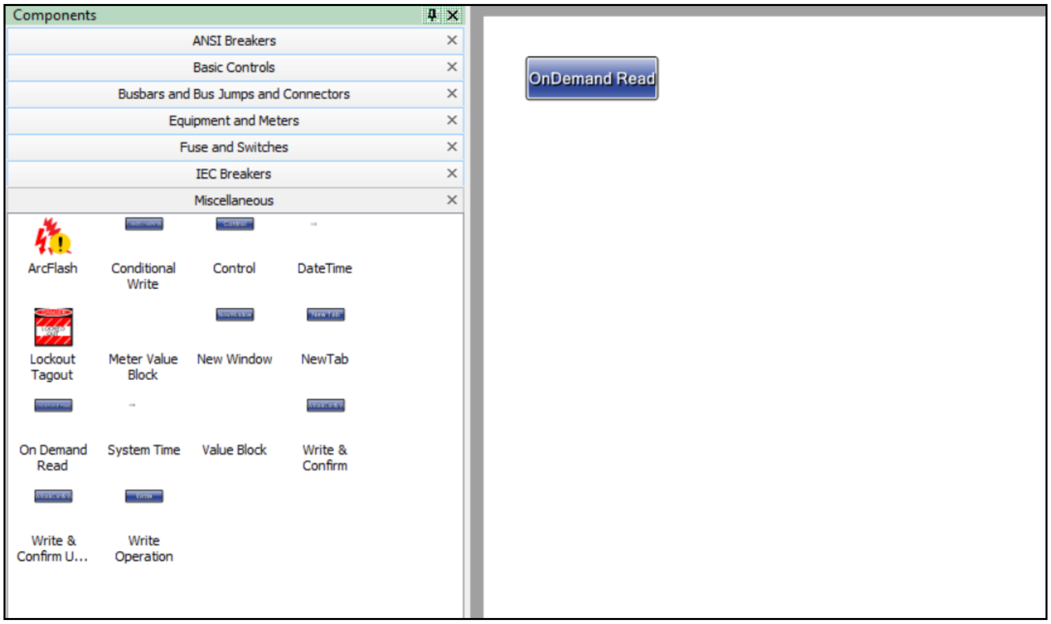
1. In the **Graphics Editor**, create a new graphic:



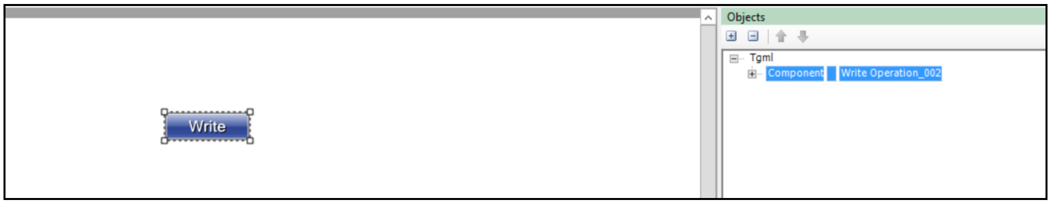
2. In the bottom left corner, click **Components**:



3. Expand **Miscellaneous**, and then drag and drop the **On Demand Read** component and bind the component based on your requirement.

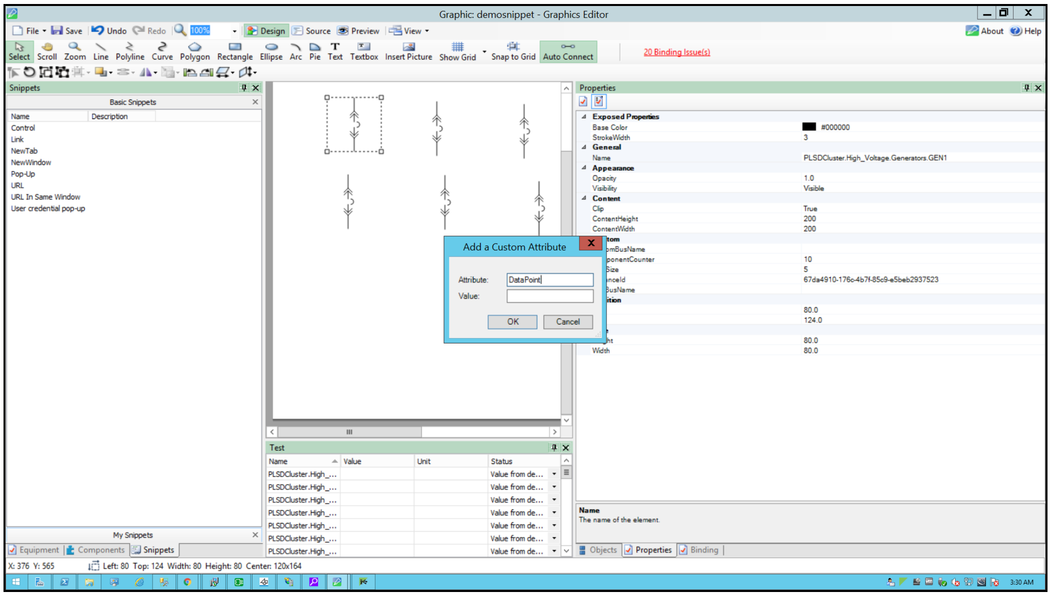


4. Go to the **Object** pane in the bottom right corner, and then click the **Component** which is binded under the TGML:



5. Go to the **Properties** pane just beside the **Object** pane, and then enter the **DataPoint** IDs of the item names separated by comma. If it is a single **Datapoint** ID, no comma is required.

   The following image is a reference for examples to read the Current A and Current B value:



Add a Custom Attribute

Attribute: DataPoint

Value:

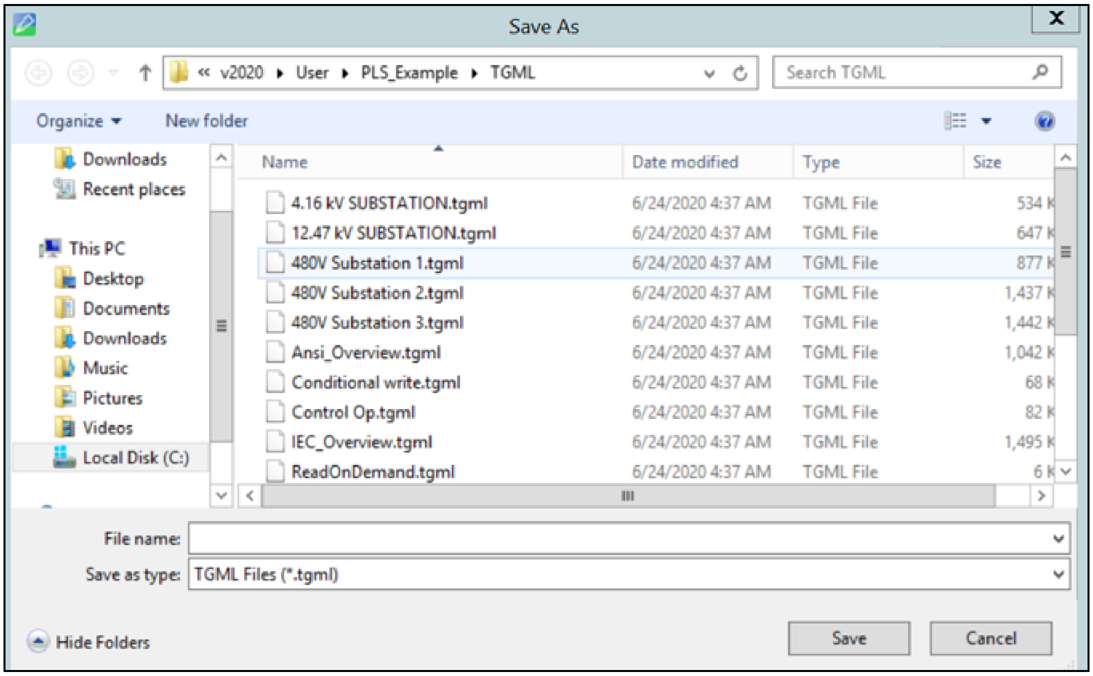
Base Color #000000
StrokeWidth 3
Name PLSDCluster.High_Voltage.Generators.GEN1
Opacity 1.0
Visibility Visible
Clip True
ContentHeight 200
ContentWidth 200

Name
The name of the element.

X: 376 Y: 565 Left: 80 Top: 124 Width: 80 Height: 80 Center: 120x164

6. Go to **File > Save As > Project TGML**.

7. Fill in the project name in the **File name** field and click **Save**:



Save As

« v2020 ▸ User ▸ PLS_Example ▸ TGML

Search TGML

Organize ▾ New folder

Downloads
Recent places
This PC
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)

Name	Date modified	Type	Size
4.16 kV SUBSTATION.tgml	6/24/2020 4:37 AM	TGML File	534 K
12.47 kV SUBSTATION.tgml	6/24/2020 4:37 AM	TGML File	647 K
480V Substation 1.tgml	6/24/2020 4:37 AM	TGML File	877 K
480V Substation 2.tgml	6/24/2020 4:37 AM	TGML File	1,437 K
480V Substation 3.tgml	6/24/2020 4:37 AM	TGML File	1,442 K
Ansi_Overview.tgml	6/24/2020 4:37 AM	TGML File	1,042 K
Conditional write.tgml	6/24/2020 4:37 AM	TGML File	68 K
Control Op.tgml	6/24/2020 4:37 AM	TGML File	82 K
IEC_Overview.tgml	6/24/2020 4:37 AM	TGML File	1,495 K
ReadOnDemand.tgml	6/24/2020 4:37 AM	TGML File	6 K

File name:

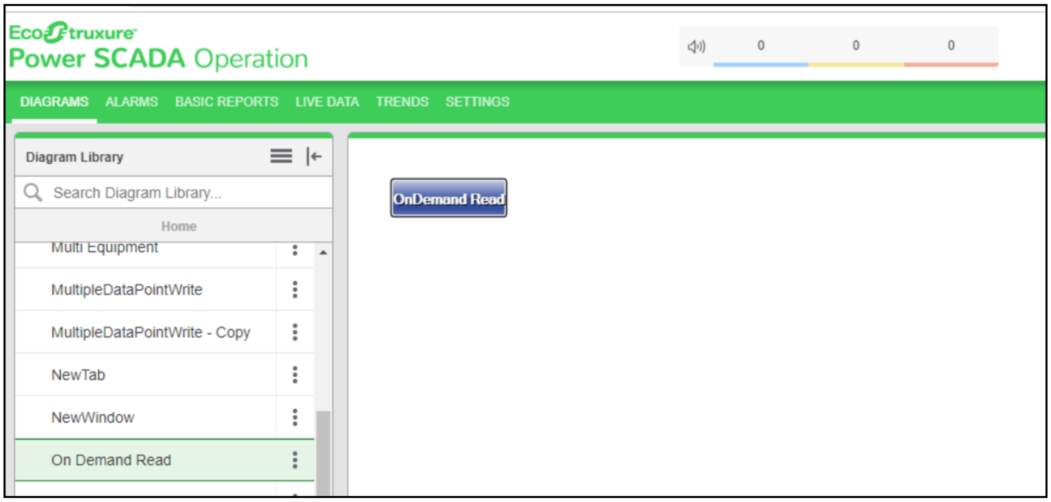
Save as type: TGML Files (*.tgml)

Hide Folders

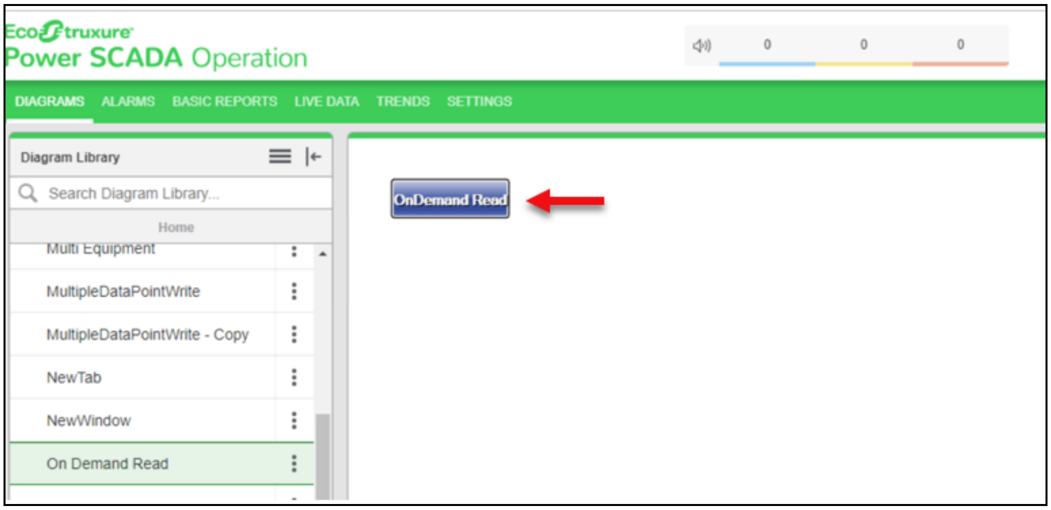
Save Cancel

To test the changes:

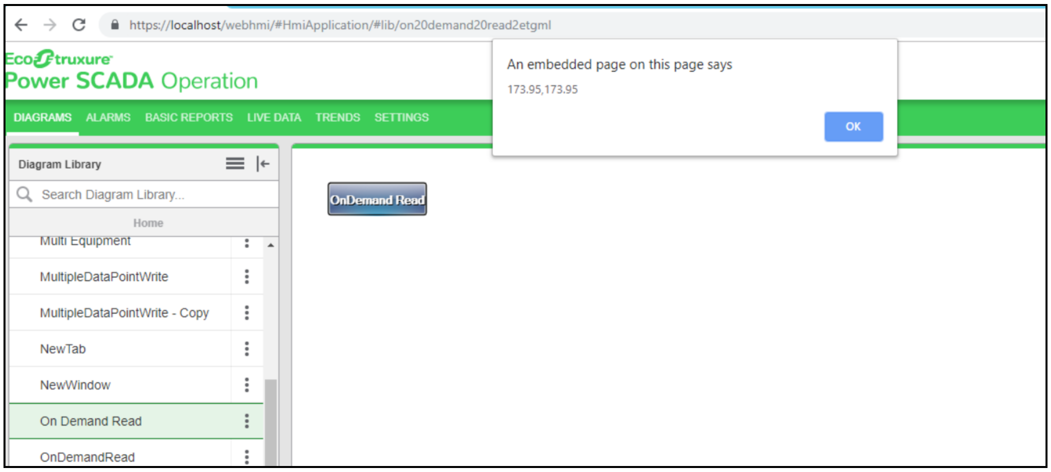
1. Log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**.)
2. Click **OnDemandRead**:



3. Click on the **Component** on the right side of the panel.



4. The value is read from PSO and displayed to you:



An embedded page on this page says

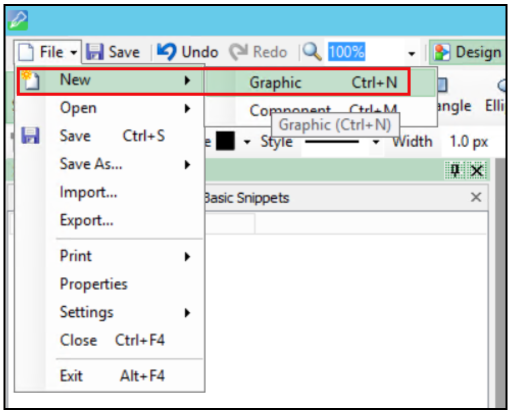
173.95,173.95

Single or Multiple DataPoint Write

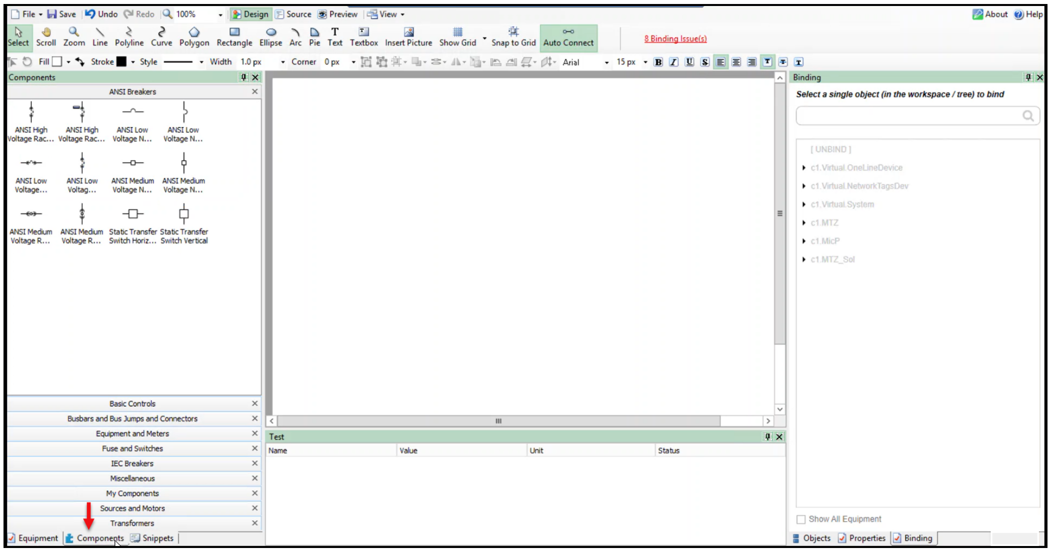
Use the Single or Multiple DataPoint Write component to perform generic write operations for a single or block write. It can be used to write multiple set-points at once, setting up tariff and command write.

To write single or multiple DataPoint:

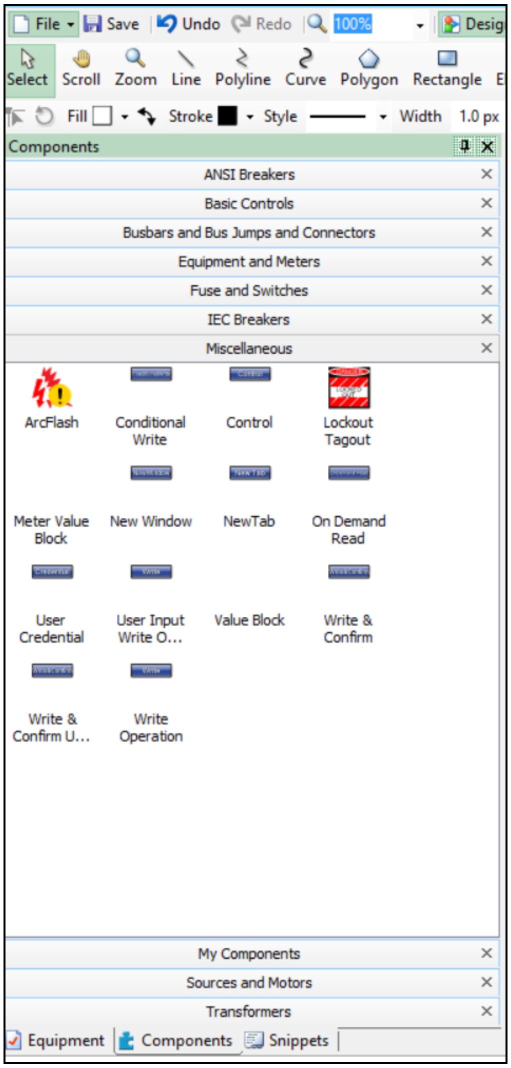
1. In the Graphics Editor, create a new graphics file:



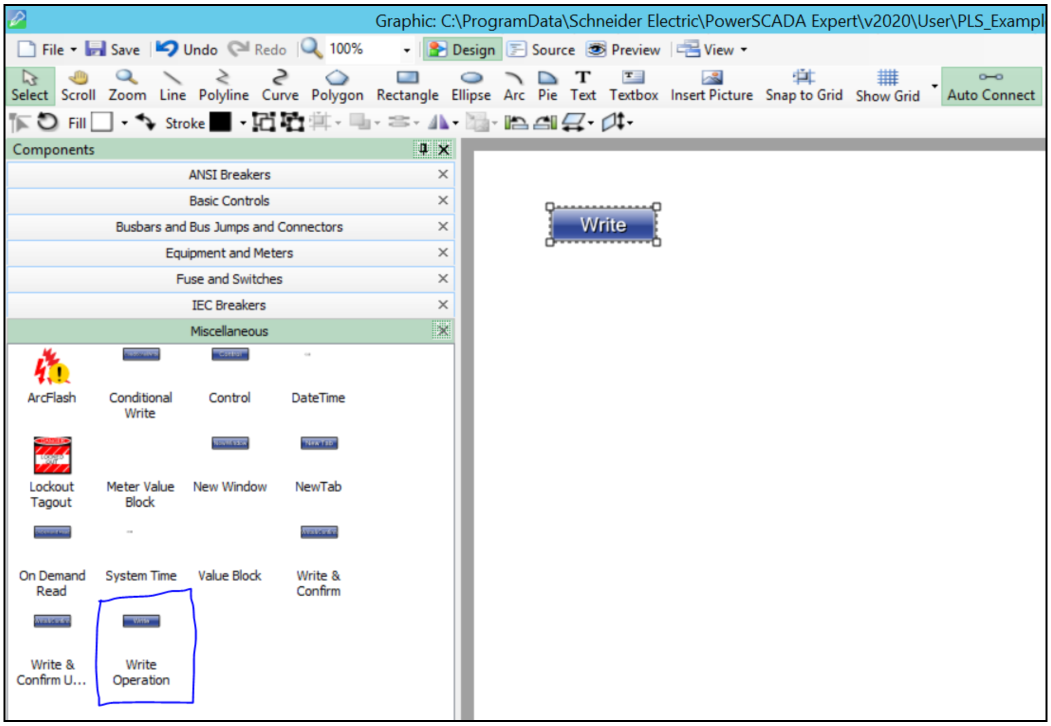
2. In the bottom right corner, go to **Components**:



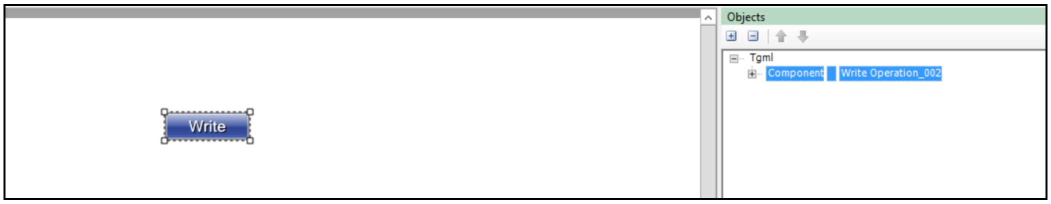
3. Click **Miscellaneous**:



4. Drag and drop a **Write Operation** component to the workspace:

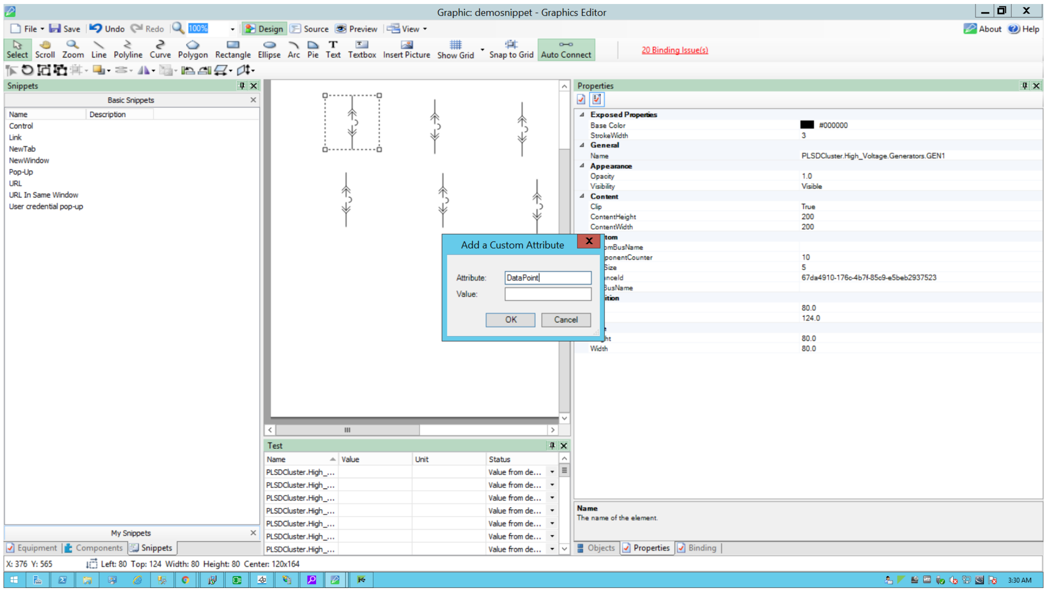


5. In the bottom right corner, click **Objects**, and then click on the write operation component:



6. Click **Properties** (beside **Objects**.)

The **DataPoint** attribute is located in the **Custom** group:



Attribute: DataPoint
Value:

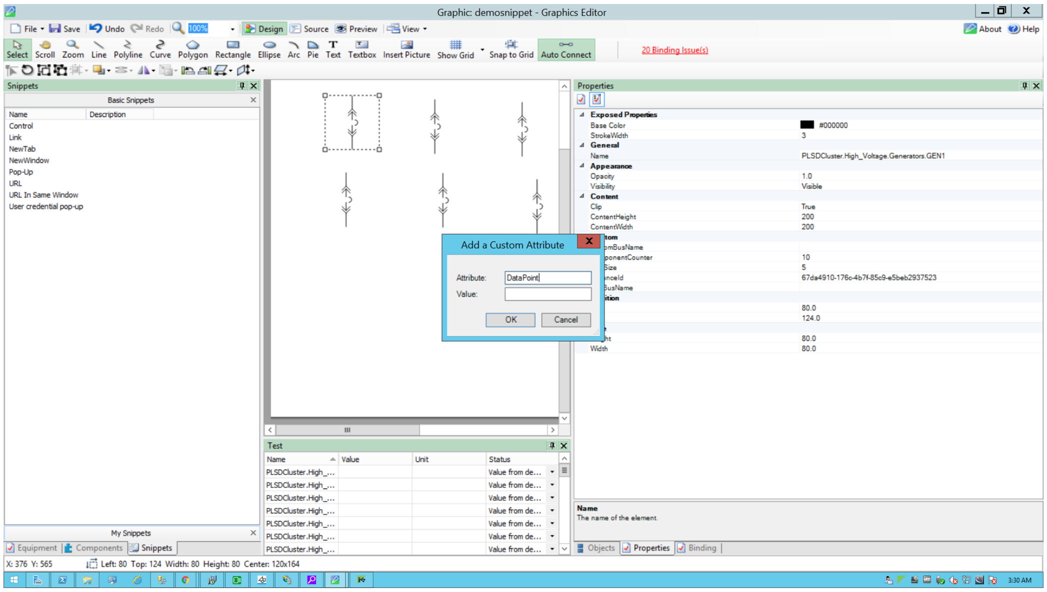
Exposed Properties
Base Color #000000
StrokeWidth 3
General
Name PLSDCluster.High_Voltage.Generators.GEN1
Appearance
Opacity 1.0
Visibility Visible
Content
Clip True
ContentHeight 200
ContentWidth 200
tom
omBusName
ponentCounter 10
Size 5
nceId 67da4910-176c-4b7f-85c9-e5beb2937523
BusName
ition
80.0
124.0
ht 80.0
Width 80.0

Name
The name of the element.

PLSDCluster.High_... Value from de...
PLSDCluster.High_... Value from de...
PLSDCluster.High_... Value from de...
PLSDCluster.High_... Value from de...
PLSDCluster.High_... Value from de...
PLSDCluster.High_... Value from de...
PLSDCluster.High_... Value from de...

X: 376 Y: 565 Left: 80 Top: 124 Width: 80 Height: 80 Center: 120x164

7. In **DataPoint**, enter the fully qualified DataPoint value. The **DataPoint** attribute is located in the **Custom** group:



Add a Custom Attribute

Attribute: DataPoint

Value:

OK Cancel

Properties

Exposed Properties	
Base Color	#000000
StrokeWidth	3
General	
Name	PLSDCluster.High_Voltage.Generators.GEN1
Appearance	
Opacity	1.0
Visibility	Visible
Content	
Clip	True
ContentHeight	200
ContentWidth	200
tom	
omBusName	
ponentCounter	10
Size	5
nceId	67da4910-176c-4b7f-85c9-e5beb2937523
BusName	
ition	
	80.0
	124.0
ht	80.0
Width	80.0

Name
The name of the element.

Test

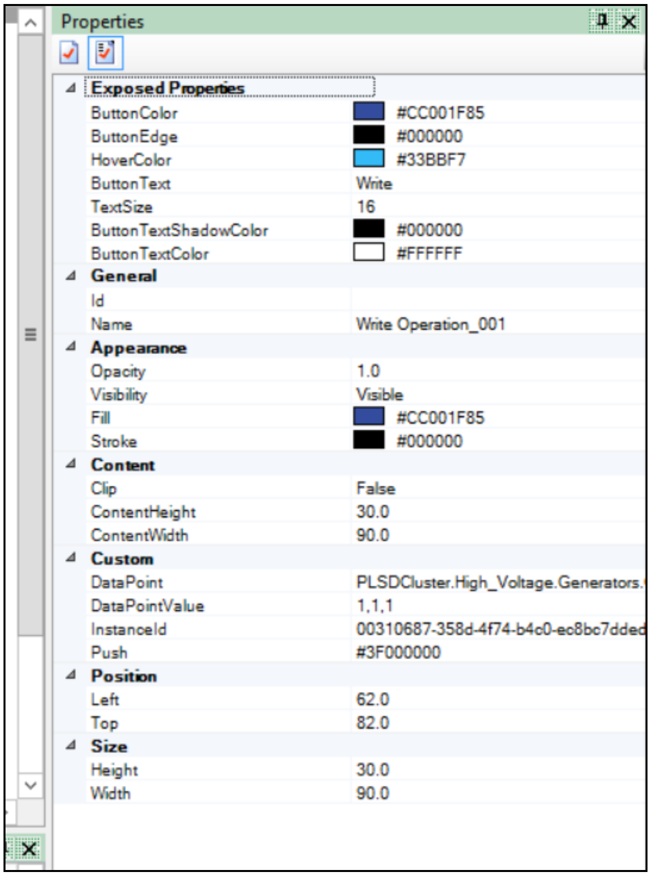
Name	Value	Unit	Status
PLSDCluster.High_...			Value from de...
PLSDCluster.High_...			Value from de...
PLSDCluster.High_...			Value from de...
PLSDCluster.High_...			Value from de...
PLSDCluster.High_...			Value from de...
PLSDCluster.High_...			Value from de...
PLSDCluster.High_...			Value from de...

X: 376 Y: 565 Left: 80 Top: 124 Width: 80 Height: 80 Center: 120x164

NOTE: Use a comma to add multiple **DataPoint** values.

8. Type the **DataPointValue** to write in the **DataPointValue** property.

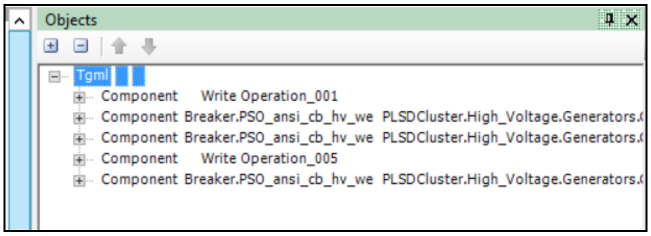
NOTE: Use a comma to add multiple **DataPointValue** values.



Properties

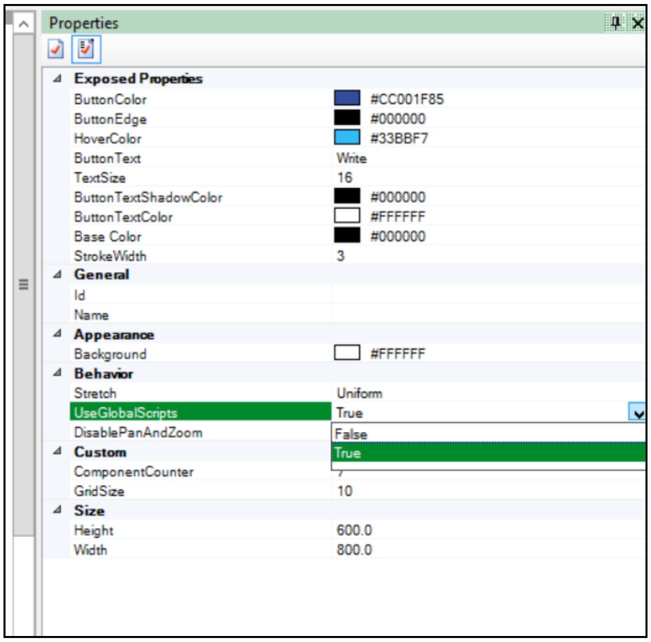
Exposed Properties	
ButtonColor	#CC001F85
ButtonEdge	#000000
HoverColor	#33BBF7
ButtonText	Write
TextSize	16
ButtonTextShadowColor	#000000
ButtonTextColor	#FFFFFF
General	
Id	
Name	Write Operation_001
Appearance	
Opacity	1.0
Visibility	Visible
Fill	#CC001F85
Stroke	#000000
Content	
Clip	False
ContentHeight	30.0
ContentWidth	90.0
Custom	
DataPoint	PLSDCluster.High_Voltage.Generators.
DataPointValue	1,1,1
InstanceId	00310687-358d-4f74-b4c0-ec8bc7dded
Push	#3F000000
Position	
Left	62.0
Top	82.0
Size	
Height	30.0
Width	90.0

9. In the **Object** pane, expand **Tgml**:



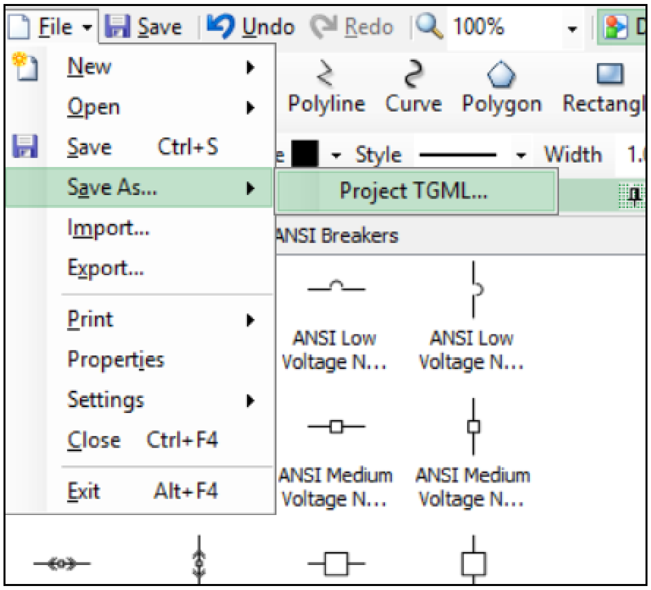
```
⊟ Tgml
  ⊞ Component    Write Operation_001
  ⊞ Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.(
  ⊞ Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.(
  ⊞ Component    Write Operation_005
  ⊞ Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.(
```

10. In **Properties**, from the **UseGlobalScripts** attribute drop-down, select **True**:



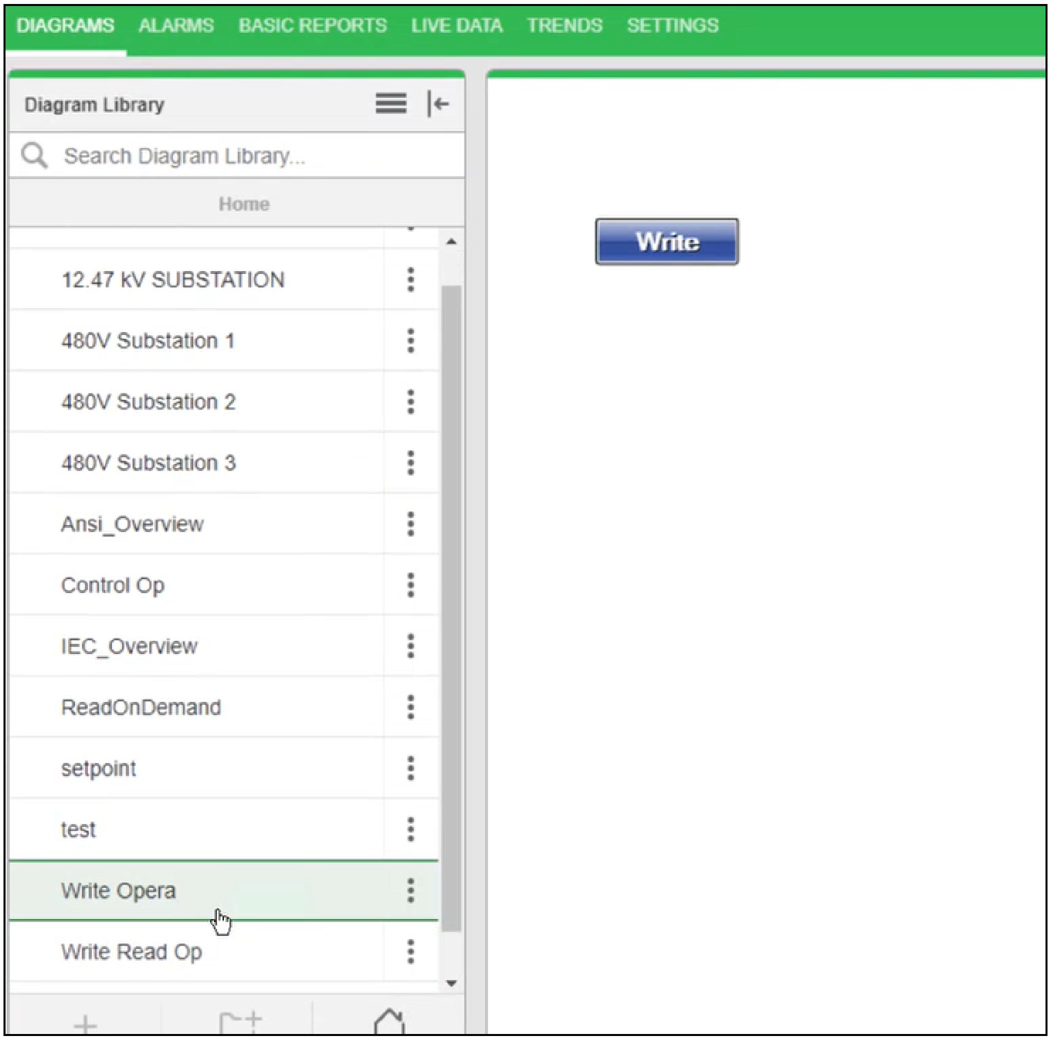
Exposed Properties	
ButtonColor	#CC001F85
ButtonEdge	#000000
HoverColor	#33BBF7
ButtonText	Write
TextSize	16
ButtonTextShadowColor	#000000
ButtonTextColor	#FFFFFF
Base Color	#000000
StrokeWidth	3
General	
Id	
Name	
Appearance	
Background	#FFFFFF
Behavior	
Stretch	Uniform
UseGlobalScripts	True
DisablePanAndZoom	False True
Custom	
ComponentCounter	7
GridSize	10
Size	
Height	600.0
Width	800.0

11. Go to **File > Save As > Project TGML**.

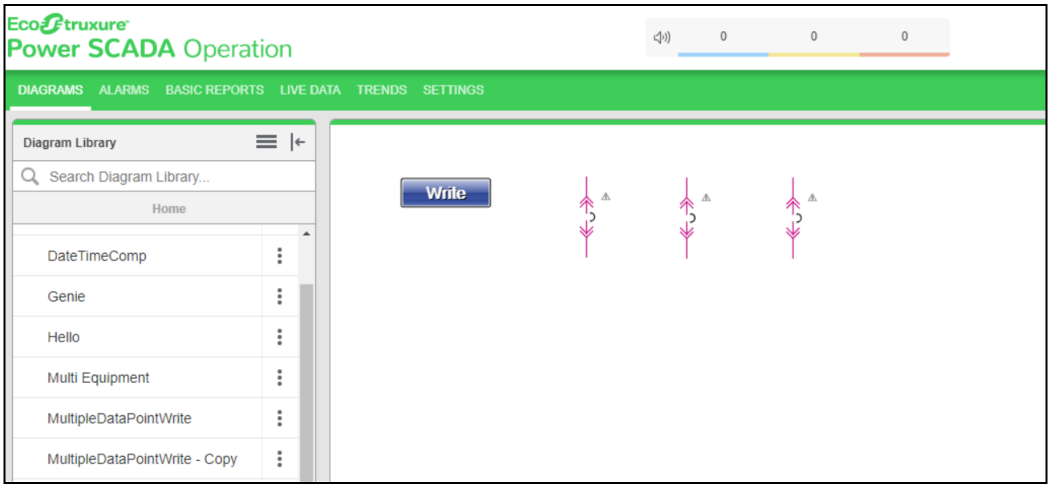


Test the changes:

1. Log in to PSO Web Applications (**https://localhost/webhmi** or **https://ipaddress/webhmi**)
2. Click the **Write Operation** diagram:

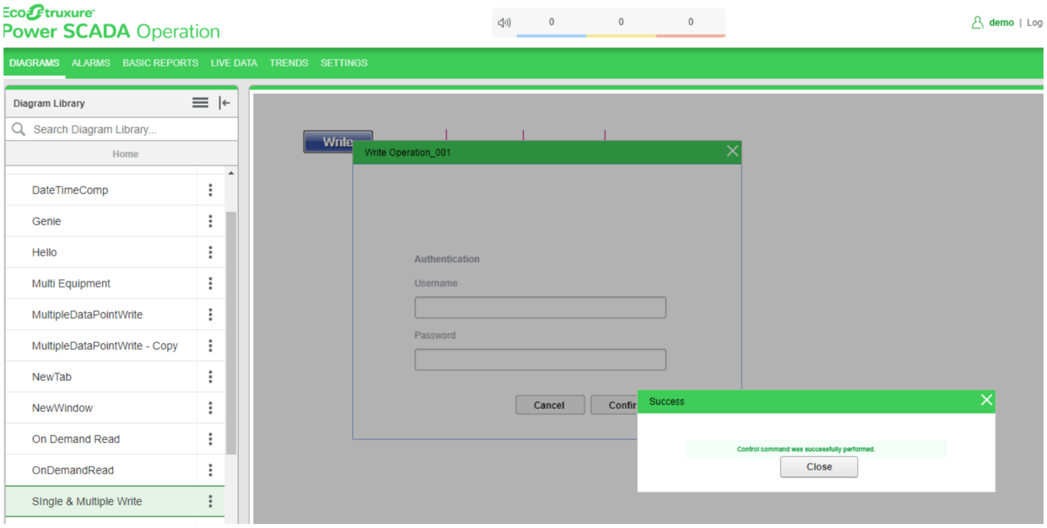


For example, the **Write Operation** closes 3 breakers:



3. Click the **Write Operation** component.
4. Enter your username and password, and then click **Confirm**. To control which components require authentication, see Turning off credential requirements for control components.

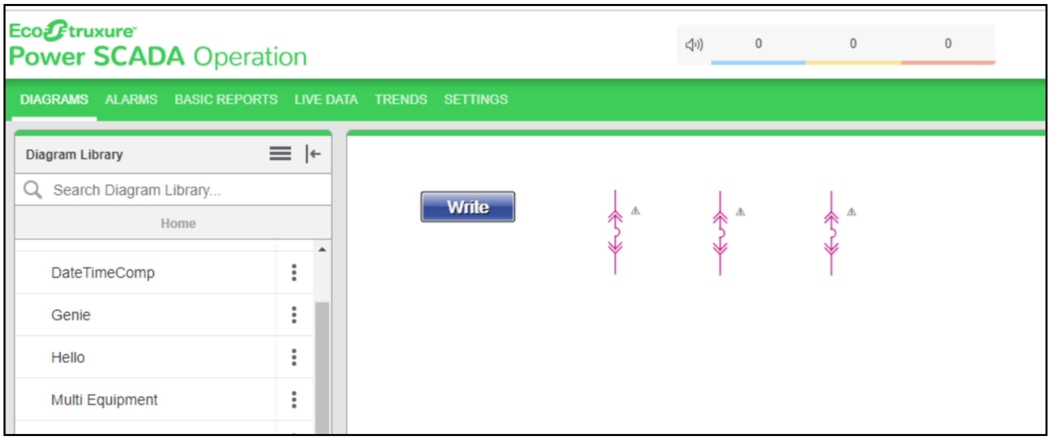
   Upon successful write operation, a **Success** dialog appears:



Control command was successfully performed.

5. Close the dialog.

   All 3 breakers are closed and the write operation is successful:

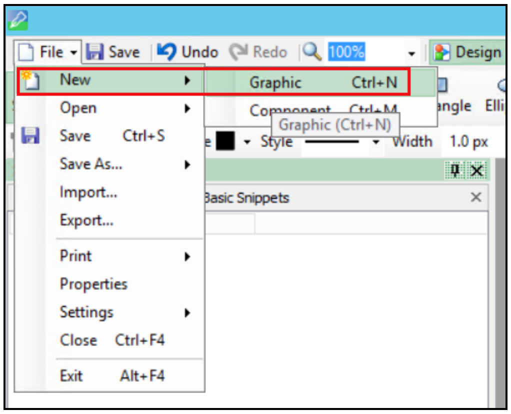


Write and Confirm

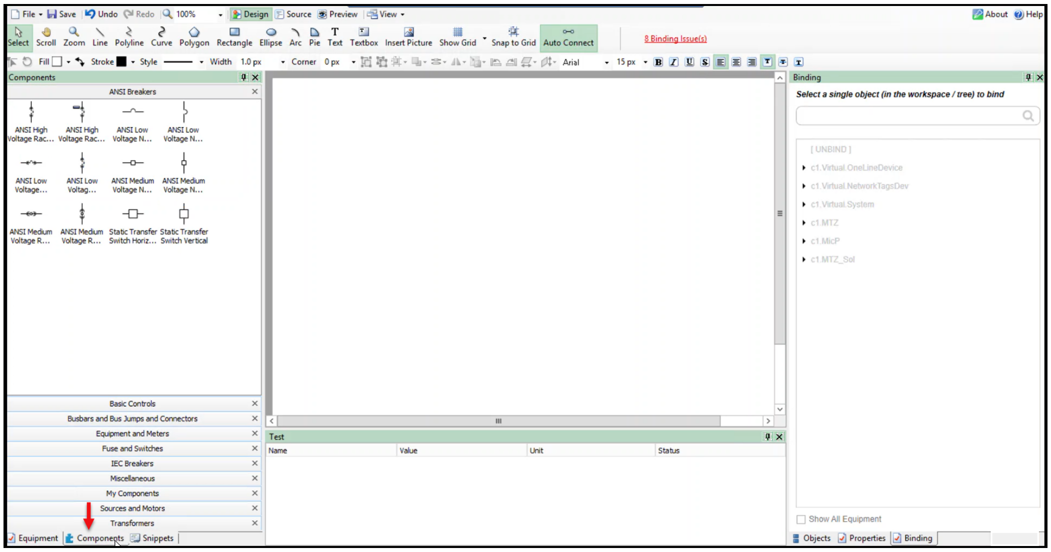
Use the Write and Confirm component to perform breaker operations, On/Off digital outputs, and set CT and PT ratios. Write and Confirm verifies and then confirms that the write operation is either successful or unsuccessful.

To use Write and Confirm:

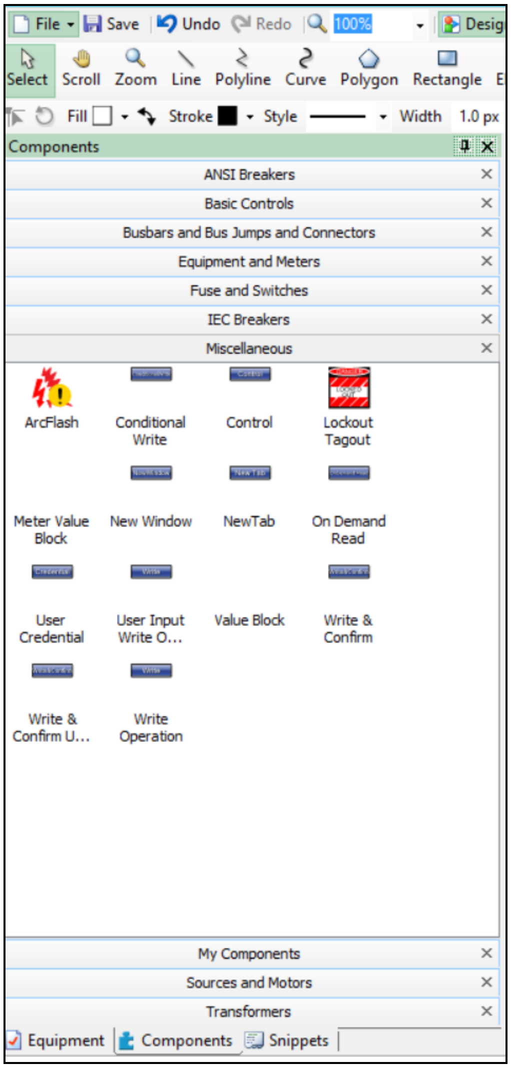
1. In the Graphics Editor, create a new graphics file:



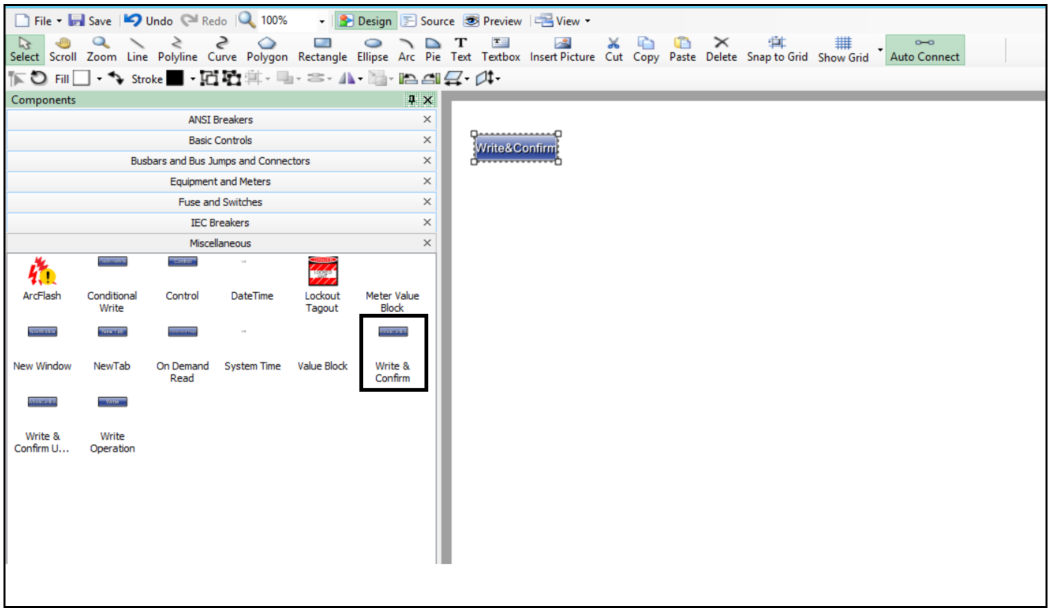
2. In the bottom left corner, click **Components**:



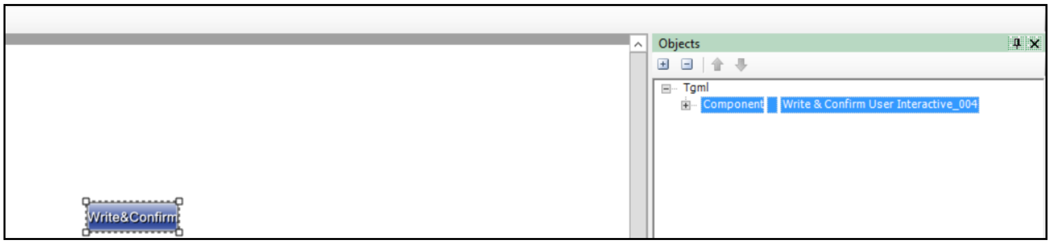
3. Click **Miscellaneous**:



4. Drag and drop the **Write and Confirm** component to the workspace:



5. In the bottom right corner, click **Object**, and then click on the **Component** which is binded under the TGML:



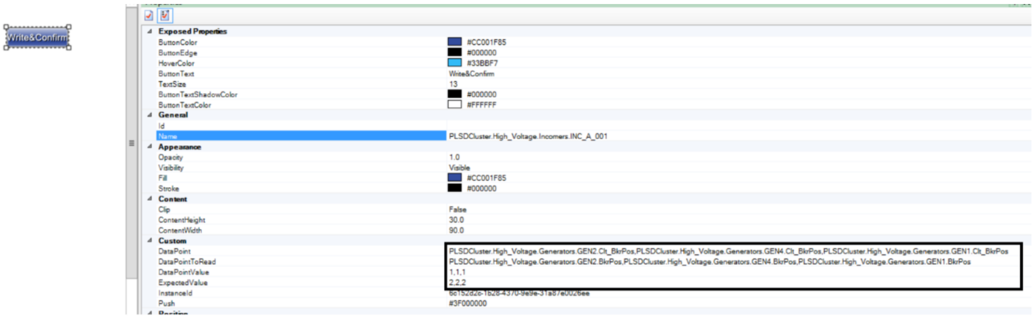
6. Go to **Properties** pane just beside the **Object** pane of the component and enter the following properties based on your requirements:

   a. **DataPoint**: Specify the fully qualified item names to do the write operation. Commas can be used as a delimiter to do the write operation. If only one DataPoint then no comma is required.

   b. **DataPointToRead**: Specify the fully qualified item names to read and verify that the item names are written correctly.

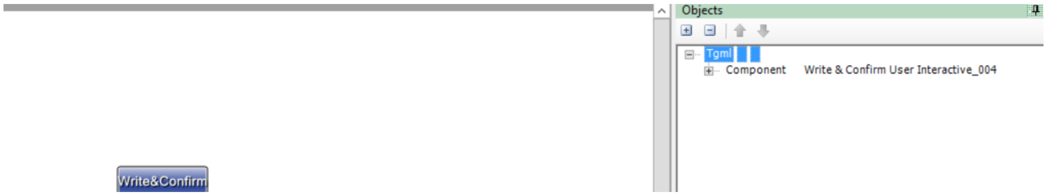
   c. **DatapointValue**:Specify the value to write which was specified in DataPoint field.

d. **Expected Value**: Specify the expected value to verify the final value.

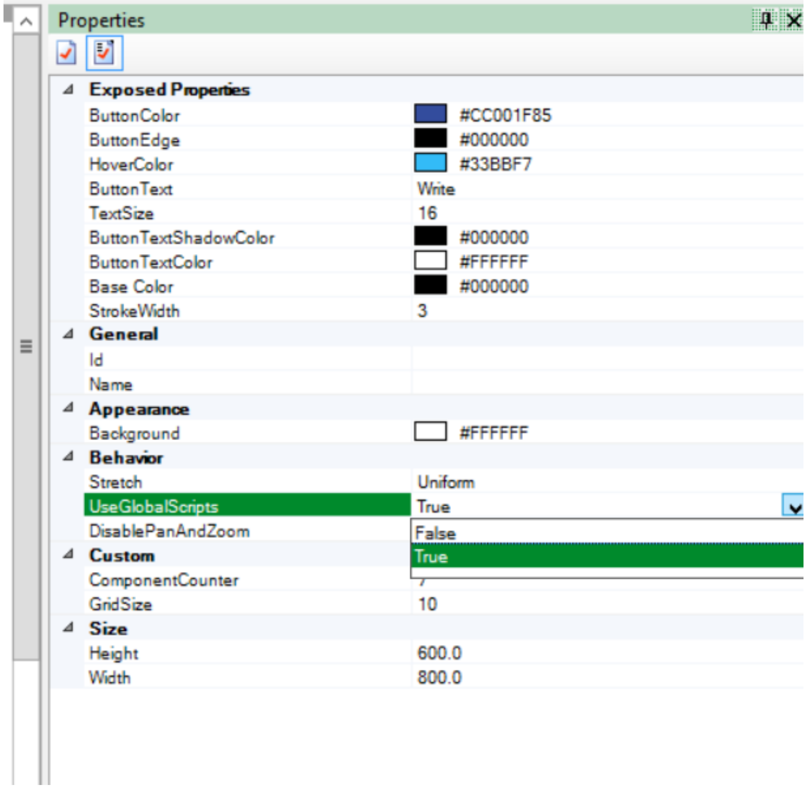


Exposed Properties	
ButtonColor	#CC001F85
ButtonEdge	#000000
HoverColor	#33BBF7
ButtonText	Write&Confirm
TextSize	13
ButtonTextShadowColor	#000000
ButtonTextColor	#FFFFFF
General	
Id	
Name	PLSDCluster.High_Voltage.Incomers.INC_A_001
Appearance	
Opacity	1.0
Visibility	Visible
Fill	#CC001F85
Stroke	#000000
Content	
Clip	False
ContentHeight	30.0
ContentWidth	90.0
Custom	
DataPoint	PLSDCluster.High_Voltage.Generators.GEN2.Clt_BkrPos,PLSDCluster.High_Voltage.Generators.GEN4.Clt_BkrPos,PLSDCluster.High_Voltage.Generators.GEN1.Clt_BkrPos
DataPointToRead	PLSDCluster.High_Voltage.Generators.GEN2.BkrPos,PLSDCluster.High_Voltage.Generators.GEN4.BkrPos,PLSDCluster.High_Voltage.Generators.GEN1.BkrPos
DataPointValue	1,1,1
ExpectedValue	2,2,2
InstanceId	6c152d2c-1b28-4370-9e9e-31a87e0026ee
Push	#3F000000

7. Go to the **Object** pane, and then click on **Tgml**:

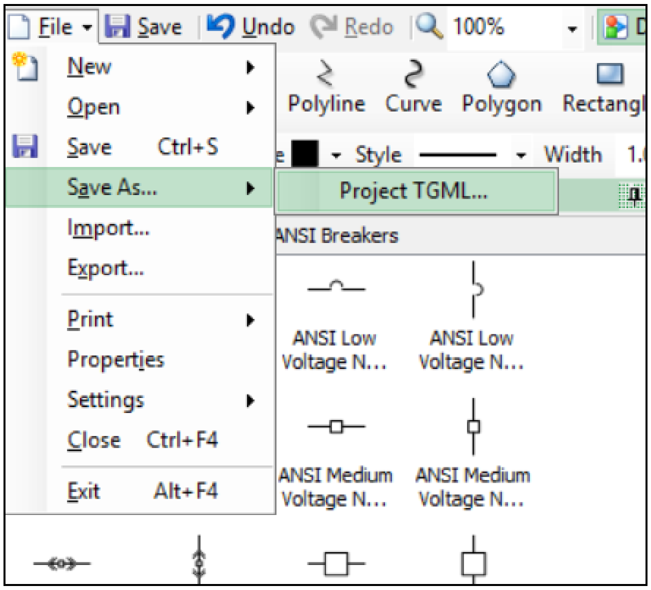


8. Go to **Properties** pane again, and from the **UseGlobalScripts** attribute drop-down select `True`:



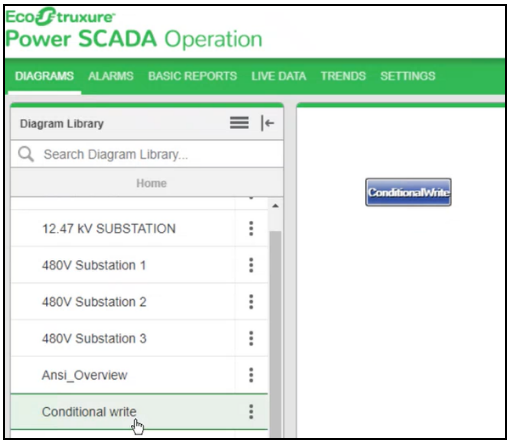
Exposed Properties	
ButtonColor	#CC001F85
ButtonEdge	#000000
HoverColor	#33BBF7
ButtonText	Write
TextSize	16
ButtonTextShadowColor	#000000
ButtonTextColor	#FFFFFF
Base Color	#000000
StrokeWidth	3
General	
Id	
Name	
Appearance	
Background	#FFFFFF
Behavior	
Stretch	Uniform
UseGlobalScripts	True
DisablePanAndZoom	
Custom	
ComponentCounter	7
GridSize	10
Size	
Height	600.0
Width	800.0

9. Go to **File > Save As > Project TGML**.

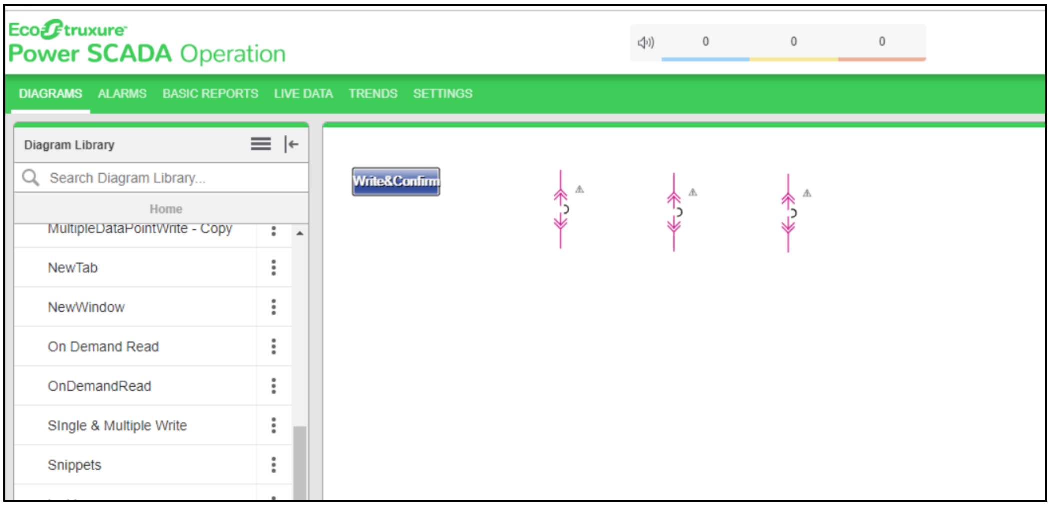


Test the changes:

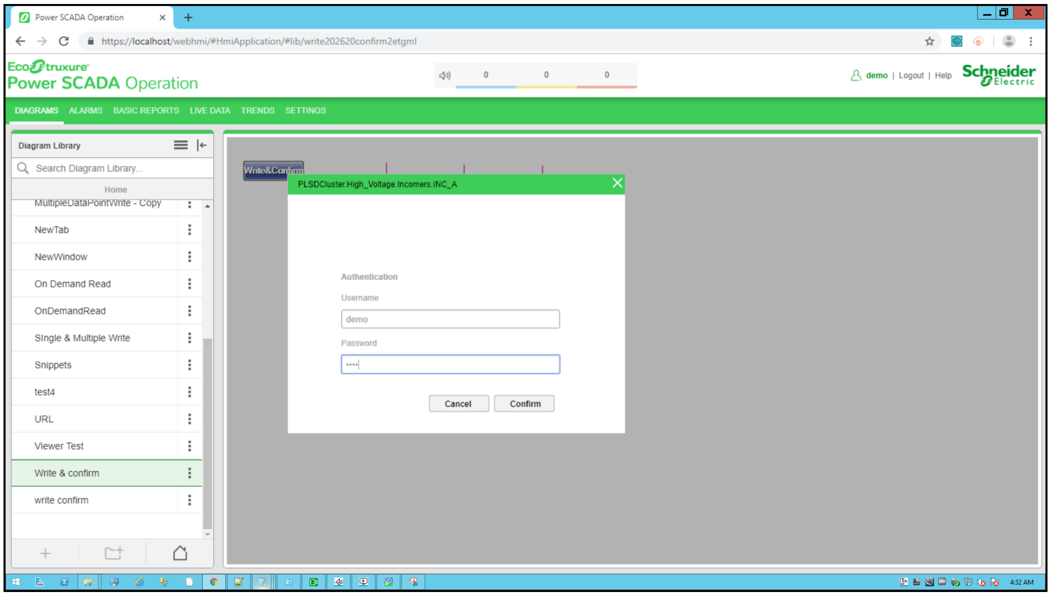
1. Log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**).
2. Click on the **Write and Confirm** component created from the **Diagram Library**:



For example, verify if the breakers operation is working properly in this feature.



3. Click on the **Write and Confirm** component. You are prompted to enter your user credentials:



PLSDCluster.High_Voltage.Incomers.INC_A

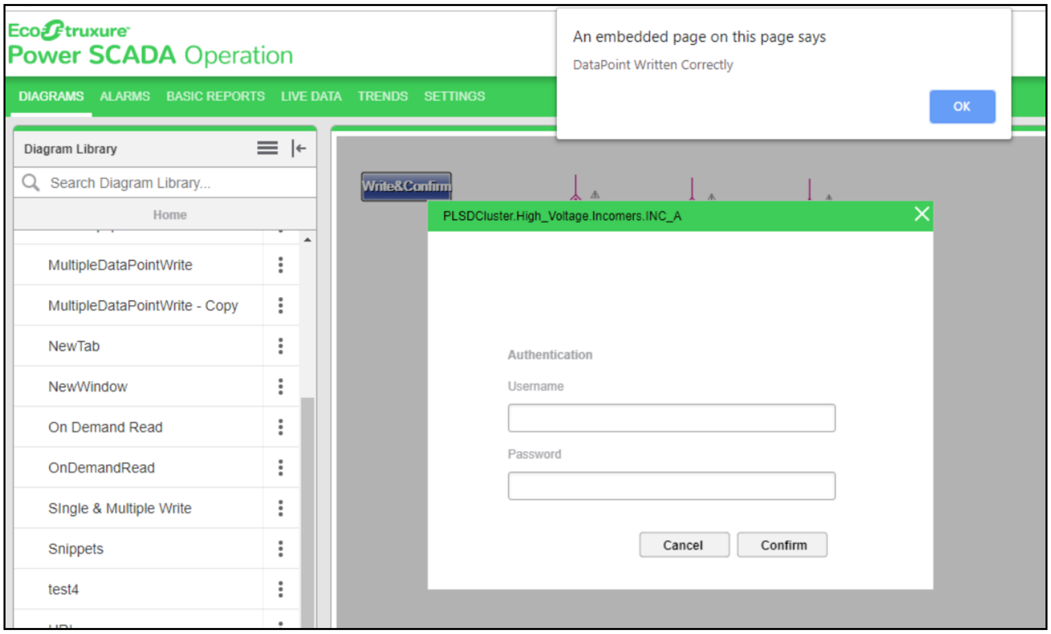
Authentication

Username: demo

Password: ****

4. Enter your username and password, and then click **Confirm**. To control which components require authentication, see Turning off credential requirements for control components.

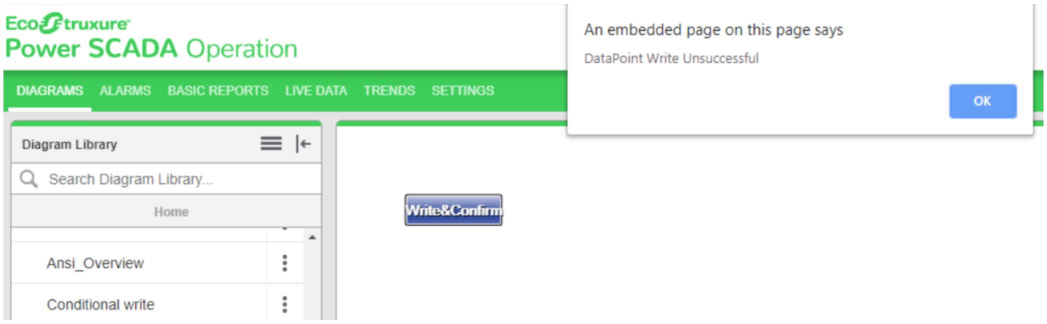
   If the write operation is successful, the **DataPoint Written Correctly** message appears:



An embedded page on this page says

DataPoint Written Correctly

If the operation is not successful, the **DataPoint Wrtiten Unsuccessful** message appears:



An embedded page on this page says

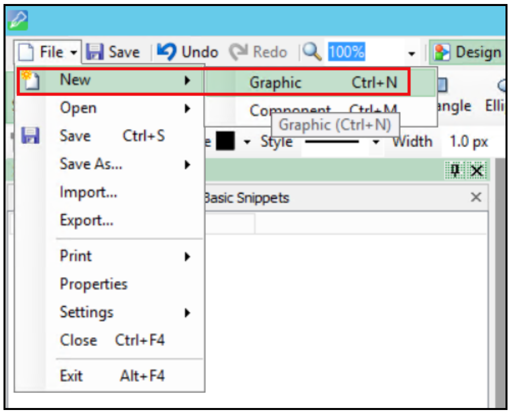
DataPoint Write Unsuccessful

Write and Confirm User Interactive

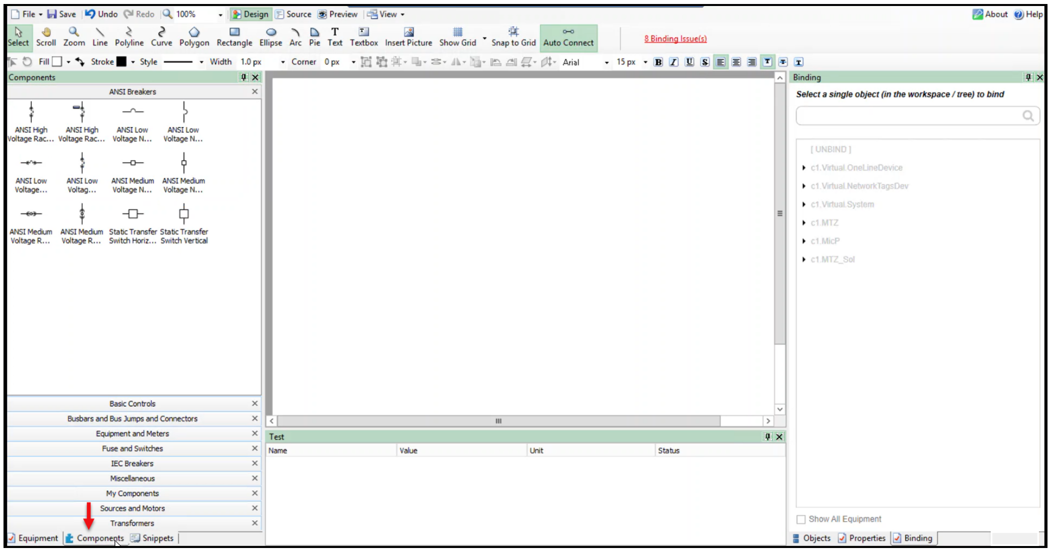
Use the Write and Confirm User Interactive component to perform breaker operations, On/Off digital outputs, set CT and PT ratios, and alarm set-points. This component prompts the user for an input (DataPoint) value, and then checks and confirms whether the value is written properly in the respective register.

To use Write and Confirm User Interactive:

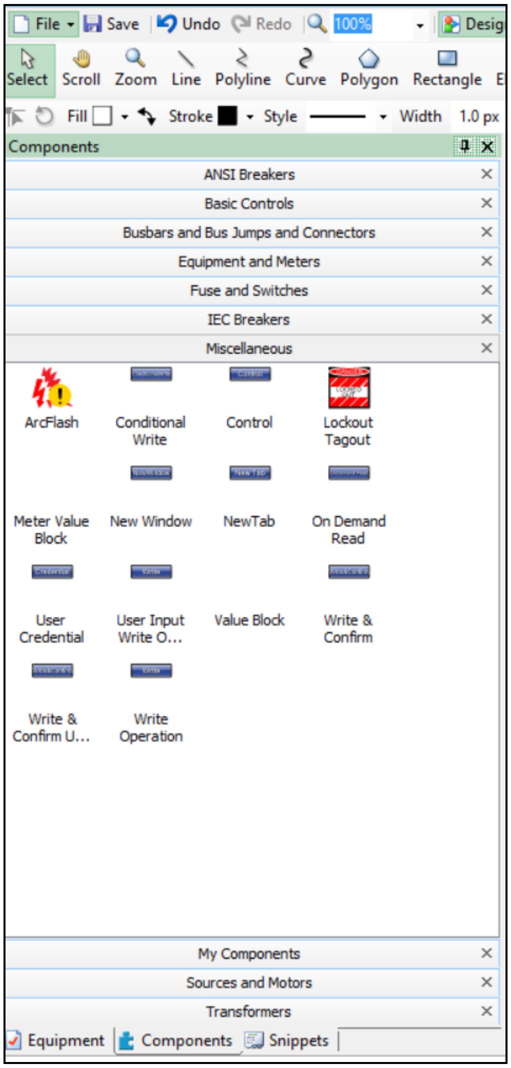
1. In the Graphics Editor, create a new graphic file:



2. In the bottom left corner, click **Components**:

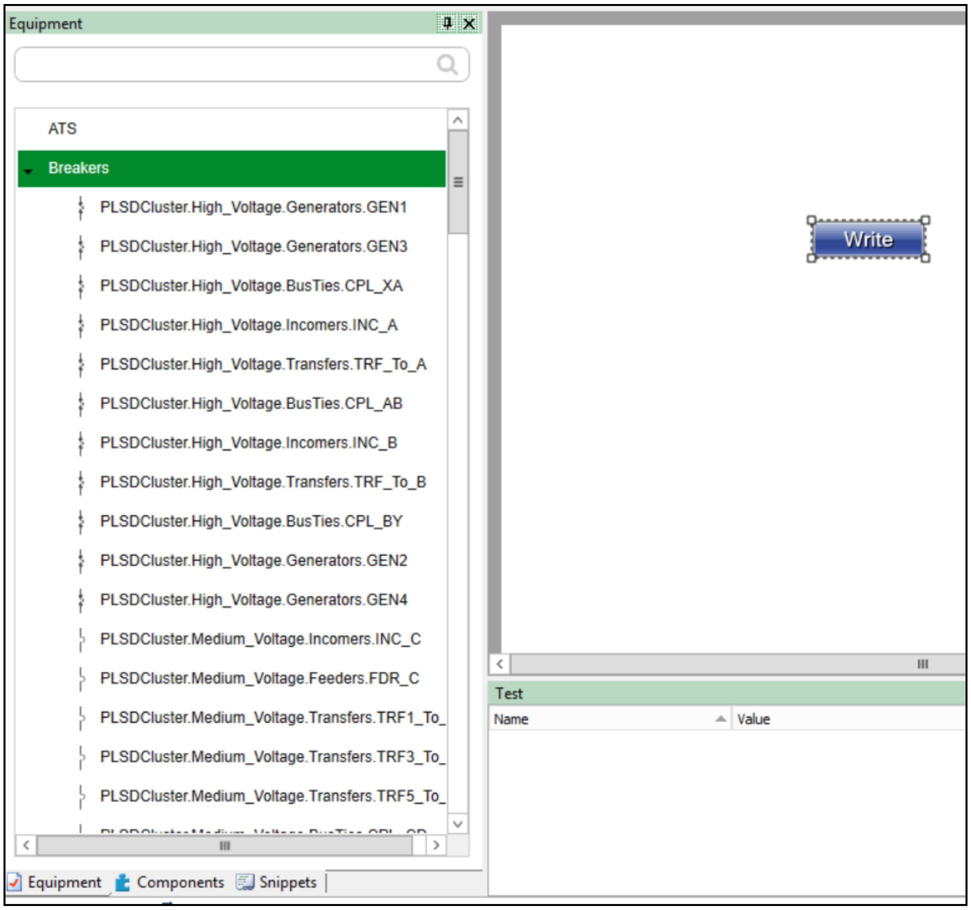


3. Expand **Miscellaneous**:



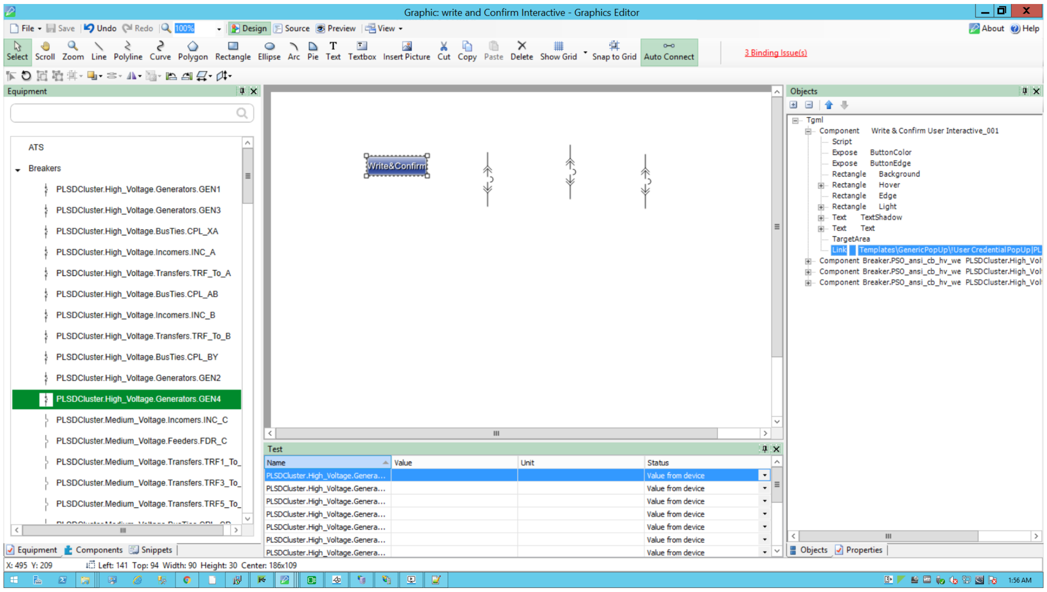
4. Drag the **Write and Confirm User Interactive** component to the workspace.

5. In the bottom left corner, click **Equipment**, and then expand **Breakers**:


Equipment
ATS
Breakers
PLSDCluster.High_Voltage.Generators.GEN1
PLSDCluster.High_Voltage.Generators.GEN3
PLSDCluster.High_Voltage.BusTies.CPL_XA
PLSDCluster.High_Voltage.Incomers.INC_A
PLSDCluster.High_Voltage.Transfers.TRF_To_A
PLSDCluster.High_Voltage.BusTies.CPL_AB
PLSDCluster.High_Voltage.Incomers.INC_B
PLSDCluster.High_Voltage.Transfers.TRF_To_B
PLSDCluster.High_Voltage.BusTies.CPL_BY
PLSDCluster.High_Voltage.Generators.GEN2
PLSDCluster.High_Voltage.Generators.GEN4
PLSDCluster.Medium_Voltage.Incomers.INC_C
PLSDCluster.Medium_Voltage.Feeders.FDR_C
PLSDCluster.Medium_Voltage.Transfers.TRF1_To_
PLSDCluster.Medium_Voltage.Transfers.TRF3_To_
PLSDCluster.Medium_Voltage.Transfers.TRF5_To_
Equipment
Components
Snippets
Write
Test
Name
Value

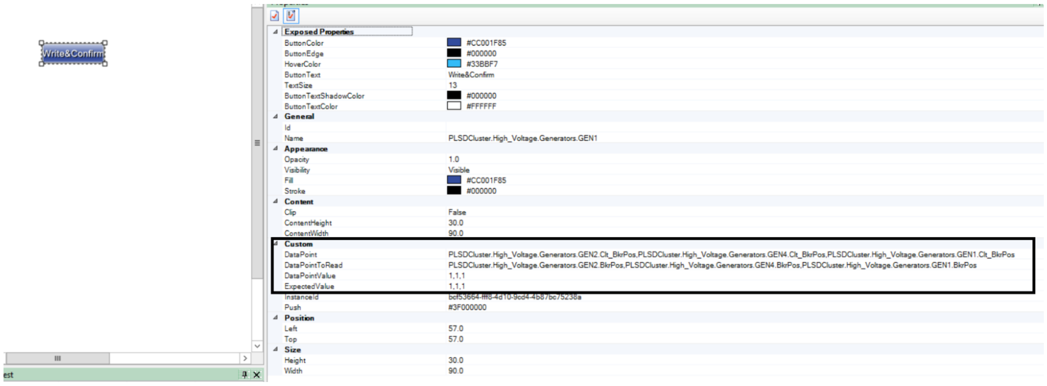
6. Drag and drop any breaker from the list of breakers to the **Graphics Editor** workspace (or multiple breakers based on your requirement).

7. In the bottom left corner, click **Component** to verify that the **Tgml** components are added correctly:


Graphic: write and Confirm Interactive - Graphics Editor
File
Save
Undo
Redo
Design
Source
Preview
View
About
Help
Select
Scroll
Zoom
Line
Polyline
Curve
Polygon
Rectangle
Ellipse
Arc
Pie
Text
Textbox
Insert Picture
Cut
Copy
Paste
Delete
Show Grid
Snap to Grid
Auto Connect
3 Binding Issue(s)
Equipment
ATS
Breakers
PLSDCluster.High_Voltage.Generators.GEN1
PLSDCluster.High_Voltage.Generators.GEN3
PLSDCluster.High_Voltage.BusTies.CPL_XA
PLSDCluster.High_Voltage.Incomers.INC_A
PLSDCluster.High_Voltage.Transfers.TRF_To_A
PLSDCluster.High_Voltage.BusTies.CPL_AB
PLSDCluster.High_Voltage.Incomers.INC_B
PLSDCluster.High_Voltage.Transfers.TRF_To_B
PLSDCluster.High_Voltage.BusTies.CPL_BY
PLSDCluster.High_Voltage.Generators.GEN2
PLSDCluster.High_Voltage.Generators.GEN4
PLSDCluster.Medium_Voltage.Incomers.INC_C
PLSDCluster.Medium_Voltage.Feeders.FDR_C
PLSDCluster.Medium_Voltage.Transfers.TRF1_To_
PLSDCluster.Medium_Voltage.Transfers.TRF3_To_
PLSDCluster.Medium_Voltage.Transfers.TRF5_To_
Equipment
Components
Snippets
Write&Confirm
Test
Name
Value
Unit
Status
Objects
Tgml
Component Write & Confirm User Interactive_001
Script
Expose ButtonColor
Expose ButtonEdge
Rectangle Background
Rectangle Hover
Rectangle Edge
Rectangle Light
Text TextShadow
Text Text
TargetArea
Link Templates\GenericPopUp\UserCredentialPopUp
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Vol
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Vol
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Vol
Objects
Properties
X: 495 Y: 209
Left: 141 Top: 94 Width: 90 Height: 30 Center: 186x109

Name	Value	Unit	Status
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device

8. In the bottom right corner, click **Properties**:

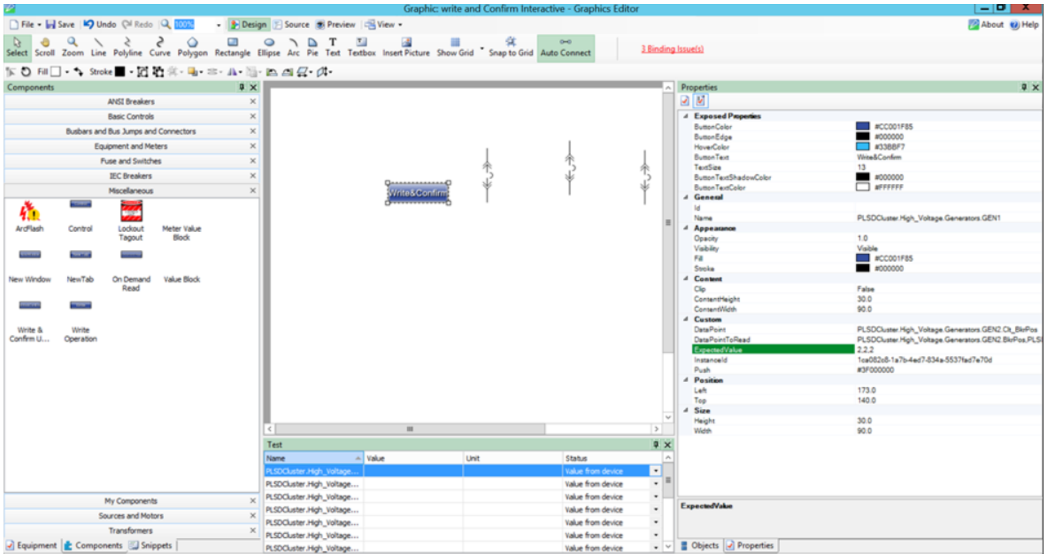


Exposed Properties	
ButtonColor	#CC001F85
ButtonEdge	#000000
HoverColor	#33BBF7
ButtonText	Write&Confirm
TextSize	13
ButtonTextShadowColor	#000000
ButtonTextColor	#FFFFFF
General	
Id	
Name	PLSDCluster.High_Voltage.Generators.GEN1
Appearance	
Opacity	1.0
Visibility	Visible
Fill	#CC001F85
Stroke	#000000
Content	
Clip	False
ContentHeight	30.0
ContentWidth	90.0
Custom	
DataPoint	PLSDCluster.High_Voltage.Generators.GEN2.Clt_BkrPos,PLSDCluster.High_Voltage.Generators.GEN4.Clt_BkrPos,PLSDCluster.High_Voltage.Generators.GEN1.Clt_BkrPos
DataPointToRead	PLSDCluster.High_Voltage.Generators.GEN2.BkrPos,PLSDCluster.High_Voltage.Generators.GEN4.BkrPos,PLSDCluster.High_Voltage.Generators.GEN1.BkrPos
DataPointValue	1,1,1
ExpectedValue	1,1,1
InstanceId	bcf53664-fff8-4d10-9cd4-4b87bc75238a
Push	#3F000000
Position	
Left	57.0
Top	57.0
Size	
Height	30.0
Width	90.0

The **DataPoint** attribute is available in the **Custom** section.

9. Add the fully qualified **DataPoint** names.

NOTE: Use a comma to add multiple **DataPoint** values.



Exposed Properties	
ButtonColor	#CC001F85
ButtonEdge	#000000
HoverColor	#33BBF7
ButtonText	Write&Confirm
TextSize	13
ButtonTextShadowColor	#000000
ButtonTextColor	#FFFFFF
General	
Id	
Name	PLSDCluster.High_Voltage.Generators.GEN1
Appearance	
Opacity	1.0
Visibility	Visible
Fill	#CC001F85
Stroke	#000000
Content	
Clip	False
ContentHeight	30.0
ContentWidth	90.0
Custom	
DataPoint	PLSDCluster.High_Voltage.Generators.GEN2.Clt_BkrPos
DataPointToRead	PLSDCluster.High_Voltage.Generators.GEN2.BkrPos,PLS
ExpectedValue	2,2,2
InstanceId	1ca082c8-1a7b-4ed7-834a-5537fad7e70d
Push	#3F000000
Position	
Left	173.0
Top	140.0
Size	
Height	30.0
Width	90.0

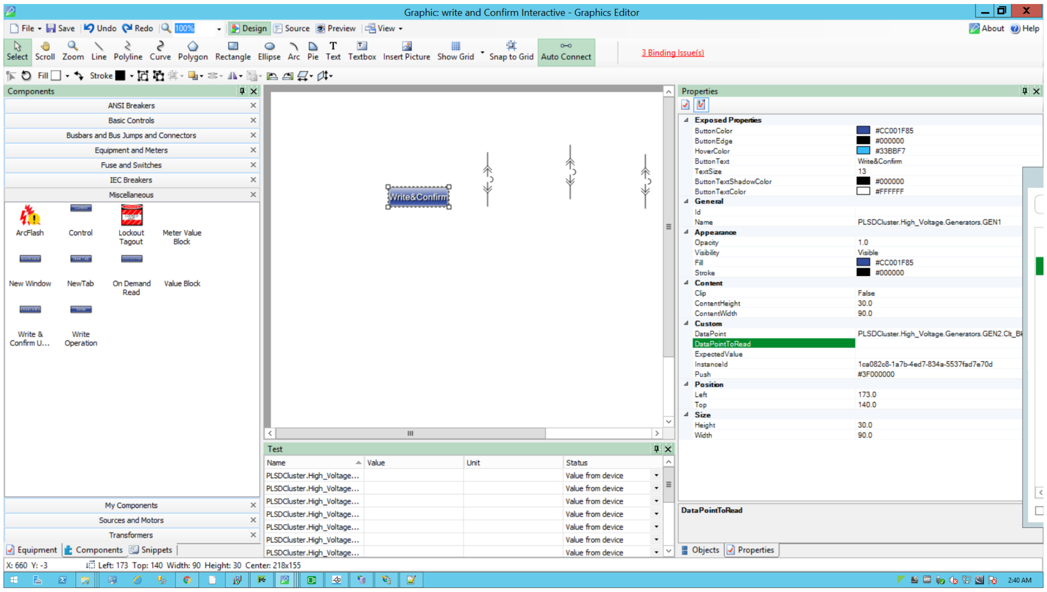
ExpectedValue

Test

Name	Value	Unit	Status
PLSDCluster.High_Voltage...			Value from device
PLSDCluster.High_Voltage...			Value from device
PLSDCluster.High_Voltage...			Value from device
PLSDCluster.High_Voltage...			Value from device
PLSDCluster.High_Voltage...			Value from device
PLSDCluster.High_Voltage...			Value from device
PLSDCluster.High_Voltage...			Value from device

10. Add the fully qualified **DataPointToRead** names.

NOTE: Use a comma to add multiple **DataPointToRead** values.



Properties

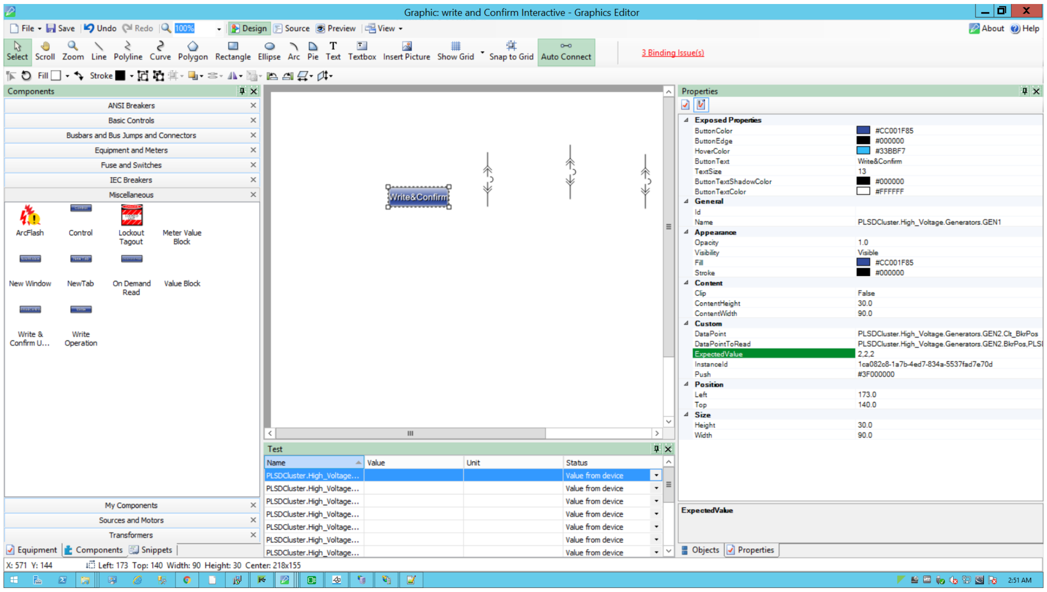
Exposed Properties	
ButtonColor	#CC001F85
ButtonEdge	#000000
HoverColor	#33BBF7
ButtonText	Write&Confirm
TextSize	13
ButtonTextShadowColor	#000000
ButtonTextColor	#FFFFFF
General	
Id	
Name	PLSDCluster.High_Voltage.Generators.GEN1
Appearance	
Opacity	1.0
Visibility	Visible
Fill	#CC001F85
Stroke	#000000
Content	
Clip	False
ContentHeight	30.0
ContentWidth	90.0
Custom	
DataPoint	PLSDCluster.High_Voltage.Generators.GEN2.Clt_Bk
DataPointToRead	
ExpectedValue	
InstanceId	1ca082c8-1a7b-4ed7-834a-5537fad7e70d
Push	#3F000000
Position	
Left	173.0
Top	140.0
Size	
Height	30.0
Width	90.0

DataPointToRead

Name	Value	Unit	Status
PLSDCluster.High_Voltage...			Value from device
PLSDCluster.High_Voltage...			Value from device
PLSDCluster.High_Voltage...			Value from device
PLSDCluster.High_Voltage...			Value from device
PLSDCluster.High_Voltage...			Value from device
PLSDCluster.High_Voltage...			Value from device
PLSDCluster.High_Voltage...			Value from device

11. Add the fully qualified **ExpectedValue**. For example: ExpectedValue (2,2,2)

NOTE: Use a comma to add multiple **ExpectedValue** values.



Properties

Exposed Properties	
ButtonColor	#CC001F85
ButtonEdge	#000000
HoverColor	#33BBF7
ButtonText	Write&Confirm
TextSize	13
ButtonTextShadowColor	#000000
ButtonTextColor	#FFFFFF
General	
Id	
Name	PLSDCluster.High_Voltage.Generators.GEN1
Appearance	
Opacity	1.0
Visibility	Visible
Fill	#CC001F85
Stroke	#000000
Content	
Clip	False
ContentHeight	30.0
ContentWidth	90.0
Custom	
DataPoint	PLSDCluster.High_Voltage.Generators.GEN2.Clt_BkrPos
DataPointToRead	PLSDCluster.High_Voltage.Generators.GEN2.BkrPos,PLSI
ExpectedValue	2,2,2
InstanceId	1ca082c8-1a7b-4ed7-834a-5537fad7e70d
Push	#3F000000
Position	
Left	173.0
Top	140.0
Size	
Height	30.0
Width	90.0

ExpectedValue

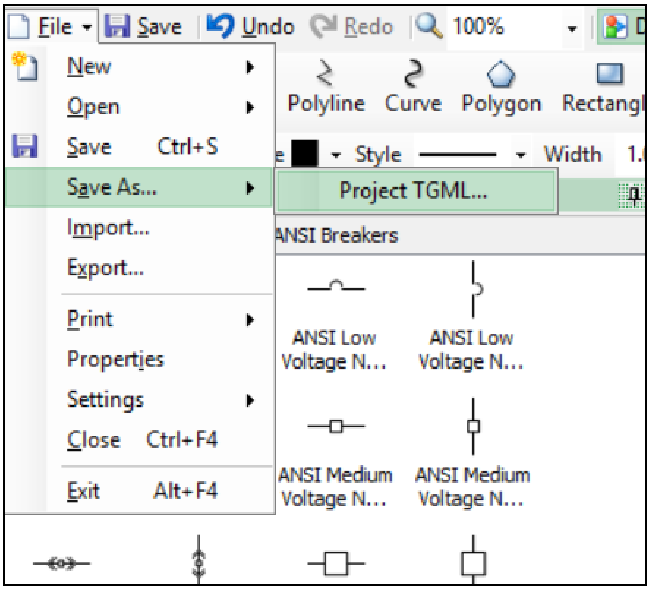
Name	Value	Unit	Status
PLSDCluster.High_Voltage...			Value from device
PLSDCluster.High_Voltage...			Value from device
PLSDCluster.High_Voltage...			Value from device
PLSDCluster.High_Voltage...			Value from device
PLSDCluster.High_Voltage...			Value from device
PLSDCluster.High_Voltage...			Value from device
PLSDCluster.High_Voltage...			Value from device

NOTE:

To close the breakers, **ExpectedValue** (2,2,2) for three breakers.

To open the breakers, **ExpectedValue** (1,1,1) for three breakers.

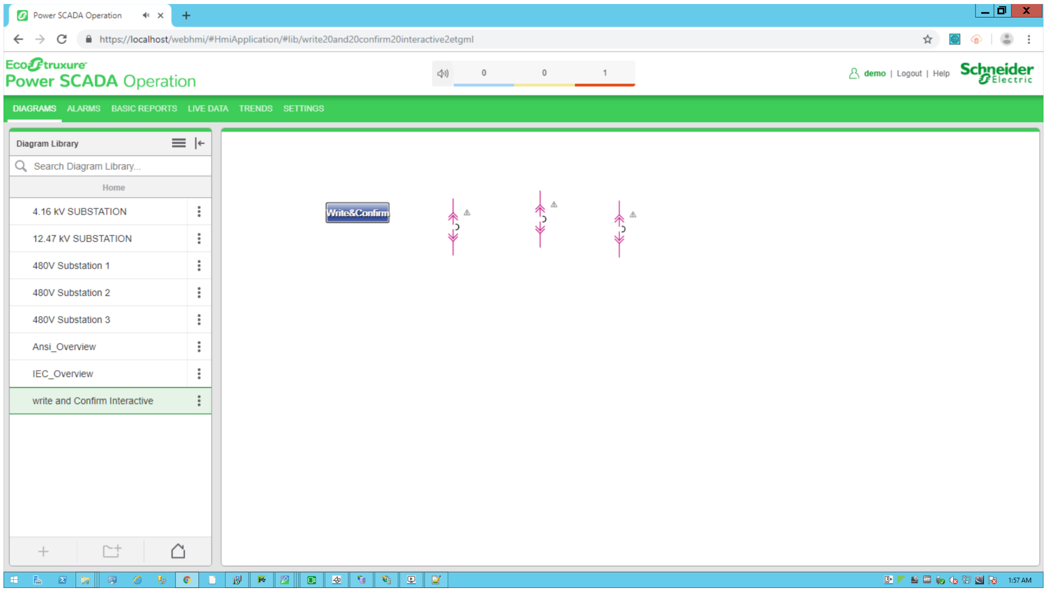
12. Go to **File > Save As > Project TGML**.



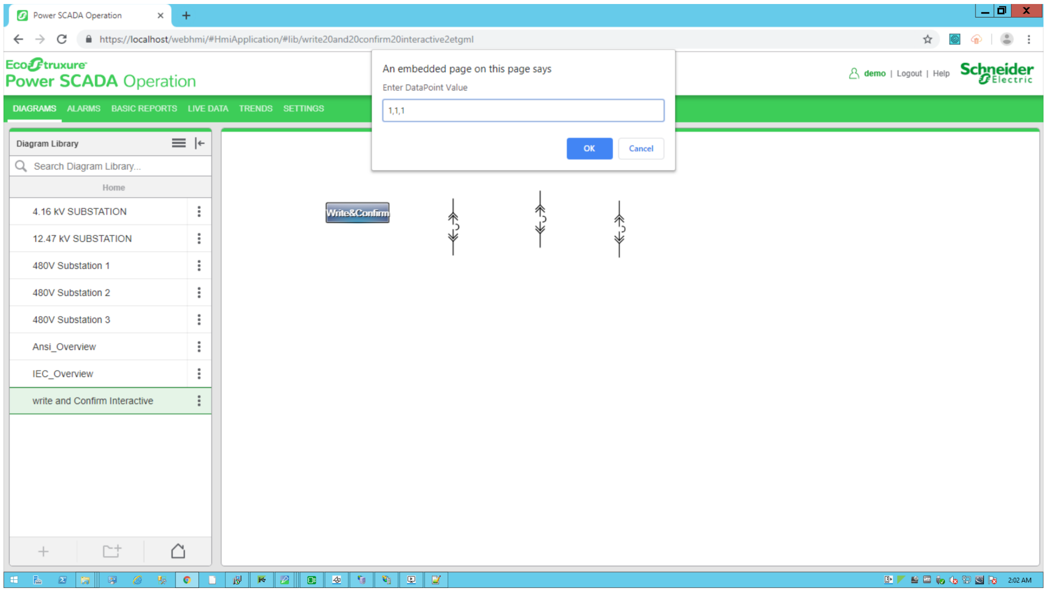
Test the changes:

1. Log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)

2. Click on **Write&ConfirmUserInteractive** component created from the **Diagram Library**:



3. Click on the **Write&Confirm** operation component, type the **DataPoint** value (for example, 1,1,1) in the displayed popup, and then click **OK**.



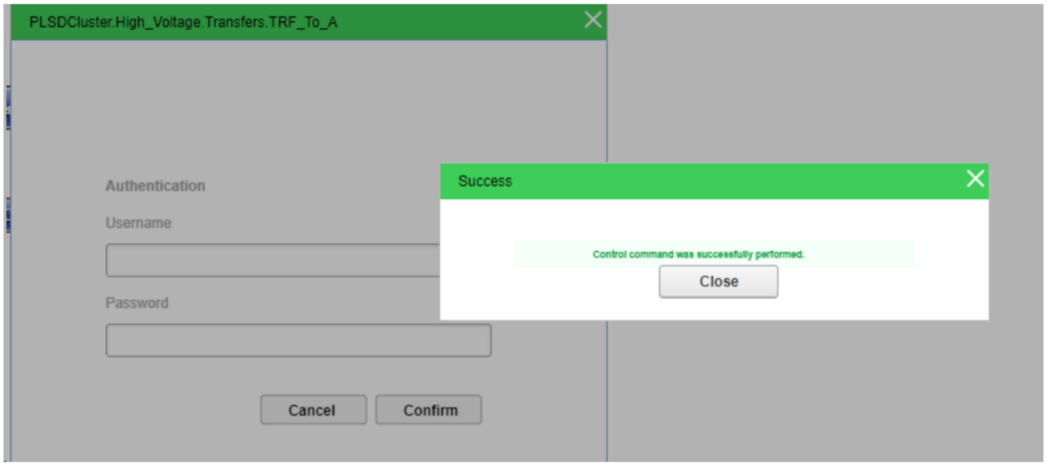
An embedded page on this page says

Enter DataPoint Value

1,1,1

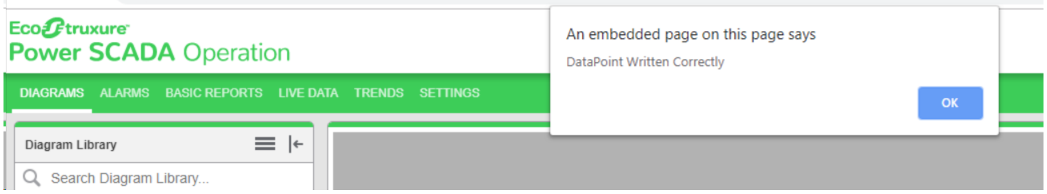
4. Type the **Username** and **Password**, and then click **Confirm**. To control which components require authentication, see Turning off credential requirements for control components.

   The success popup message is displayed:



Control command was successfully performed.

5. Click **Close**. The following pop up appears:


EcoStruxure
Power SCADA Operation
DIAGRAMS ALARMS BASIC REPORTS LIVE DATA TRENDS SETTINGS
Diagram Library
Search Diagram Library...
OK

An embedded page on this page says

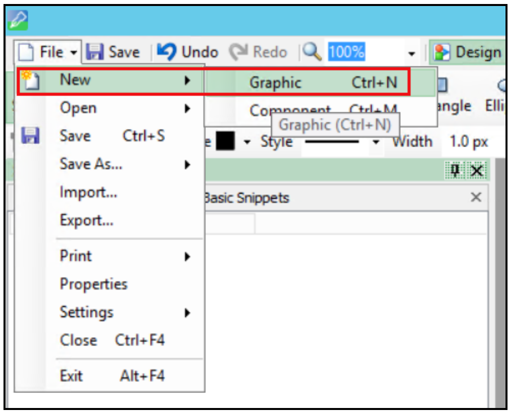
DataPoint Written Correctly

User Input Write Operation

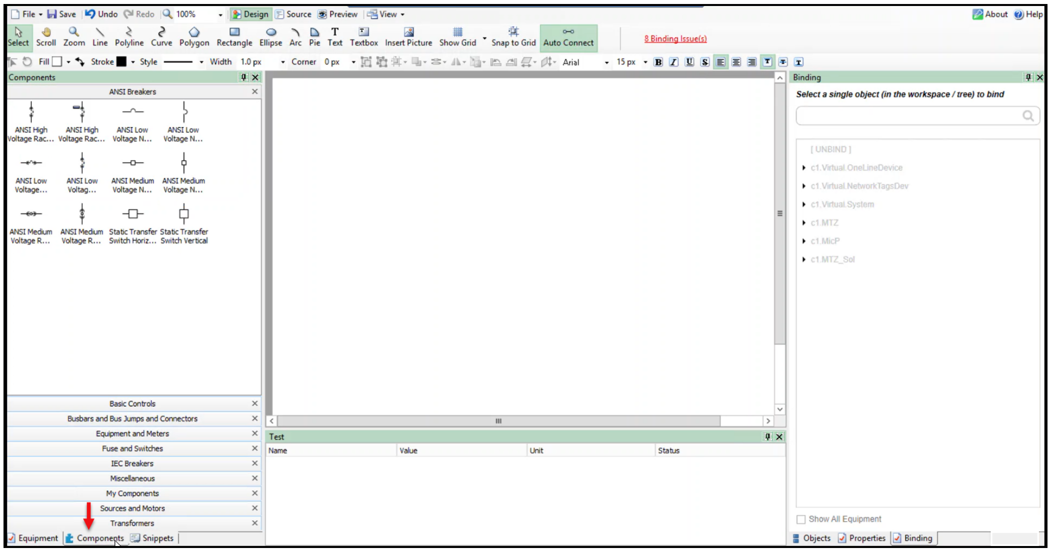
Use the User Input Write component to write set-point register values by prompting the user to input the value.

To use User Write Input Write Operation:

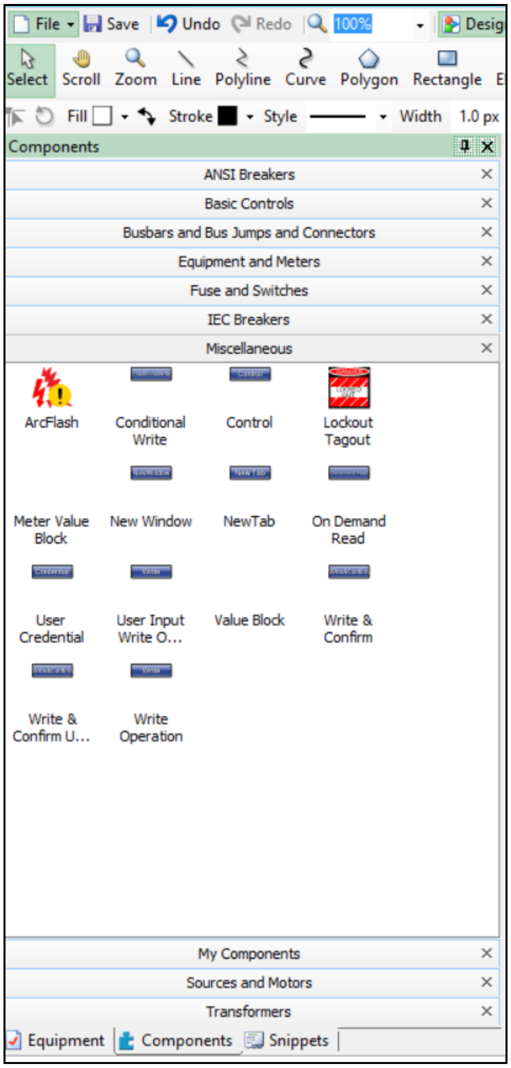
1. In the Graphics Editor, create a new graphics file:



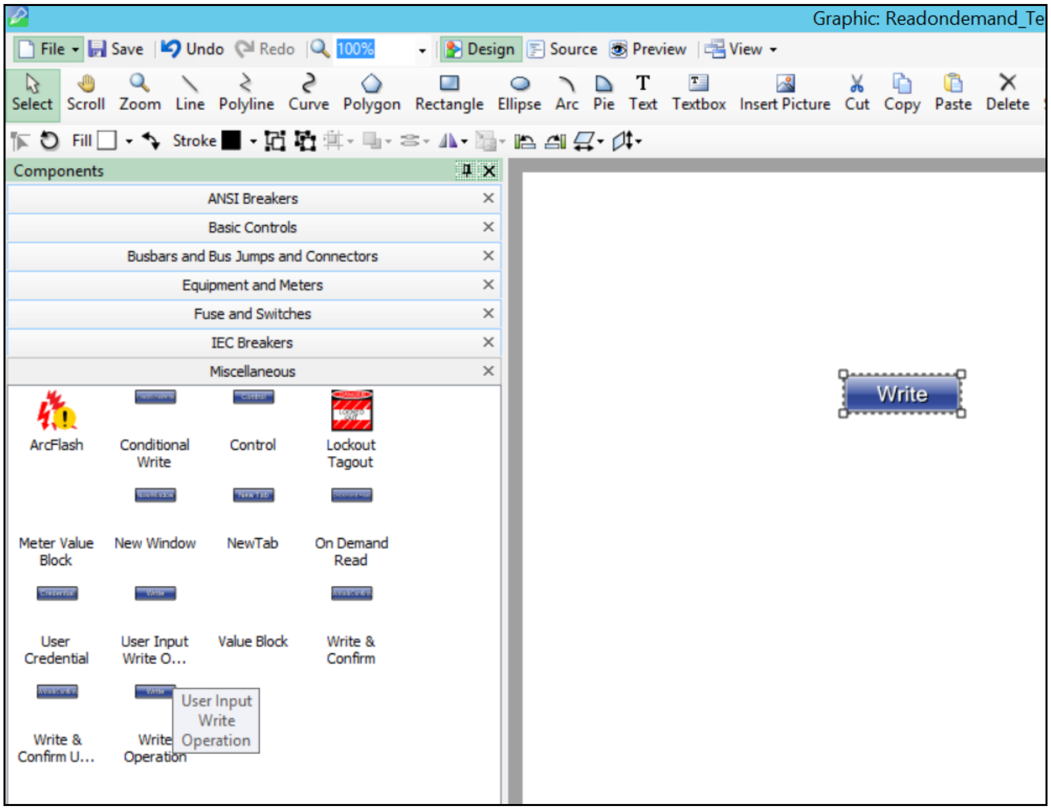
2. In the bottom right corner, go to **Components**:



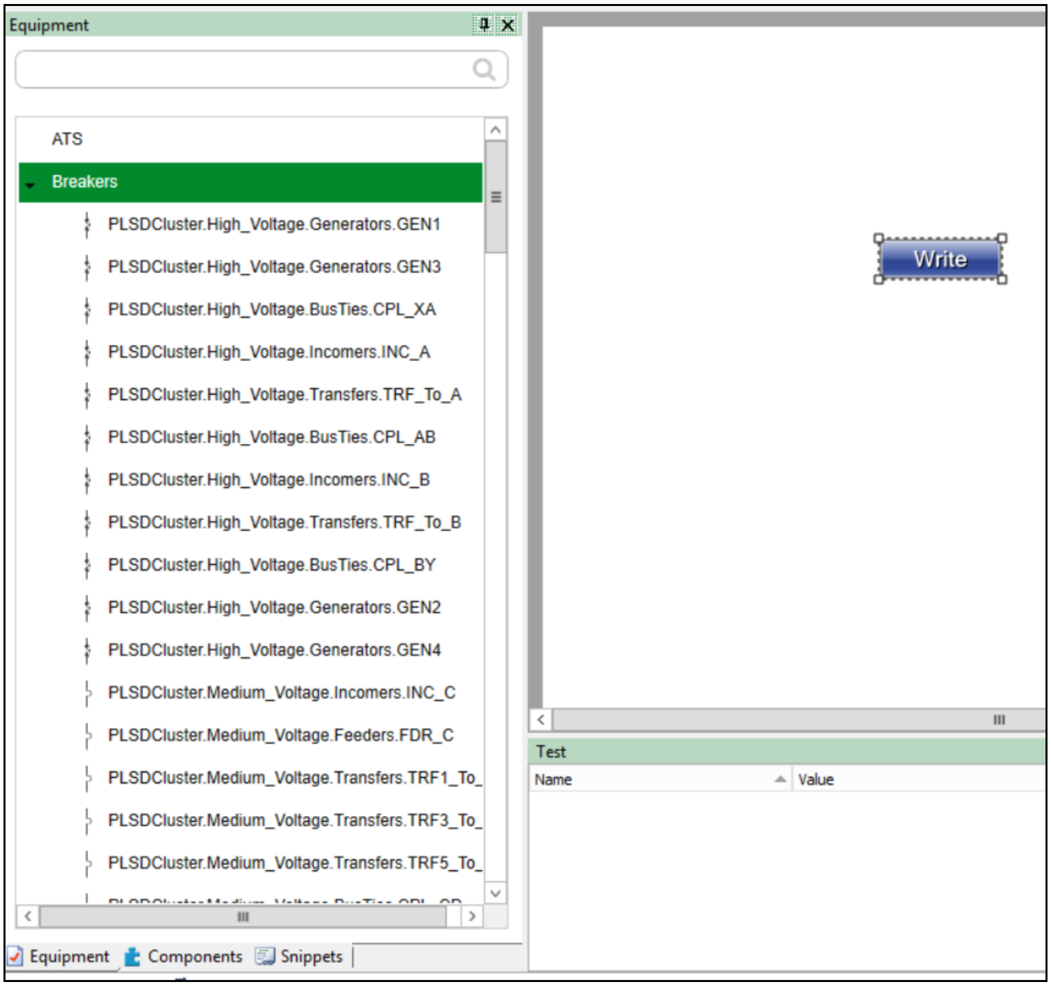
3. In the **Components** pane, click on the **Miscellaneous** tab:



4. Drag and drop the **User Input Write Operation** to the workspace:

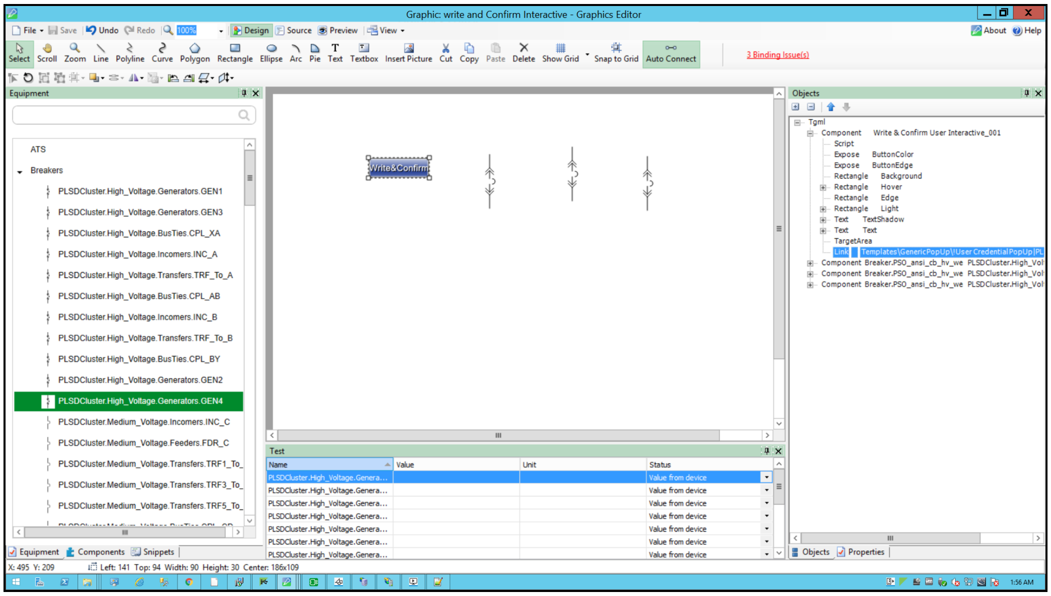


5. In the bottom left corner, go to the **Equipment** pane, and then click **Breakers**:


Equipment
ATS
Breakers
PLSDCluster.High_Voltage.Generators.GEN1
PLSDCluster.High_Voltage.Generators.GEN3
PLSDCluster.High_Voltage.BusTies.CPL_XA
PLSDCluster.High_Voltage.Incomers.INC_A
PLSDCluster.High_Voltage.Transfers.TRF_To_A
PLSDCluster.High_Voltage.BusTies.CPL_AB
PLSDCluster.High_Voltage.Incomers.INC_B
PLSDCluster.High_Voltage.Transfers.TRF_To_B
PLSDCluster.High_Voltage.BusTies.CPL_BY
PLSDCluster.High_Voltage.Generators.GEN2
PLSDCluster.High_Voltage.Generators.GEN4
PLSDCluster.Medium_Voltage.Incomers.INC_C
PLSDCluster.Medium_Voltage.Feeders.FDR_C
PLSDCluster.Medium_Voltage.Transfers.TRF1_To_
PLSDCluster.Medium_Voltage.Transfers.TRF3_To_
PLSDCluster.Medium_Voltage.Transfers.TRF5_To_
Equipment
Components
Snippets
Write
Test
Name
Value

6. Drag and drop any breaker from the list of Breakers to the **Graphics Editor** workspace. You can add multiple breakers as per the requirement to demonstrate the write operation using the component.

Example: In the following image, 3 breakers are added:



Equipment:

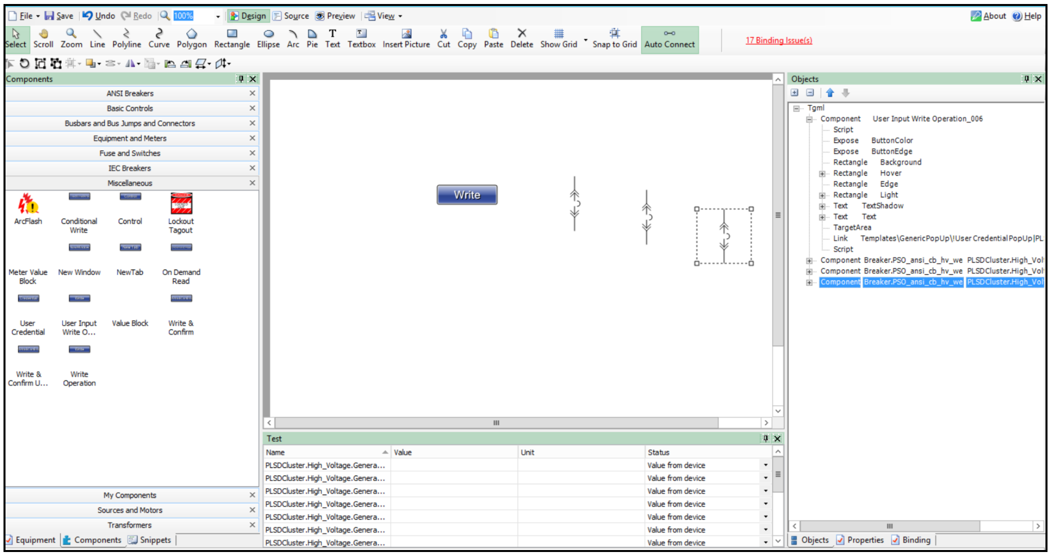
- ATS
- Breakers
  - PLSDCluster.High_Voltage.Generators.GEN1
  - PLSDCluster.High_Voltage.Generators.GEN3
  - PLSDCluster.High_Voltage.BusTies.CPL_XA
  - PLSDCluster.High_Voltage.Incomers.INC_A
  - PLSDCluster.High_Voltage.Transfers.TRF_To_A
  - PLSDCluster.High_Voltage.BusTies.CPL_AB
  - PLSDCluster.High_Voltage.Incomers.INC_B
  - PLSDCluster.High_Voltage.Transfers.TRF_To_B
  - PLSDCluster.High_Voltage.BusTies.CPL_BY
  - PLSDCluster.High_Voltage.Generators.GEN2
  - PLSDCluster.High_Voltage.Generators.GEN4
  - PLSDCluster.Medium_Voltage.Incomers.INC_C
  - PLSDCluster.Medium_Voltage.Feeders.FDR_C
  - PLSDCluster.Medium_Voltage.Transfers.TRF1_To_
  - PLSDCluster.Medium_Voltage.Transfers.TRF3_To_
  - PLSDCluster.Medium_Voltage.Transfers.TRF5_To_

Objects:

```
Tgml
  Component   Write & Confirm User Interactive_001
    Script
    Expose      ButtonColor
    Expose      ButtonEdge
    Rectangle   Background
    Rectangle   Hover
    Rectangle   Edge
    Rectangle   Light
    Text        TextShadow
    Text        Text
    TargetArea
    Link        Templates\GenericPopUp\!User CredentialPopUp|PL
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Vol
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Vol
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Vol
```

Name	Value	Unit	Status
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device

7. In the bottom left corner, click **Component** to verify that the **Tgml** components are added correctly:

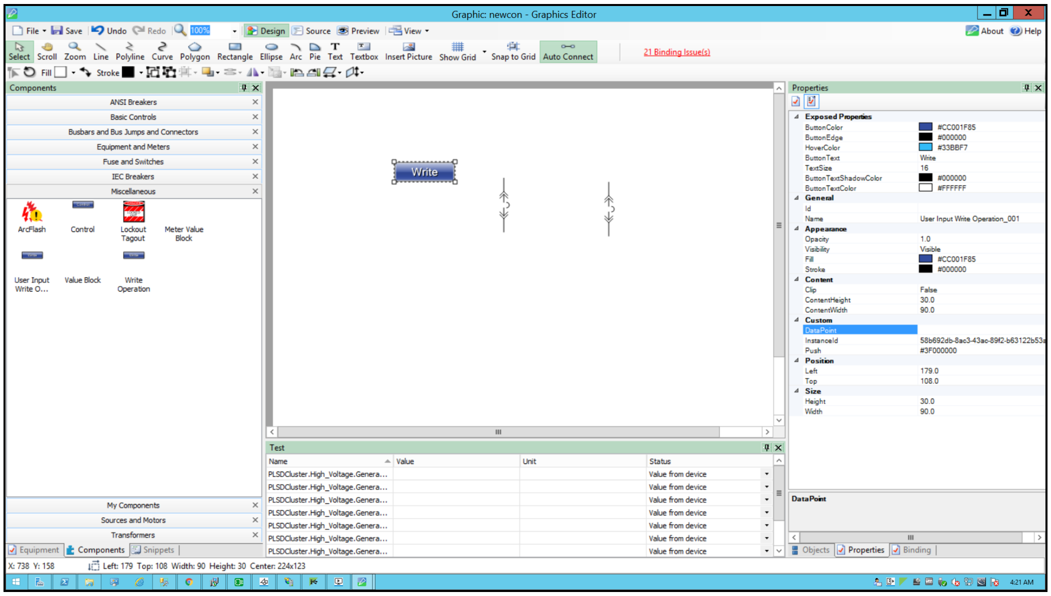


Objects:

```
Tgml
  Component   User Input Write Operation_006
    Script
    Expose      ButtonColor
    Expose      ButtonEdge
    Rectangle   Background
    Rectangle   Hover
    Rectangle   Edge
    Rectangle   Light
    Text        TextShadow
    Text        Text
    TargetArea
    Link        Templates\GenericPopUp\!User CredentialPopUp|PL
    Script
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Vol
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Vol
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Vol
```

Name	Value	Unit	Status
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device

8. In the bottom right corner, click **Properties**:



Properties

Exposed Properties	
ButtonColor	#CC001F85
ButtonEdge	#000000
HoverColor	#33BBF7
ButtonText	Write
TextSize	16
ButtonTextShadowColor	#000000
ButtonTextColor	#FFFFFF
General	
Id	
Name	User Input Write Operation_001
Appearance	
Opacity	1.0
Visibility	Visible
Fill	#CC001F85
Stroke	#000000
Content	
Clip	False
ContentHeight	30.0
ContentWidth	90.0
Custom	
DataPoint	
InstanceId	58b692db-8ac3-43ac-89f2-b63122b53a
Push	#3F000000
Position	
Left	179.0
Top	108.0
Size	
Height	30.0
Width	90.0

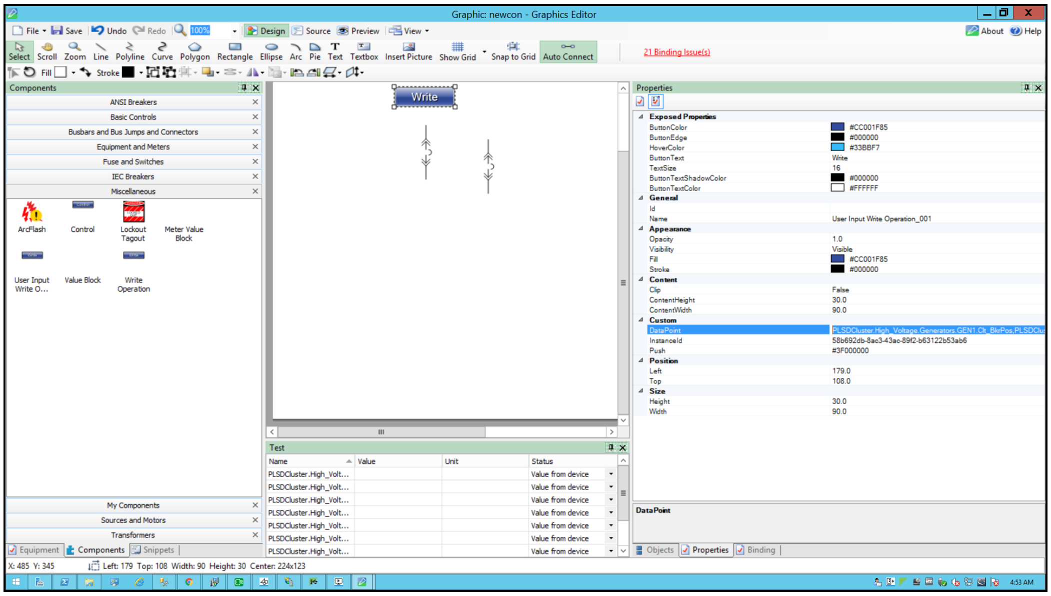
DataPoint

Test

Name	Value	Unit	Status
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device
PLSDCluster.High_Voltage.Genera...			Value from device

X: 738 Y: 158 Left: 179 Top: 108 Width: 90 Height: 30 Center: 224x123

9. Add the fully qualified **DataPoint** names. The **DataPoint** attribute is located in the **Custom** group:



Properties

Exposed Properties	
ButtonColor	#CC001F85
ButtonEdge	#000000
HoverColor	#33BBF7
ButtonText	Write
TextSize	16
ButtonTextShadowColor	#000000
ButtonTextColor	#FFFFFF
General	
Id	
Name	User Input Write Operation_001
Appearance	
Opacity	1.0
Visibility	Visible
Fill	#CC001F85
Stroke	#000000
Content	
Clip	False
ContentHeight	30.0
ContentWidth	90.0
Custom	
DataPoint	PLSDCluster.High_Voltage.Generators.GEN1.Clt_BkrPos,PLSDClu
InstanceId	58b692db-8ac3-43ac-89f2-b63122b53ab6
Push	#3F000000
Position	
Left	179.0
Top	108.0
Size	
Height	30.0
Width	90.0

DataPoint

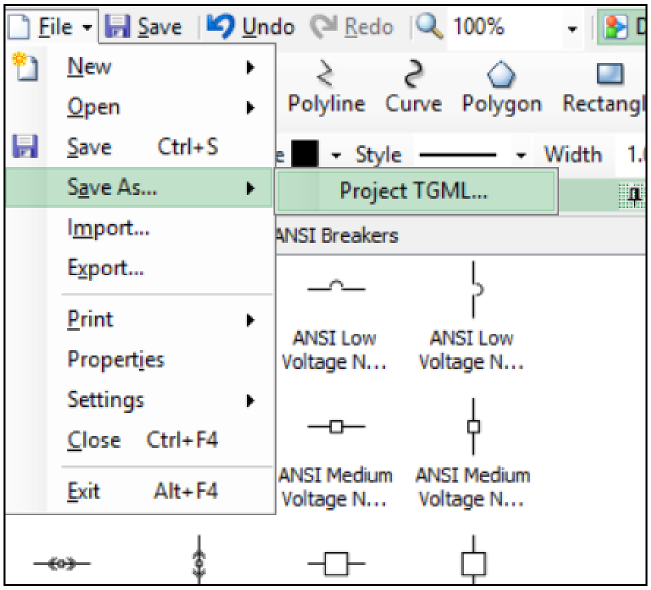
Test

Name	Value	Unit	Status
PLSDCluster.High_Volt...			Value from device
PLSDCluster.High_Volt...			Value from device
PLSDCluster.High_Volt...			Value from device
PLSDCluster.High_Volt...			Value from device
PLSDCluster.High_Volt...			Value from device
PLSDCluster.High_Volt...			Value from device
PLSDCluster.High_Volt...			Value from device

X: 485 Y: 345 Left: 179 Top: 108 Width: 90 Height: 30 Center: 224x123

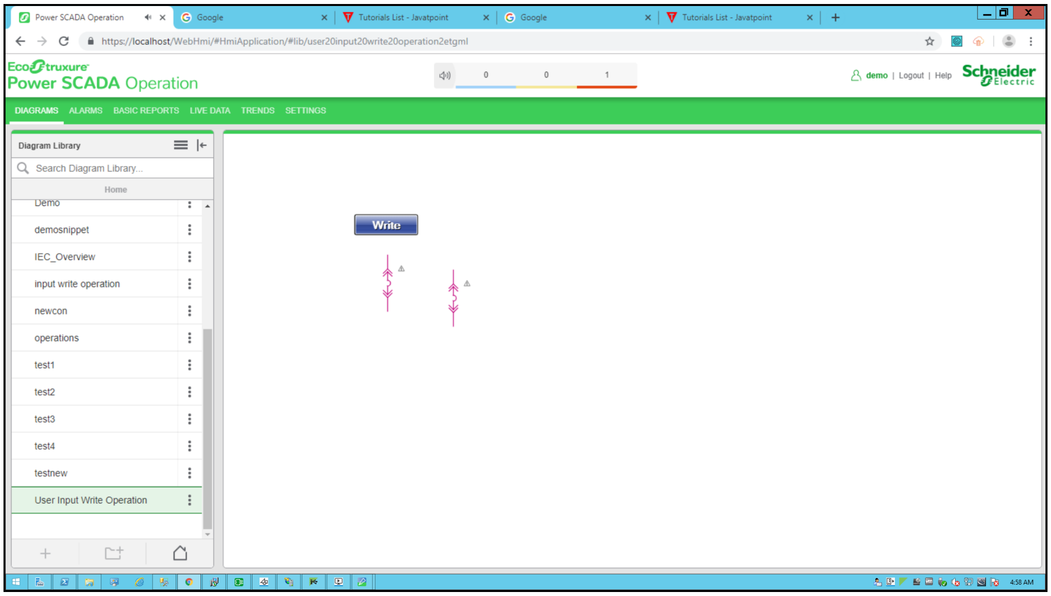
NOTE: Use a comma to add multiple **DataPoint** values.

10. Go to **File > Save As > Project TGML**.

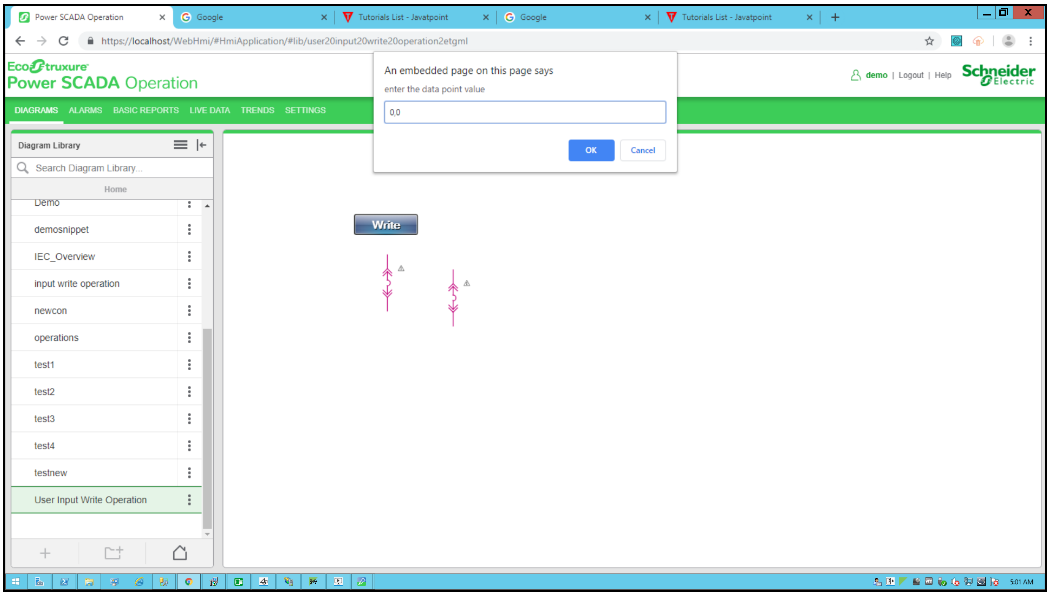


Test the changes:

1. Log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)
2. Click on **UserInputWriteOperation** component created from the **Diagram Library**:



3. Click on **Write** component to do the write operation.
4. For example, the 2 breakers are closed in the below screen. Type the datapoint value from popup to open the breakers and click **OK**. (Datapoint value given is 0,0).

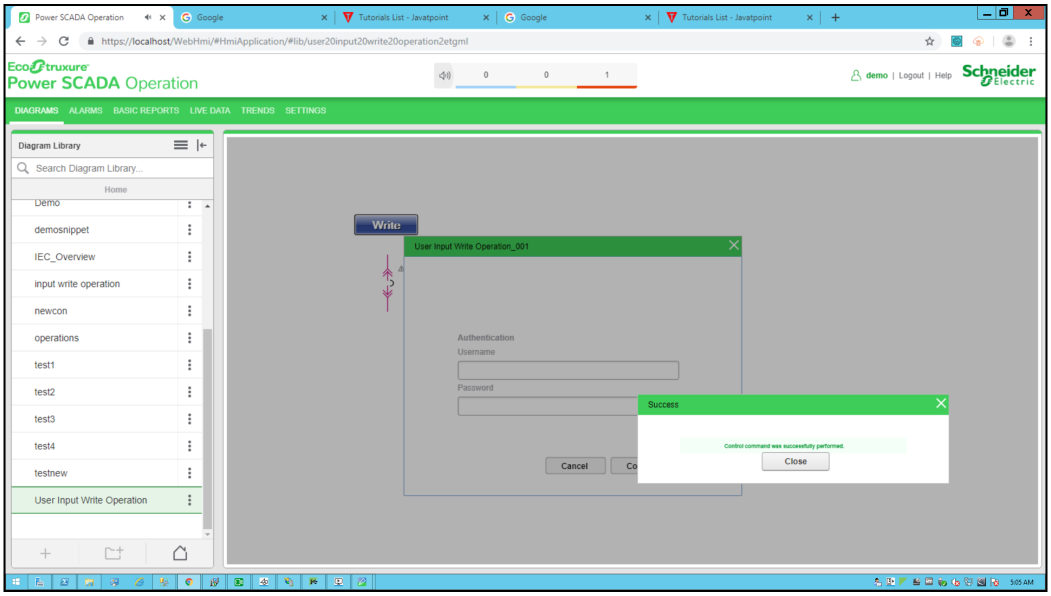


An embedded page on this page says

enter the data point value

0,0

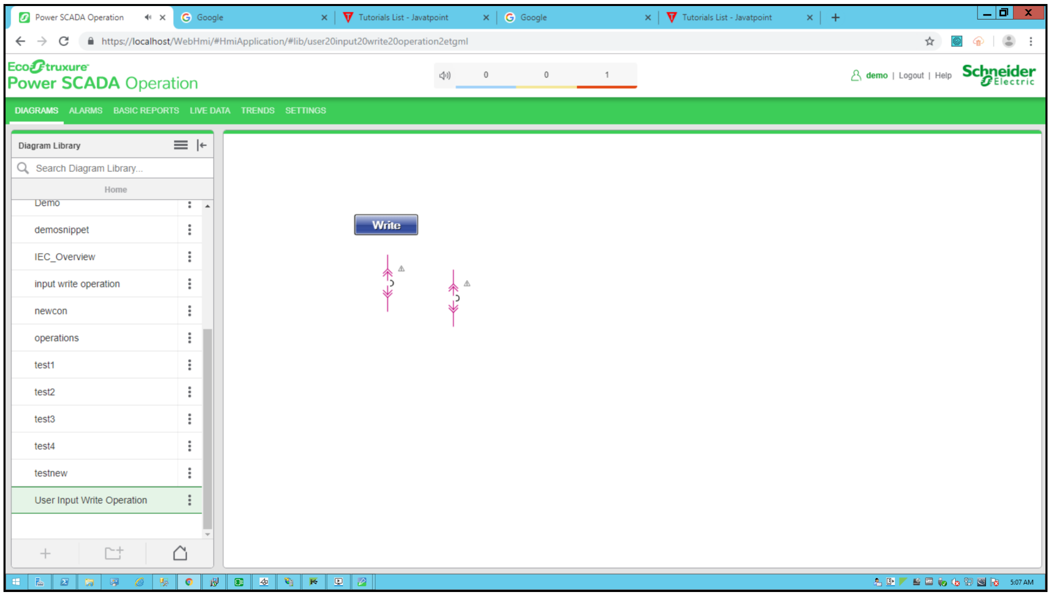
5. Enter your user name and password, and then click **Confirm**. To control which components require authentication, see Turning off credential requirements for control components.



Control command was successfully performed.

6. Close the pop up.

User credential pop up and two breakers are in open status:

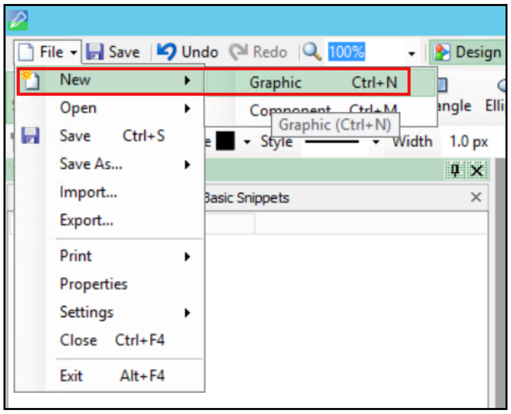


Analog Write Operation

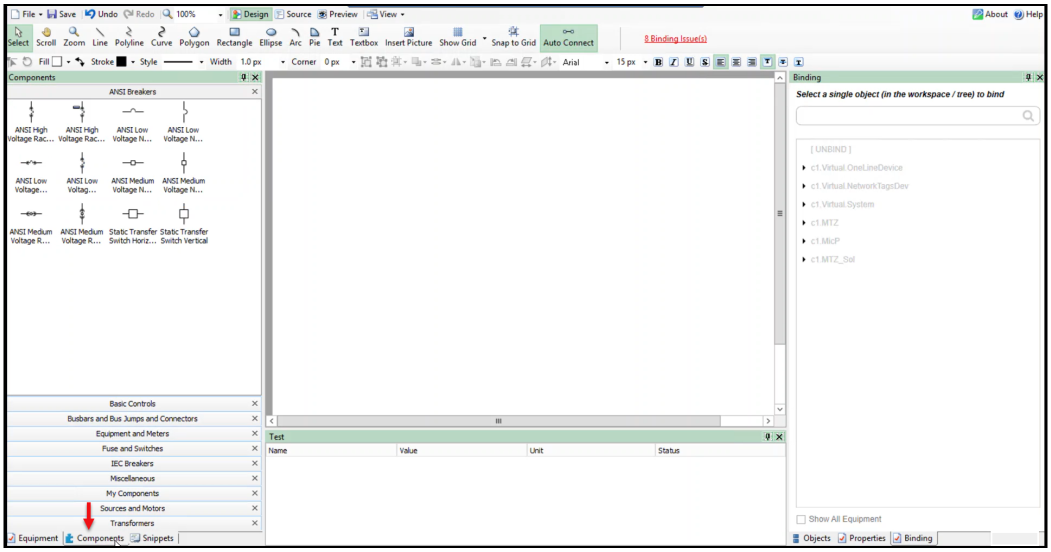
Use the Analog Write component to update or configure set-point registers for alarms. You can also use Alalog Write to set up temperature, scaling values, and CT and PT ratios.

Follow the steps below to create write and confirm user interactive.

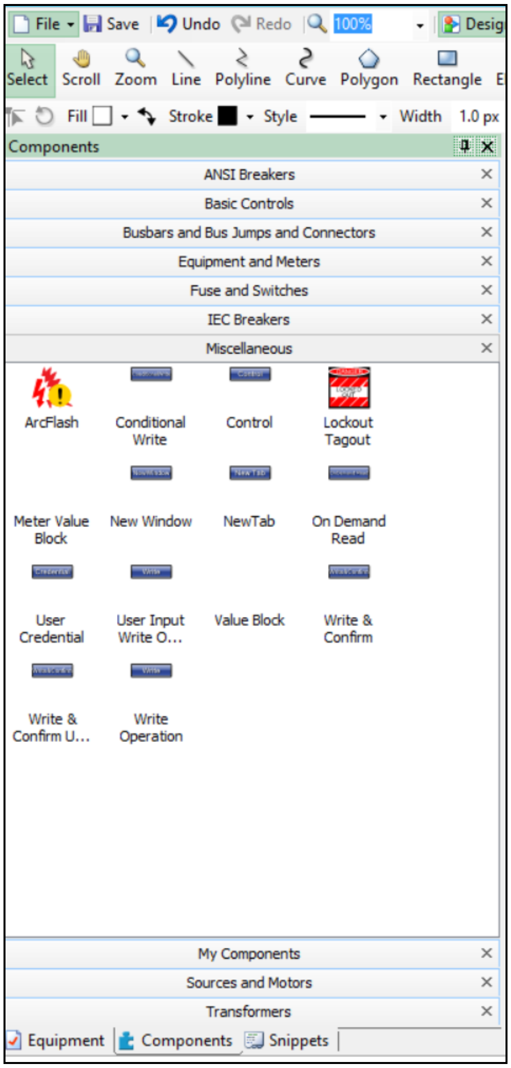
1. In the Graphics Editor, create a new graphic.



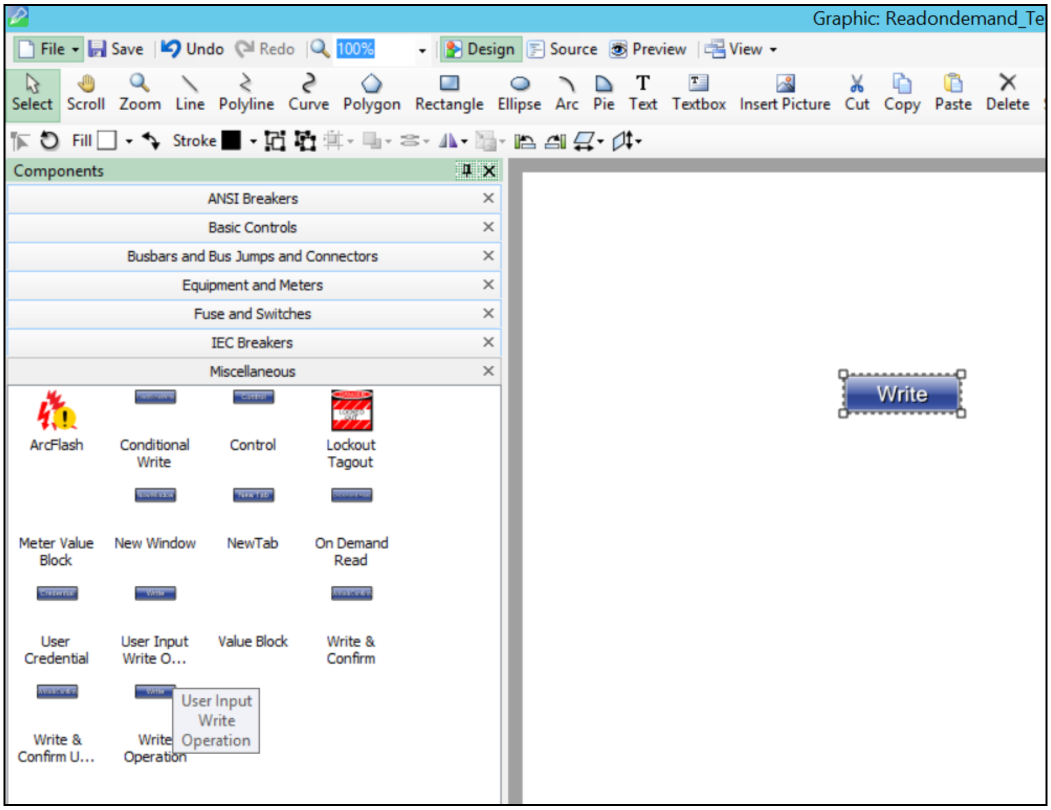
2. In the bottom left corner, click **Components**.



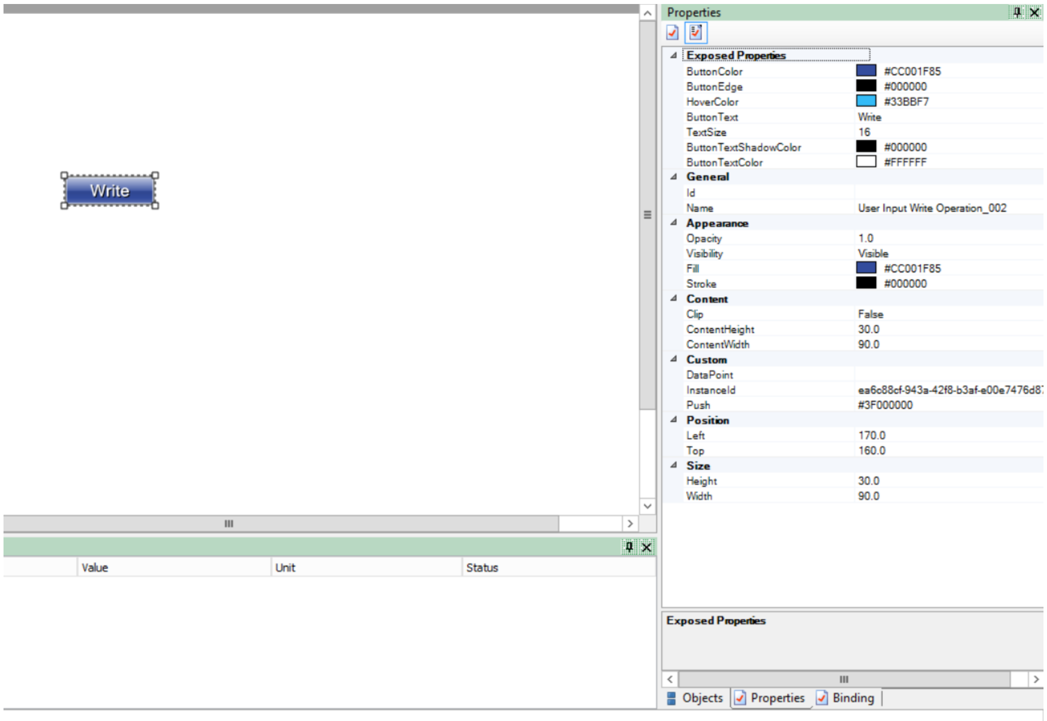
3. Click on **Miscellaneous** tab within the **Components** pane.



4. Drag and drop the **User Input Write Operation** component from the **Miscellaneous** section to the workspace. For example:



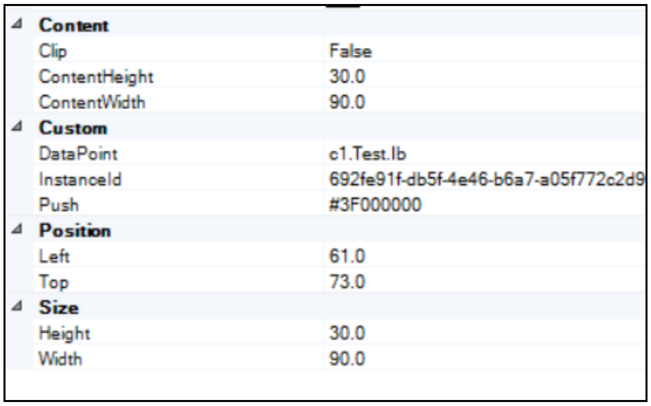
5. In the bottom right corner, click **Properties**.



Exposed Properties	
ButtonColor	#CC001F85
ButtonEdge	#000000
HoverColor	#33BBF7
ButtonText	Write
TextSize	16
ButtonTextShadowColor	#000000
ButtonTextColor	#FFFFFF
General	
Id	
Name	User Input Write Operation_002
Appearance	
Opacity	1.0
Visibility	Visible
Fill	#CC001F85
Stroke	#000000
Content	
Clip	False
ContentHeight	30.0
ContentWidth	90.0
Custom	
DataPoint	
InstanceId	ea6c88cf-943a-42f8-b3af-e00e7476d87
Push	#3F000000
Position	
Left	170.0
Top	160.0
Size	
Height	30.0
Width	90.0

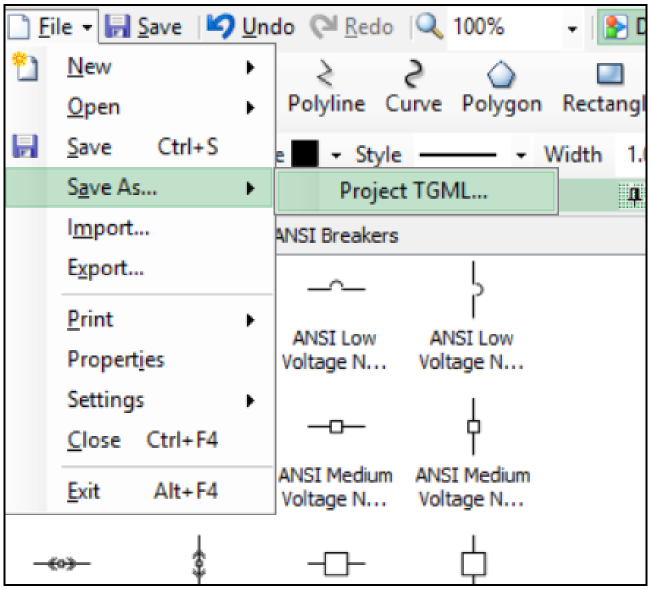
Exposed Properties

6. Add the fully qualified **DataPoint** name values as follows:



Content	
Clip	False
ContentHeight	30.0
ContentWidth	90.0
Custom	
DataPoint	c1.Test.lb
InstanceId	692fe91f-db5f-4e46-b6a7-a05f772c2d9
Push	#3F000000
Position	
Left	61.0
Top	73.0
Size	
Height	30.0
Width	90.0

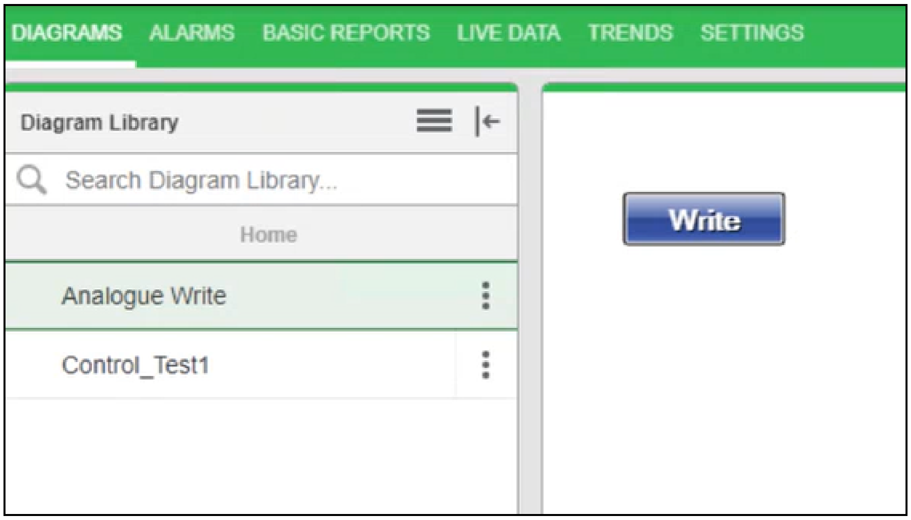
7. Go to **File > Save As > Project TGML**.



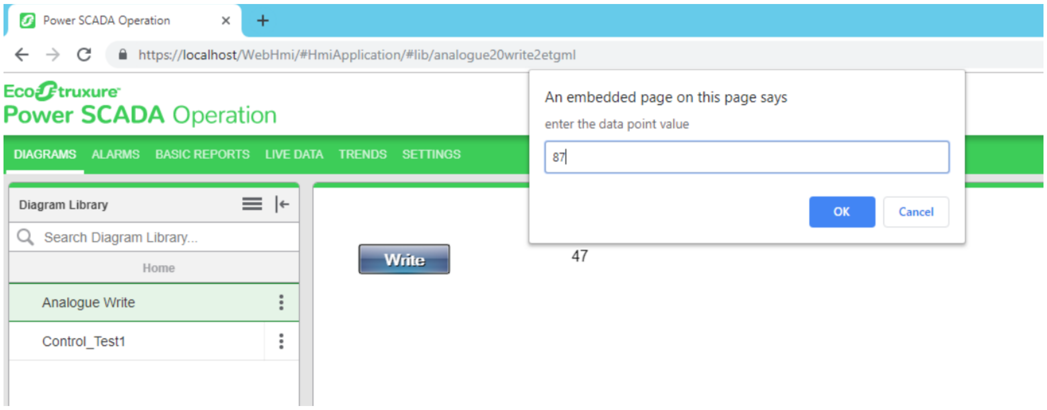
Test the changes:

1. Log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)

2. Click on **Analog Write** component created from the **Diagram Library**:



3. Click on **Write** operation component, type the **Data point** value in the displayed popup and click on **OK**.

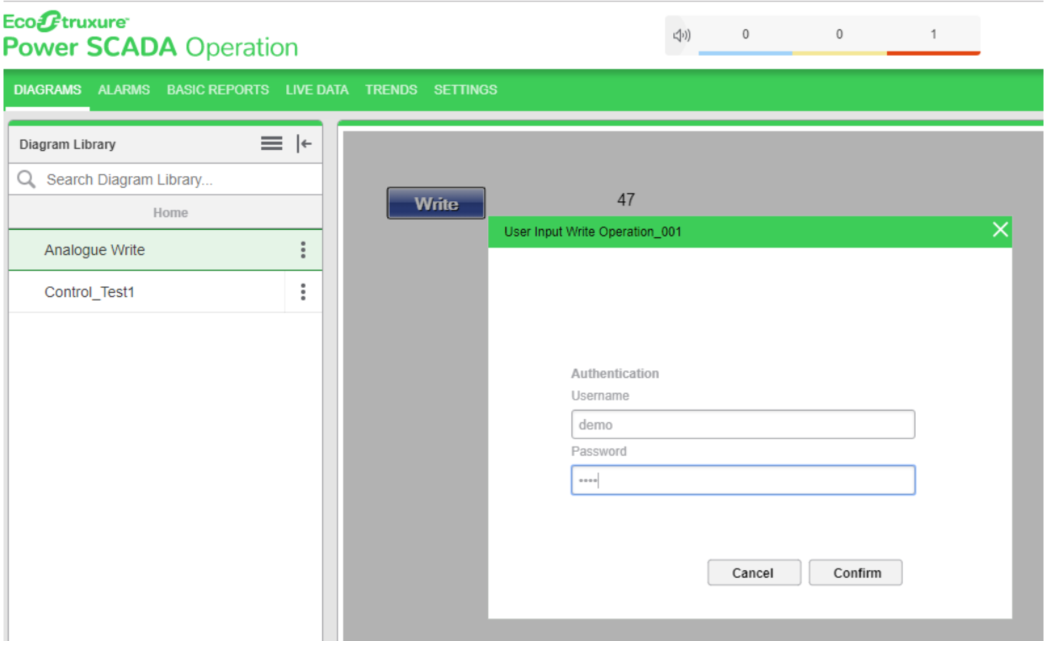


An embedded page on this page says

enter the data point value

87

4. Type the **Username** and **Password** and click on **Confirm**. To control which components require authentication, see Turning off credential requirements for control components.



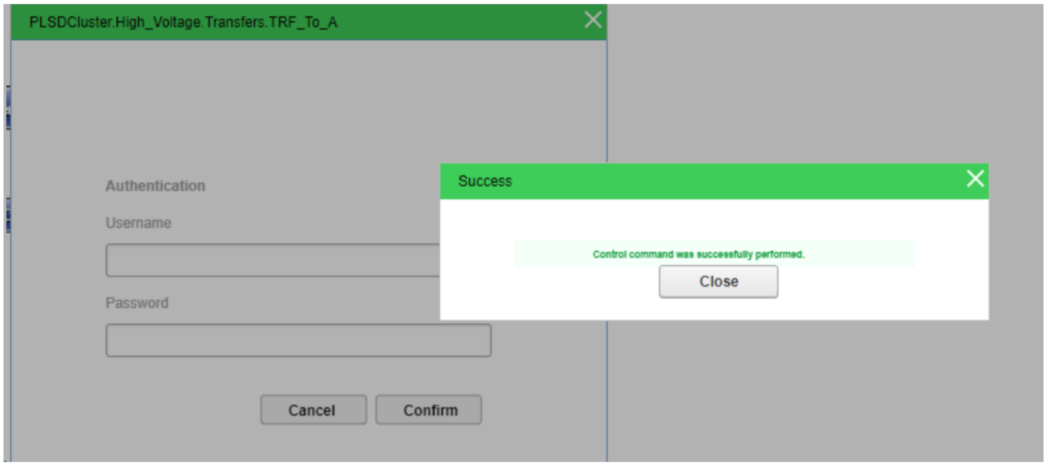
User Input Write Operation_001

Authentication

Username: demo

Password: ••••

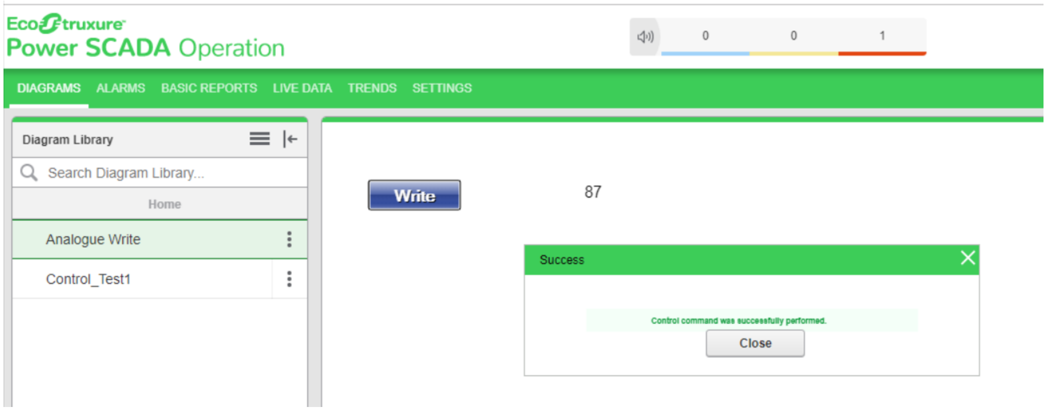
The success popup message is displayed:



PLSDCluster.High_Voltage.Transfers.TRF_To_A

Control command was successfully performed.

5. Click **Close** in the pop up. The datapoint value 87 is displayed:



Control command was successfully performed.

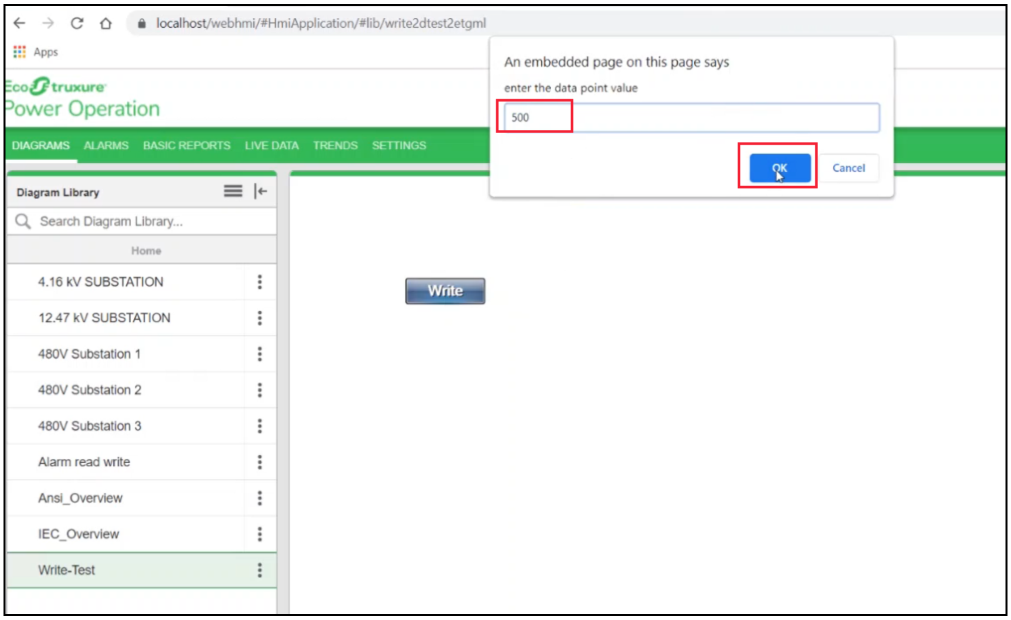
Read and Write Alarm Properties

Configure TGML to write the properties of alarms using web graphics. Configure TGML to display different alarm properties on web graphics.

Write alarm properties using TGML

You can write alarm properties, such as DataPoints.

1. In Graphics Editor, click on the **Components** tab.
2. Expand **Miscellaneous**, click **User Input Write Operation**.
3. While holding down CTRL, drag and drop the **User Input Write Operation** button to the work area.
4. In the Properties pane, enter the fully qualified DataPoint value. For example, DataPoint: PLSDCluster.High_Voltage.Generators.GEN1.Ia.High includes the cluster name, equipment name, source name, and item name, followed by the Alarm property.
5. Save the graphic file.
6. In WebHMI, click on the **DIAGRAMS** tab.
7. In the Diagram Library, click the saved graphic file.
8. In the dialog, enter the DataPoint value you want assigned to the alarm property and click **OK**.



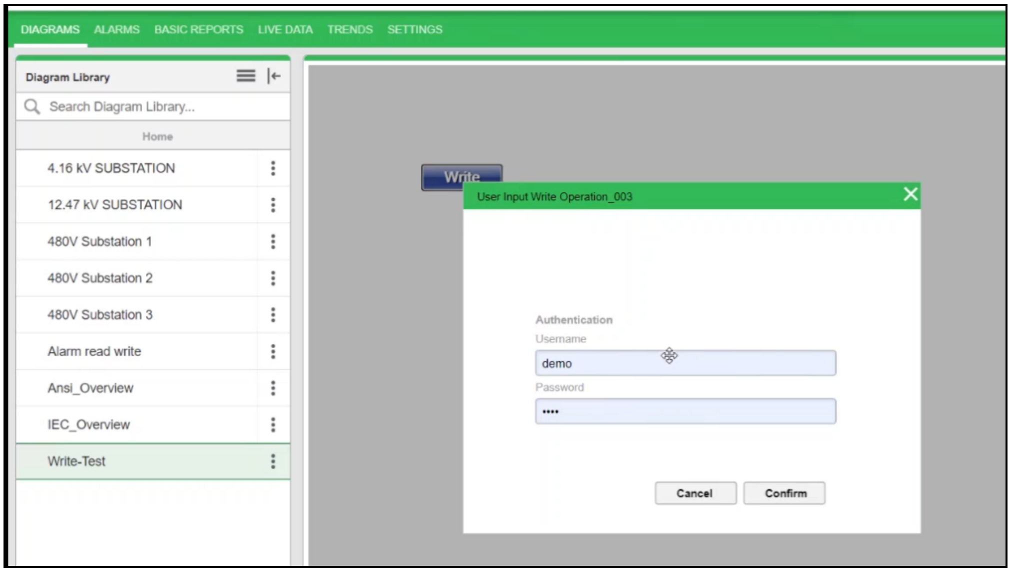
localhost/webhmi/#HmiApplication/#lib/write2dtest2etgml

An embedded page on this page says

enter the data point value

500

9. In the Write Operation dialog, enter credentials and click **Confirm**.



Username: demo

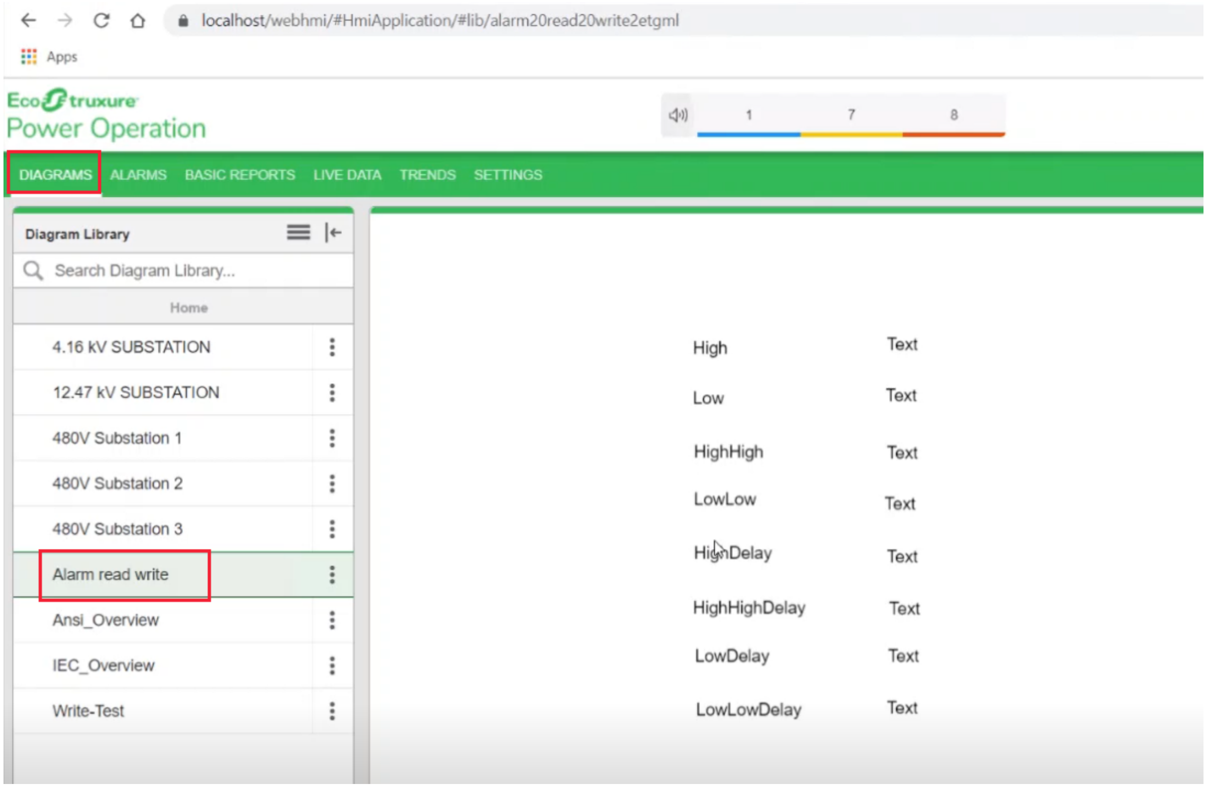
Password: ••••

Read alarm properties using TGML

You can view the alarm property values on the DIAGRAMS tab.

1. In Graphics Editor, select the **Text** tool from the menu, place the cursor on the work area, and enter a name to act as a default.
2. In the Properties pane, click **Text**.
3. Right-click **Text** > **New** > **Bind**.
4. Click **Bind** > **Properties**.
5. Enter a property value.

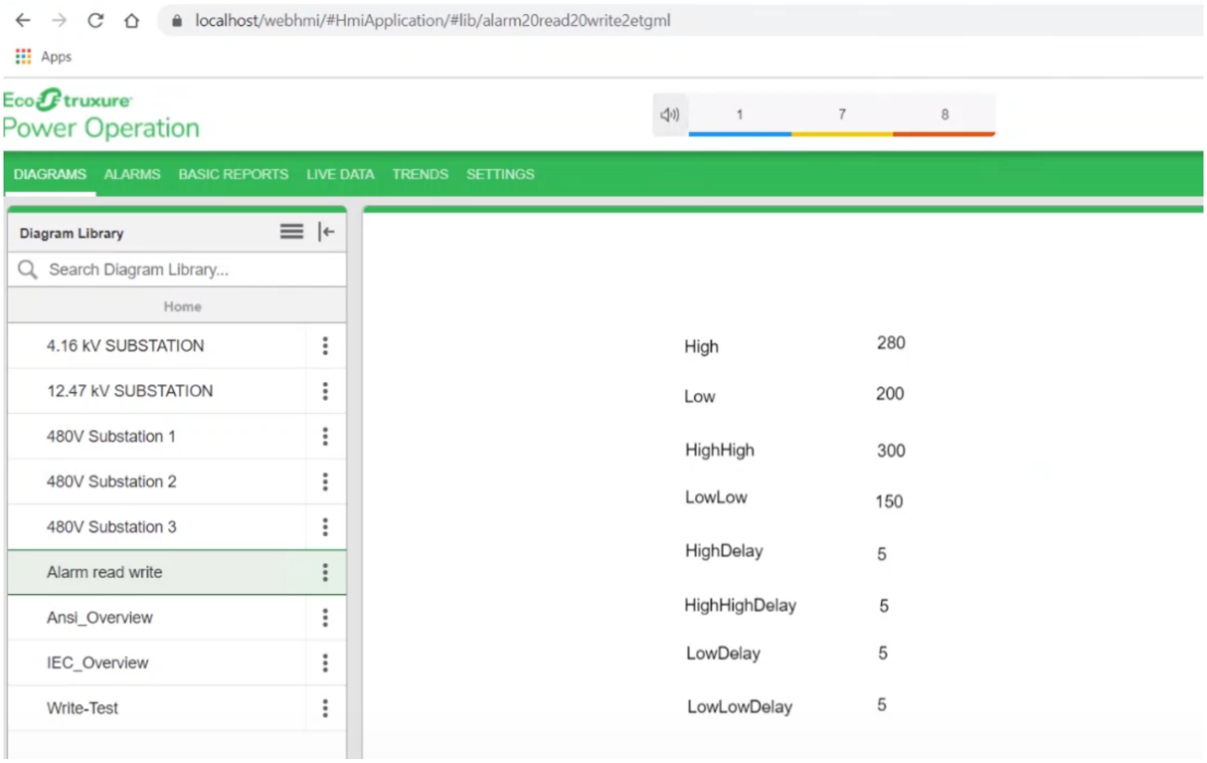
   For example: PLSDCluster.High_Voltage.Generators.GEN1.Ia.High.
6. Save the graphic file.
7. In WebHMI, click on the **DIAGRAMS** tab.
8. In the Diagram Library, click the saved graphic file.



High	Text
Low	Text
HighHigh	Text
LowLow	Text
HighDelay	Text
HighHighDelay	Text
LowDelay	Text
LowLowDelay	Text

NOTE: Values will be displayed from the **Read API**.

Property values will be displayed based on how they were defined during the Write operation.



High	280
Low	200
HighHigh	300
LowLow	150
HighDelay	5
HighHighDelay	5
LowDelay	5
LowLowDelay	5

Alarm property keywords

- High High - HighHigh
- High High Delay - HHDelay
- Low - Low
- Low Delay - LDelay
- Low Low - LowLow
- Low Low Delay - LLDelay

Linked TGML graphics

You can create TGML graphics that, when clicked by the user, can open other items, including diagrams, pop ups, and web pages. This section includes examples that demonstrate how you can configure TGML graphic components to link to other items.

- "Creating TGML graphic pop-ups" on page 454
- "Configuring a NewTab component" on page 463
- "Configuring a NewWindow component" on page 465
- "Opening links from TGML components" on page 467
- "Invoke function" on page 483
- "Adding a diagram to the menu bar" on page 500

TGML snippet examples

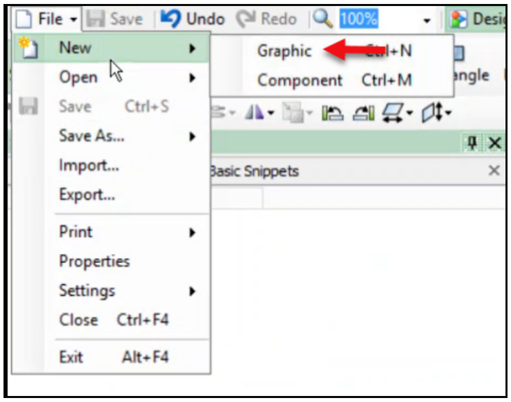
A lot of the functionality for opening other items is predefined in TGML snippets. See "TGML snippet examples" on page 504 for examples on how to use the snippets to create linked TGML graphics.

Creating TGML graphic pop-ups

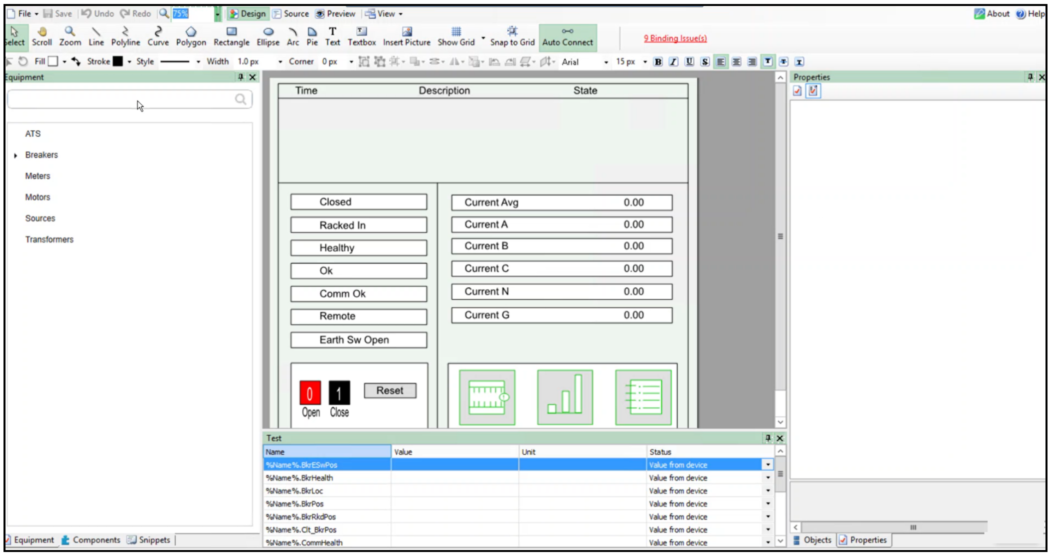
You can add a pop-up to a TGML graphic. When a user clicks on a device in a TGML graphics page that has a configured pop-up, a pop-up is displaayed. This topic lists the steps to add a pop-up to a device, and includes an example to illustrate how to create a TGML graphic pop-up.

To create a TGML graphic pop-up:

1. Open the Graphics Editor: Go to **Start > Power SCADA Operation > Graphics Editor**.
2. Go to **File > New > Graphic**.



3. Design the graphics based on the PSO user requirement.



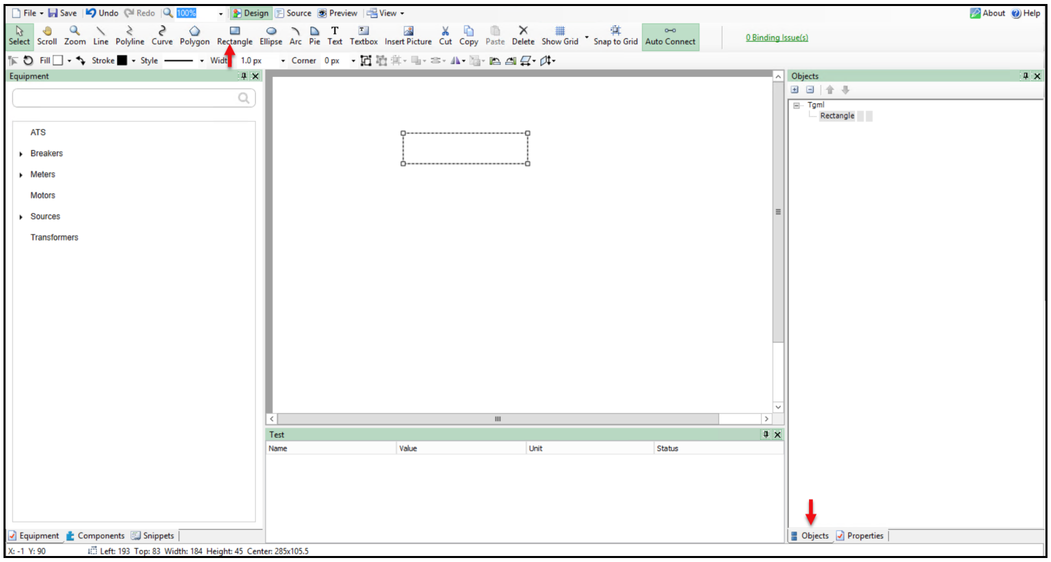
Name	Value	Unit	Status
%Name%.BkrESwPos			Value from device
%Name%.BkrHealth			Value from device
%Name%.BkrLoc			Value from device
%Name%.BkrPos			Value from device
%Name%.BkrRkdPos			Value from device
%Name%.Clt_BkrPos			Value from device
%Name%.CommHealth			Value from device

NOTE:
Newly created TGML graphic pop-ups can be used for all the devices.

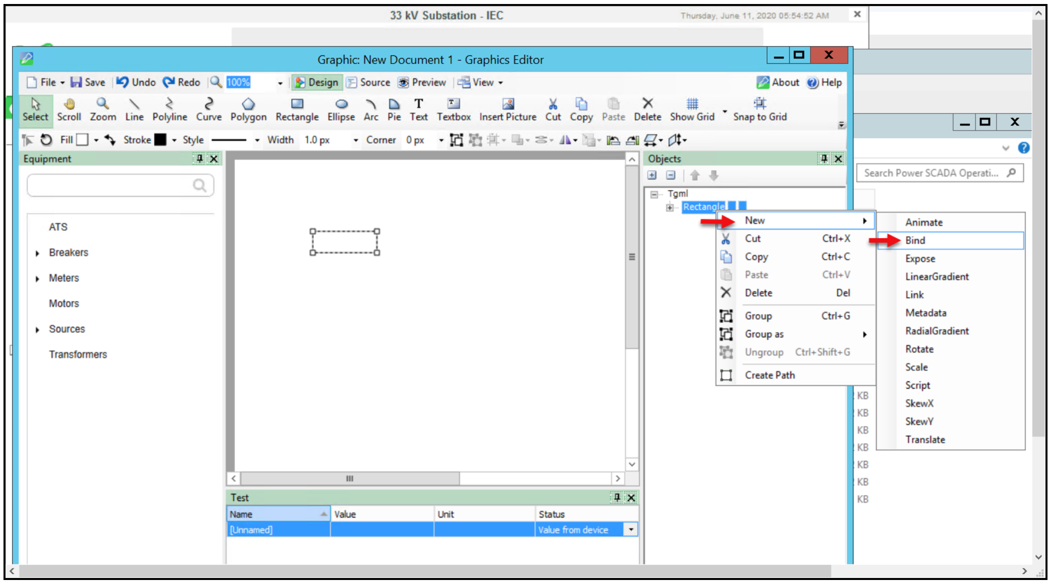
When the user clicks on a breaker in the PSO Web Applications, the pop-up displays the same for all the components, but the values will be different based on the breaker.

Example: The following steps demonstrate how to create a rectangle box and bind it.

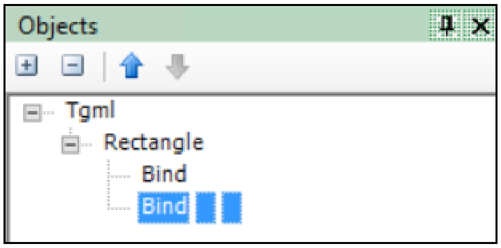
1. Click **Rectangle** from the top menu bar, draw on the workspace and click the **Objects** tab.



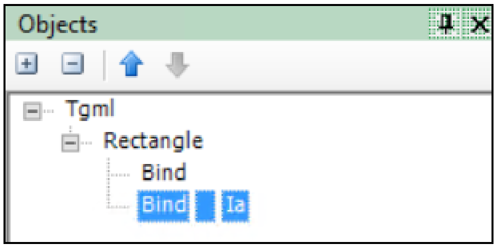
2. Go to the TGML > Right click on **Rectangle**, click **New**, and select **Bind**.



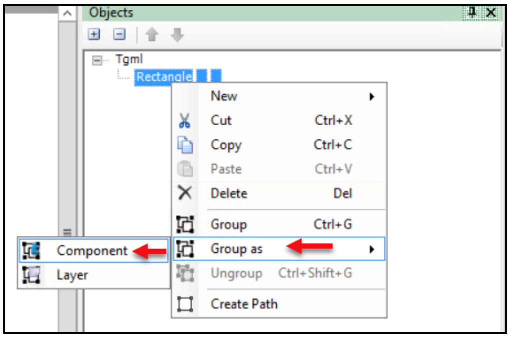
The following screen is displayed:



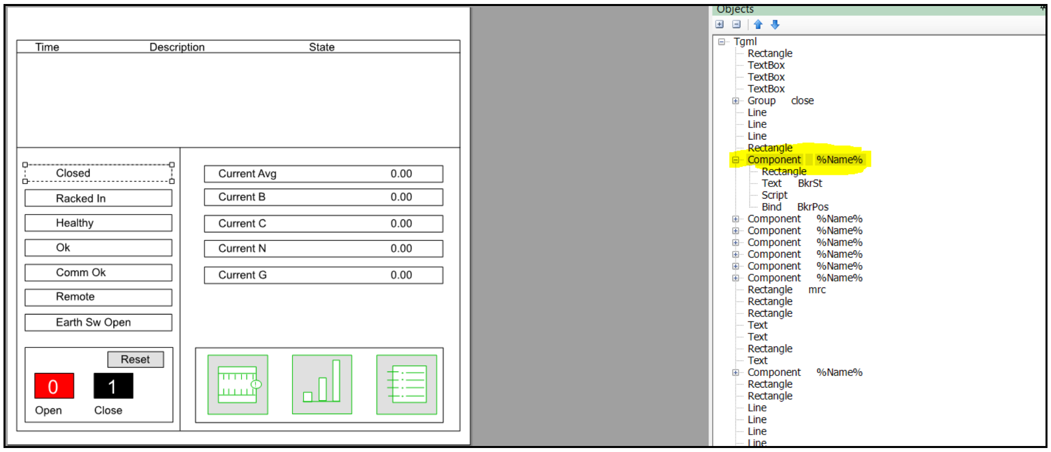
3. Double-click **Bind** and then type the required item name.



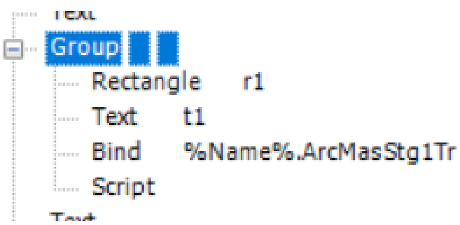
4. Right click on **Rectangle**, click **Group as**, and select **Component**.



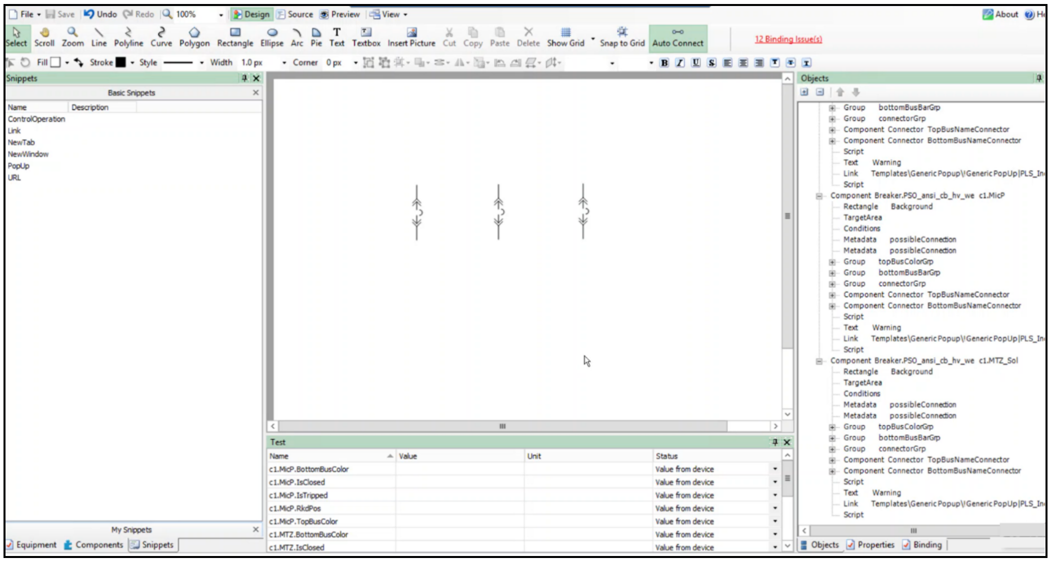
5. Rename the component as **%Name%** to make the TGML graphic concept work as the device details are inherited from the parent.



6. If the direct bind names is used without component, rename the bind to **%Name%.BindName**. Refer to the following image. During initiating the instance for the PopUp,**%Name%** will be replaced with the device name only in these 2 cases.



7. Create PSO user required components in the TGML new page, drag and drop **PopUp** snippet on the components and update the **Link** (type the file name of the PopUp with path if required). See **Pop-Up Snippet** feature for details.



Name	Description
ControlOperation	
Link	
NewTab	
NewWindow	
PopUp	
URL	

Test

Name	Value	Unit	Status
c1.MicP.BottomBusColor			Value from device
c1.MicP.IsClosed			Value from device
c1.MicP.IsTripped			Value from device
c1.MicP.RkdPos			Value from device
c1.MicP.TopBusColor			Value from device
c1.MTZ.BottomBusColor			Value from device
c1.MTZ.IsClosed			Value from device

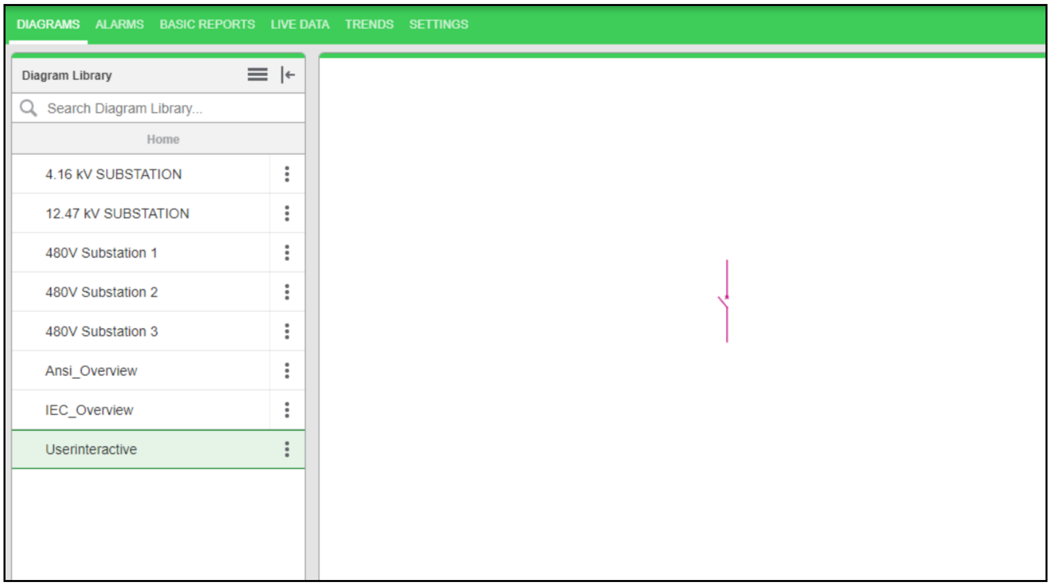
Objects

```
    Group     bottomBusBarGrp
    Group     connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Script
    Text      Warning
    Link      Templates\Generic Popup\!Generic PopUp|PLS_In
    Script
Component Breaker.PSO_ansi_cb_hv_we  c1.MicP
    Rectangle  Background
    TargetArea
    Conditions
    Metadata   possibleConnection
    Metadata   possibleConnection
    Group     topBusColorGrp
    Group     bottomBusBarGrp
    Group     connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Script
    Text      Warning
    Link      Templates\Generic Popup\!Generic PopUp|PLS_In
    Script
Component Breaker.PSO_ansi_cb_hv_we  c1.MTZ_Sol
    Rectangle  Background
    TargetArea
    Conditions
    Metadata   possibleConnection
    Metadata   possibleConnection
    Group     topBusColorGrp
    Group     bottomBusBarGrp
    Group     connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Script
    Text      Warning
    Link      Templates\Generic Popup\!Generic PopUp|PLS_In
    Script
```

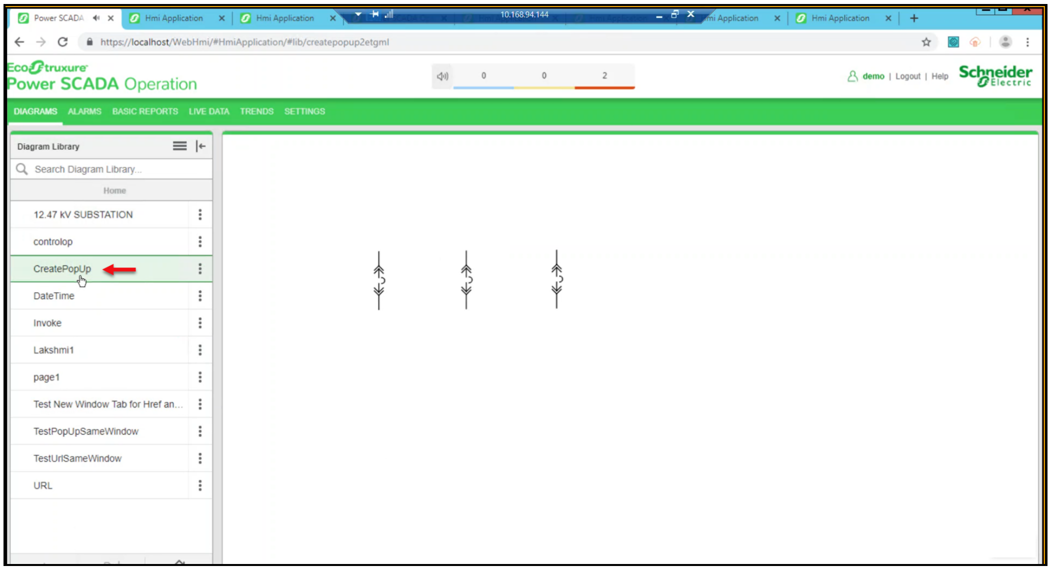
8. Save the TGML file.

9. Log in to the PSO Web Applications.

   The following screen is displayed.

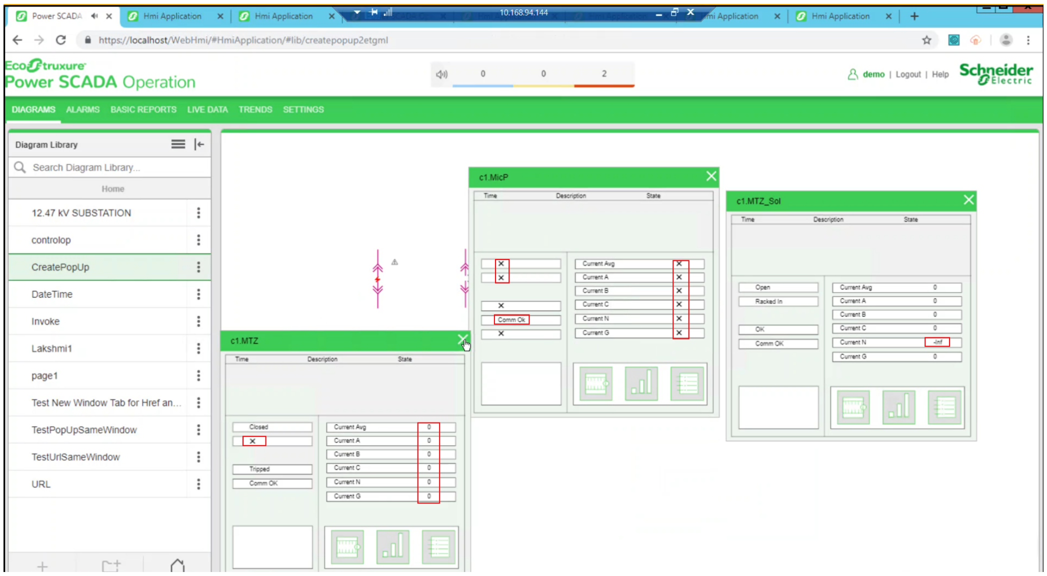


10. Click on created TGML file in the **Diagram Library** menu.


https://localhost/WebHmi/#HmiApplication/#lib/createpopup2etgml
EcoStruxure Power SCADA Operation
demo | Logout | Help
DIAGRAMS ALARMS BASIC REPORTS LIVE DATA TRENDS SETTINGS
Diagram Library
Search Diagram Library...
Home
12.47 kV SUBSTATION
controlop
CreatePopUp
DateTime
Invoke
Lakshmi1
page1
Test New Window Tab for Href an...
TestPopUpSameWindow
TestUrlSameWindow
URL

11. Click on each breaker.

The following screen is displayed with the created pop-up for each breaker:

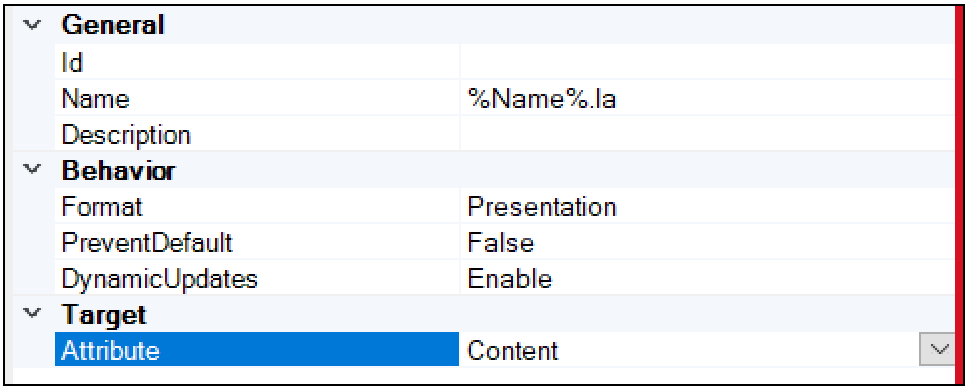


NOTE:
The pop-up values are different for each breaker.

If the bind name is not present in PSO > Item name, the designed fields of pop-up are hidden as shown in the above image.

Rendering error conditions in PopUp using presentation value

In the bind properties, the Presentation values are **Format = Presentation**, **Attribute = Content**:



General	
Id	
Name	%Name%.Ia
Description	
Behavior	
Format	Presentation
PreventDefault	False
DynamicUpdates	Enable
Target	
Attribute	Content

The following table explains the error rendering with the respective place holder that appears in the output Pop-Up as shown in the above screen:

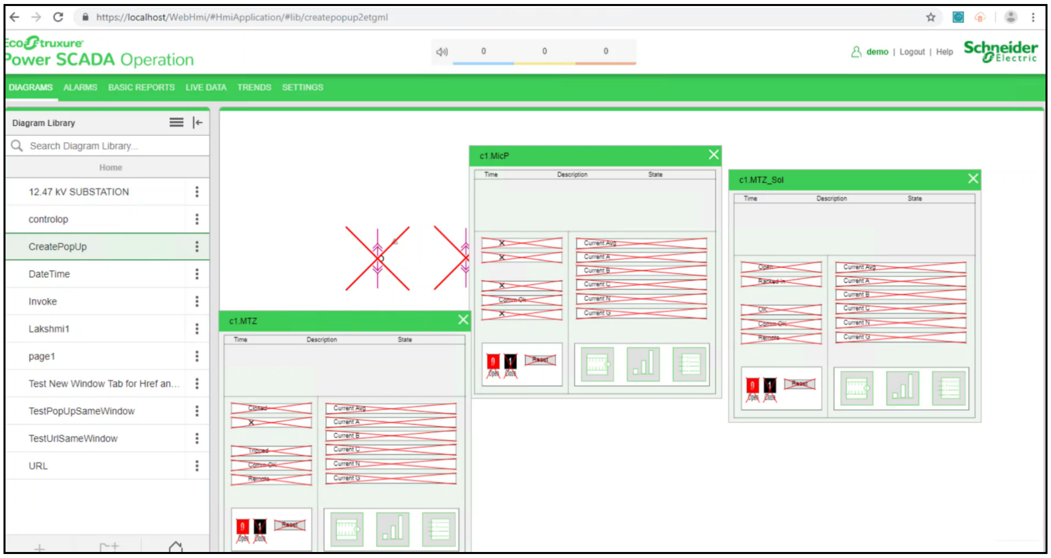
Tags Missing in Profile	Place Holder Should Disappear
Comm Loss for Tags	✕
NA Value	-INF
Quality is good	Show Value returned by API
PSO not running	Default Comm Loss
Unknown	⚠

Rendering error conditions in PopUp using getValue() function in script

The following code example explains the **getValue()** function in script:

```
function change(evt)
{
    var val = evt.getValue().split(",")[0];
    var dateTime = evt.getValue().split(",")[1];
    var quality = evt.getValue().split(",")[2];
    var comp = evt.getCurrentTarget();

    //Hiding the placeholder
    if(quality == "0" && dateTime == "0"){
        comp.setAttribute("Visibility", "Hidden");
    }

    //Setting the value as -inf
    else if(val == "-Infinity"){
        comp.setAttribute("Visibility","Visible");
        comp.getChild("BrkSt").setAttribute("Content","-Inf");
    }

    //Setting the value to x
    else if(quality == "2"){
        comp.setAttribute("Visibility","Visible");
        comp.getChild("BrkSt").setAttribute("Content","x");
    }
    //Setting the value in case of good quality
    else{
        comp.setAttribute("Visibility","Visible");
        if(val == "1"){
            evt.getCurrentTarget().getChild("BrkSt").setAttribute
("Content","Open");
        }
        if(val == "0"){
            evt.getCurrentTarget().getChild("BrkSt").setAttribute
("Content","Close");
        }
}
```

For example, when PSO is shut down, the default **comm loss** is shown below:



Invoking a PopUp

1. In the Graphics Editor, drag and drop any component onto the workspace.
2. Drag the PopUp snippet onto the component in the workspace, and then save the file.

   The following code example details the code to invoke a PopUp.

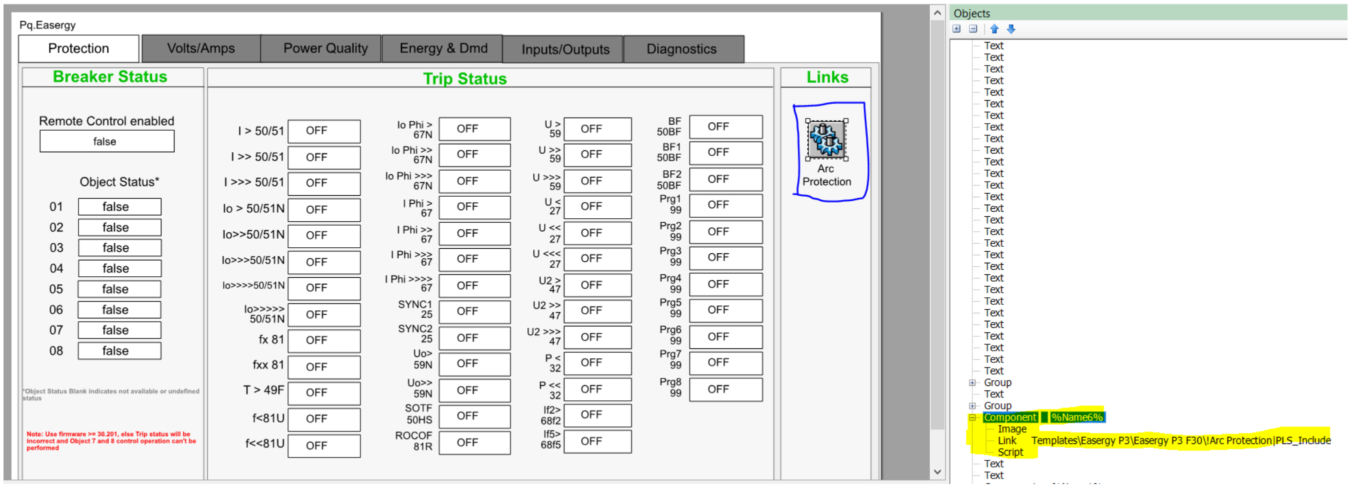
```
functionclick(evt)
{
    var componentName = evt.getCurrentTarget().getAttribute("Name");
    var connector = evt.getCurrentTarget().getElementsByTagName("Link");
    var instanceId = evt.getCurrentTarget().getAttribute("InstanceId");
    var title = componentName;
    var customExpose = evt.getCurrentTarget().getAttribute("SubstituteNames");
    //Height & width can be configurable by the user
    var width = 370;
    var height = 370;
    var show TitleBar = "Yes";

    for(var i=0;i< connector.length;i++) {
        var connectorName = connector.item (i).getAttribute("Name");
        invoke(connectorName,"Type = PopUp | ComponentName =" + componentName +
" | InstanceID =" + instanceId + " | Title=" + title + " | Width=" + width + " |
Height=" + height + " | ShowTitleBar =" + showTitleBar + " | CustomExpose=" +
customExpose);
    }
}
```

For more details see the "Invoke function" on page 483 and the "TGML snippet examples" on page 504.

WebReach Diagram Design

Refer all the steps from the above section on **How to design popup** for the below screen WebReach Diagram.

- Rename the link to the template as shown in the following image as it should be available in **PLS_Include** Project.
- The link will be used to navigate from one template to other template. The link can be used from snippets.
- If the template is available in a different project, then change the project name instead of **PLS_Include** to the project where the TGML is present. Refer to the following image.
- An Invoke method can be used in the navigation Link script. For the parameters details, see "Invoke function attributes" on page 492 .



Remote Control enabled: false

Object Status*
01 false
02 false
03 false
04 false
05 false
06 false
07 false
08 false

*Object Status Blank indicates not available or undefined status

Note: Use firmware >= 30.201, else Trip status will be incorrect and Object 7 and 8 control operation can't be performed

I > 50/51 OFF
I >> 50/51 OFF
I >>> 50/51 OFF
Io > 50/51N OFF
Io>>50/51N OFF
Io>>>50/51N OFF
Io>>>>50/51N OFF
Io>>>>> 50/51N OFF
fx 81 OFF
fxx 81 OFF
T > 49F OFF
f<81U OFF
f<<81U OFF

Io Phi > 67N OFF
Io Phi >> 67N OFF
Io Phi >>> 67N OFF
I Phi > 67 OFF
I Phi >> 67 OFF
I Phi >>> 67 OFF
I Phi >>>> 67 OFF
SYNC1 25 OFF
SYNC2 25 OFF
Uo> 59N OFF
Uo>> 59N OFF
SOTF 50HS OFF
ROCOF 81R OFF

U > 59 OFF
U >> 59 OFF
U >>> 59 OFF
U < 27 OFF
U << 27 OFF
U <<< 27 OFF
U2 > 47 OFF
U2 >> 47 OFF
U2 >>> 47 OFF
P < 32 OFF
P << 32 OFF
If2> 68f2 OFF
If5> 68f5 OFF

BF 50BF OFF
BF1 50BF OFF
BF2 50BF OFF
Prg1 99 OFF
Prg2 99 OFF
Prg3 99 OFF
Prg4 99 OFF
Prg5 99 OFF
Prg6 99 OFF
Prg7 99 OFF
Prg8 99 OFF

```
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Text
Group
Text
Group
Component  %Name6%
    Image
    Link      Templates\Easergy P3\Easergy P3 F30\!Arc Protection|PLS_Include
    Script
Text
Text
```

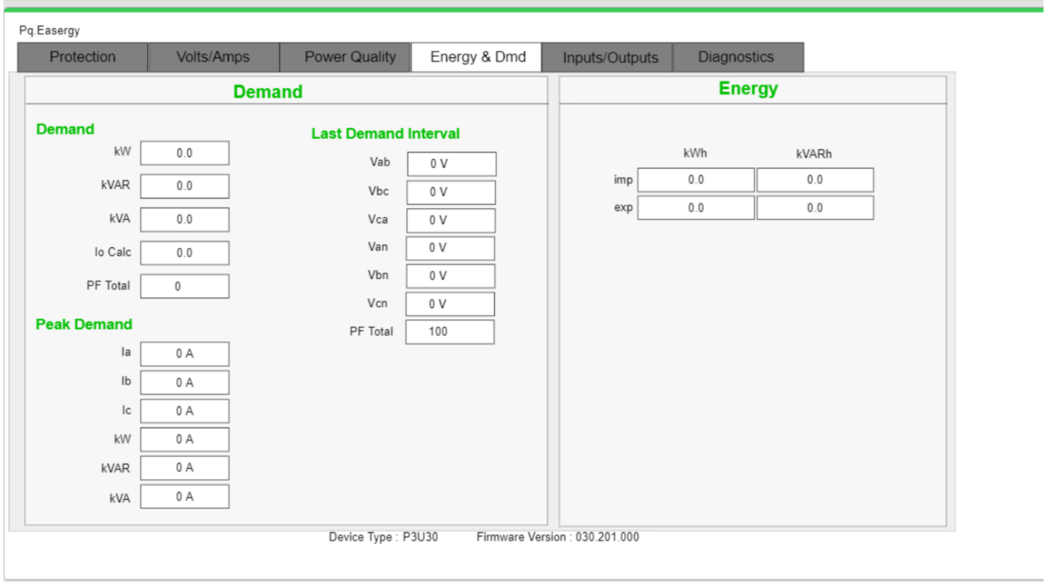
Rendering error conditions in WebReach Diagrams

The following table explains the error rendering scenarios with respective place holder which appears in the WebReach diagram:

Tags Missing in Profile	⊘

Comm Loss for Tags	✖
NA Value	-INF
Quality is good	Show Value returned by API
PSO not running	Default Comm Loss
Unknown	⚠

Refer to the following WebReach diagram output screen:


Pq.Easergy

Protection | Volts/Amps | Power Quality | Energy & Dmd | Inputs/Outputs | Diagnostics

Demand

Demand

kW	0.0
kVAR	0.0
kVA	0.0
Io Calc	0.0
PF Total	0

Peak Demand

Ia	0 A
Ib	0 A
Ic	0 A
kW	0 A
kVAR	0 A
kVA	0 A

Last Demand Interval

Vab	0 V
Vbc	0 V
Vca	0 V
Van	0 V
Vbn	0 V
Vcn	0 V
PF Total	100

Energy

	kWh	kVARh
imp	0.0	0.0
exp	0.0	0.0

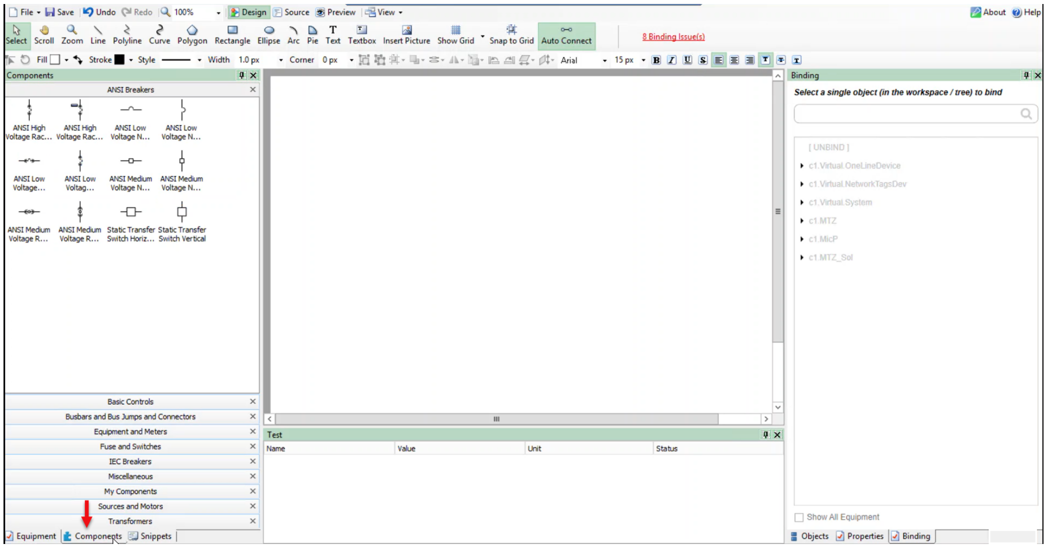
Device Type : P3U30 Firmware Version : 030.201.000

Configuring a NewTab component

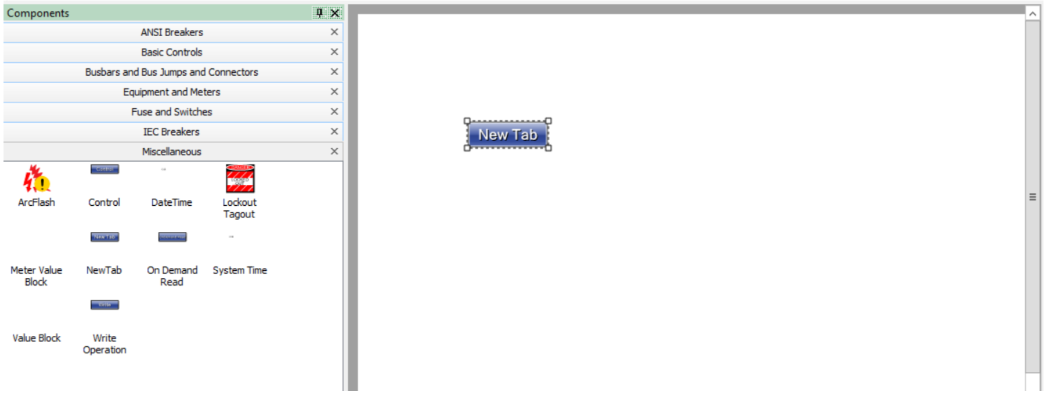
Refer to Configuration for the NewTab PopUp in the Snippet documentation section.

To configure a NewTab component:

1. In the Graphics Editor, click **Components**:

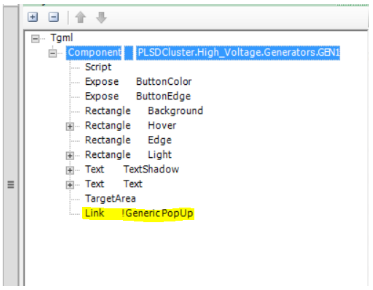


2. Drag and drop any component as per your requirement in the workspace. The example used here is NewTab:



3. With the component selected in the workspace, at the bottom right corner, click **Objects**.

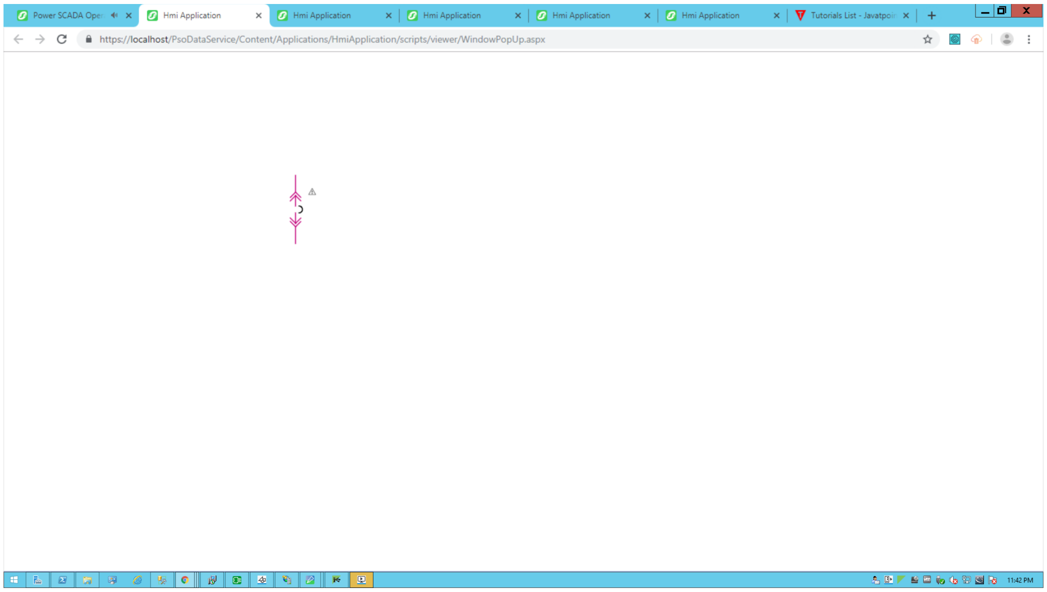
4. Expand the component in objects window, enter the file name in the **Link** which will open the new window in PSO Web Applications, and then save the file.



To test the changes:

1. Log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)

2. Click on the **NewTab** component which you have named. It opens the destination TGML in a new browser tab:

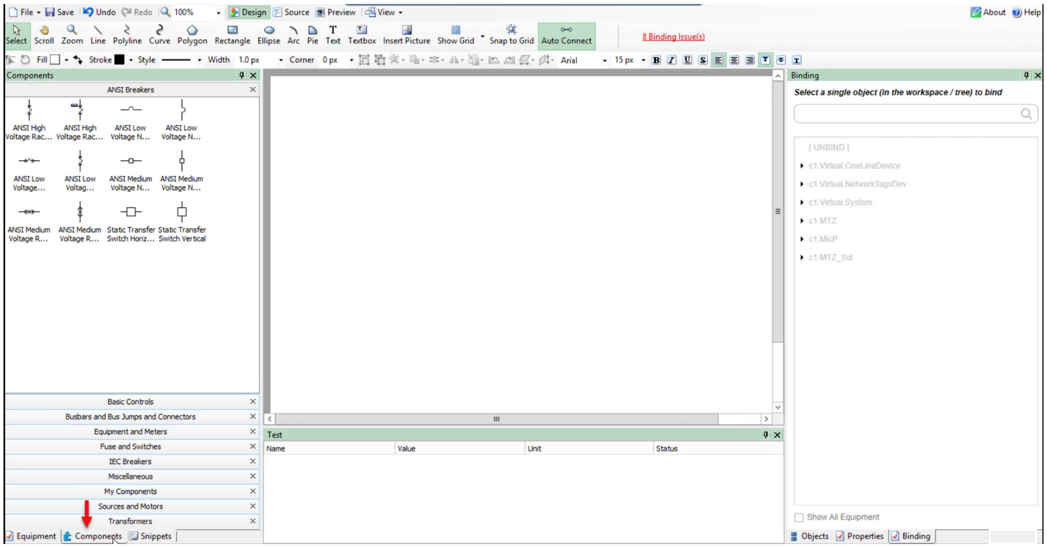

Power SCADA Ope
Hmi Application
Tutorials List - Javatpoi
https://localhost/PsoDataService/Content/Applications/HmiApplication/scripts/viewer/WindowPopUp.aspx
11:42 PM

Configuring a NewWindow component

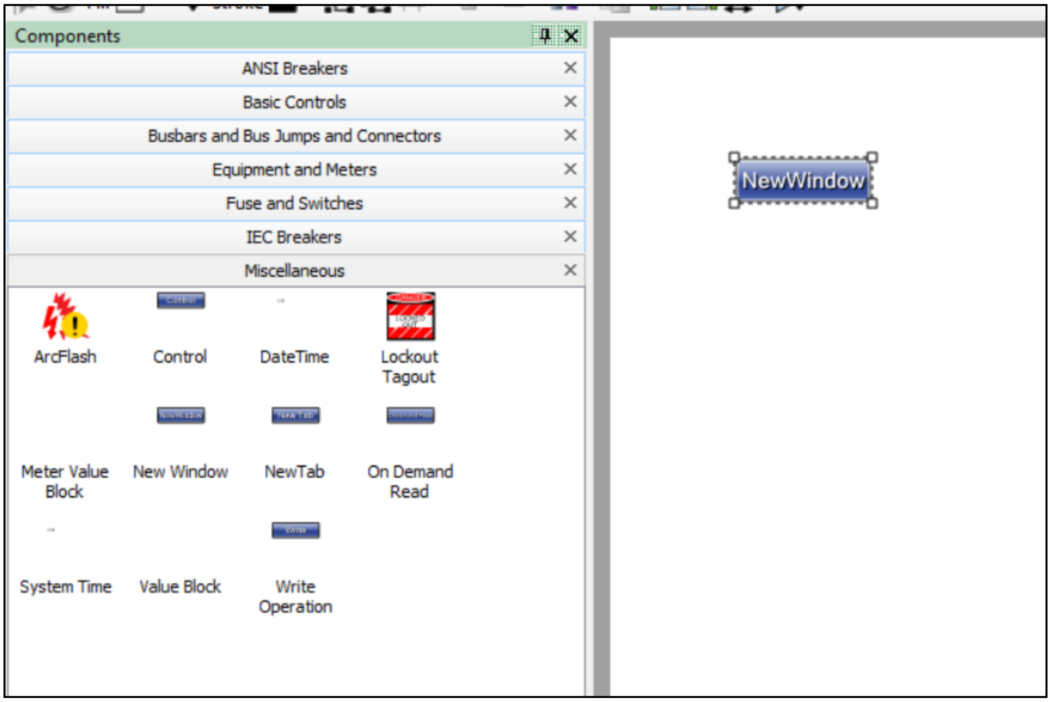
Refer to Configuration for the NewWindow PopUp in the Snippet documentation section.

To configure a NewWindow component:

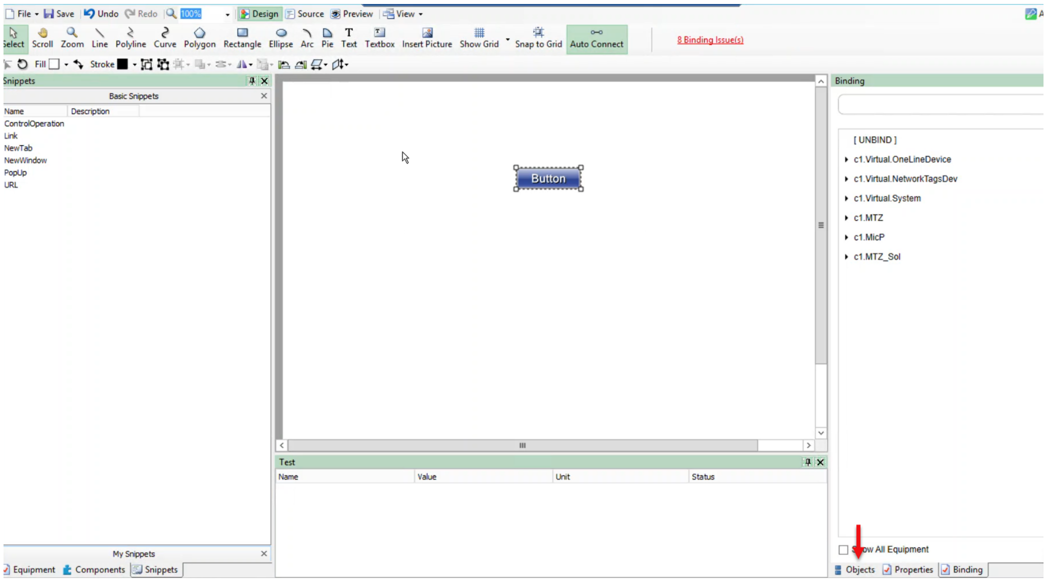
1. Open the Graphics Editor: Go to Start > Power SCADA Operation > Graphics Editor.
2. In the bottom left corner, click **Components**. Refer NewWindow Snippet documentation for more details.



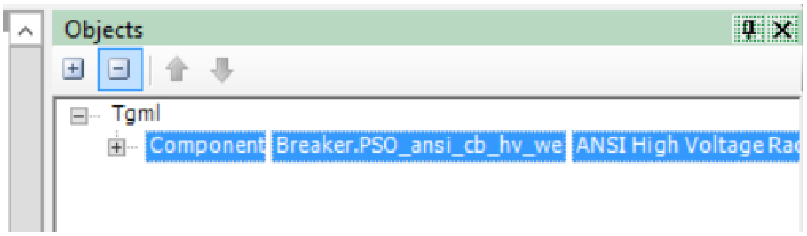
3. Drag and drop a component onto the workspace based on your requirement. For example:



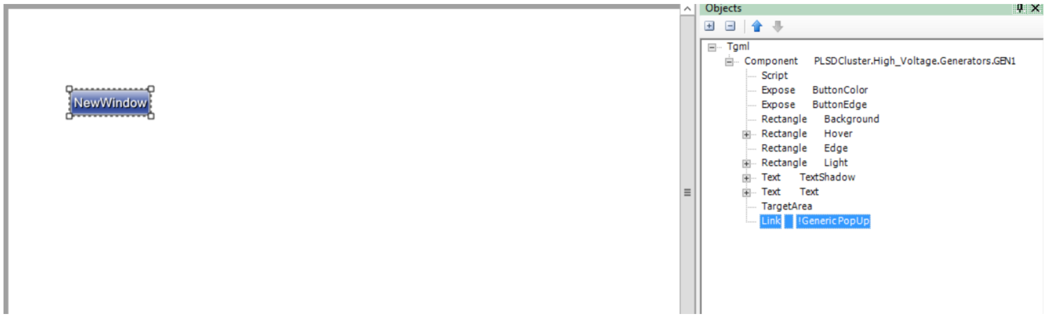
4. In the bottom right corner, click its **Objects**.



5. Expand the **TGML component** node.



6. Enter the file name in the Link which opens the new window output in the Web Applications page.

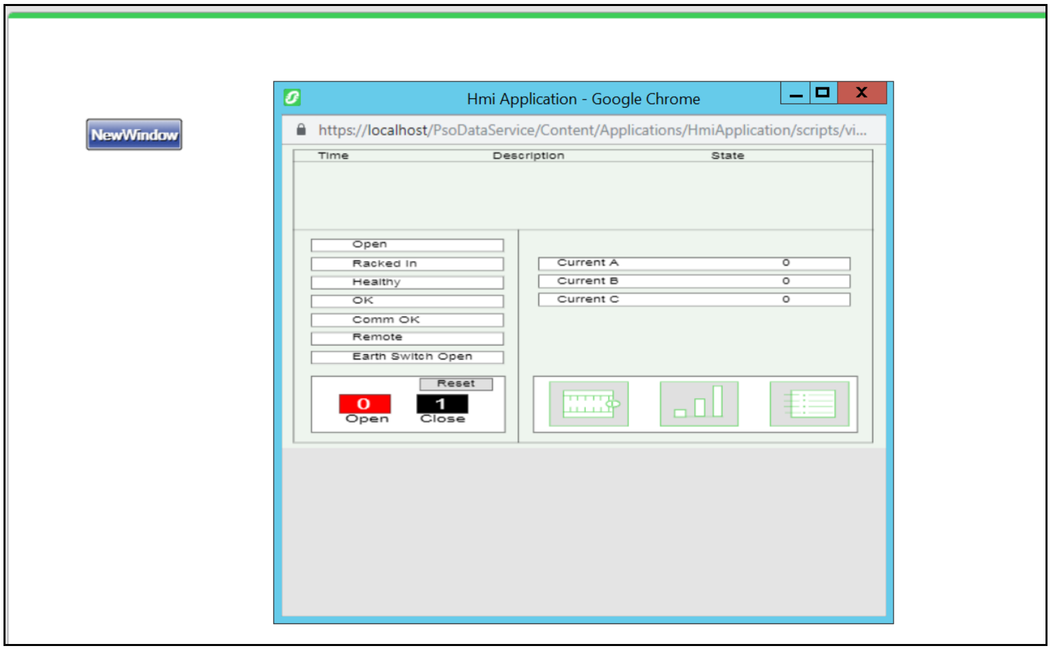


```
Tgml
  Component   PLSDCluster.High_Voltage.Generators.GEN1
    Script
    Expose     ButtonColor
    Expose     ButtonEdge
    Rectangle  Background
    Rectangle  Hover
    Rectangle  Edge
    Rectangle  Light
    Text       TextShadow
    Text       Text
    TargetArea
    Link       !GenericPopUp
```

7. Click **File > Save**, enter a file name, and then save the file.

To test the changes:

1. Log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)
2. Click on the component which you have named. It opens the destination TGML in a new browser window:

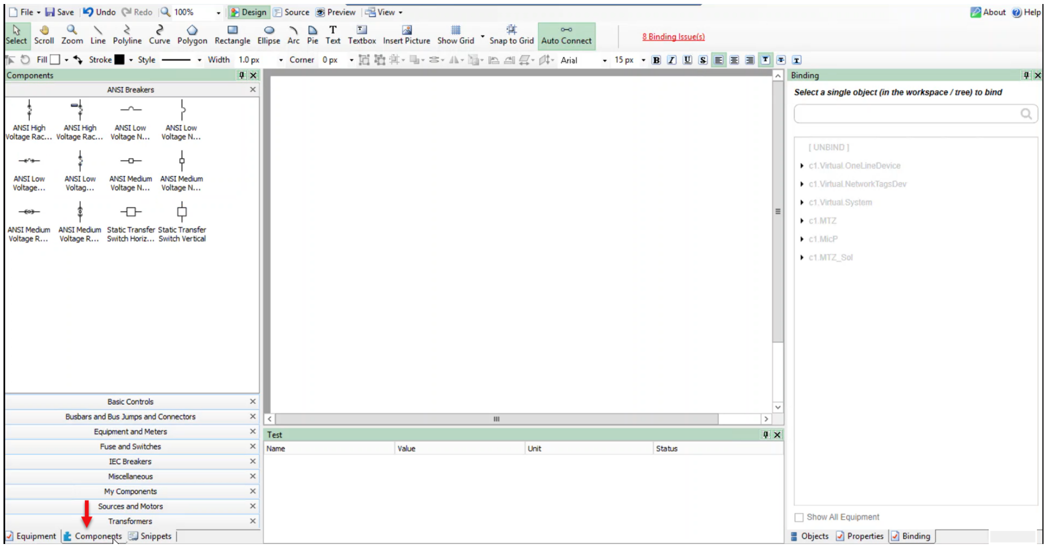


Opening links from TGML components

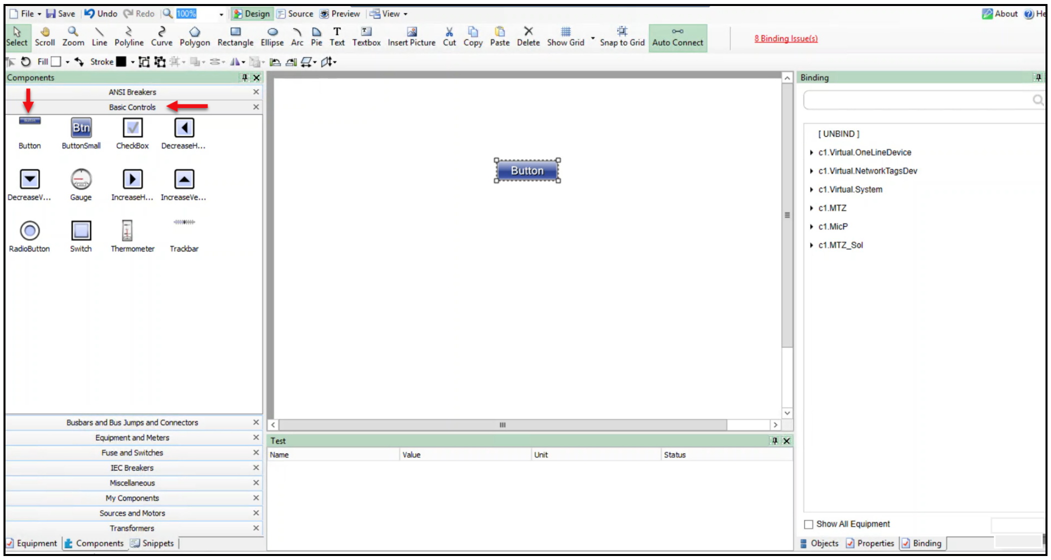
You can open a URL link from a TGML component. This topic uses an example to illustrate how you can configure a TGML component to open a URL.

To open a link from a component:

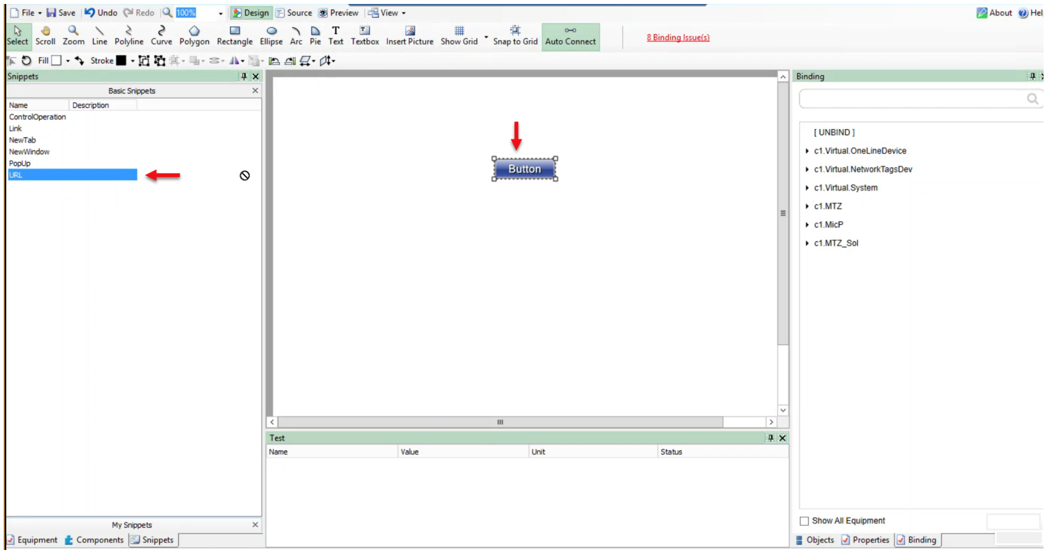
1. Open the Graphics Editor: Go to Start > Power SCADA Operation > Graphics Editor.
2. At the bottom left, click **Components**.


File Save Undo Redo 100% Design Source Preview View About Help
Select Scroll Zoom Line Polyline Curve Polygon Rectangle Ellipse Arc Pie Text Textbox Insert Picture Show Grid Snap to Grid Auto Connect
8 Binding Issue(s)
Components
ANSI Breakers
Basic Controls
Busbars and Bus Jumps and Connectors
Equipment and Meters
Fuse and Switches
IEC Breakers
Miscellaneous
My Components
Sources and Motors
Transformers
Equipment Components Snippets
Test
Name Value Unit Status
Binding
Select a single object (in the workspace / tree) to bind
[UNBIND]
c1.Virtual.OneLineDevice
c1.Virtual.NetworkTagsDev
c1.Virtual.System
c1.MTZ
c1.MicP
c1.MTZ_Sol
Show All Equipment
Objects Properties Binding

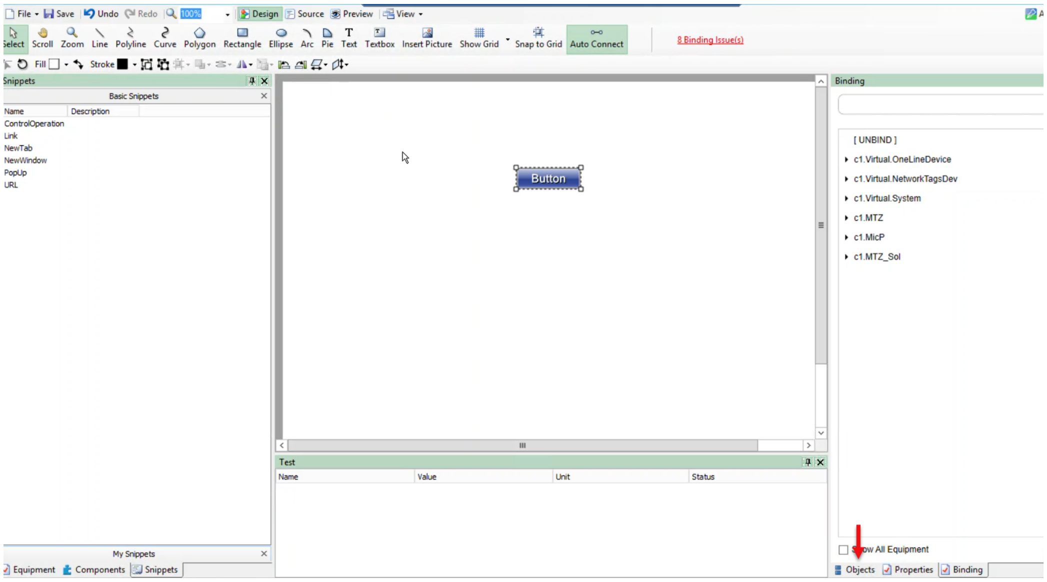
3. Expand the **Basic Controls** tab, and then drag and drop the button components to the workspace.



4. At the bottom left, click **Snippets**, and then drag and drop **URL** onto the **Button** component in the workspace.

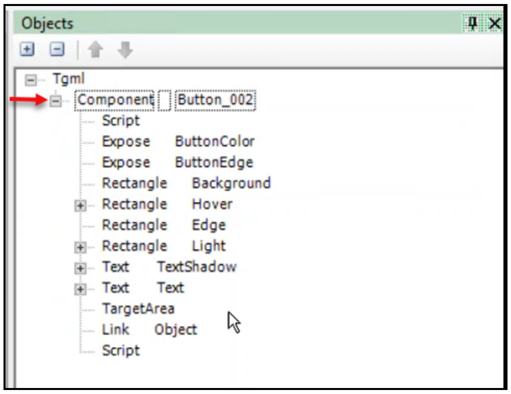


5. At the bottom right, click **Objects**.



6. Define the Link attribute value:

   a. Expand the **Component** node to find the **Link** attribute within the button.

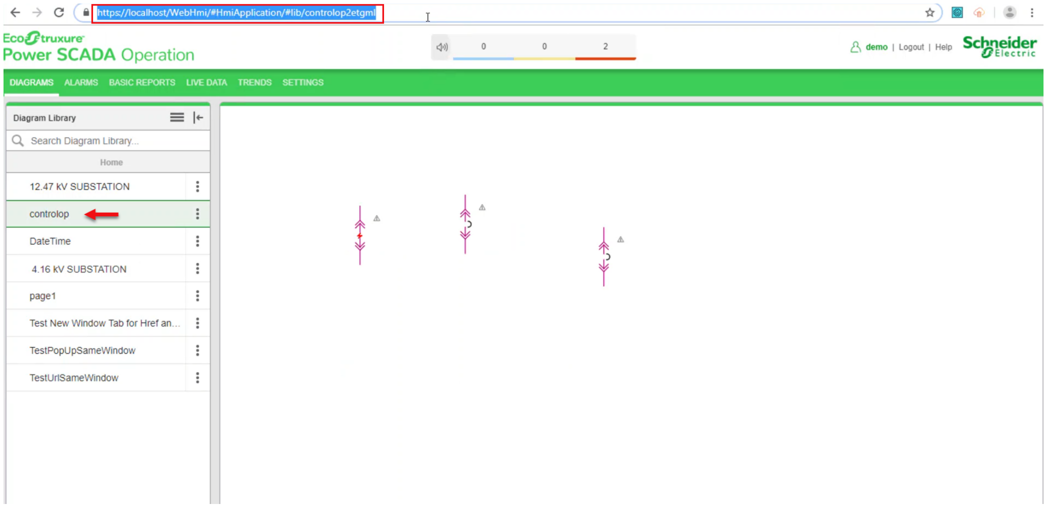


Objects

```
Tgml
  Component  Button_002
    Script
    Expose     ButtonColor
    Expose     ButtonEdge
    Rectangle  Background
    Rectangle  Hover
    Rectangle  Edge
    Rectangle  Light
    Text       TextShadow
    Text       Text
    TargetArea
    Link       Object
    Script
```

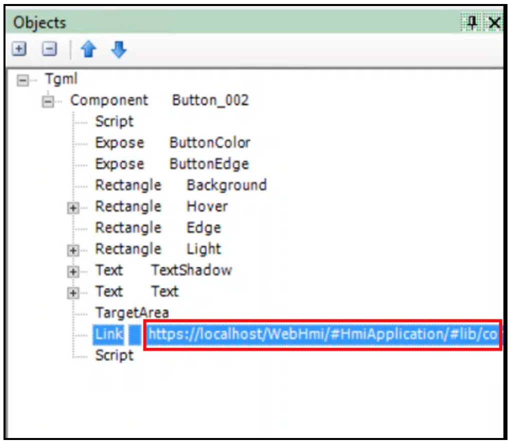
   b. Log in to the PSOWeb Applications (**https://localhost/webhmi** or **https://IPaddress/webhmi**).

   c. For example, in **Diagram Library** click on **controlop**, and then copy the highlighted URL.



https://localhost/WebHmi/#HmiApplication/#lib/controlop2etgml

   d. Go back to **Graphics Editor TGML** page, double-click on **Link > Object**, and then paste the copied URL.



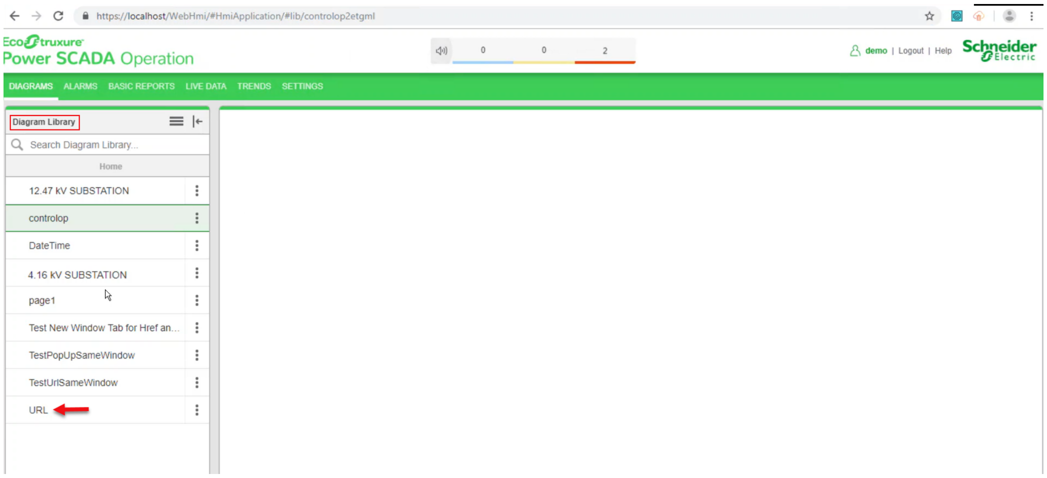
Objects

- Tgml
  - Component Button_002
    - Script
    - Expose ButtonColor
    - Expose ButtonEdge
    - Rectangle Background
    - Rectangle Hover
    - Rectangle Edge
    - Rectangle Light
    - Text TextShadow
    - Text Text
    - TargetArea
    - Link https://localhost/WebHmi/#HmiApplication/#lib/co
    - Script

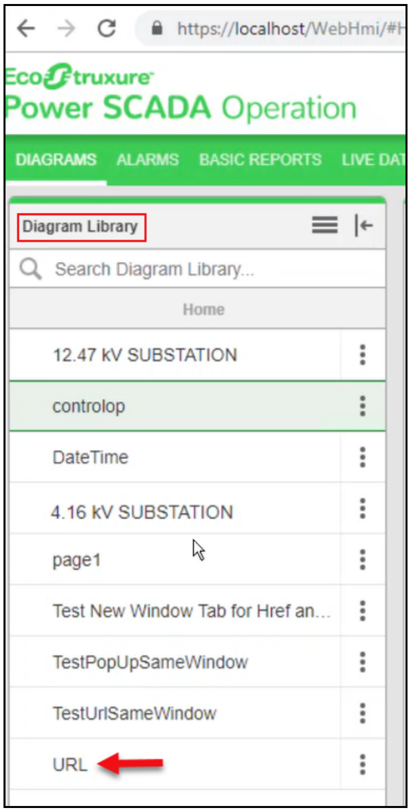
e. Click **Save** to save the TGML file (for example, file name is saved as **URL**)

To test the changes:

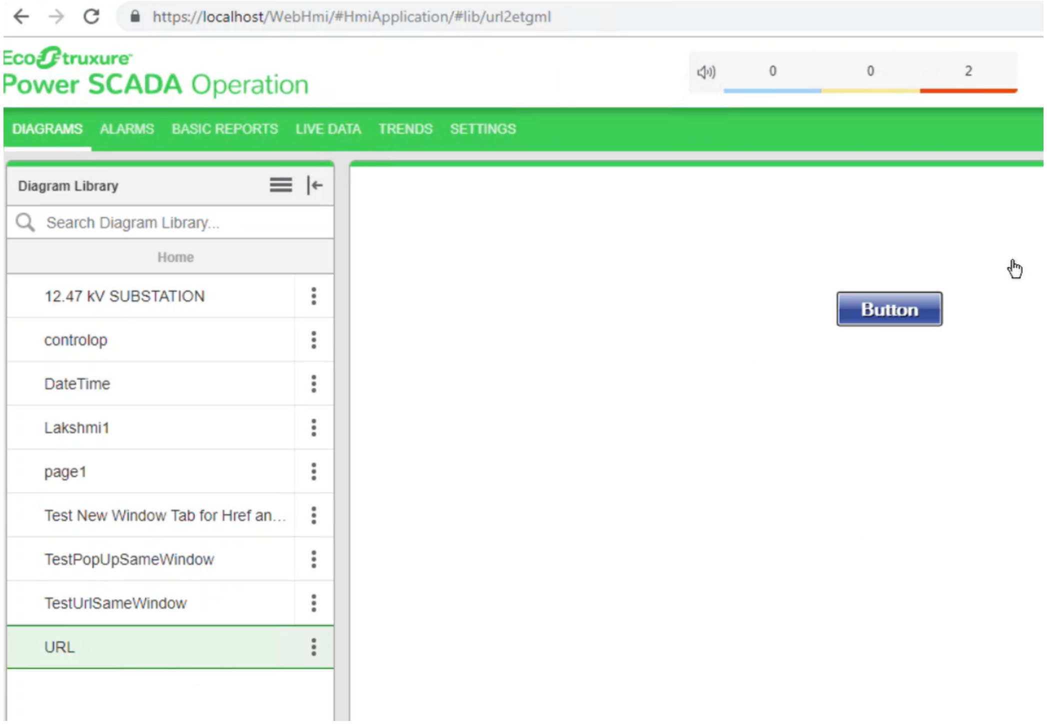
1. Go back to PSO Web Applications page that was already open, and then refresh the page.
2. **URL** saved graphic file name is displayed in the **Diagram Library** menu.



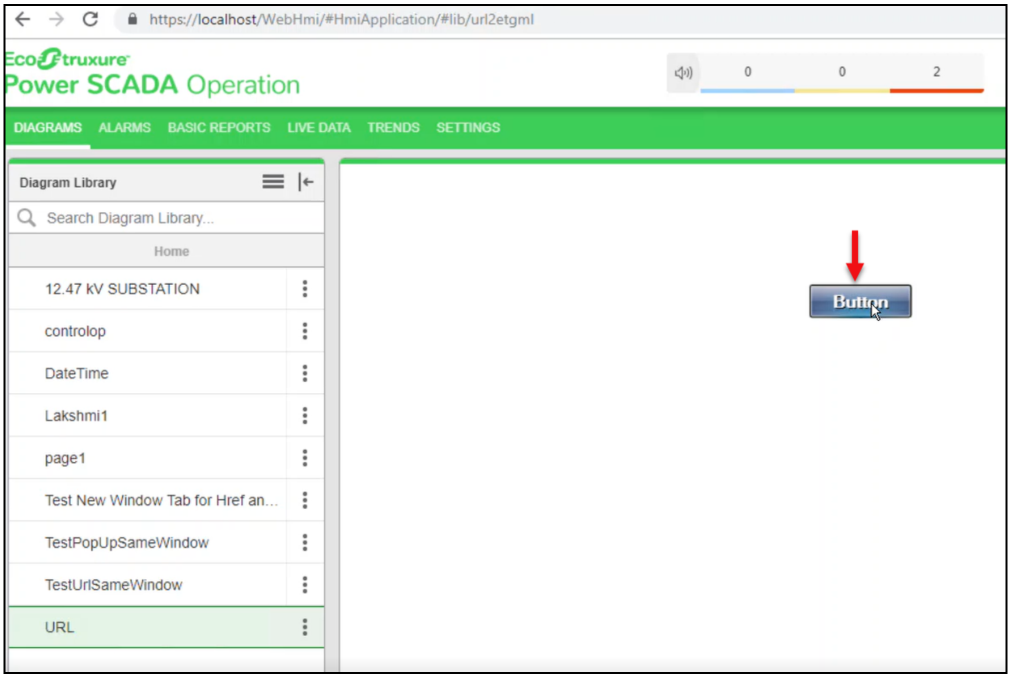
3. Click on **URL** in the **Diagram Library** menu.



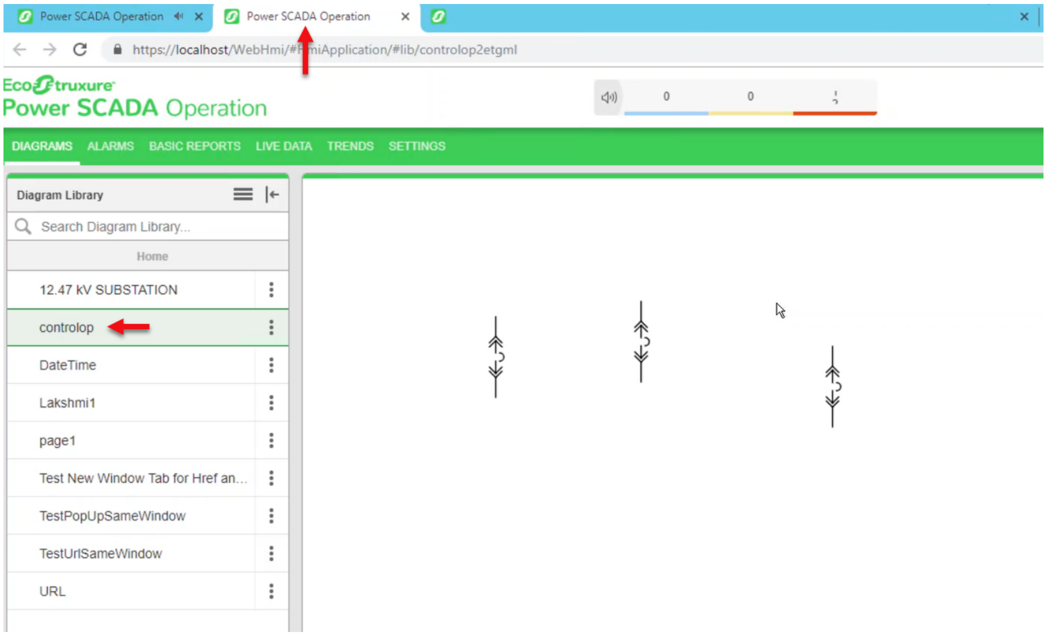
The following screen is displayed:


https://localhost/WebHmi/#HmiApplication/#lib/url2etgml
EcoStruxure
Power SCADA Operation
0
0
2
DIAGRAMS ALARMS BASIC REPORTS LIVE DATA TRENDS SETTINGS
Diagram Library
Search Diagram Library...
Home
12.47 kV SUBSTATION
controlop
DateTime
Lakshmi1
page1
Test New Window Tab for Href an...
TestPopUpSameWindow
TestUrlSameWindow
URL
Button

4. Click **Button**.



The following screen is displayed with the linked **URL** of **controlop** in a new tab:

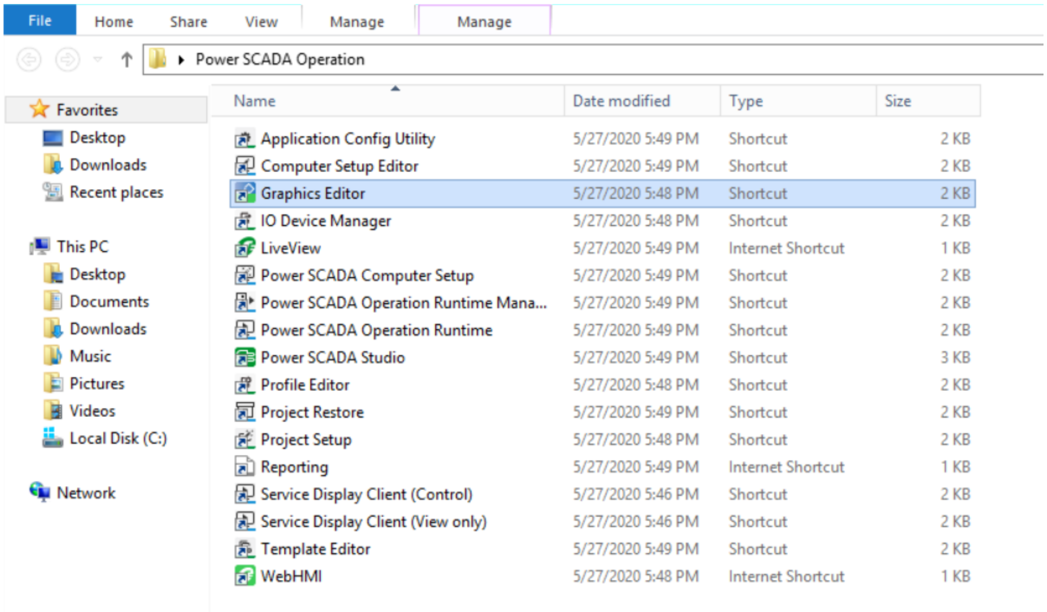


Opening URL links in Web Applications

You can open URL links from TGML graphics.

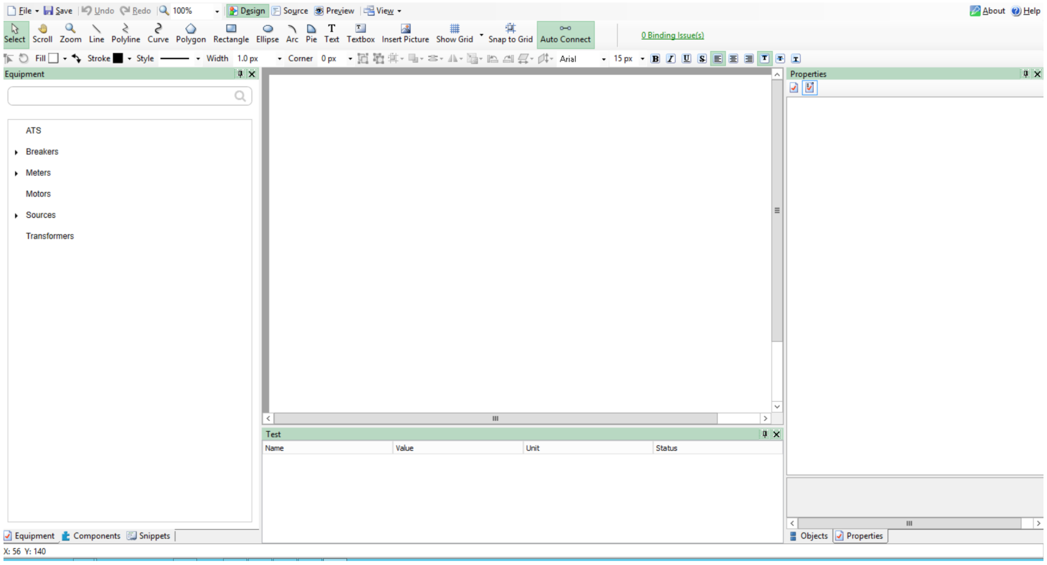
While you can add a URL link to any TGML component, in this example a button component is created and then linked to a URL. When the button is clicked in Web Applications, the URL displays.

1. Open the Graphics Editor from this location **C:\Users\Public\Desktop\Power SCADA Operation** by clicking on the **Graphics Editor** Icon.

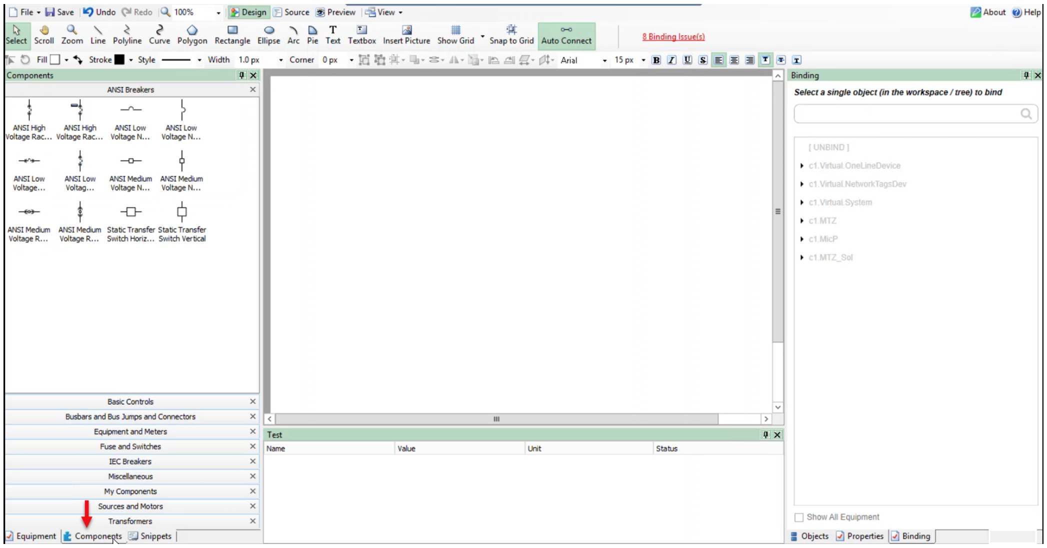


Name	Date modified	Type	Size
Application Config Utility	5/27/2020 5:49 PM	Shortcut	2 KB
Computer Setup Editor	5/27/2020 5:49 PM	Shortcut	2 KB
Graphics Editor	5/27/2020 5:48 PM	Shortcut	2 KB
IO Device Manager	5/27/2020 5:48 PM	Shortcut	2 KB
LiveView	5/27/2020 5:49 PM	Internet Shortcut	1 KB
Power SCADA Computer Setup	5/27/2020 5:49 PM	Shortcut	2 KB
Power SCADA Operation Runtime Mana...	5/27/2020 5:49 PM	Shortcut	2 KB
Power SCADA Operation Runtime	5/27/2020 5:49 PM	Shortcut	2 KB
Power SCADA Studio	5/27/2020 5:49 PM	Shortcut	3 KB
Profile Editor	5/27/2020 5:48 PM	Shortcut	2 KB
Project Restore	5/27/2020 5:49 PM	Shortcut	2 KB
Project Setup	5/27/2020 5:48 PM	Shortcut	2 KB
Reporting	5/27/2020 5:49 PM	Internet Shortcut	1 KB
Service Display Client (Control)	5/27/2020 5:46 PM	Shortcut	2 KB
Service Display Client (View only)	5/27/2020 5:46 PM	Shortcut	2 KB
Template Editor	5/27/2020 5:49 PM	Shortcut	2 KB
WebHMI	5/27/2020 5:48 PM	Internet Shortcut	1 KB

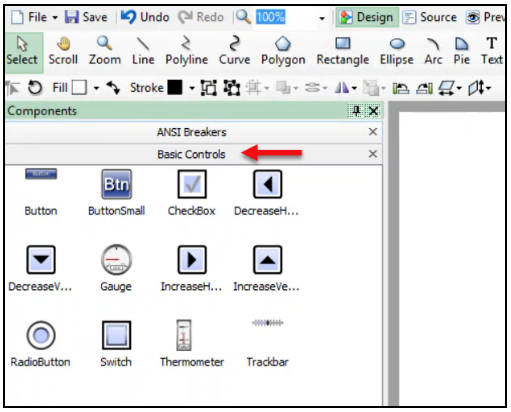
The following screen is displayed.



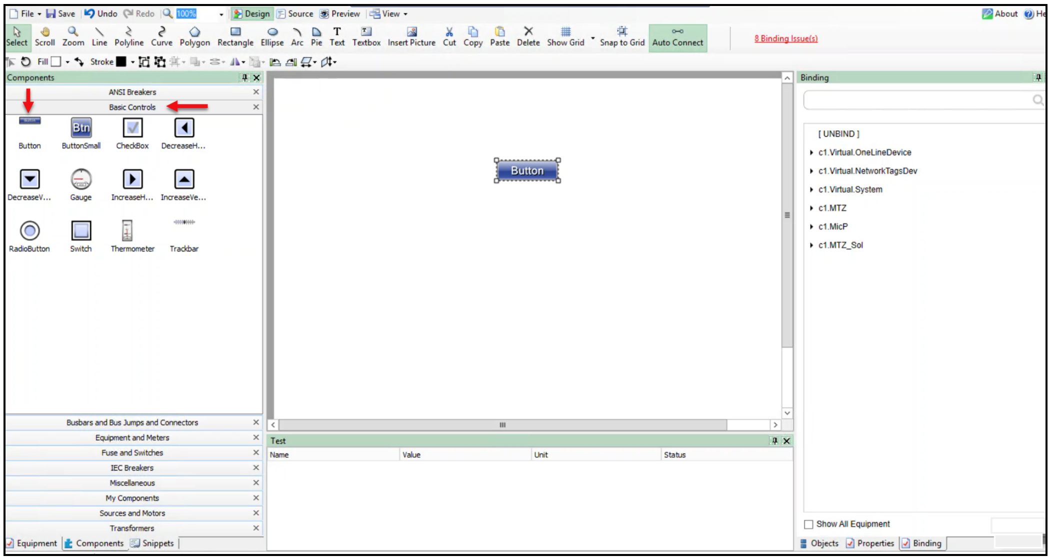
2. In the bottom left corner, click **Components**:



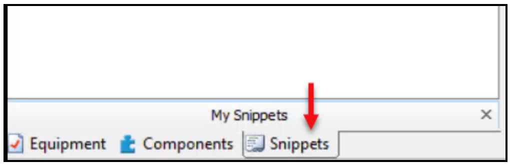
3. Click **Basic Controls**:



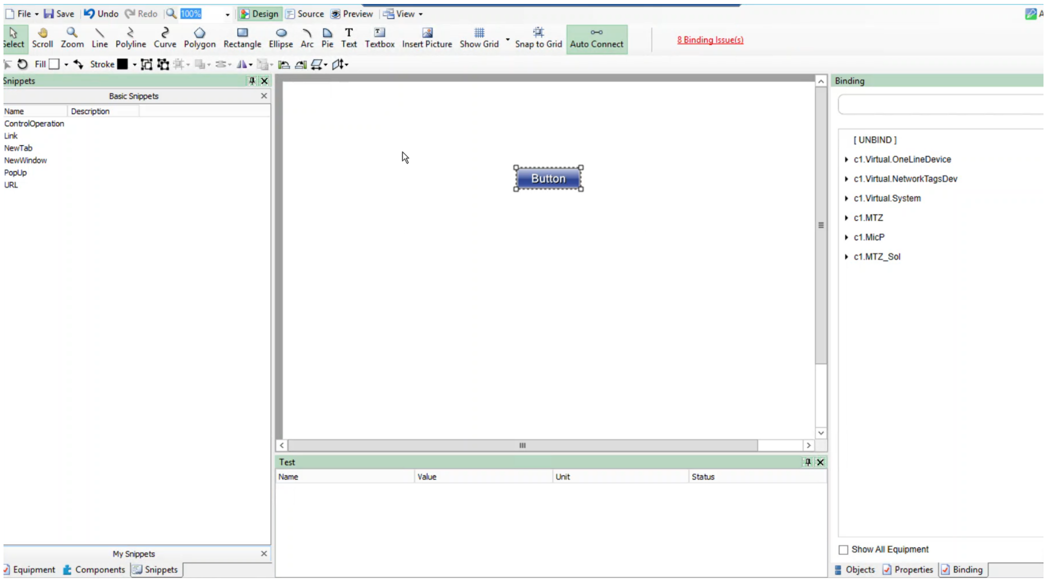
4. Drag and drop a **Button** component to the workspace:



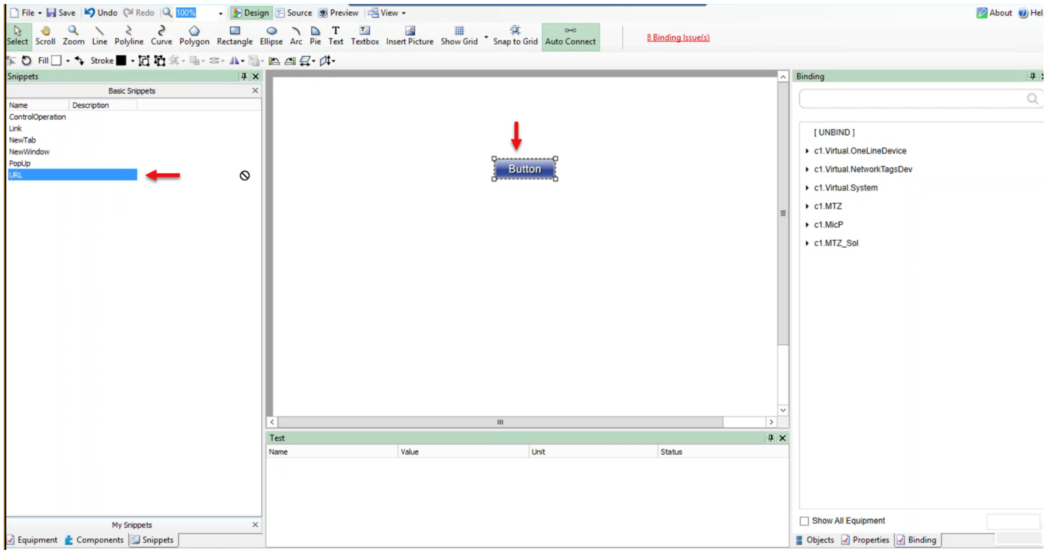
5. Click **Snippets**:



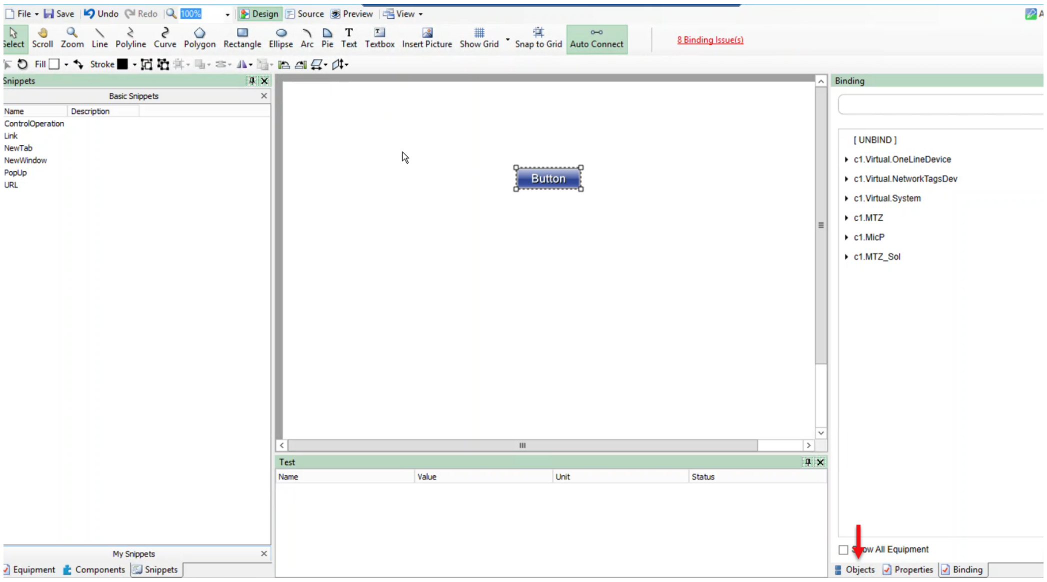
The following screen is displayed:



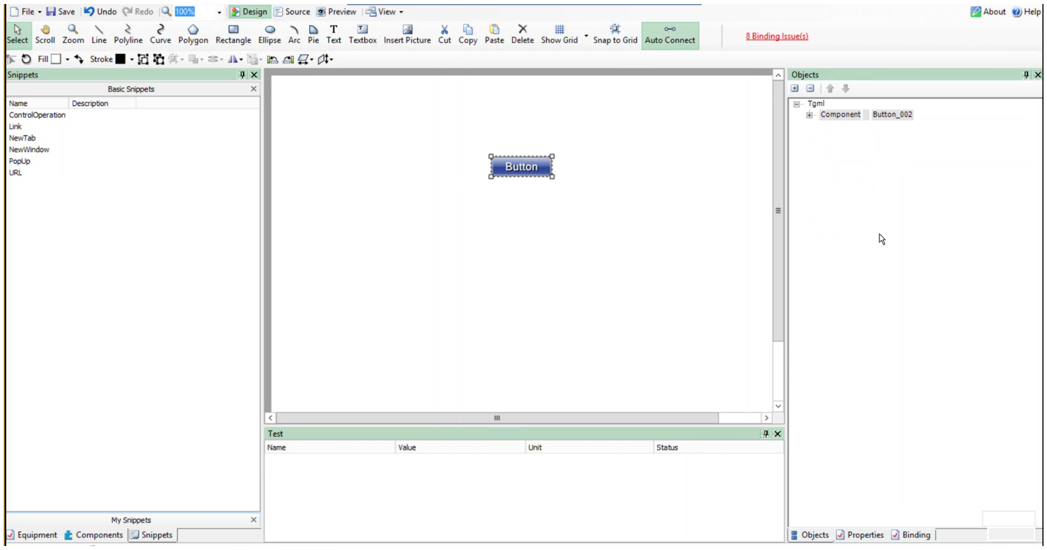
6. Drag and drop **URL** onto the **Button** component in the workspace.



7. In the bottom right corner, click **Objects**:

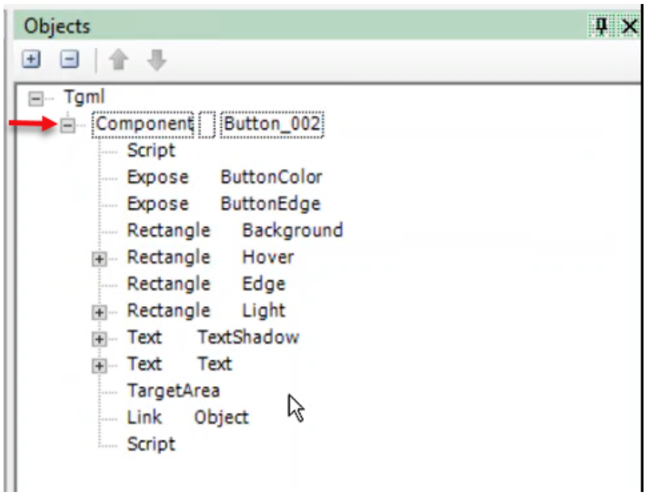


The following screen is displayed:



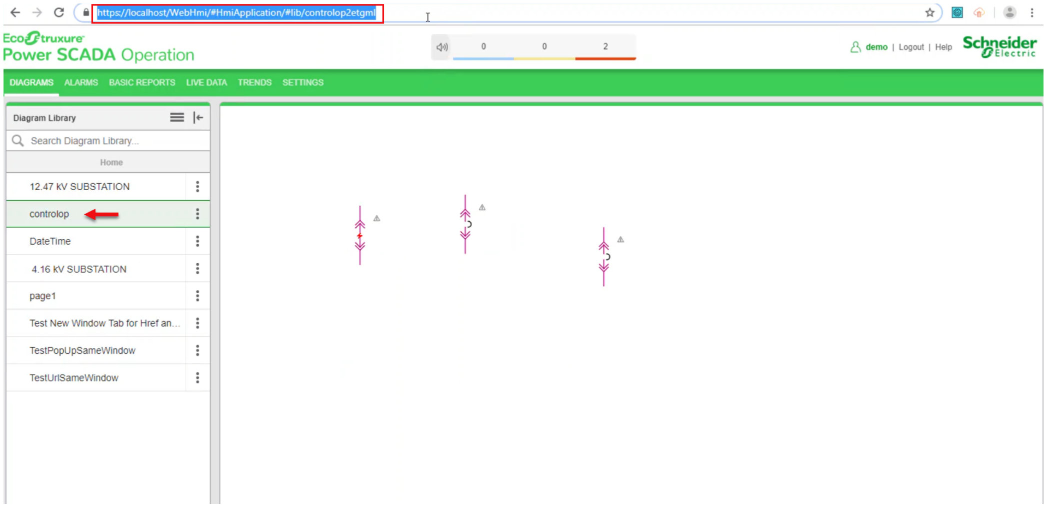
8. Set the Link attribute value:

   a. Expand the **Component** plus box to find the **Link** attribute within the button:

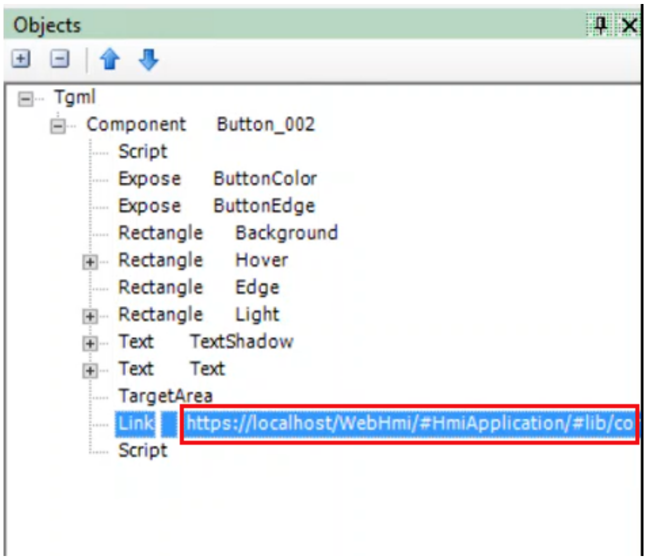


   b. Log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)

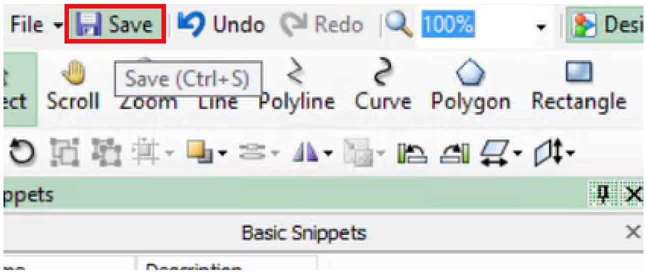
      For example, click on **controlop** in **Diagram Library** menu, and then copy the highlighted URL:


https://localhost/WebHmi/#HmiApplication/#lib/controlop2etgml

c. Go back to **Graphics Editor TGML** page, double-click on **Link > Object**, and then paste the copied URL:



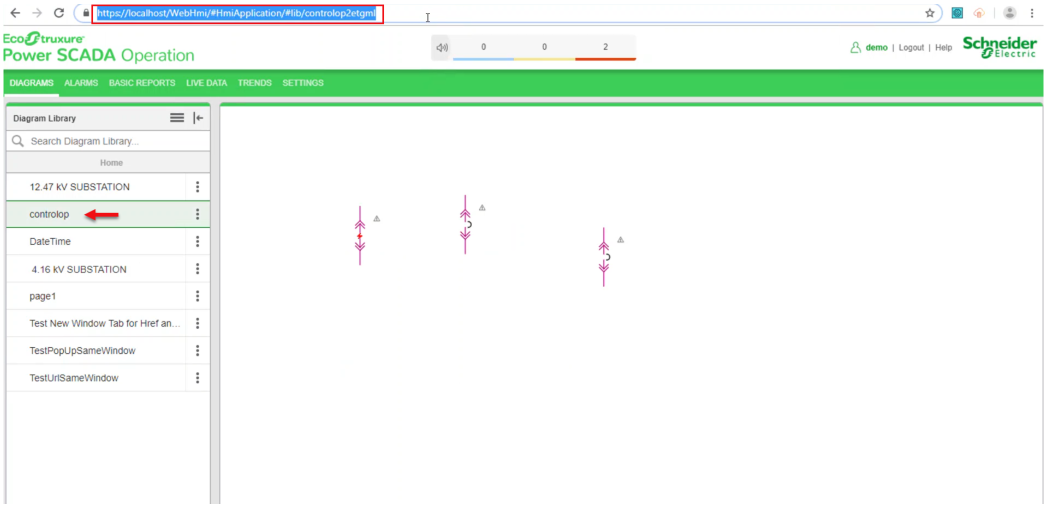
9. Click **Save** to save the TGML file. For example: URL



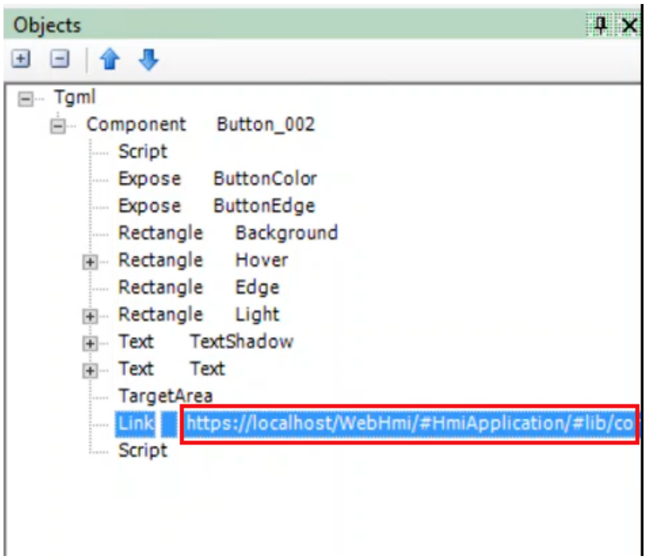
Test the changes:

1. Log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)

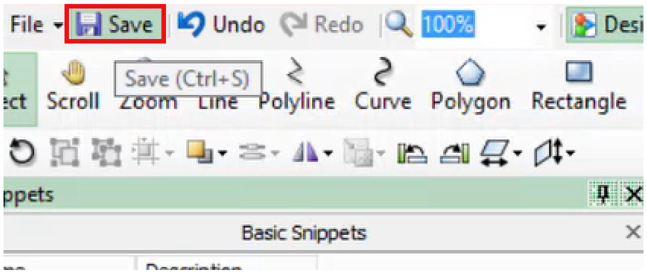
For example, click on **controlop** in **Diagram Library** menu, and then copy the highlighted URL.



2. Go back to **Graphics Editor TGML** page, double-click on **Link > Object**, and then paste the copied URL.

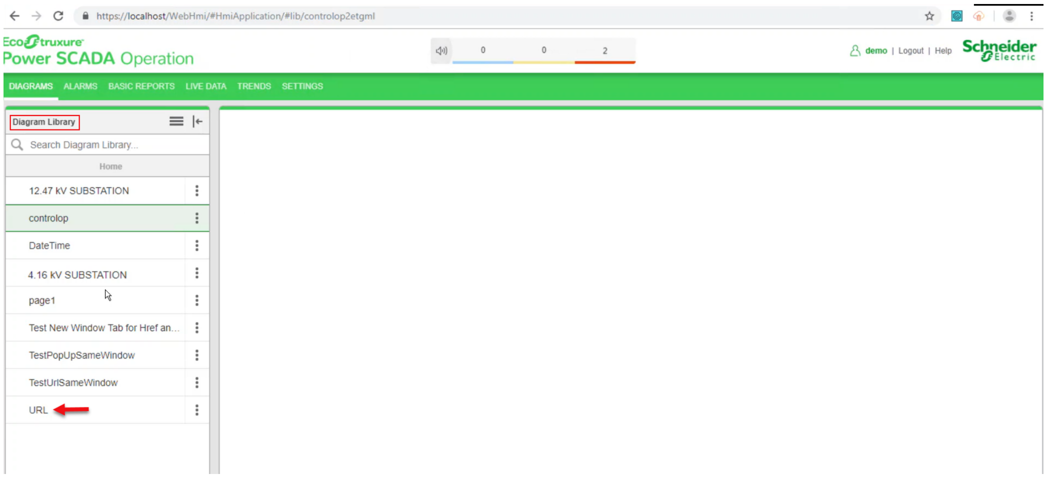


3. Click **Save** to save the TGML file. For example: URL

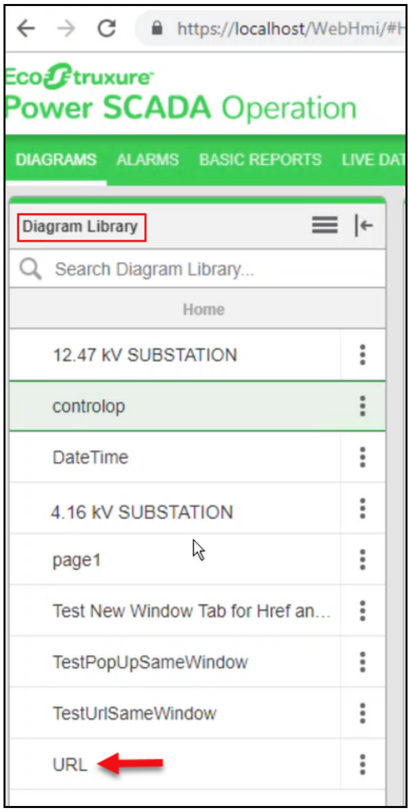


4. Go back to PSO Web Applications page which was already open, and then refresh the page.

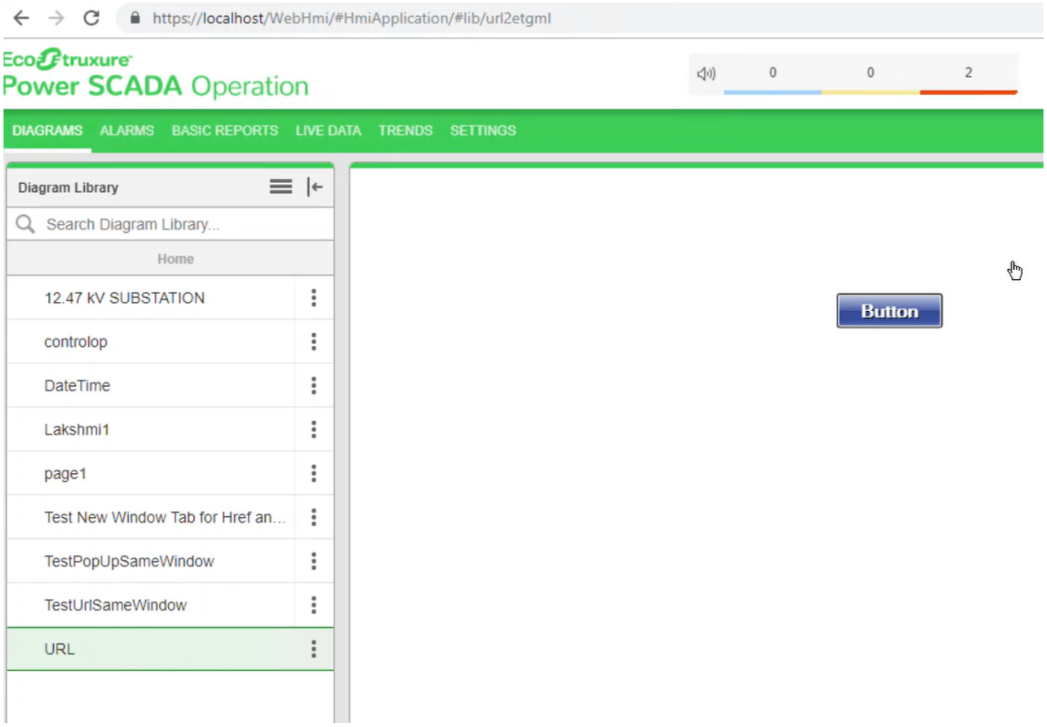
URL saved graphic file name is displayed in the **Diagram Library** menu.



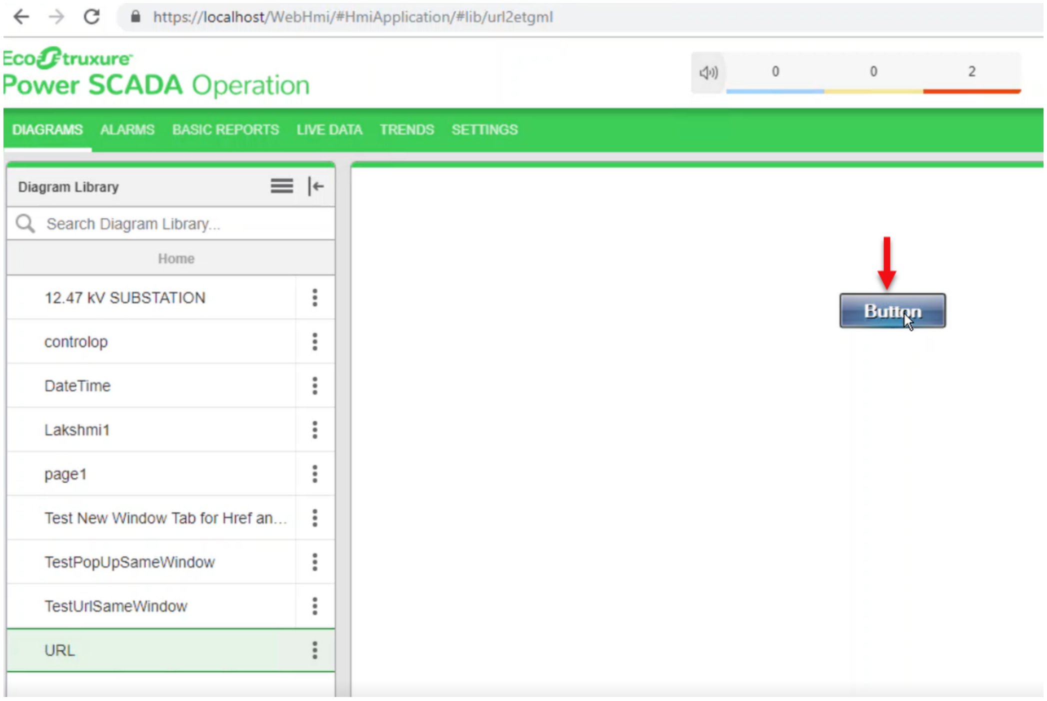
5. In the **Diagram Library** menu, click **URL**:



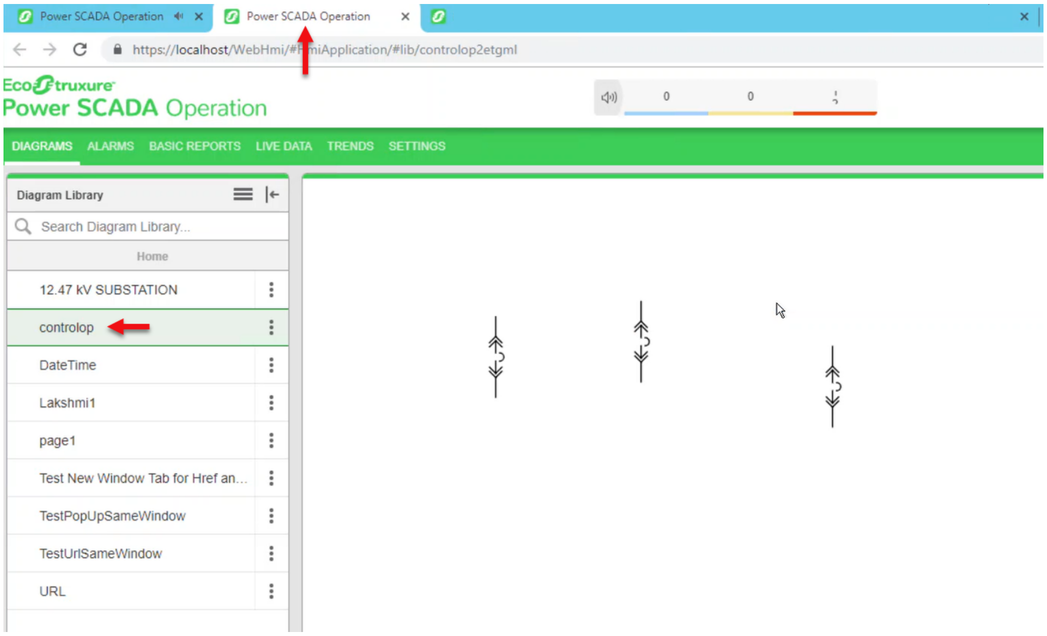
The following screen is displayed:



6. Click the **Button**:



The following output screen is displayed with the linked **URL** of **controlop** in new tab.



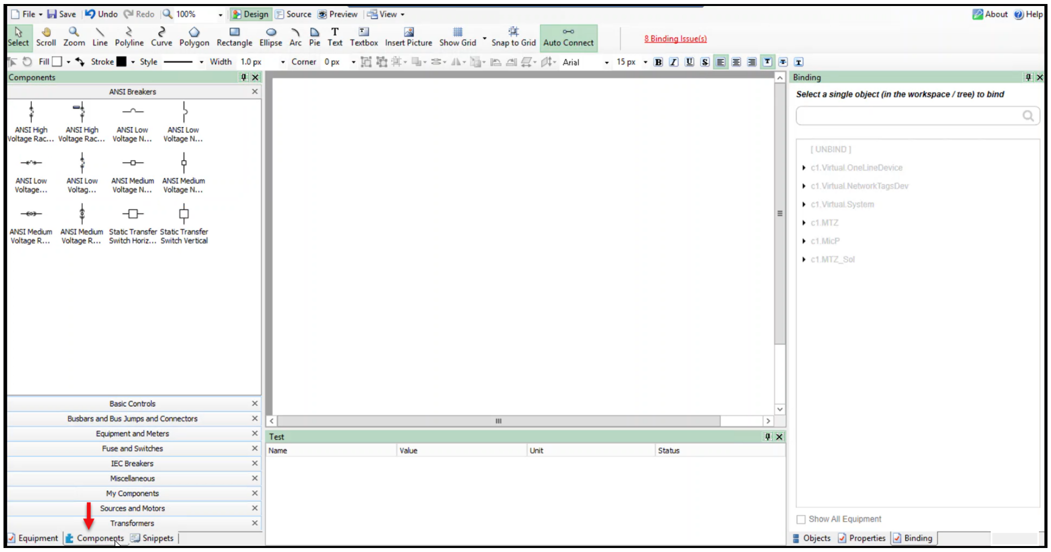
Invoke function

You can configure TGML graphic components in the Graphic Editor to open a linked target object to a target location. You can do this using the invoke function in a script. When you perform a specific action like PopUp, NewWindow, Link, NewTab, URL, Control on the component, the `invoke` function is used.

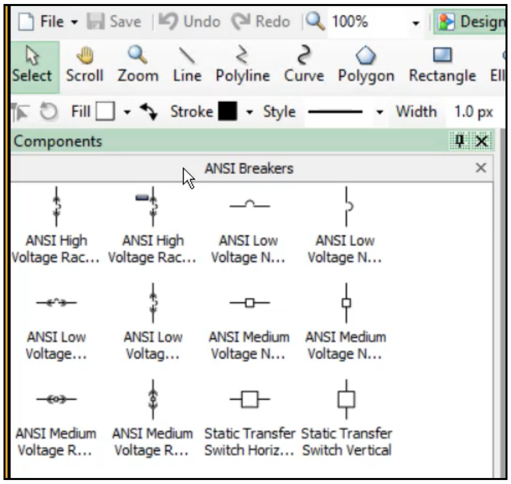
This topic includes an example to illustrate how to use the invoke function, and also details the invoke function and attributes.

To use the invoke function to open a pop-up:

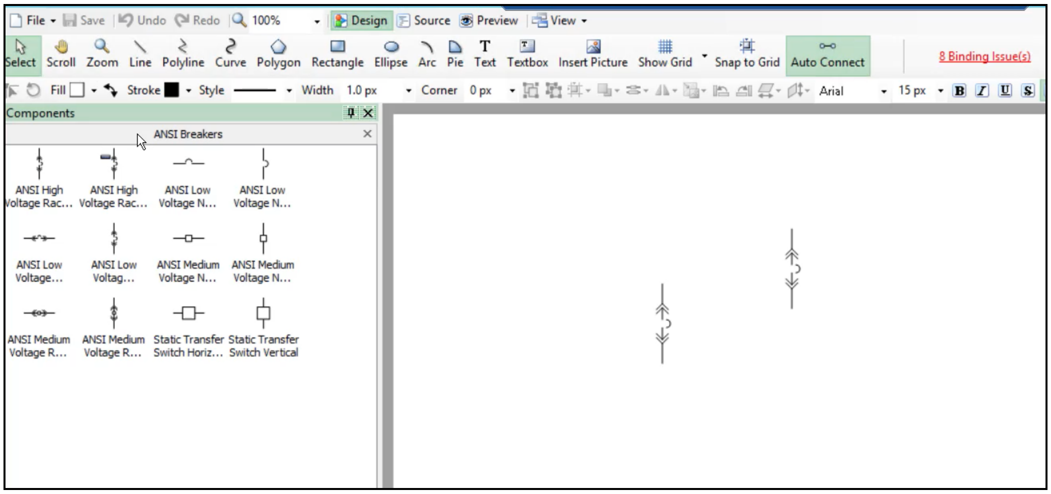
1. Open the Graphics Editor: Go to Start > Power SCADA Operation > Graphics Editor.
2. In the bottom left corner, click **Components**.



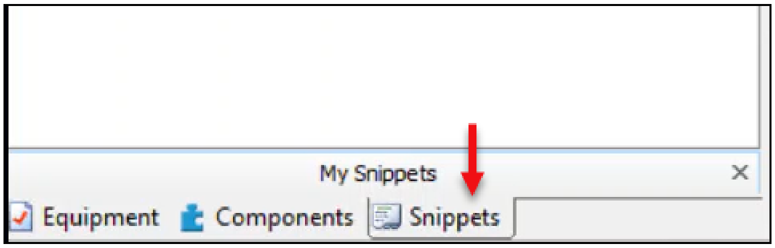
3. Click on **ANSI Breakers**.



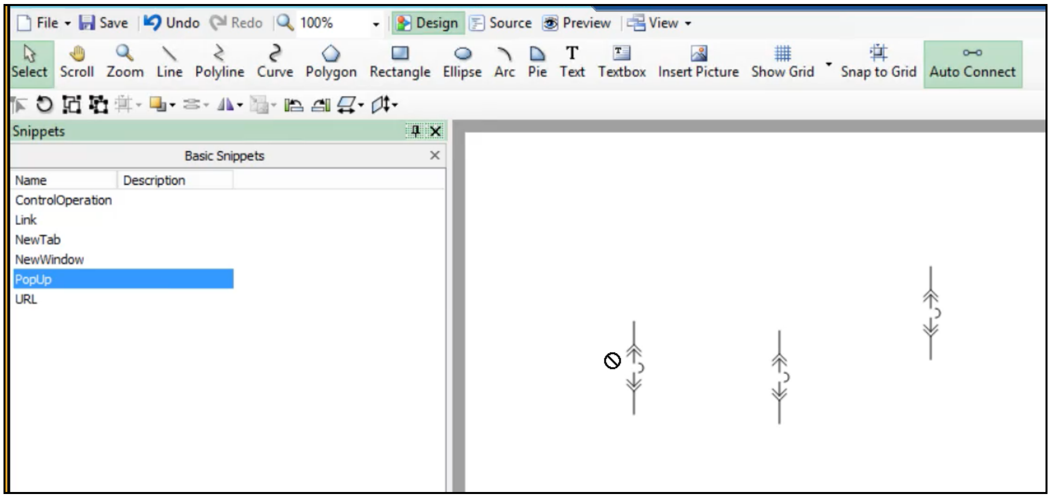
4. Drag and drop **Breakers** onto the workspace.



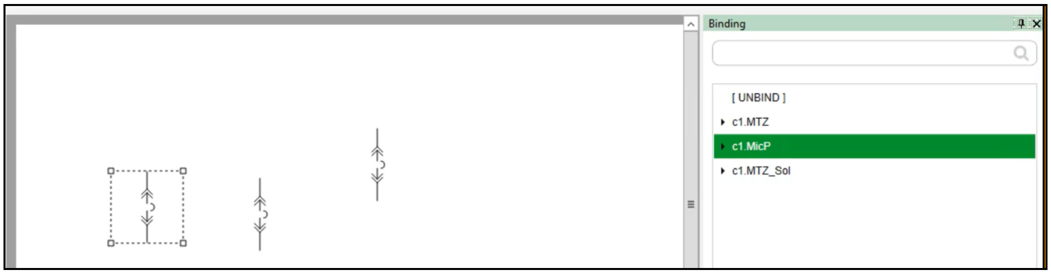
5. In the bottom left corner, click **Snippets**.



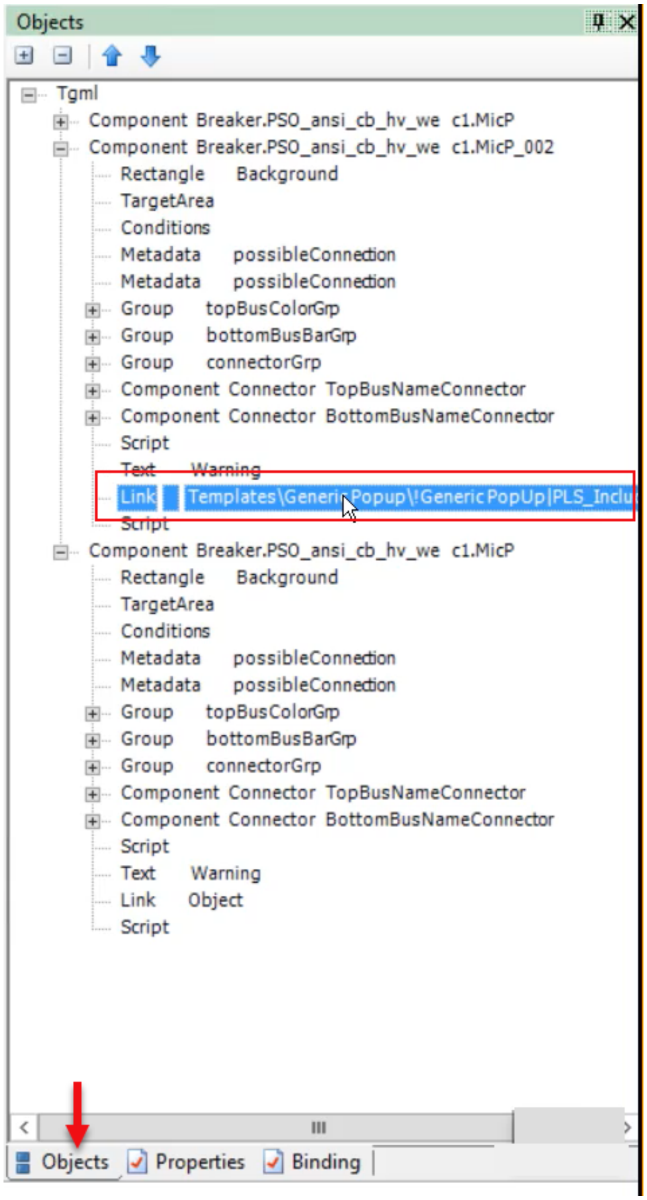
6. Drag and drop a **PopUp** snippet onto the breakers.



7. Select the **Breaker** and click on required breaker name to bind.



8. In the bottom right corner, click **Objects**, click on each breaker components and type the path in the **Link**:



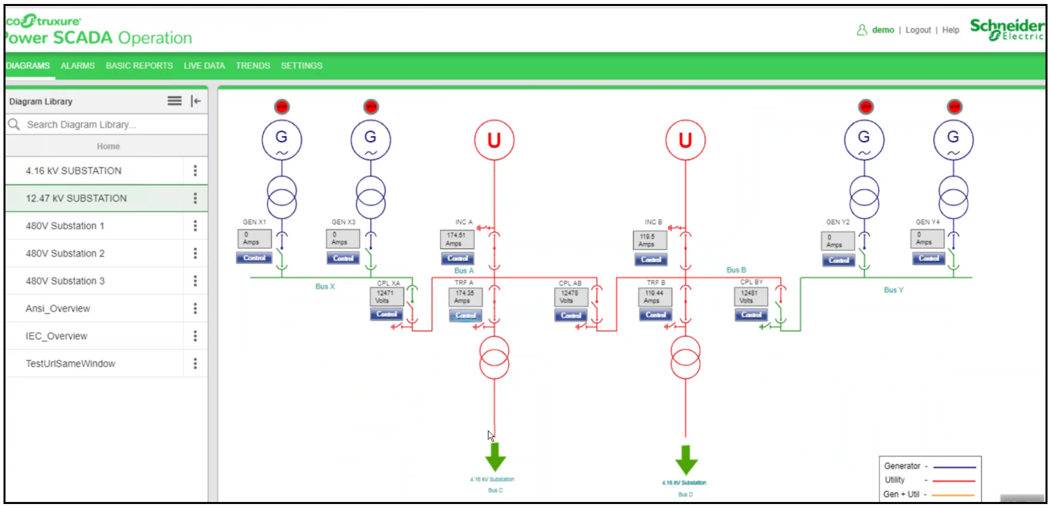
```
Tgml
    Component Breaker.PSO_ansi_cb_hv_we c1.MicP
    Component Breaker.PSO_ansi_cb_hv_we c1.MicP_002
        Rectangle   Background
        TargetArea
        Conditions
        Metadata   possibleConnection
        Metadata   possibleConnection
        Group   topBusColorGrp
        Group   bottomBusBarGrp
        Group   connectorGrp
        Component Connector TopBusNameConnector
        Component Connector BottomBusNameConnector
        Script
        Text   Warning
        Link   Templates\GenericPopup\!GenericPopUp|PLS_Inclu
        Script
    Component Breaker.PSO_ansi_cb_hv_we c1.MicP
        Rectangle   Background
        TargetArea
        Conditions
        Metadata   possibleConnection
        Metadata   possibleConnection
        Group   topBusColorGrp
        Group   bottomBusBarGrp
        Group   connectorGrp
        Component Connector TopBusNameConnector
        Component Connector BottomBusNameConnector
        Script
        Text   Warning
        Link   Object
        Script
```

9. Save the TGML graphic file.

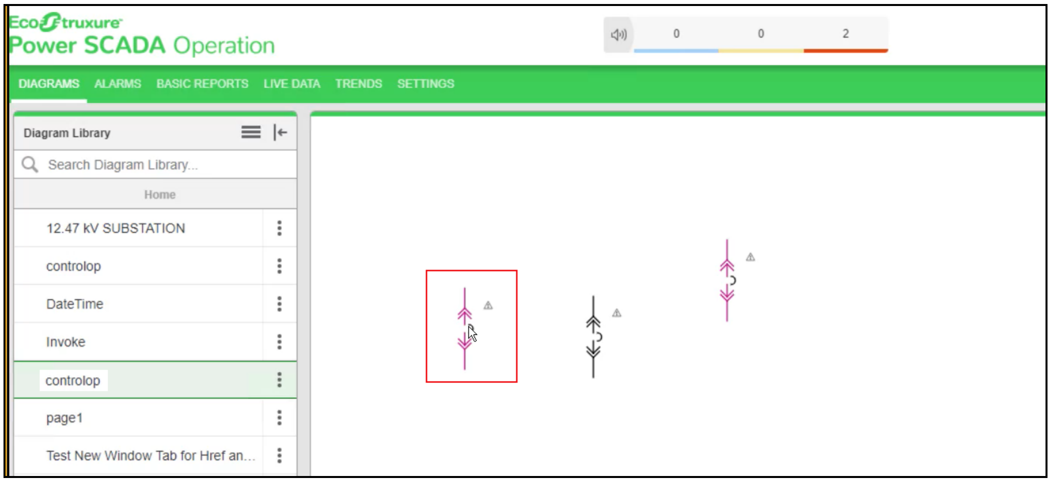
To test the changes:

1. Log in to PSOWeb Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)

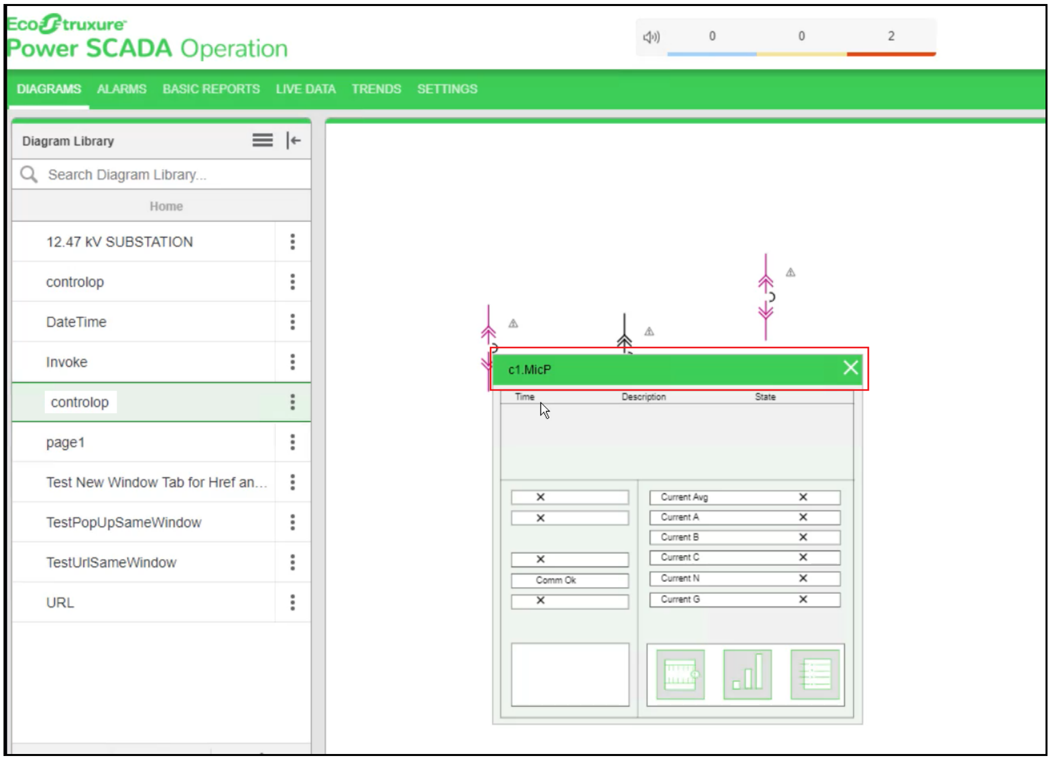
   The following screen is displayed:



2. Click on **Breaker**.


EcoStruxure Power SCADA Operation
0
0
2
DIAGRAMS ALARMS BASIC REPORTS LIVE DATA TRENDS SETTINGS
Diagram Library
Search Diagram Library...
Home
12.47 kV SUBSTATION
controlop
DateTime
Invoke
controlop
page1
Test New Window Tab for Href an...

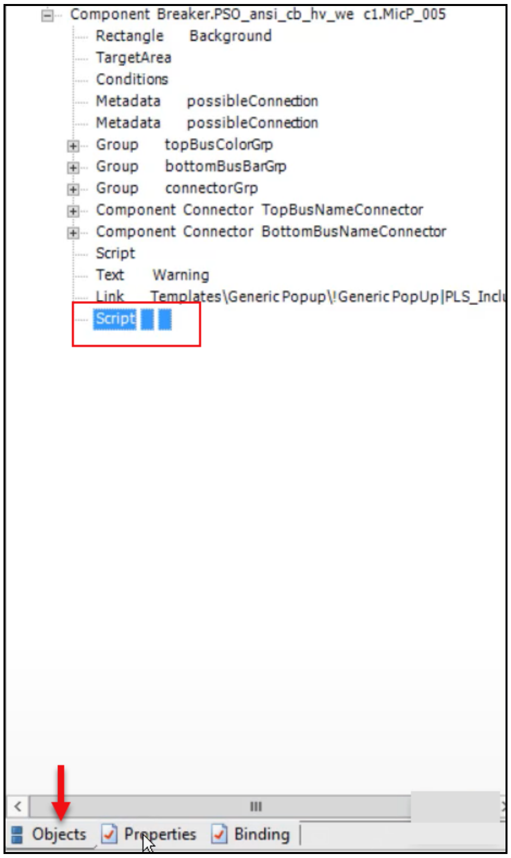
The following screen is displayed with title bar:



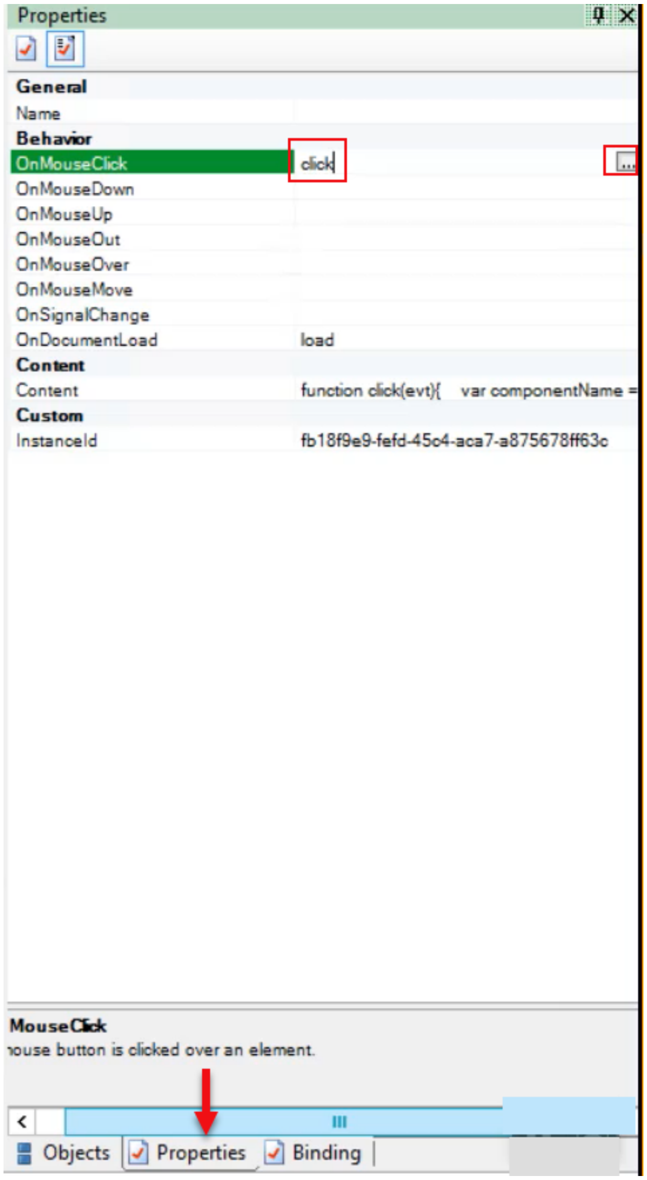
Invoke function description

1. In the Graphics Editor, go back to TGML graphics page that was created already.

2. In the bottom right corner, click **Object**, and then click on **Script** for each component.


Component Breaker.PSO_ansi_cb_hv_we c1.MicP_005
Rectangle Background
TargetArea
Conditions
Metadata possibleConnection
Metadata possibleConnection
Group topBusColorGrp
Group bottomBusBarGrp
Group connectorGrp
Component Connector TopBusNameConnector
Component Connector BottomBusNameConnector
Script
Text Warning
Link Templates\Generic Popup\!Generic PopUp|PLS_Inclu
Script
Objects
Properties
Binding

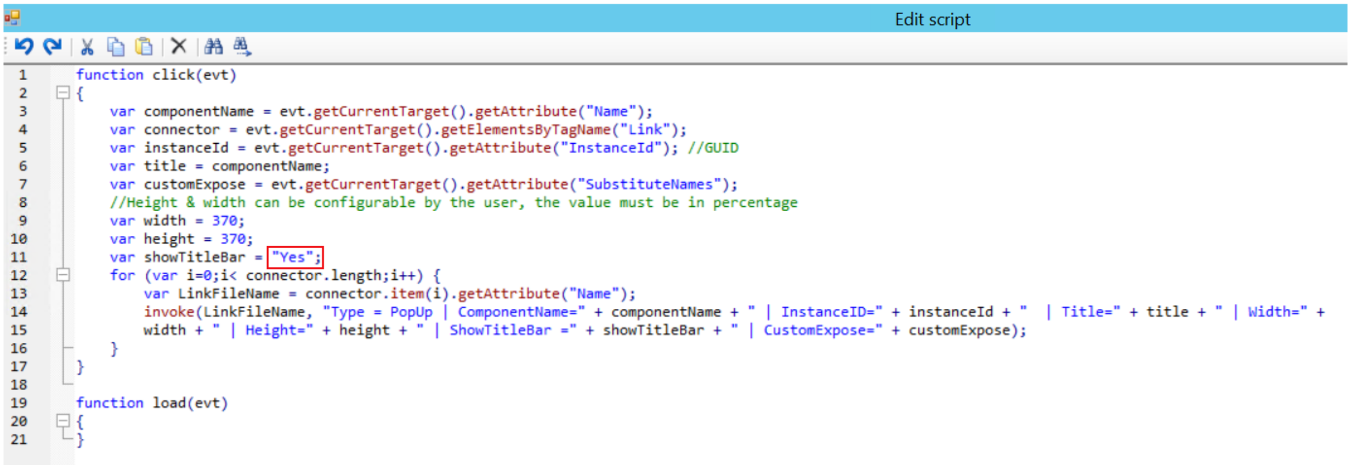
3. In the bottom right corner, click **Properties**, click on **OnMouseClick** script, and then click on the ellipsis.



General	
Name	
Behavior	
OnMouseClick	click
OnMouseDown	
OnMouseUp	
OnMouseOut	
OnMouseOver	
OnMouseMove	
OnSignalChange	
OnDocumentLoad	load
Content	
Content	function click(evt){ var componentName =
Custom	
InstanceId	fb18f9e9-fefd-45c4-aca7-a875678ff63c

MouseClick
nouse button is clicked over an element.

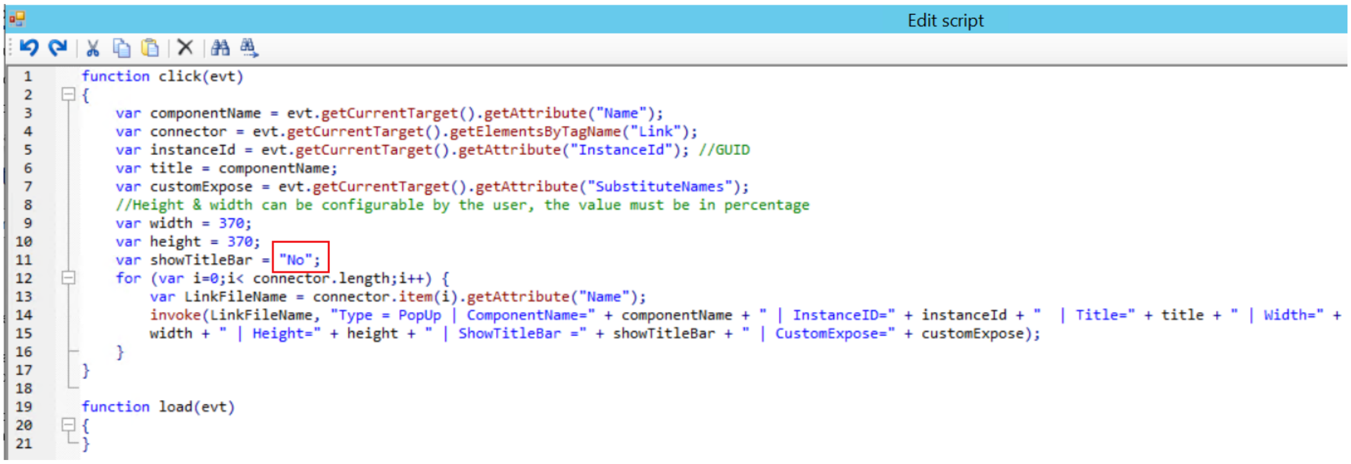
The following screen is displayed.



```
function click(evt)
{
    var componentName = evt.getCurrentTarget().getAttribute("Name");
    var connector = evt.getCurrentTarget().getElementsByTagName("Link");
    var instanceId = evt.getCurrentTarget().getAttribute("InstanceId"); //GUID
    var title = componentName;
    var customExpose = evt.getCurrentTarget().getAttribute("SubstituteNames");
    //Height & width can be configurable by the user, the value must be in percentage
    var width = 370;
    var height = 370;
    var showTitleBar = "Yes";
    for (var i=0;i< connector.length;i++) {
        var LinkFileName = connector.item(i).getAttribute("Name");
        invoke(LinkFileName, "Type = PopUp | ComponentName=" + componentName + " | InstanceID=" + instanceId + "  | Title=" + title + " | Width=" +
        width + " | Height=" + height + " | ShowTitleBar =" + showTitleBar + " | CustomExpose=" + customExpose);
    }
}

function load(evt)
{
}
```

NOTE: The invoke function integrates for all snippets by default. You can customize this function based on your requirements.

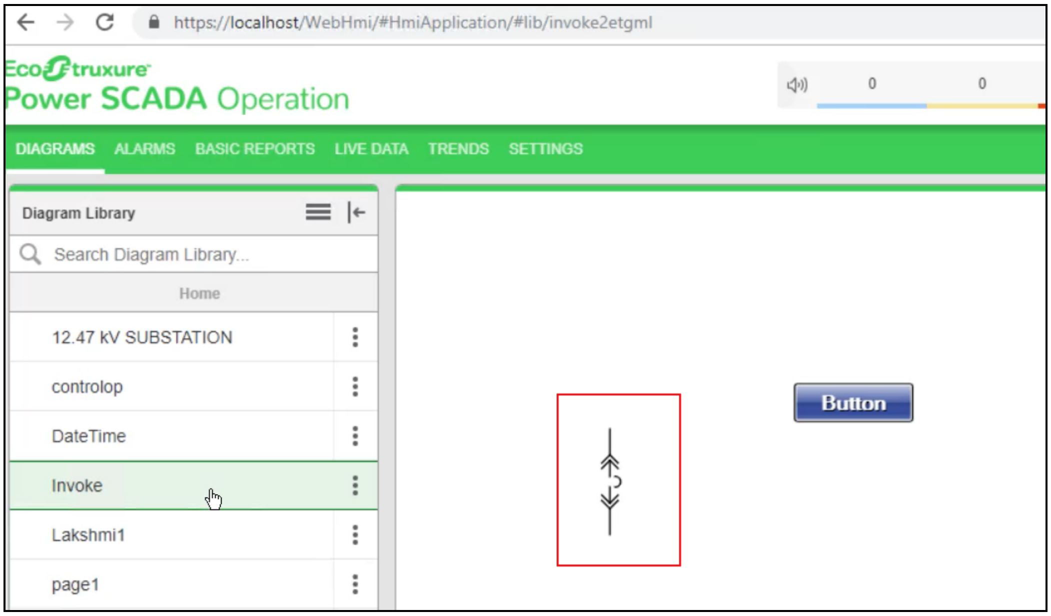
For example: if the title bar name in the web page PopUp output does not need to be displayed, type **No** in **showTitleBar** variable, and then click **OK**.



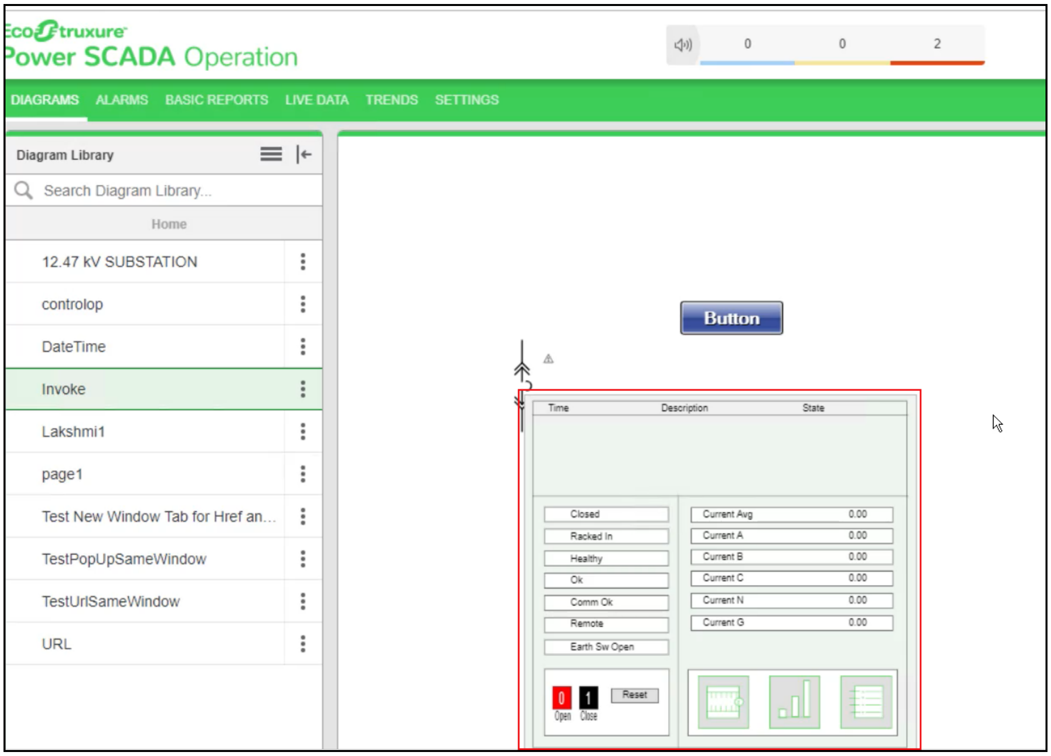
```
function click(evt)
{
    var componentName = evt.getCurrentTarget().getAttribute("Name");
    var connector = evt.getCurrentTarget().getElementsByTagName("Link");
    var instanceId = evt.getCurrentTarget().getAttribute("InstanceId"); //GUID
    var title = componentName;
    var customExpose = evt.getCurrentTarget().getAttribute("SubstituteNames");
    //Height & width can be configurable by the user, the value must be in percentage
    var width = 370;
    var height = 370;
    var showTitleBar = "No";
    for (var i=0;i< connector.length;i++) {
        var LinkFileName = connector.item(i).getAttribute("Name");
        invoke(LinkFileName, "Type = PopUp | ComponentName=" + componentName + " | InstanceID=" + instanceId + "  | Title=" + title + " | Width=" +
        width + " | Height=" + height + " | ShowTitleBar =" + showTitleBar + " | CustomExpose=" + customExpose);
    }
}

function load(evt)
{
}
```

4. Go back to Web Applications page that was already opened.

5. Click on **Breaker**:



The following screen is displayed with no title bar:



Invoke function attributes

The following table lists the `invoke` function attributes for each snippet type:

NOTE: The `invoke` function attributes vary depending on the snippet type.

Snippet type	Syntax	Attributes Description
PopUp	invoke (LinkFileName, "Type = PopUp | ComponentName=" + componentName + " | InstanceID=" + instanceId + ” | Title=" + title + " | Width=" + width + " | Height=" + height + " | ShowTitleBar =" + showTitleBar + " | CustomExpose=" + customExpose); **NOTE:** LinkFileName: Extracting the file name from the Link	**Required Attributes for snippet type Pop-Up** **Type** :Type name displays based on the snippet type selection (Type = PopUp). **ComponentName**: Component name is the device name. **Title**: This attribute will enable or disable the title bar. **Width**: Used to set the width of the PopUp. **Height**: Used to set the height of the PopUp. **InstanceID**: It is auto generated new instance id for each component. **Optional Attributes for snippet type Pop-Up** **ShowTitleBar**: Displays the Title Bar in the target pane when set to **Yes**. **CustomExpose**: This attribute contains multi equipment data.

Snippet type	Syntax	Attributes Description
NewWindow	invoke (LinkFileName, "Type = NewWindow | ComponentName=" + componentName + " | Title=" + title + " | Width=" + width + " | Height=" + height + " | CustomExpose=" + customExpose); **NOTE:** LinkFileName: Extracting the file name from the Link	**Required Attributes for snippet type NewWindow** **Type**:Type name displays based on the snippet type selection (Type = NewWindow). **ComponentName**: Component name is the device name. **Title**: This attribute will enable or disable the title bar. **Width**: Used to set the width of the NewWindow. **Height**: Used to set the height of the NewWindow. **InstanceID**: It is auto generated new instance id for each component PopUp selection. **Optional Attributes for snippet type NewWindow** **CustomExpose**: This attribute contains multi equipment data.
URL	invoke(LinkFileName, "Type = Href | HrefSameWindow = "+sameWindow); **NOTE:** LinkFileName: Extracting the file name from the Link	**Required Attributes for snippet type URL** **Type**: Type name displays based on the snippet type selection (Type = Href). **Optional Attributes for snippet type URL** **HrefSameWindow**: When this value is No, the URL opens in new window. Syntax: var sameWindow = "No".

Snippet type	Syntax	Attributes Description
Link	invoke (LinkFileName, "Type = Link | ComponentName=" + componentName + " | CustomExpose=" + customExpose); **NOTE:** LinkFileName: Extracting the file name from the Link	**Required Attributes for snippet Type Link** **Type** :Type name displays based on the snippet type selection (Type = Link). **ComponentName**: Component name is the device name.
NewTab	invoke (LinkFileName, "Type = NewTab | ComponentName=" + componentName + " | Title=" + title + " | CustomExpose=" + customExpose); **NOTE:** LinkFileName: Extracting the file name from the Link	**Required Attributes for snippet type NewTab** **Type** :Type name displays based on the snippet type selection (Type = NewTab). **ComponentName**: Component name is the device name. **Title**: This attribute will enable or disable the title bar. **Optional Attributes for snippet Type NewTab** **CustomExpose**: This attribute contains multi equipment data.

Snippet type	Syntax	Attributes Description
URL in Same Window	invoke(LinkFileName, "Type = Href | HrefSameWindow = "+sameWindow); **NOTE:** LinkFileName: Extracting the file name from the Link	**Required Attributes for snippet type URL in Same Window** **Type** :Type name displays based on the snippet type selection (Type = Href). **Optional Attributes for snippet Type URL in Same Window** **HrefSameWindow**: Update the variable sameWindow as Yes to display URL in the samewindow or update the variable sameWindow as No to display URL in the different window. **Syntax**: var sameWindow = "Yes";

Snippet type	Syntax	Attributes Description
Control	invoke(LinkFileName, "Type = PopUp | ComponentName=" + componentName + " | InstanceID=" + instanceId + " | DataPoint = "+ dataPoint +"| Title=" + title + " | Width=" + width + " | Height=" + height + " | ShowTitleBar =" + showTitleBar + " | ShowUnamePwd =" + showUnamePwd +" | UserCredBottom = "+usercredbottom +" | UserCredLeft = "+ usercredleft+" | UserCredWidth = "+ usercredwidth +" | UserCredHeight = "+usercredheight +" | UserCredBackColor = "+usercredbackcolor+" | UnamePwdWidth = "+unamepwdwidth+" | UnamePwdColor = "+unamepwdcolor); **NOTE:** LinkFileName: Extracting the file name from the Link	**Required Attributes for snippet type Pop-Up** **Type** :Type name displays based on the snippet type selection (Type = PopUp). **ComponentName**: Component name is the device name. **Title**: This attribute will enable or disable the title bar. **Width**: Used to set the width of the Control. **Height**: Used to set the height of the Control. **InstanceID**: It is auto generated new instance id for each component. **DataPoint**: It is the item name to do the write operation. **UserCredBottom**: The vertical position of a positioned element. Sets the bottom of the user credential PopUp. **UserCredLeft**: The horizontal position of a positioned element. Sets the left of the user credential PopUp. **UserCredWidth**: Sets the width of the user credential PopUp. **UserCredHeight**: Sets the height of the user credential PopUp. **UserCredBackColor**: Sets the background color of the user credential PopUp. **UnamePwdWidth**: Sets the width user credential PopUp username and password. **UnamePwdColor**: Sets the color user credential PopUp username

Snippet type	Syntax	Attributes Description
		and password. **Optional Attributes for snippet Type Pop-Up** **ShowTitleBar**: Displays the title bar in the target pane when set to **Yes**.

Snippet type	Syntax	Attributes Description
User credential Pop-Up	invoke(LinkFileName, "Type = PopUp | ComponentName=" + componentName + " | InstanceID=" + instanceId + " | DataPoint = "+ dataPoint +"| Title=" + title + " | Width=" + width + " | Height=" + height + " | ShowTitleBar =" + showTitleBar + " | ShowUnamePwd =" + showUnamePwd +" | UserCredBottom = "+usercredbottom +" | UserCredLeft = "+ usercredleft+" | UserCredWidth = "+ usercredwidth +" | UserCredHeight = "+usercredheight +" | UserCredBackColor = "+usercredbackcolor+" | UnamePwdWidth = "+unamepwdwidth+" | UnamePwdColor = "+unamepwdcolor); **NOTE:** LinkFileName: Extracting the file name from the Link	**Required Attributes for snippet Type User credential PopUp** **Type** :Type name displays based on the snippet type selection (Type = PopUp). **ComponentName**: Component name is the device name. **Title**: This attribute will enable or disable the title bar. **Width**: Used to set the width of the User credential Pop-Up. **Height**: Used to set the height of the User credential Pop-Up. **InstanceID**: It is auto generated new instance id for each component. **DataPoint**: It is the item name to do the write operation. **UserCredBottom**: The vertical position of a positioned element. Sets the bottom of the user credential PopUp. **UserCredLeft**: The horizontal position of a positioned element. Sets the left of the user credential PopUp. **UserCredWidth**: Sets the width of the user credential PopUp. **UserCredHeight**: Sets the height of the user credential PopUp. **UserCredBackColor**: Sets the background color of the user credential PopUp. **UnamePwdWidth**: Sets the width user credential PopUp username and password. **UnamePwdColor**: Sets the color user credential PopUp username

Snippet type	Syntax	Attributes Description
		and password. **Optional Attributes for snippet Type User credential PopUp** **ShowTitleBar**: Displays the title bar in the target pane when set to **Yes**.

Adding a diagram to the menu bar

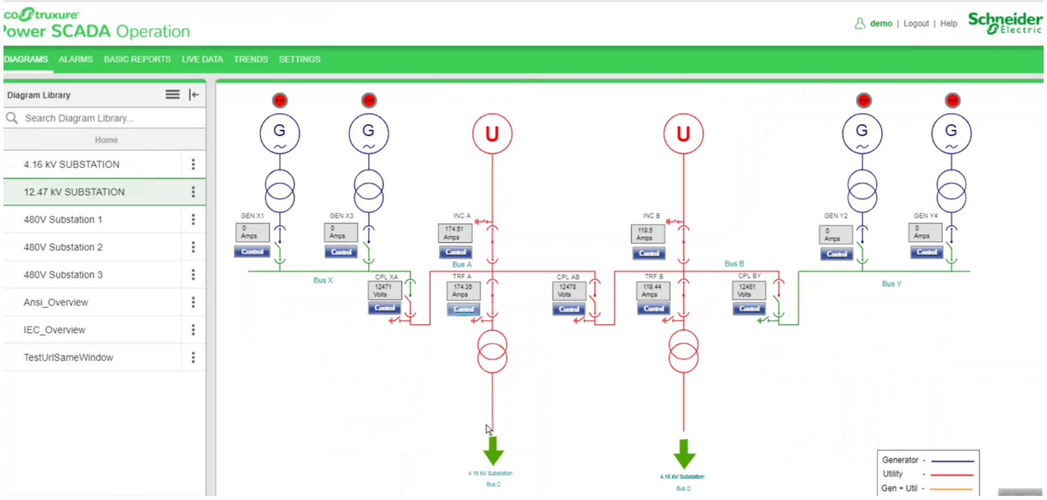
You can add a diagram to the menu bar and then use it to navigate to diagrams.

This topic uses an example to demonstrate how to accomplish this.

To add a diagram to the menu bar:

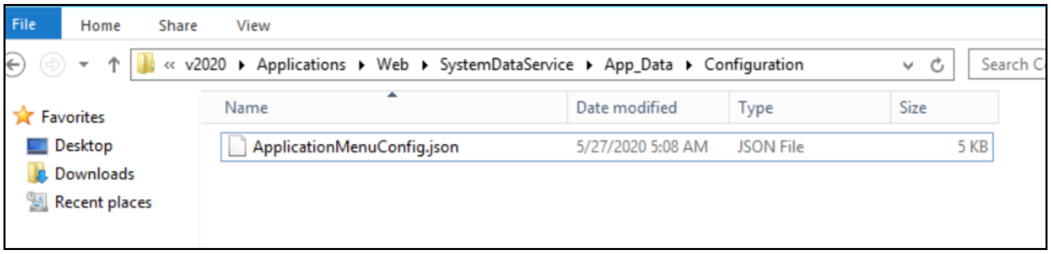
1. Log in to PSO Web Applications (**https//localhost/webhml or ipaddress/webhmi**).

   The following screen is displayed.



2. Example: A user wants to display **480V Substation 3** of **Diagram Library Panel** in the menu bar:

   a. Go to **File path: \Program Files (x86)\Schneider Electric\Power ScadaOperation\v2020\Applications\Web\SystemDataService\App_Data\Configuration** as shown below.



« v2020 ▸ Applications ▸ Web ▸ SystemDataService ▸ App_Data ▸ Configuration

Name	Date modified	Type	Size
ApplicationMenuConfig.json	5/27/2020 5:08 AM	JSON File	5 KB

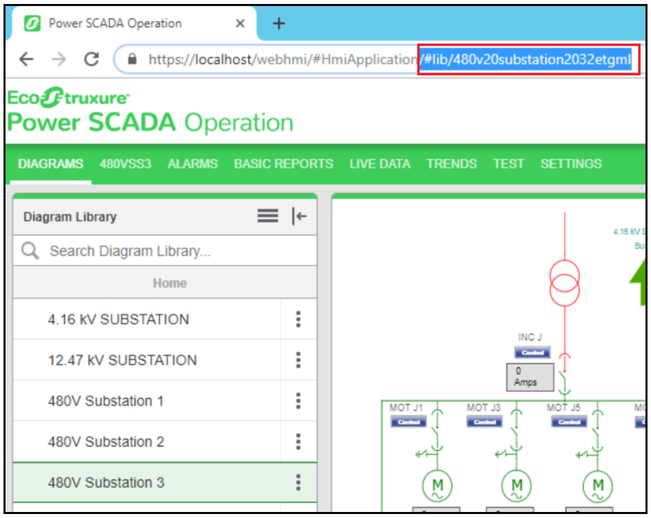
b. In a text editor such as Notepad, open `ApplicationMenuConfig.json`.

c. Enter the code lines 22 to 31 for diagram navigation from new menu after `HmiApplication` code:

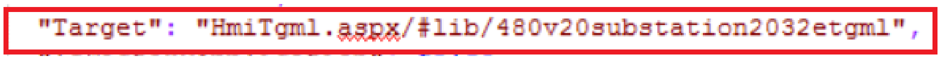
```
 1  "ApplicationsSettings": [
 2  {
 3  "Id": "HmiApplication",
 4  "Description": "",
 5  "DisplayName": "Diag_Application_Title",
 6  "ResourceSet": "HmiApplication",
 7  "Enabled": true,
 8  "Target": "HmiTgml.aspx",
 9  "IsFactoryApplication": false,
10  "RequiredPrivilege": null
11  },
12  {
13  "Id": "Alarms",
14  "Description": "",
15  "DisplayName": "AV_App_Title",
16  "ResourceSet": "AlarmViewer",
17  "Enabled": true,
18  "Target": "Alarms",
19  "IsFactoryApplication": false,
20  "RequiredPrivilege": "AlarmViewer.AccessApplication"
21  },
22  {
23  "Id": "480VSS3",
24  "Description": "",
25  "DisplayName": "480VSS3",
26  "ResourceSet": "HmiApplication",
27  "Enabled": true,
28  "Target": "HmiTgml.aspx/#lib/480v20substation2032etgml",
29  "IsFactoryApplication": false,
30  "RequiredPrivilege": null
31  },
```

3. To get the `Target` value:

a. In PSO Web Applications (**https://localhost/webhmi** or **ipadress/webhmi**), from the Diagram Library click **480V Substation 3**.

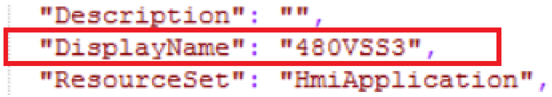


b. Copy the highlighted text from the URL as shown above, and then paste in the `Target` field in the JSON file as follows:



```
"Target": "HmiTgml.aspx/#lib/480v20substation2032etgml",
```

c. For `DisplayName`, enter the required name to display in the menu bar.

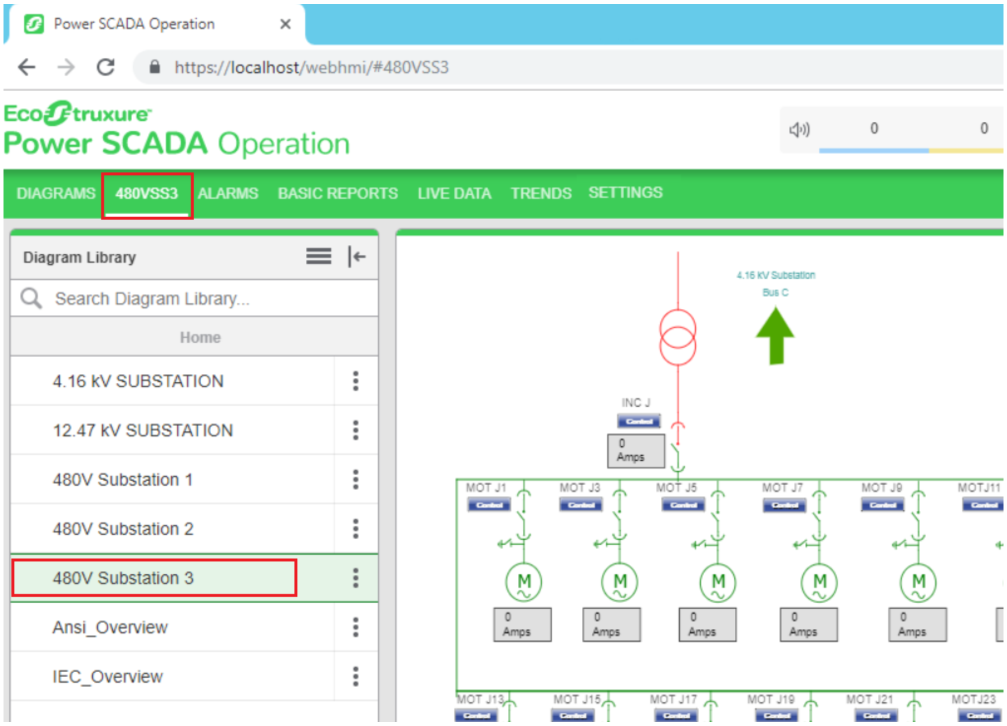


```
"Description": "",
"DisplayName": "480VSS3",
"ResourceSet": "HmiApplication",
```

d. Save and close the JSON file.

4. Go back to PSO Web Applications, and then refresh the browser to display the newly-added menu.

The following image is displayed with the newly-added **480VSS3** menu of **480V Substation 3**:



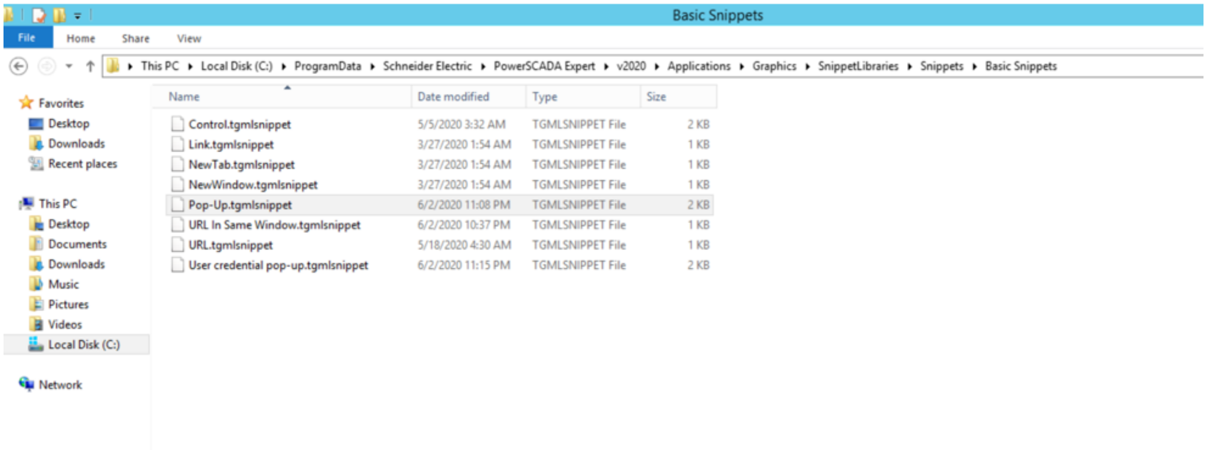
TGML snippets

A snippet is TGML code that stores a **behavior** for reuse. Several common behaviors are stored in the **Snippets** pane.

A snippet can be dragged and dropped onto an object in the Graphics Editor workspace. Objects can be copied, modified, created and then saved as new snippets in the library.

If you want to add a snippet in Graphics Editor, in Windows Explorer navigate to (**..\ProgramData\Schneider Electric\PowerSCADA Expert\v2020\Applications\Graphics\SnippetLibraries\Snippets\Basic Snippets**).

Create a TGMLSNIPPET file and configure it based on your requirements.


Basic Snippets
File
Home
Share
View
This PC ▸ Local Disk (C:) ▸ ProgramData ▸ Schneider Electric ▸ PowerSCADA Expert ▸ v2020 ▸ Applications ▸ Graphics ▸ SnippetLibraries ▸ Snippets ▸ Basic Snippets
Favorites
Desktop
Downloads
Recent places
This PC
Desktop
Documents
Downloads
Music
Pictures
Videos
Local Disk (C:)
Network

Name	Date modified	Type	Size
Control.tgmlsnippet	5/5/2020 3:32 AM	TGMLSNIPPET File	2 KB
Link.tgmlsnippet	3/27/2020 1:54 AM	TGMLSNIPPET File	1 KB
NewTab.tgmlsnippet	3/27/2020 1:54 AM	TGMLSNIPPET File	1 KB
NewWindow.tgmlsnippet	3/27/2020 1:54 AM	TGMLSNIPPET File	1 KB
Pop-Up.tgmlsnippet	6/2/2020 11:08 PM	TGMLSNIPPET File	2 KB
URL In Same Window.tgmlsnippet	6/2/2020 10:37 PM	TGMLSNIPPET File	1 KB
URL.tgmlsnippet	5/18/2020 4:30 AM	TGMLSNIPPET File	1 KB
User credential pop-up.tgmlsnippet	6/2/2020 11:15 PM	TGMLSNIPPET File	2 KB

This section includes example "TGML snippet examples" on page 504 that demonstrate how to use the snippet types.

For more information on snippets, see "Snippets Overview" on page 1179.

TGML snippet examples

This section provides examples that demonstrate how to use snippets in TGML graphics. Follow these examples to create TGML graphic snippet behavior in your project.

TGML snippet examples:

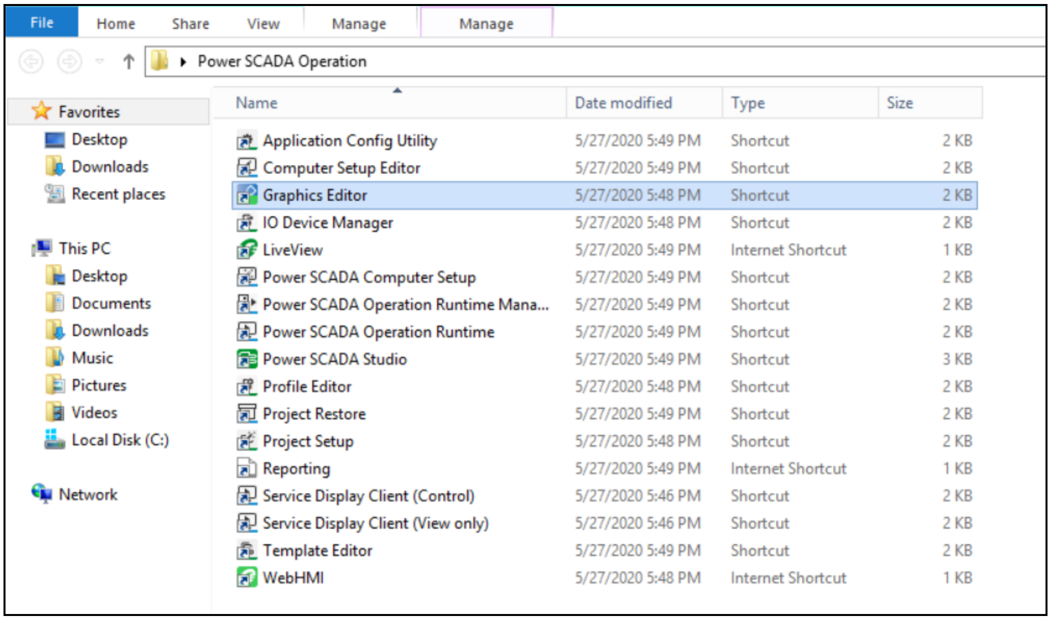
- "Control snippet example" on page 506
- "Link snippet example" on page 519
- "NewTab snippet example" on page 525
- "NewWindow Snippet" on page 531
- "PopUp snippet example" on page 536
- "URL snippet example" on page 543
- "URL in Same Window" on page 548

Prerequisites

To follow the TGML snippet examples, you need to have a graphic file with either a binded component or an equipment in the workspace.

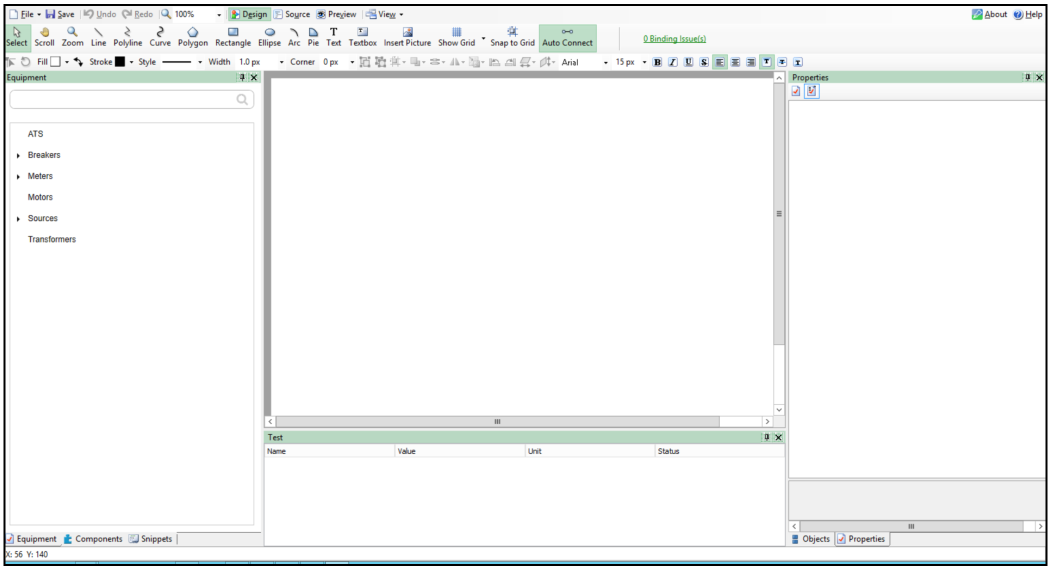
To create a TGML graphic file with a binded component or an equipment in the workspace:

1. Open the **Graphics Editor** from this location **C:\Users\Public\Desktop\Power SCADA Operation**, or by clicking on the **Graphics Editor** Icon.

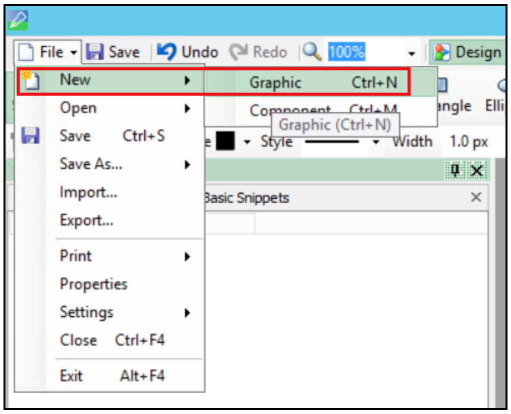


Name	Date modified	Type	Size
Application Config Utility	5/27/2020 5:49 PM	Shortcut	2 KB
Computer Setup Editor	5/27/2020 5:49 PM	Shortcut	2 KB
Graphics Editor	5/27/2020 5:48 PM	Shortcut	2 KB
IO Device Manager	5/27/2020 5:48 PM	Shortcut	2 KB
LiveView	5/27/2020 5:49 PM	Internet Shortcut	1 KB
Power SCADA Computer Setup	5/27/2020 5:49 PM	Shortcut	2 KB
Power SCADA Operation Runtime Mana...	5/27/2020 5:49 PM	Shortcut	2 KB
Power SCADA Operation Runtime	5/27/2020 5:49 PM	Shortcut	2 KB
Power SCADA Studio	5/27/2020 5:49 PM	Shortcut	3 KB
Profile Editor	5/27/2020 5:48 PM	Shortcut	2 KB
Project Restore	5/27/2020 5:49 PM	Shortcut	2 KB
Project Setup	5/27/2020 5:48 PM	Shortcut	2 KB
Reporting	5/27/2020 5:49 PM	Internet Shortcut	1 KB
Service Display Client (Control)	5/27/2020 5:46 PM	Shortcut	2 KB
Service Display Client (View only)	5/27/2020 5:46 PM	Shortcut	2 KB
Template Editor	5/27/2020 5:49 PM	Shortcut	2 KB
WebHMI	5/27/2020 5:48 PM	Internet Shortcut	1 KB

The following screen is displayed.



2. Click **File > New > Graphic**:



3. Add the components to the workspace:
   a. At the bottom left corner, click **Components**.
   b. Select any component and then drag and drop it onto the workspace.

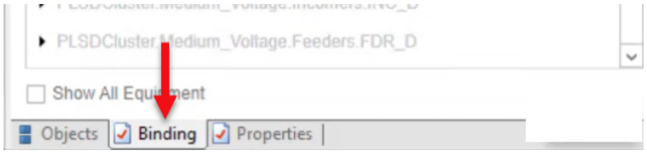
      OR

      At the bottom left corner, click **Equipment**, and then click **Breakers**.
   c. Drag and drop any breaker (from List of breakers) onto the workspace based on your requirement.

4. Bind the components:

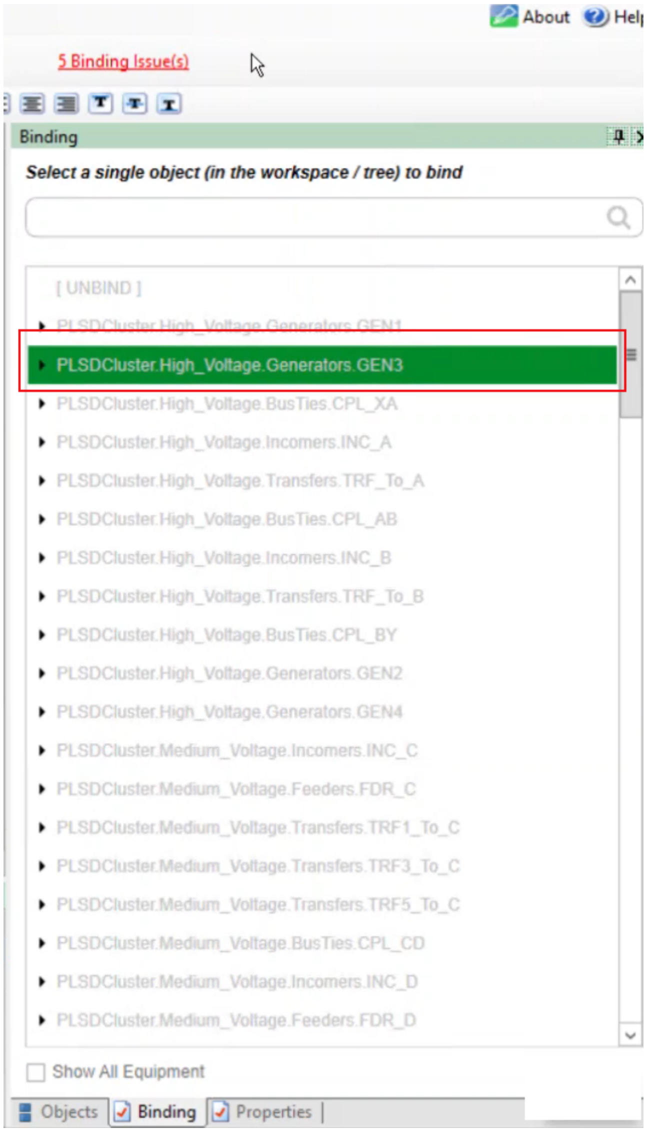
NOTE: If you dragged a **Component** onto the workspace, this step is required. However, if you dragged an **Equipment** onto the workspace, you can proceed to the snippet examples.

a. At the bottom right corner, click **Binding**.



PLSDCluster.Medium_Voltage.Feeders.FDR_D

b. Select a component or device to bind to the selected component. For example:



Select a single object (in the workspace / tree) to bind

[UNBIND]
PLSDCluster.High_Voltage.Generators.GEN1
PLSDCluster.High_Voltage.Generators.GEN3
PLSDCluster.High_Voltage.BusTies.CPL_XA
PLSDCluster.High_Voltage.Incomers.INC_A
PLSDCluster.High_Voltage.Transfers.TRF_To_A
PLSDCluster.High_Voltage.BusTies.CPL_AB
PLSDCluster.High_Voltage.Incomers.INC_B
PLSDCluster.High_Voltage.Transfers.TRF_To_B
PLSDCluster.High_Voltage.BusTies.CPL_BY
PLSDCluster.High_Voltage.Generators.GEN2
PLSDCluster.High_Voltage.Generators.GEN4
PLSDCluster.Medium_Voltage.Incomers.INC_C
PLSDCluster.Medium_Voltage.Feeders.FDR_C
PLSDCluster.Medium_Voltage.Transfers.TRF1_To_C
PLSDCluster.Medium_Voltage.Transfers.TRF3_To_C
PLSDCluster.Medium_Voltage.Transfers.TRF5_To_C
PLSDCluster.Medium_Voltage.BusTies.CPL_CD
PLSDCluster.Medium_Voltage.Incomers.INC_D
PLSDCluster.Medium_Voltage.Feeders.FDR_D

Control snippet example

Control snippets control equipment and circuit breakers, and can change device states.

NOTE: Only authorized users can perform control operation.

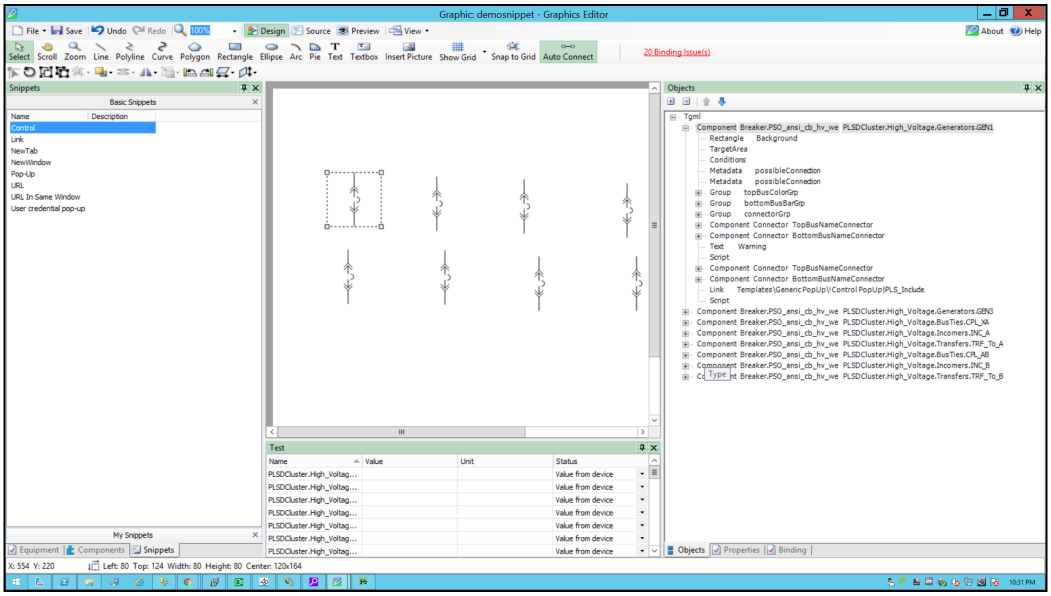
This topic uses an example to illustrate how to configure a Control snippet.

Prerequisites

This example uses a graphic file that already has a binded component or equipment in the workspace. For more information on how to prepare the TGML graphic snippet examples, see "Prerequisites" on page 504.

To create a Control snippet:

1. Click on **Snippet** pane in the bottom left corner and click on **Control**.
2. Drag and drop the **Control** snippet over the selected component in the workspace:



Name	Description
Control	
Link	
NewTab	
NewWindow	
Pop-Up	
URL	
URL In Same Window	
User credential pop-up	

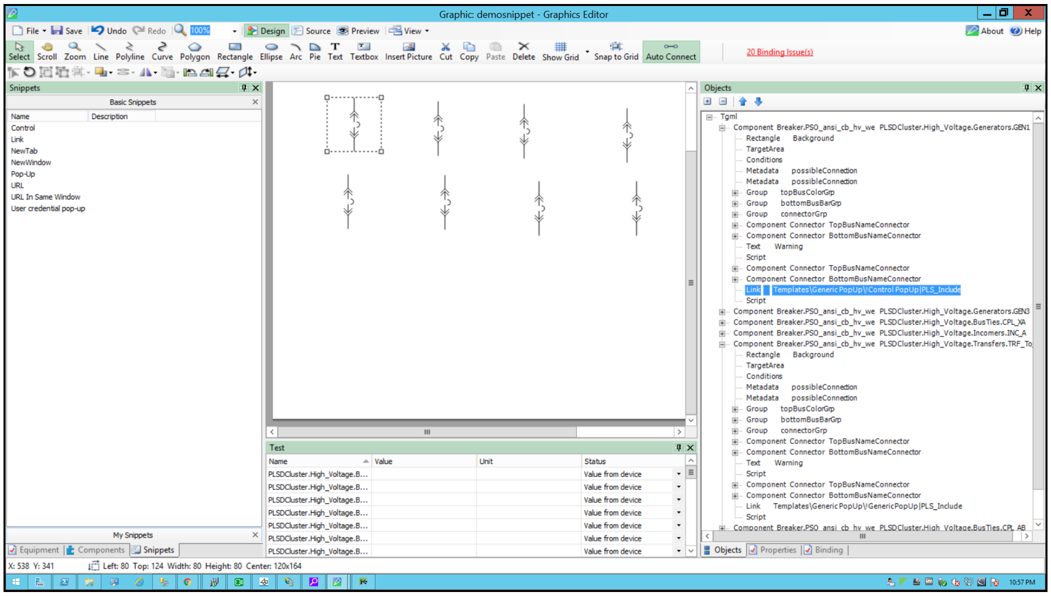
```
Tgml
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Generators.GEN1
    Rectangle   Background
    TargetArea
    Conditions
    Metadata   possibleConnection
    Metadata   possibleConnection
    Group   topBusColorGrp
    Group   bottomBusBarGrp
    Group   connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Text   Warning
    Script
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Link   Templates\GenericPopUp\!Control PopUp|PLS_Include
    Script
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Generators.GEN3
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.BusTies.CPL_XA
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Incomers.INC_A
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Transfers.TRF_To_A
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.BusTies.CPL_AB
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Incomers.INC_B
  Co Type nt Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Transfers.TRF_To_B
```

Name	Value	Unit	Status
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device

3. Click on **Objects** pane in the bottom right corner, and click on **+** to open the TGML.

   Two additional properties appears: **Link** and **Script**.

4. Add popup name in the link by default as generic popup name.

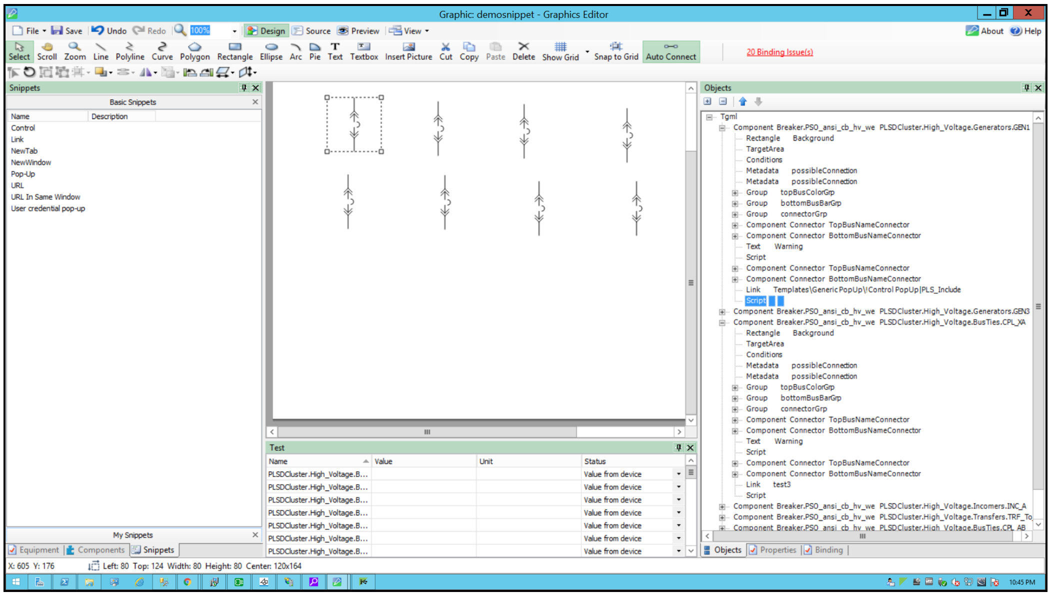


Name	Description
Control	
Link	
NewTab	
NewWindow	
Pop-Up	
URL	
URL In Same Window	
User credential pop-up	

```
Tgml
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Generators.GEN1
    Rectangle   Background
    TargetArea
    Conditions
    Metadata   possibleConnection
    Metadata   possibleConnection
    Group   topBusColorGrp
    Group   bottomBusBarGrp
    Group   connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Text   Warning
    Script
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Link   Templates\GenericPopUp\!Control PopUp|PLS_Include
    Script
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Generators.GEN3
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.BusTies.CPL_XA
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Incomers.INC_A
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Transfers.TRF_To
    Rectangle   Background
    TargetArea
    Conditions
    Metadata   possibleConnection
    Metadata   possibleConnection
    Group   topBusColorGrp
    Group   bottomBusBarGrp
    Group   connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Text   Warning
    Script
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Link   Templates\GenericPopUp\!GenericPopUp|PLS_Include
    Script
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.BusTies.CPL_AB
```

Name	Value	Unit	Status
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device

5. Click on **Script**:

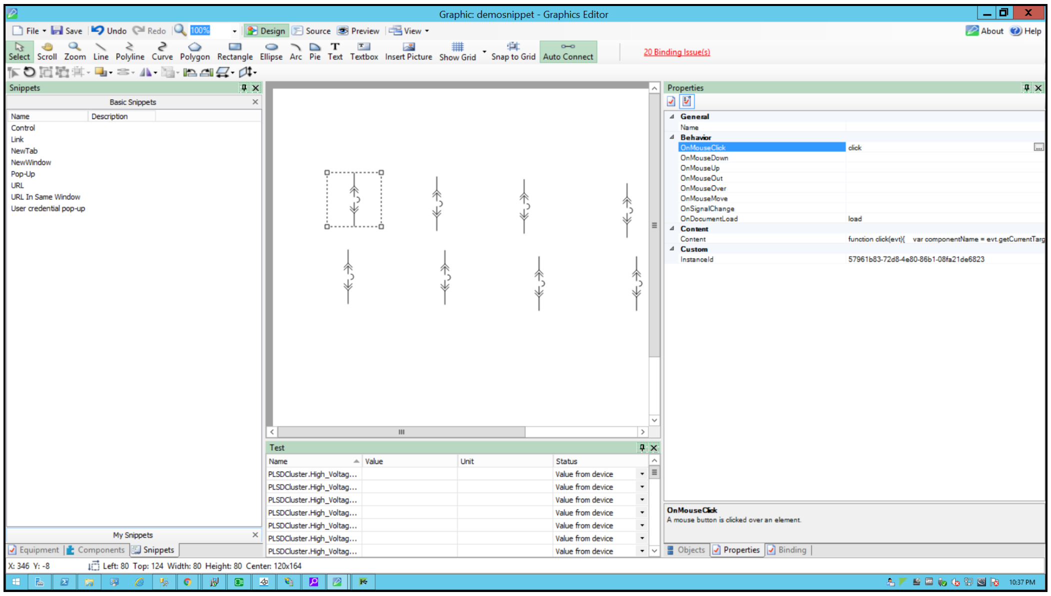


Name	Value	Unit	Status
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device

```
Tgml
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.GEN1
    Rectangle    Background
    TargetArea
    Conditions
    Metadata    possibleConnection
    Metadata    possibleConnection
    Group    topBusColorGrp
    Group    bottomBusBarGrp
    Group    connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Text    Warning
    Script
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Link    Templates\Generic PopUp\!Control PopUp|PLS_Include
    Script
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.GEN3
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.BusTies.CPL_XA
    Rectangle    Background
    TargetArea
    Conditions
    Metadata    possibleConnection
    Metadata    possibleConnection
    Group    topBusColorGrp
    Group    bottomBusBarGrp
    Group    connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Text    Warning
    Script
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Link    test3
    Script
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Incomers.INC_A
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Transfers.TRF_To
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.BusTies.CPL_AB
```

6. In the bottom right corner, click **Properties**, and then expand **Behavior**.

7. Click the ellipsis button in **OnMouseClick**:



Name	Value	Unit	Status
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device

Properties

General	
Name	
Behavior	
OnMouseClick	click
OnMouseDown	
OnMouseUp	
OnMouseOut	
OnMouseOver	
OnMouseMove	
OnSignalChange	
OnDocumentLoad	load
Content	
Content	function click(evt){ var componentName = evt.getCurrentTarg
Custom	
InstanceId	57961b83-72d8-4e80-86b1-08fa21de6823

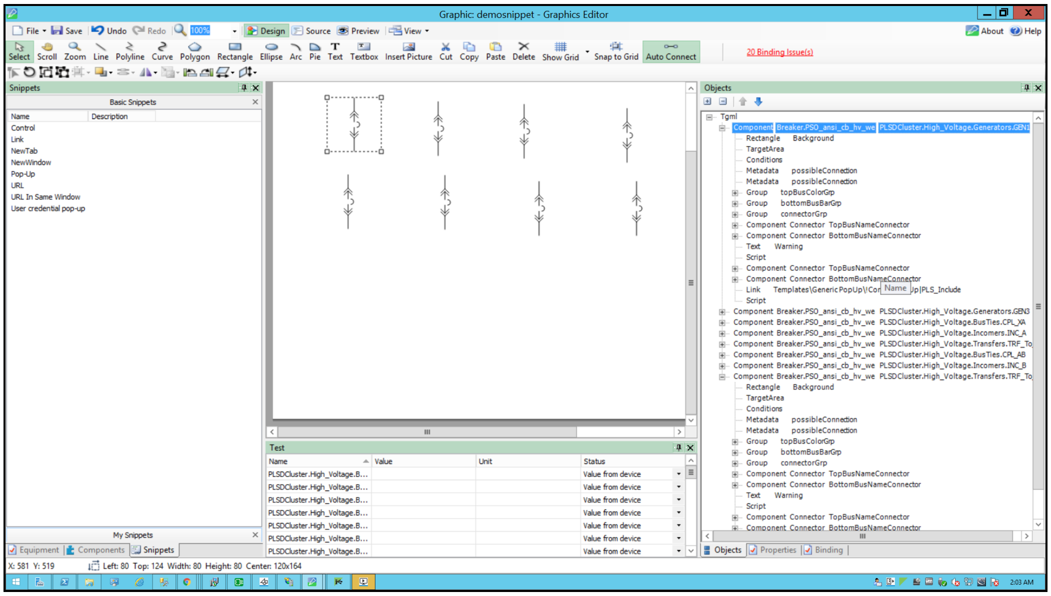
OnMouseClick
A mouse button is clicked over an element.

8. In the script window, use the following script to configure the control snippet and then close the window.

```
function click(evt)
{

    // componentName is name of the component based on the component
selection we will fetch the component name
    var componentName = evt.getCurrentTarget().getAttribute("Name");
    //Collecting the links from the Component
    var Link = evt.getCurrentTarget().getElementsByTagName("Link");

    //InstanceId-It is auto generating id each component pop up selection it
will create new instance id
    var instanceId = evt.getCurrentTarget().getAttribute("InstanceId");

    //dataPoint is the item name to do the write operation
    var dataPoint = componentName+"."+evt.getCurrentTarget().getAttribute
("DataPoint");

    //title is component name use for showing the title
    var title = componentName;

    //CustomExpose-If two breakers are internally connected (means multi
equipment)
    var customExpose = evt.getCurrentTarget().getAttribute
("SubstituteNames");

    //Height & width can be configurable by the user
    var popUpWidth = evt.getCurrentTarget().getAttribute("PopUpWidth");
    var popUpHeight = evt.getCurrentTarget().getAttribute("PopUpHeight");

    //Sets the width of the window in pixels
    var width = (popUpWidth == "")? 576:popUpWidth;

    //height: Sets the height of the window in pixels
    var height = (popUpHeight == "")? 525:popUpHeight;

    //showTitleBar: Displays the Title Bar in the target pane when set to
Yes
    var showTitleBar = "Yes";

    //showUnamePwd: Displays the Username and Password in the target pane
when set to Yes
    var showUnamePwd = "Yes";

    //usercredbottom: The vertical position of a positioned element. Sets
the position bottom  of the user credential popup
    var usercredbottom = 25;

    //usercredleft: The horizontal position of a positioned element. Sets
the position left of the user credential popup
    var usercredleft = 15;

    //usercredbackcolor: Sets the background color of the user credential
popup
    var usercredbackcolor = "white";

    //usercredwidth: : Sets the height of the user credential popup
    var usercredwidth = 65;

    //usercredheight:Sets the height  of the user credential popup
    var usercredheight = 24;

    //unamepwdwidth: Sets the width  user credential popup username and
password
    var unamepwdwidth = 100;

    //unamepwdcolor: Sets the color user credential popup username and
password
```

```
    var unamepwdcolor = "#9FA0A4";

    for (var i=0;i< Link.length;i++)
    {
        //LinkFileName : Extracting the file name from the Link
        var LinkFileName = Link.item(i).getAttribute("Name");

        //With invoke function you can configure the graphic component in
Graphics Editor
        //to open a linked target object in a target location when you
perform a
        //control action on the component
        invoke(LinkFileName, "Type = PopUp | ComponentName=" + componentName
+ " | InstanceID=" + instanceId + "  | DataPoint = "+ dataPoint +"| Title="
+ title + " |Width=" + width + " | Height=" + height + " | ShowTitleBar =" +
showTitleBar + " |ShowUnamePwd =" + showUnamePwd +" | UserCredBottom =
"+usercredbottom +" |UserCredLeft = "+ usercredleft+" | UserCredWidth = "+
usercredwidth +" |UserCredHeight = "+usercredheight +" | UserCredBackColor =
"+usercredbackcolor+" |UnamePwdWidth = "+unamepwdwidth+" | UnamePwdColor =
"+unamepwdcolor);
    }
}

function load(evt)
{
}
```

9. In the bottom right corner, click **Objects**, and then click **Component** inside the Tgml:



Name	Description
Control	
Link	
NewTab	
NewWindow	
Pop-Up	
URL	
URL In Same Window	
User credential pop-up	

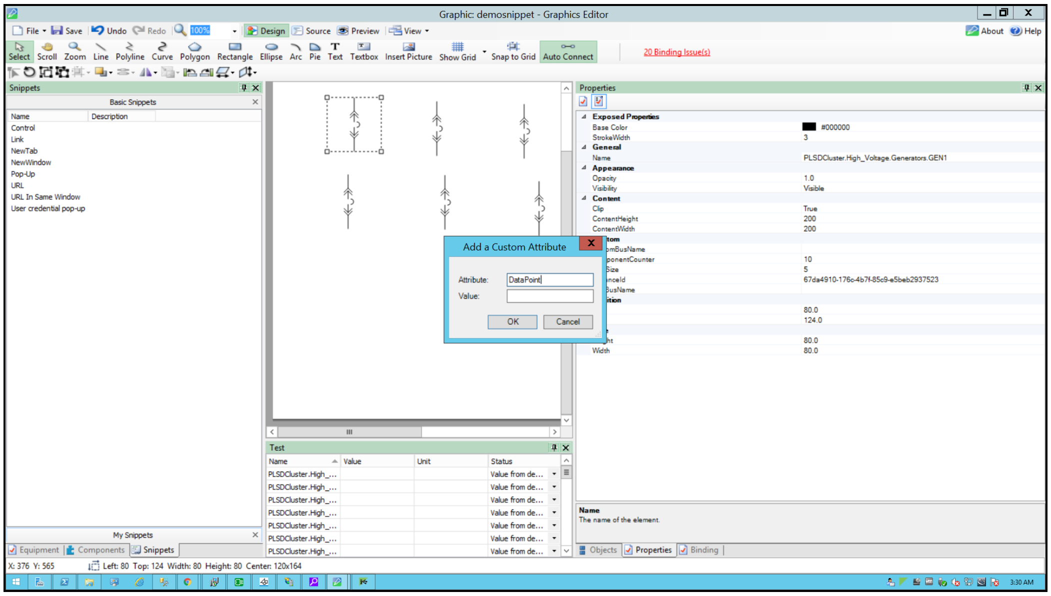
```
Tgml
  Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.GEN1
    Rectangle   Background
    TargetArea
    Conditions
    Metadata   possibleConnection
    Metadata   possibleConnection
    Group   topBusColorGrp
    Group   bottomBusBarGrp
    Group   connectorGrp
    Component  Connector  TopBusNameConnector
    Component  Connector  BottomBusNameConnector
    Text   Warning
    Script
    Component  Connector  TopBusNameConnector
    Component  Connector  BottomBusNameConnector
    Link   Templates\GenericPopUp\!Cor    Jp|PLS_Include
    Script
  Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.GEN3
  Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.BusTies.CPL_XA
  Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Incomers.INC_A
  Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Transfers.TRF_To
  Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.BusTies.CPL_AB
  Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Incomers.INC_B
  Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Transfers.TRF_To
    Rectangle   Background
    TargetArea
    Conditions
    Metadata   possibleConnection
    Metadata   possibleConnection
    Group   topBusColorGrp
    Group   bottomBusBarGrp
    Group   connectorGrp
    Component  Connector  TopBusNameConnector
    Component  Connector  BottomBusNameConnector
    Text   Warning
    Script
    Component  Connector  TopBusNameConnector
    Component  Connector  BottomBusNameConnector
```

Name	Value	Unit	Status
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device

X: 581 Y: 519 Left: 80 Top: 124 Width: 80 Height: 80 Center: 120x164

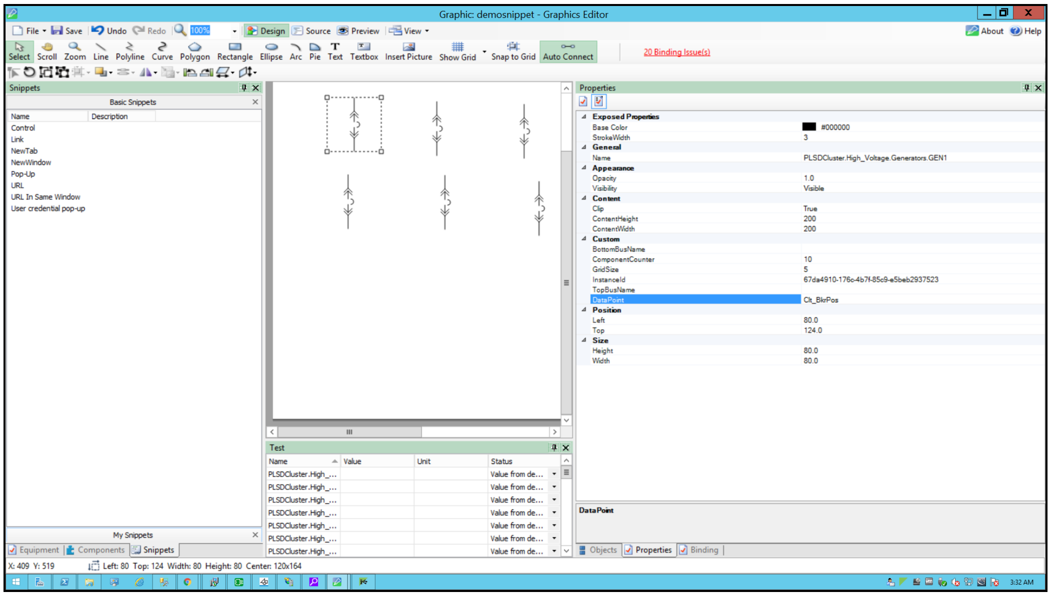
10. In the bottom right corner, click **Properties**, click the **Custom** attribute, and then right-click to select **Add**.

11. For **Attribute**, enter `DataPoint`, and then click **OK**:



Attribute: DataPoint
Value:

The following screen is displayed with the added **DataPoint** attribute in the **Custom** group:

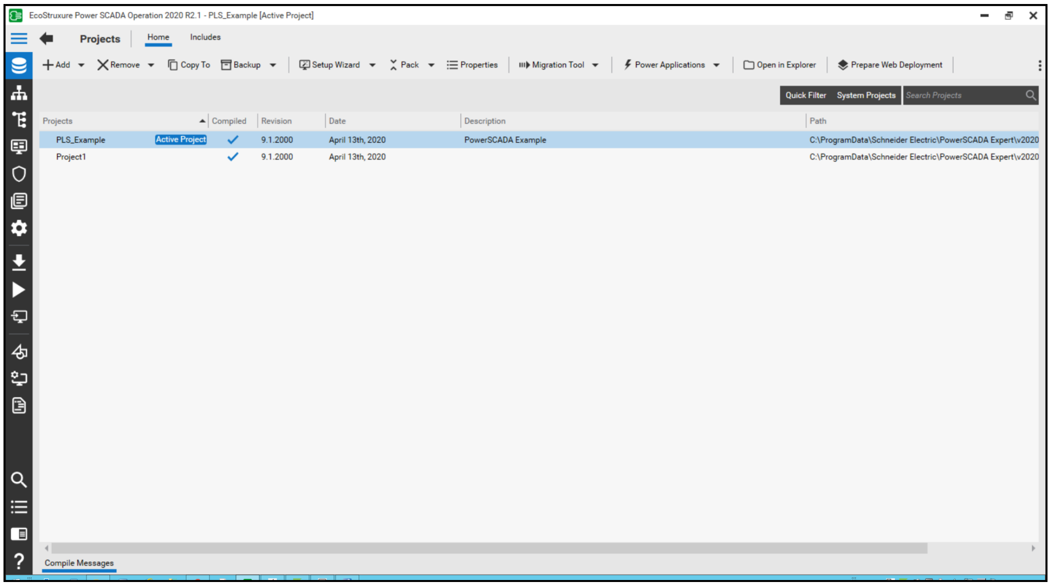


Exposed Properties	
Base Color	#000000
StrokeWidth	3
General	
Name	PLSDCluster.High_Voltage.Generators.GEN1
Appearance	
Opacity	1.0
Visibility	Visible
Content	
Clip	True
ContentHeight	200
ContentWidth	200
Custom	
BottomBusName	
ComponentCounter	10
GridSize	5
InstanceId	67da4910-176c-4b7f-85c9-e5beb2937523
TopBusName	
DataPoint	Clt_BkrPos
Position	
Left	80.0
Top	124.0
Size	
Height	80.0
Width	80.0

12. Get the DataPoint value:

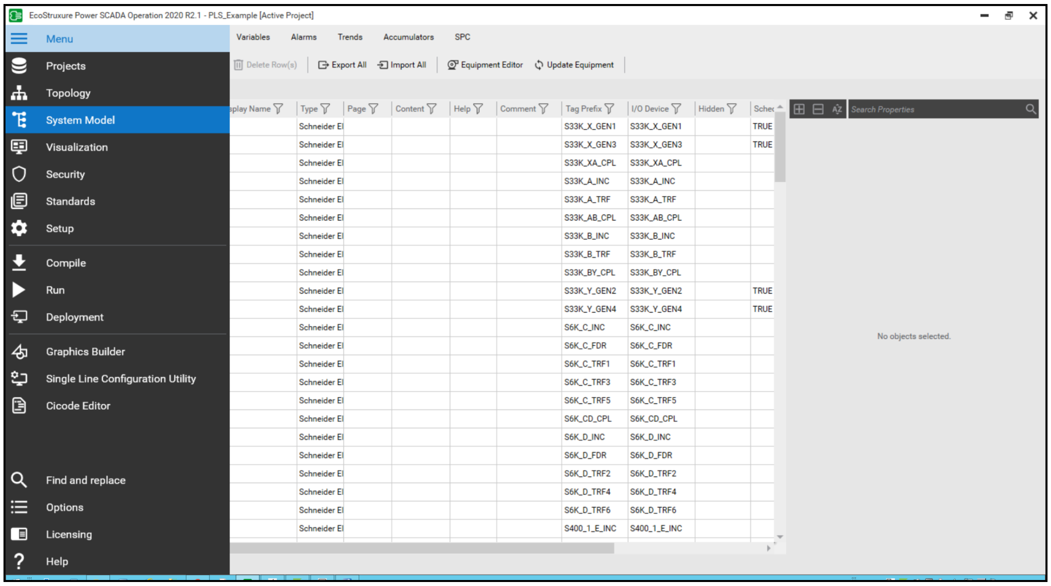
a. Open Power SCADA Studio.

b. Go to the **Active** project.



Projects		Compiled	Revision	Date	Description	Path
PLS_Example	Active Project	✓	9.1.2000	April 13th, 2020	PowerSCADA Example	C:\ProgramData\Schneider Electric\PowerSCADA Expert\v2020
Project1		✓	9.1.2000	April 13th, 2020		C:\ProgramData\Schneider Electric\PowerSCADA Expert\v2020

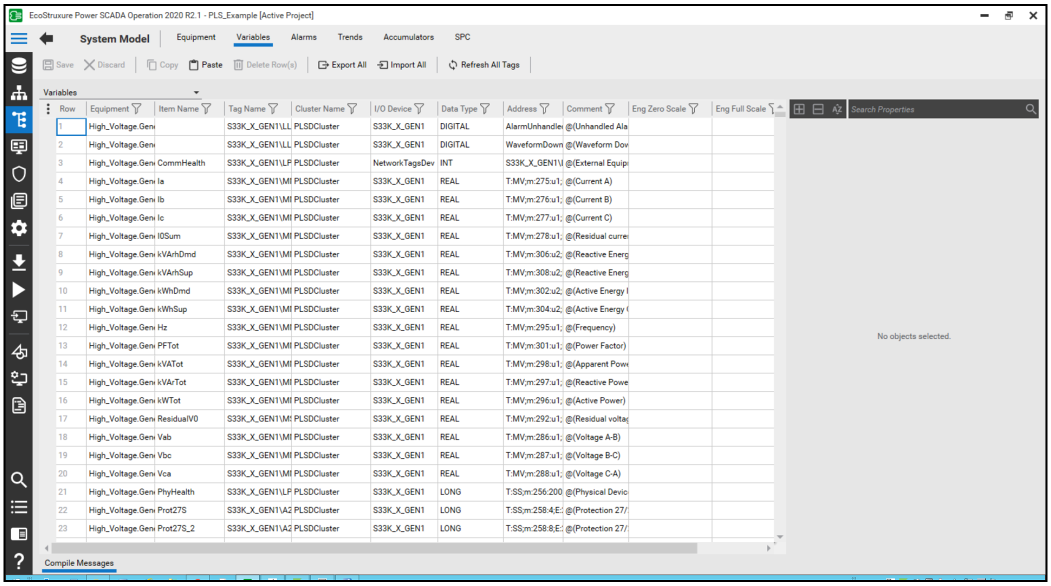
c. Click **System Model**.



splay Name	Type	Page	Content	Help	Comment	Tag Prefix	I/O Device	Hidden	Schec
	Schneider El					S33K_X_GEN1	S33K_X_GEN1		TRUE
	Schneider El					S33K_X_GEN3	S33K_X_GEN3		TRUE
	Schneider El					S33K_XA_CPL	S33K_XA_CPL		
	Schneider El					S33K_A_INC	S33K_A_INC		
	Schneider El					S33K_A_TRF	S33K_A_TRF		
	Schneider El					S33K_AB_CPL	S33K_AB_CPL		
	Schneider El					S33K_B_INC	S33K_B_INC		
	Schneider El					S33K_B_TRF	S33K_B_TRF		
	Schneider El					S33K_BY_CPL	S33K_BY_CPL		
	Schneider El					S33K_Y_GEN2	S33K_Y_GEN2		TRUE
	Schneider El					S33K_Y_GEN4	S33K_Y_GEN4		TRUE
	Schneider El					S6K_C_INC	S6K_C_INC		
	Schneider El					S6K_C_FDR	S6K_C_FDR		
	Schneider El					S6K_C_TRF1	S6K_C_TRF1		
	Schneider El					S6K_C_TRF3	S6K_C_TRF3		
	Schneider El					S6K_C_TRF5	S6K_C_TRF5		
	Schneider El					S6K_CD_CPL	S6K_CD_CPL		
	Schneider El					S6K_D_INC	S6K_D_INC		
	Schneider El					S6K_D_FDR	S6K_D_FDR		
	Schneider El					S6K_D_TRF2	S6K_D_TRF2		
	Schneider El					S6K_D_TRF4	S6K_D_TRF4		
	Schneider El					S6K_D_TRF6	S6K_D_TRF6		
	Schneider El					S400_1_E_INC	S400_1_E_INC		

No objects selected.

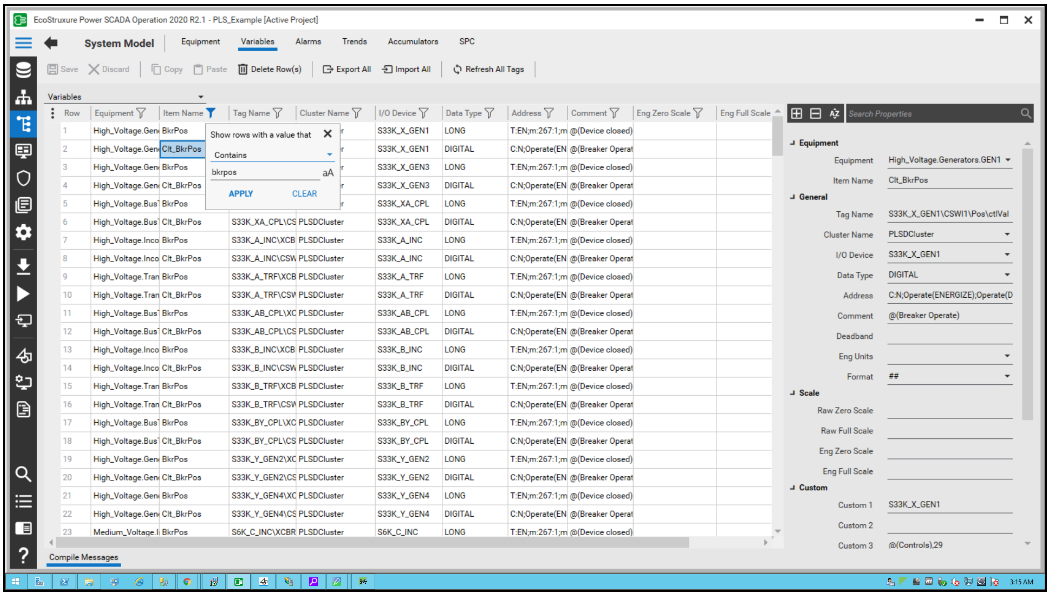
d. Click on **Variable** tab.



Row	Equipment	Item Name	Tag Name	Cluster Name	I/O Device	Data Type	Address	Comment	Eng Zero Scale	Eng Full Scale
1	High_Voltage.Gen		S33K_X_GEN1\LL	PLSDCluster	S33K_X_GEN1	DIGITAL	AlarmUnhandle	@(Unhandled Ala		
2	High_Voltage.Gen		S33K_X_GEN1\LL	PLSDCluster	S33K_X_GEN1	DIGITAL	WaveformDown	@(Waveform Dov		
3	High_Voltage.Gen	CommHealth	S33K_X_GEN1\LP	PLSDCluster	NetworkTagsDev	INT	S33K_X_GEN1\	@(External Equip		
4	High_Voltage.Gen	Ia	S33K_X_GEN1\MI	PLSDCluster	S33K_X_GEN1	REAL	T:MV;m:275:u1;	@(Current A)		
5	High_Voltage.Gen	Ib	S33K_X_GEN1\MI	PLSDCluster	S33K_X_GEN1	REAL	T:MV;m:276:u1;	@(Current B)		
6	High_Voltage.Gen	Ic	S33K_X_GEN1\MI	PLSDCluster	S33K_X_GEN1	REAL	T:MV;m:277:u1;	@(Current C)		
7	High_Voltage.Gen	I0Sum	S33K_X_GEN1\MI	PLSDCluster	S33K_X_GEN1	REAL	T:MV;m:278:u1;	@(Residual curre		
8	High_Voltage.Gen	kVArhDmd	S33K_X_GEN1\MI	PLSDCluster	S33K_X_GEN1	REAL	T:MV;m:306:u2;	@(Reactive Energ		
9	High_Voltage.Gen	kVArhSup	S33K_X_GEN1\MI	PLSDCluster	S33K_X_GEN1	REAL	T:MV;m:308:u2;	@(Reactive Energ		
10	High_Voltage.Gen	kWhDmd	S33K_X_GEN1\MI	PLSDCluster	S33K_X_GEN1	REAL	T:MV;m:302:u2;	@(Active Energy I		
11	High_Voltage.Gen	kWhSup	S33K_X_GEN1\MI	PLSDCluster	S33K_X_GEN1	REAL	T:MV;m:304:u2;	@(Active Energy (		
12	High_Voltage.Gen	Hz	S33K_X_GEN1\MI	PLSDCluster	S33K_X_GEN1	REAL	T:MV;m:295:u1;	@(Frequency)		
13	High_Voltage.Gen	PFTot	S33K_X_GEN1\MI	PLSDCluster	S33K_X_GEN1	REAL	T:MV;m:301:u1;	@(Power Factor)		
14	High_Voltage.Gen	kVATot	S33K_X_GEN1\MI	PLSDCluster	S33K_X_GEN1	REAL	T:MV;m:298:u1;	@(Apparent Powe		
15	High_Voltage.Gen	kVArTot	S33K_X_GEN1\MI	PLSDCluster	S33K_X_GEN1	REAL	T:MV;m:297:u1;	@(Reactive Powe		
16	High_Voltage.Gen	kWTot	S33K_X_GEN1\MI	PLSDCluster	S33K_X_GEN1	REAL	T:MV;m:296:u1;	@(Active Power)		
17	High_Voltage.Gen	ResidualV0	S33K_X_GEN1\MS	PLSDCluster	S33K_X_GEN1	REAL	T:MV;m:292:u1;	@(Residual voltag		
18	High_Voltage.Gen	Vab	S33K_X_GEN1\MI	PLSDCluster	S33K_X_GEN1	REAL	T:MV;m:286:u1;	@(Voltage A-B)		
19	High_Voltage.Gen	Vbc	S33K_X_GEN1\MI	PLSDCluster	S33K_X_GEN1	REAL	T:MV;m:287:u1;	@(Voltage B-C)		
20	High_Voltage.Gen	Vca	S33K_X_GEN1\MI	PLSDCluster	S33K_X_GEN1	REAL	T:MV;m:288:u1;	@(Voltage C-A)		
21	High_Voltage.Gen	PhyHealth	S33K_X_GEN1\LP	PLSDCluster	S33K_X_GEN1	LONG	T:SS;m:256:200	@(Physical Devic		
22	High_Voltage.Gen	Prot27S	S33K_X_GEN1\A2	PLSDCluster	S33K_X_GEN1	LONG	T:SS;m:258:4;E:	@(Protection 27/		
23	High_Voltage.Gen	Prot27S_2	S33K_X_GEN1\A2	PLSDCluster	S33K_X_GEN1	LONG	T:SS;m:258:8;E:	@(Protection 27/		

e. Search for the required item name.

f. Click **Apply**.



Show rows with a value that

Contains

bkrpos aA

APPLY CLEAR

Row	Equipment	Item Name	Tag Name	Cluster Name	I/O Device	Data Type	Address	Comment	Eng Zero Scale	Eng Full Scale
1	High_Voltage.Gen	BkrPos		r	S33K_X_GEN1	LONG	T:EN;m:267:1;m	@(Device closed)		
2	High_Voltage.Gen	Clt_BkrPos		r	S33K_X_GEN1	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
3	High_Voltage.Gen	BkrPos		r	S33K_X_GEN3	LONG	T:EN;m:267:1;m	@(Device closed)		
4	High_Voltage.Gen	Clt_BkrPos		r	S33K_X_GEN3	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
5	High_Voltage.Bus	BkrPos		r	S33K_XA_CPL	LONG	T:EN;m:267:1;m	@(Device closed)		
6	High_Voltage.Bus	Clt_BkrPos	S33K_XA_CPL\CS	PLSDCluster	S33K_XA_CPL	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
7	High_Voltage.Inco	BkrPos	S33K_A_INC\XCB	PLSDCluster	S33K_A_INC	LONG	T:EN;m:267:1;m	@(Device closed)		
8	High_Voltage.Inco	Clt_BkrPos	S33K_A_INC\CSW	PLSDCluster	S33K_A_INC	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
9	High_Voltage.Tran	BkrPos	S33K_A_TRF\XCB	PLSDCluster	S33K_A_TRF	LONG	T:EN;m:267:1;m	@(Device closed)		
10	High_Voltage.Tran	Clt_BkrPos	S33K_A_TRF\CSW	PLSDCluster	S33K_A_TRF	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
11	High_Voltage.Bus	BkrPos	S33K_AB_CPL\XC	PLSDCluster	S33K_AB_CPL	LONG	T:EN;m:267:1;m	@(Device closed)		
12	High_Voltage.Bus	Clt_BkrPos	S33K_AB_CPL\CS	PLSDCluster	S33K_AB_CPL	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
13	High_Voltage.Inco	BkrPos	S33K_B_INC\XCB	PLSDCluster	S33K_B_INC	LONG	T:EN;m:267:1;m	@(Device closed)		
14	High_Voltage.Inco	Clt_BkrPos	S33K_B_INC\CSW	PLSDCluster	S33K_B_INC	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
15	High_Voltage.Tran	BkrPos	S33K_B_TRF\XCB	PLSDCluster	S33K_B_TRF	LONG	T:EN;m:267:1;m	@(Device closed)		
16	High_Voltage.Tran	Clt_BkrPos	S33K_B_TRF\CSW	PLSDCluster	S33K_B_TRF	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
17	High_Voltage.Bus	BkrPos	S33K_BY_CPL\XC	PLSDCluster	S33K_BY_CPL	LONG	T:EN;m:267:1;m	@(Device closed)		
18	High_Voltage.Bus	Clt_BkrPos	S33K_BY_CPL\CS	PLSDCluster	S33K_BY_CPL	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
19	High_Voltage.Gen	BkrPos	S33K_Y_GEN2\XC	PLSDCluster	S33K_Y_GEN2	LONG	T:EN;m:267:1;m	@(Device closed)		
20	High_Voltage.Gen	Clt_BkrPos	S33K_Y_GEN2\CS	PLSDCluster	S33K_Y_GEN2	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
21	High_Voltage.Gen	BkrPos	S33K_Y_GEN4\XC	PLSDCluster	S33K_Y_GEN4	LONG	T:EN;m:267:1;m	@(Device closed)		
22	High_Voltage.Gen	Clt_BkrPos	S33K_Y_GEN4\CS	PLSDCluster	S33K_Y_GEN4	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
23	Medium_Voltage.I	BkrPos	S6K_C_INC\XCBR	PLSDCluster	S6K_C_INC	LONG	T:EN;m:267:1;m	@(Device closed)		

Equipment

- Equipment: High_Voltage.Generators.GEN1
- Item Name: Clt_BkrPos

General

- Tag Name: S33K_X_GEN1\CSWI1\Pos\ctlVal
- Cluster Name: PLSDCluster
- I/O Device: S33K_X_GEN1
- Data Type: DIGITAL
- Address: C:N;Operate(ENERGIZE);Operate(D
- Comment: @(Breaker Operate)
- Deadband:
- Eng Units:
- Format: ##

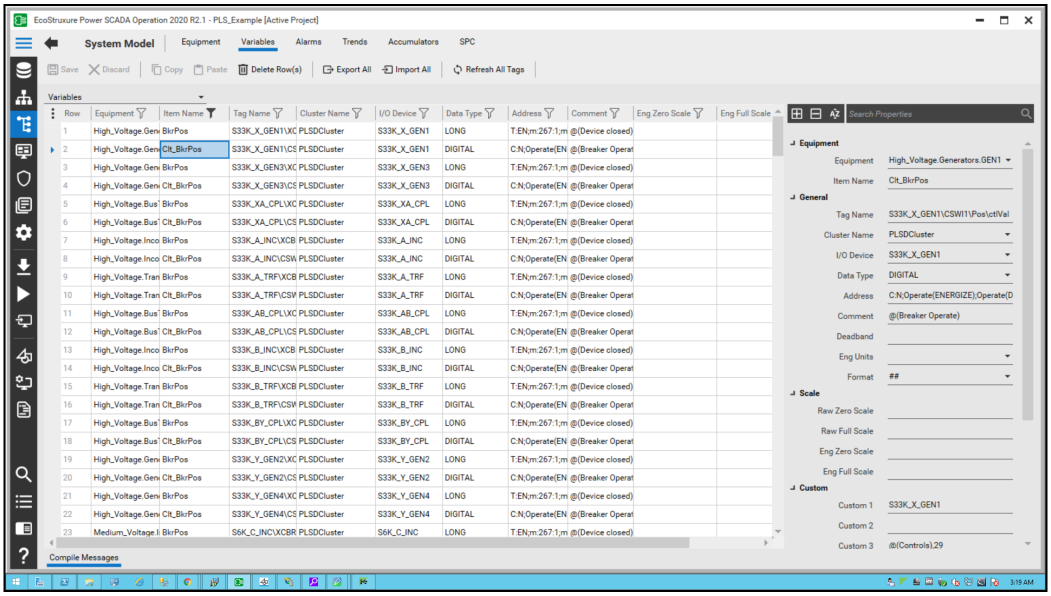
Scale

- Raw Zero Scale:
- Raw Full Scale:
- Eng Zero Scale:
- Eng Full Scale:

Custom

- Custom 1: S33K_X_GEN1
- Custom 2:
- Custom 3: @(Controls),29

13. Copy the item name (DataPoint name) and check if the Data Type is digital:



EcoStruxure Power SCADA Operation 2020 R2.1 - PLS_Example [Active Project]

Row	Equipment	Item Name	Tag Name	Cluster Name	I/O Device	Data Type	Address	Comment	Eng Zero Scale	Eng Full Scale
1	High_Voltage.Gen	BkrPos	S33K_X_GEN1\XC	PLSDCluster	S33K_X_GEN1	LONG	T:EN;m:267:1;m	@(Device closed)		
2	High_Voltage.Gen	Clt_BkrPos	S33K_X_GEN1\CS	PLSDCluster	S33K_X_GEN1	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
3	High_Voltage.Gen	BkrPos	S33K_X_GEN3\XC	PLSDCluster	S33K_X_GEN3	LONG	T:EN;m:267:1;m	@(Device closed)		
4	High_Voltage.Gen	Clt_BkrPos	S33K_X_GEN3\CS	PLSDCluster	S33K_X_GEN3	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
5	High_Voltage.Bus	BkrPos	S33K_XA_CPL\XC	PLSDCluster	S33K_XA_CPL	LONG	T:EN;m:267:1;m	@(Device closed)		
6	High_Voltage.Bus	Clt_BkrPos	S33K_XA_CPL\CS	PLSDCluster	S33K_XA_CPL	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
7	High_Voltage.Inco	BkrPos	S33K_A_INC\XCB	PLSDCluster	S33K_A_INC	LONG	T:EN;m:267:1;m	@(Device closed)		
8	High_Voltage.Inco	Clt_BkrPos	S33K_A_INC\CSW	PLSDCluster	S33K_A_INC	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
9	High_Voltage.Tran	BkrPos	S33K_A_TRF\XCB	PLSDCluster	S33K_A_TRF	LONG	T:EN;m:267:1;m	@(Device closed)		
10	High_Voltage.Tran	Clt_BkrPos	S33K_A_TRF\CSW	PLSDCluster	S33K_A_TRF	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
11	High_Voltage.Bus	BkrPos	S33K_AB_CPL\XC	PLSDCluster	S33K_AB_CPL	LONG	T:EN;m:267:1;m	@(Device closed)		
12	High_Voltage.Bus	Clt_BkrPos	S33K_AB_CPL\CS	PLSDCluster	S33K_AB_CPL	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
13	High_Voltage.Inco	BkrPos	S33K_B_INC\XCB	PLSDCluster	S33K_B_INC	LONG	T:EN;m:267:1;m	@(Device closed)		
14	High_Voltage.Inco	Clt_BkrPos	S33K_B_INC\CSW	PLSDCluster	S33K_B_INC	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
15	High_Voltage.Tran	BkrPos	S33K_B_TRF\XCB	PLSDCluster	S33K_B_TRF	LONG	T:EN;m:267:1;m	@(Device closed)		
16	High_Voltage.Tran	Clt_BkrPos	S33K_B_TRF\CSW	PLSDCluster	S33K_B_TRF	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
17	High_Voltage.Bus	BkrPos	S33K_BY_CPL\XC	PLSDCluster	S33K_BY_CPL	LONG	T:EN;m:267:1;m	@(Device closed)		
18	High_Voltage.Bus	Clt_BkrPos	S33K_BY_CPL\CS	PLSDCluster	S33K_BY_CPL	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
19	High_Voltage.Gen	BkrPos	S33K_Y_GEN2\XC	PLSDCluster	S33K_Y_GEN2	LONG	T:EN;m:267:1;m	@(Device closed)		
20	High_Voltage.Gen	Clt_BkrPos	S33K_Y_GEN2\CS	PLSDCluster	S33K_Y_GEN2	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
21	High_Voltage.Gen	BkrPos	S33K_Y_GEN4\XC	PLSDCluster	S33K_Y_GEN4	LONG	T:EN;m:267:1;m	@(Device closed)		
22	High_Voltage.Gen	Clt_BkrPos	S33K_Y_GEN4\CS	PLSDCluster	S33K_Y_GEN4	DIGITAL	C:N;Operate(EN	@(Breaker Operat		
23	Medium_Voltage.I	BkrPos	S6K_C_INC\XCBR	PLSDCluster	S6K_C_INC	LONG	T:EN;m:267:1;m	@(Device closed)		

Search Properties

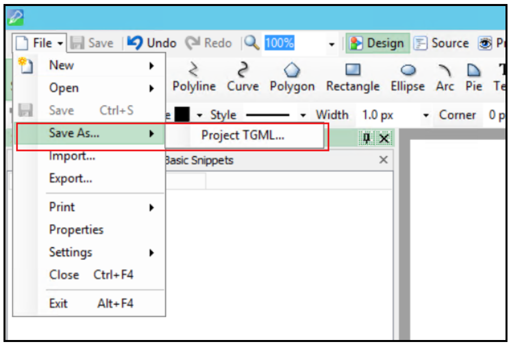
Equipment
- Equipment: High_Voltage.Generators.GEN1
- Item Name: Clt_BkrPos

General
- Tag Name: S33K_X_GEN1\CSWI1\Pos\ctlVal
- Cluster Name: PLSDCluster
- I/O Device: S33K_X_GEN1
- Data Type: DIGITAL
- Address: C:N;Operate(ENERGIZE);Operate(D
- Comment: @(Breaker Operate)
- Deadband:
- Eng Units:
- Format: ##

Scale
- Raw Zero Scale:
- Raw Full Scale:
- Eng Zero Scale:
- Eng Full Scale:

Custom
- Custom 1: S33K_X_GEN1
- Custom 2:
- Custom 3: @(Controls),29

14. Go to **File** > **Save As** > **Project TGML**.

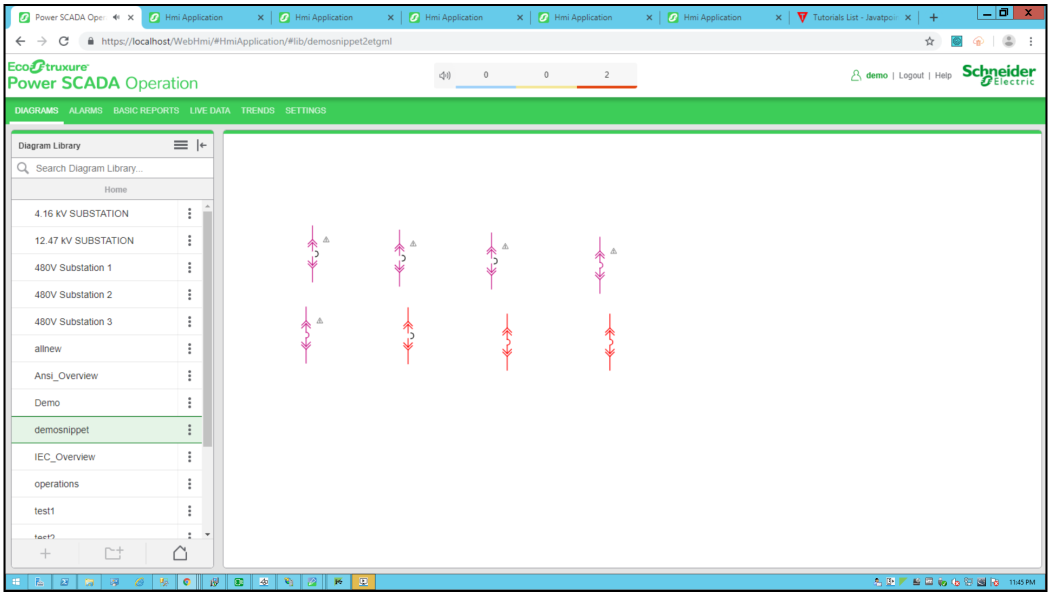


15. Enter a file name, and then click **Save**.

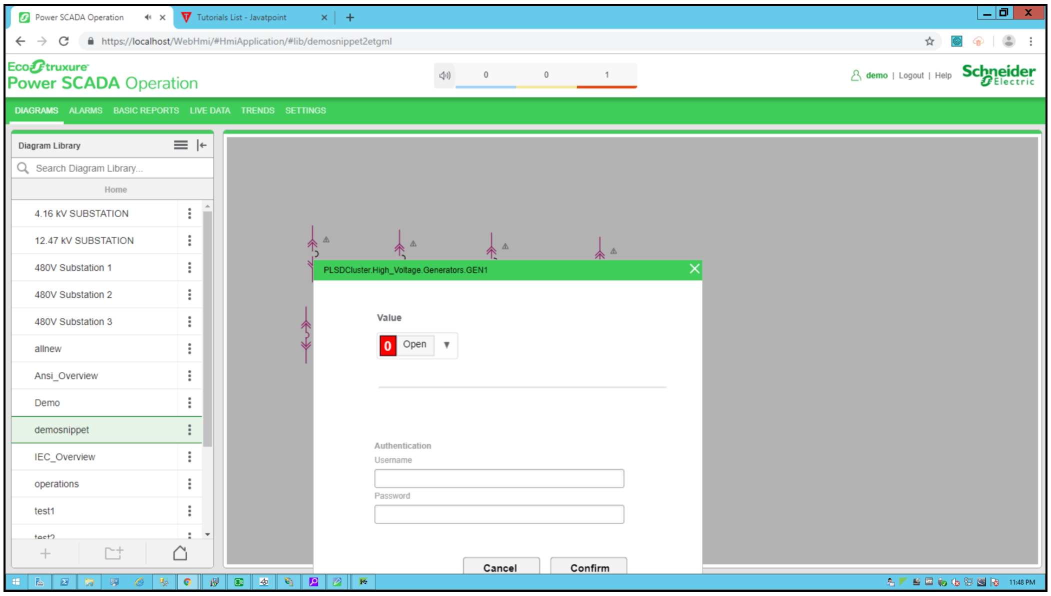
To view the snippet behavior:

1. In a web browser, log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)

2. Select the new TGML file from the **Diagram Library** from the left panel:



3. Click on the breaker:



PLSDCluster.High_Voltage.Generators.GEN1

Value

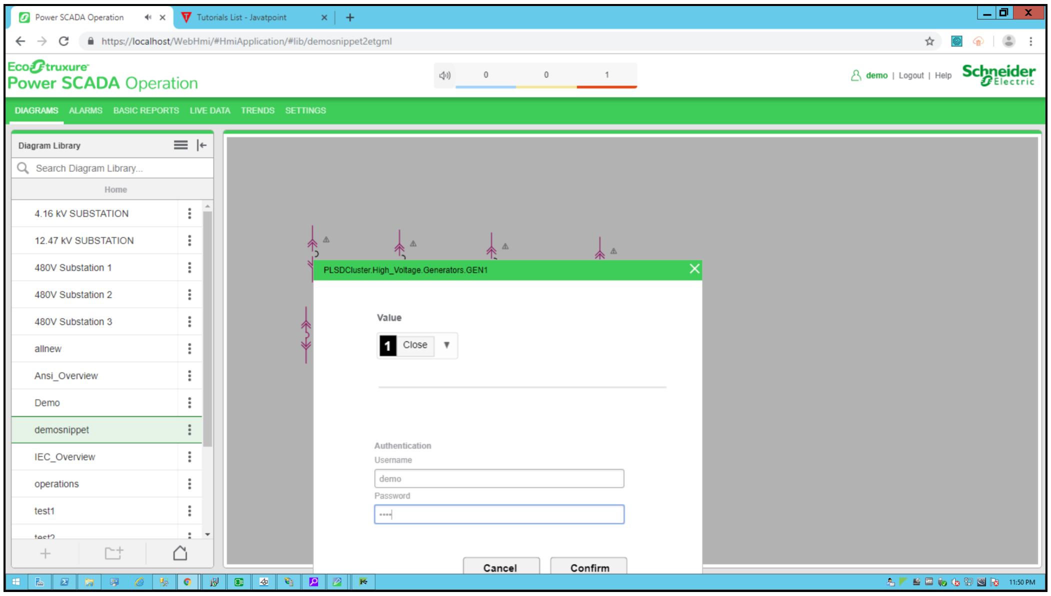
0 Open

Authentication

Username

Password

4. Select a value (**Open** or **Close**) to perform the operation.

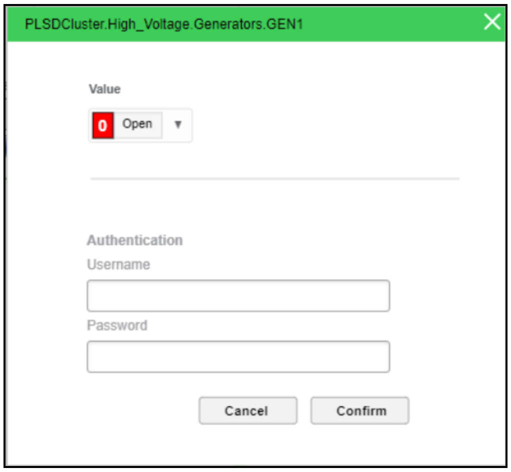


PLSDCluster.High_Voltage.Generators.GEN1

Value
1 Close

Authentication
Username
demo
Password
••••

5. Enter your **Username** and **Password**.



PLSDCluster.High_Voltage.Generators.GEN1

Value
0 Open

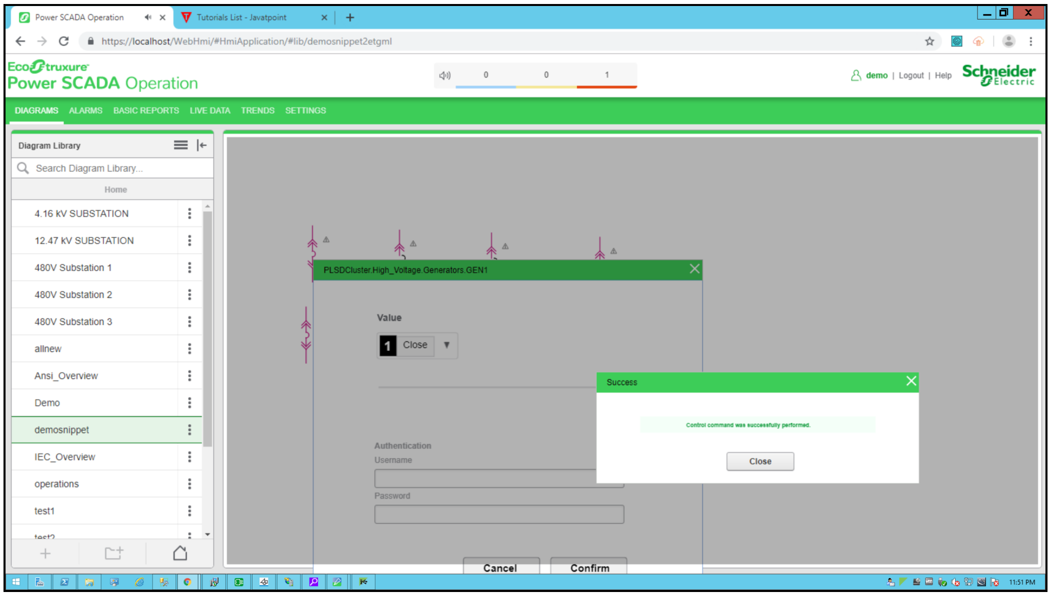
Authentication
Username

Password

6. Click **Confirm**.

One of the following dialog boxes appear:

- **Success**: This popup appears when the selected value (**Open** or **Close**) is updated on the device successfully.



PLSDCluster.High_Voltage.Generators.GEN1

Value

1 Close

Authentication

Username

Password

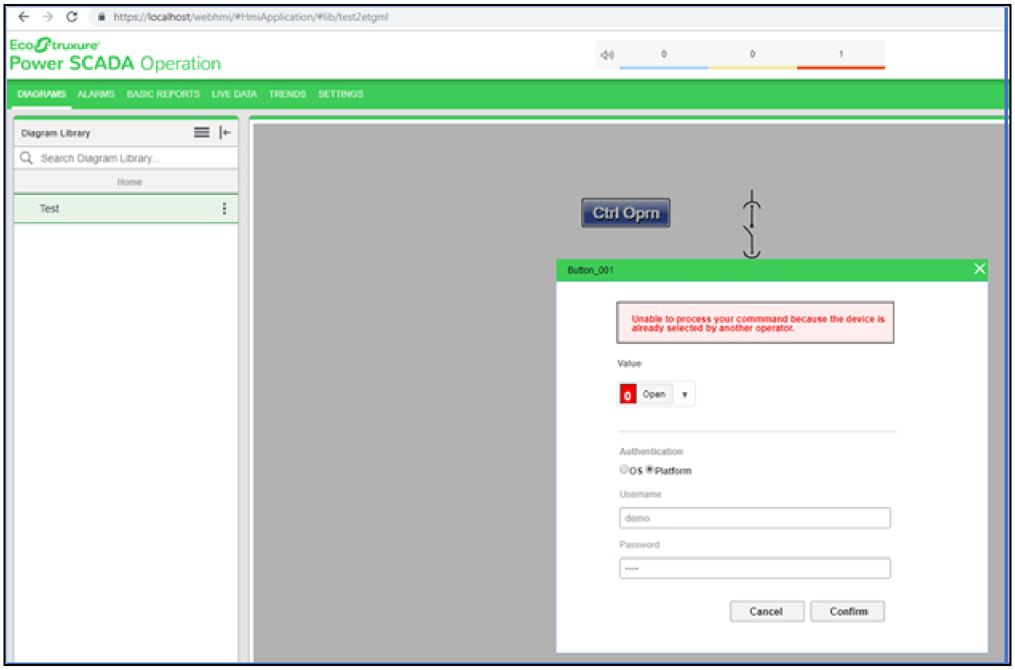
Cancel Confirm

Success

Control command was successfully performed.

Close

- **Unsuccessful**: This popup appears when the operation is failed due to the following reasons:
  - The device is already in the selected state.
  - The device is selected by another user.
  - Another device problem or issue exists.



Button_001

Unable to process your command because the device is already selected by another operator.

Value

0 Open

Authentication

○ OS ● Platform

Username

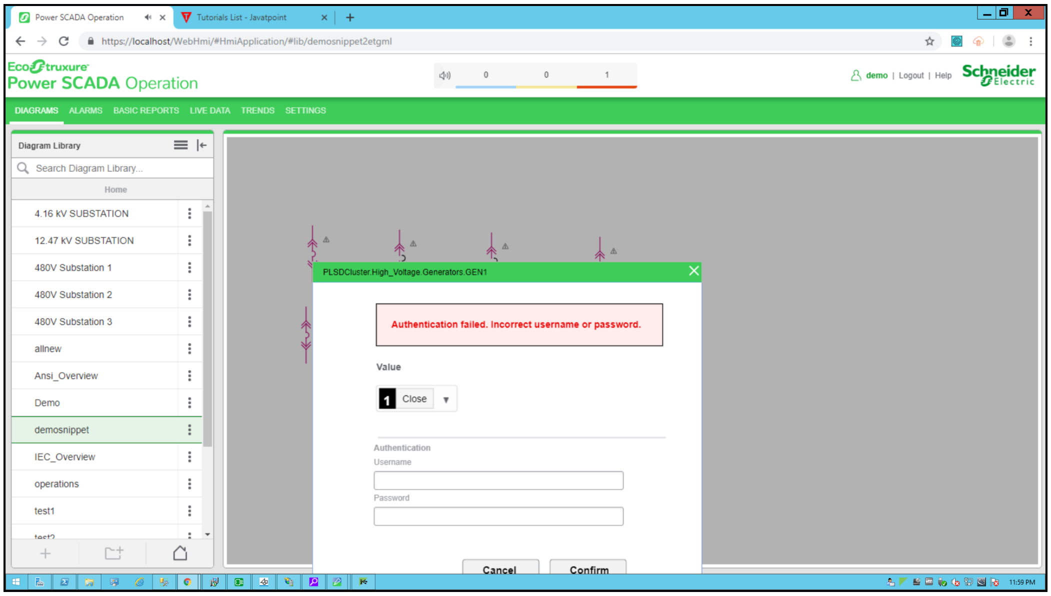
demo

Password

****

Cancel Confirm

– **Authentication failed**: This appears when the provided credentials are not valid.



PLSDCluster.High_Voltage.Generators.GEN1

Authentication failed. Incorrect username or password.

Value

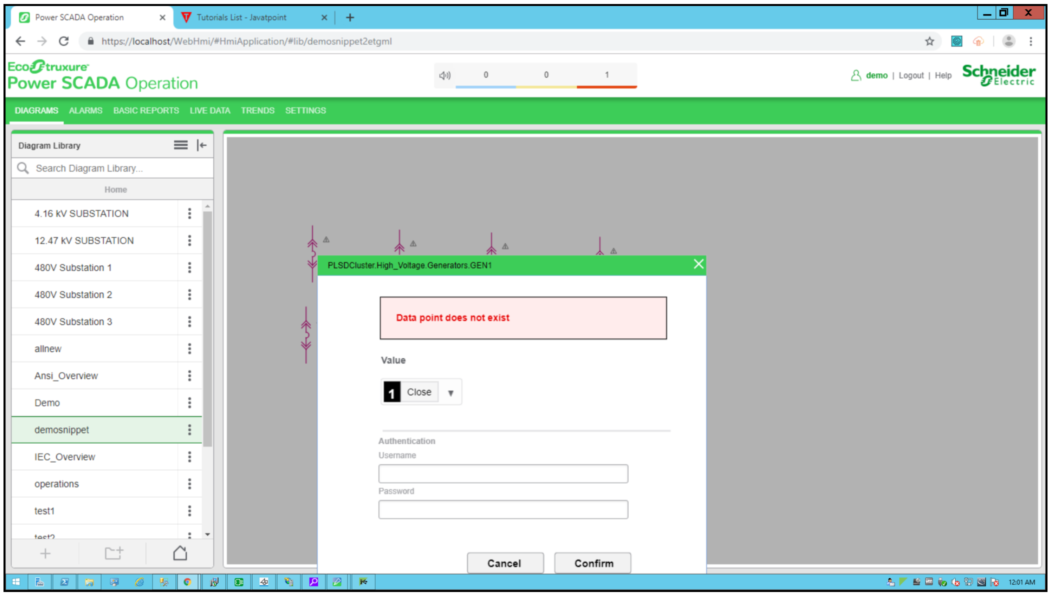
1 Close

Authentication

Username

Password

– **Data point does not exist:** This appears when the provided tag names are not correct.



PLSDCluster.High_Voltage.Generators.GEN1

Data point does not exist

Value

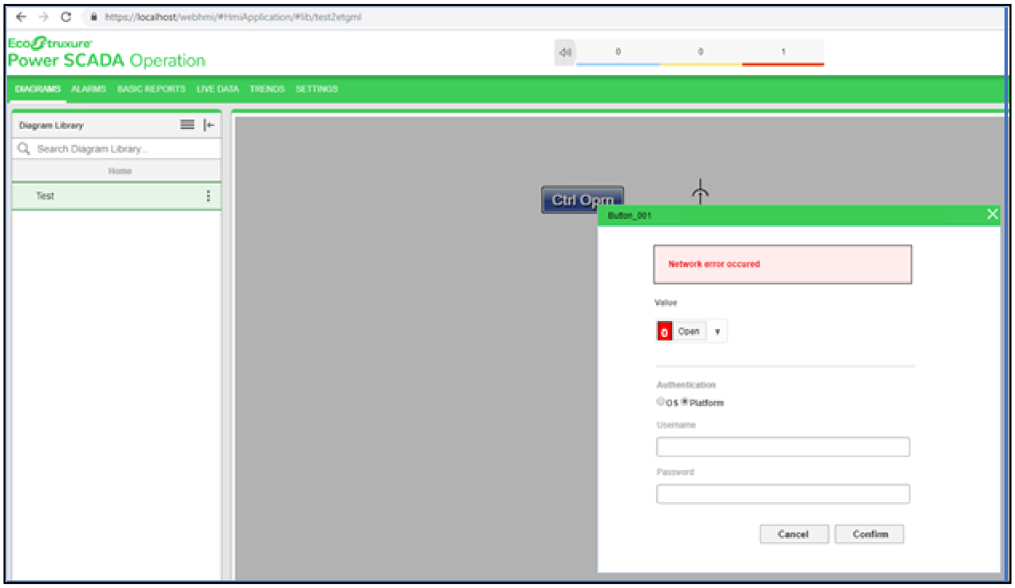
1 Close

Authentication

Username

Password

- **Network error**: This appears when there are network related issues.



Network error occured

Link snippet example

When you click a TGML graphic that has a configured Link snippet, another TGML page is displayed. Typically, you would use link snippets to navigate between TGML pages.

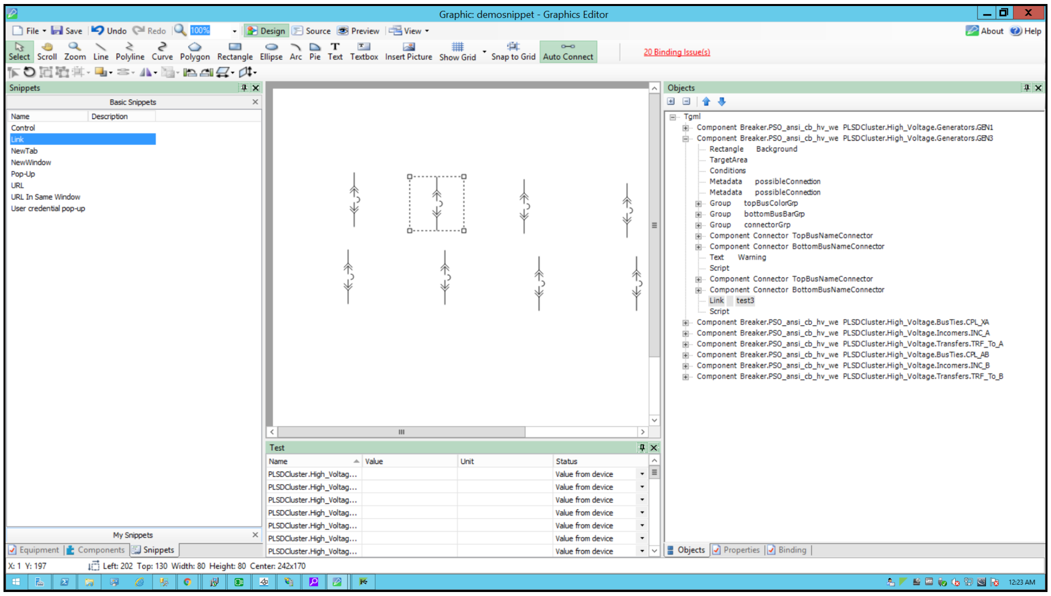
This topic uses an example to illustrate how to configure a Link snippet.

Prerequisites

This example uses a graphic file that already has a binded component or equipment in the workspace. For more information on how to prepare the TGML graphic snippet examples, see "Prerequisites" on page 504.

To create a Link snippet:

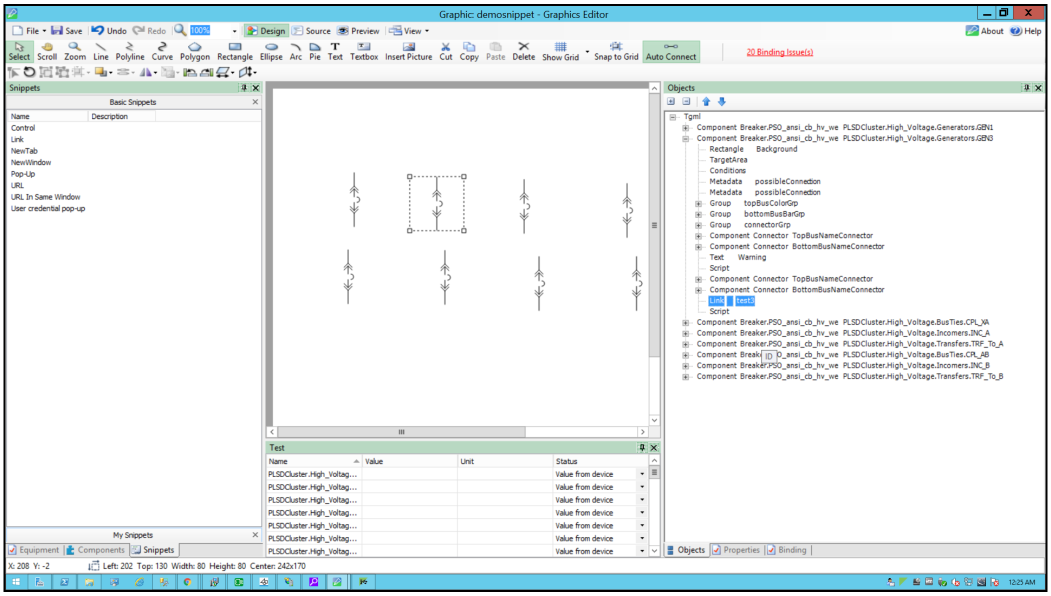
1. In the left bottom corner, click **Snippet**, and then select **Link**.
2. Drag and drop the **Link** snippet over the selected component in the workspace. For example:


Graphic: demosnippet - Graphics Editor
File Save Undo Redo 100% Design Source Preview View About Help
Select Scroll Zoom Line Polyline Curve Polygon Rectangle Ellipse Arc Pie Text Textbox Insert Picture Show Grid Snap to Grid Auto Connect
20 Binding Issue(s)
Snippets
Basic Snippets
Name Description
Control
Link
NewTab
NewWindow
Pop-Up
URL
URL In Same Window
User credential pop-up
My Snippets
Equipment Components Snippets
Objects
Tgml
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Generators.GEN1
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Generators.GEN3
Rectangle Background
TargetArea
Conditions
Metadata possibleConnection
Metadata possibleConnection
Group topBusColorGrp
Group bottomBusBarGrp
Group connectorGrp
Component Connector TopBusNameConnector
Component Connector BottomBusNameConnector
Text Warning
Script
Component Connector TopBusNameConnector
Component Connector BottomBusNameConnector
Link test3
Script
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.BusTies.CPL_XA
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Incomers.INC_A
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Transfers.TRF_To_A
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.BusTies.CPL_AB
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Incomers.INC_B
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Transfers.TRF_To_B
Test
Name Value Unit Status
PLSDCluster.High_Voltag... Value from device
Objects Properties Binding
X: 1 Y: 197 Left: 202 Top: 130 Width: 80 Height: 80 Center: 242x170
12:23 AM

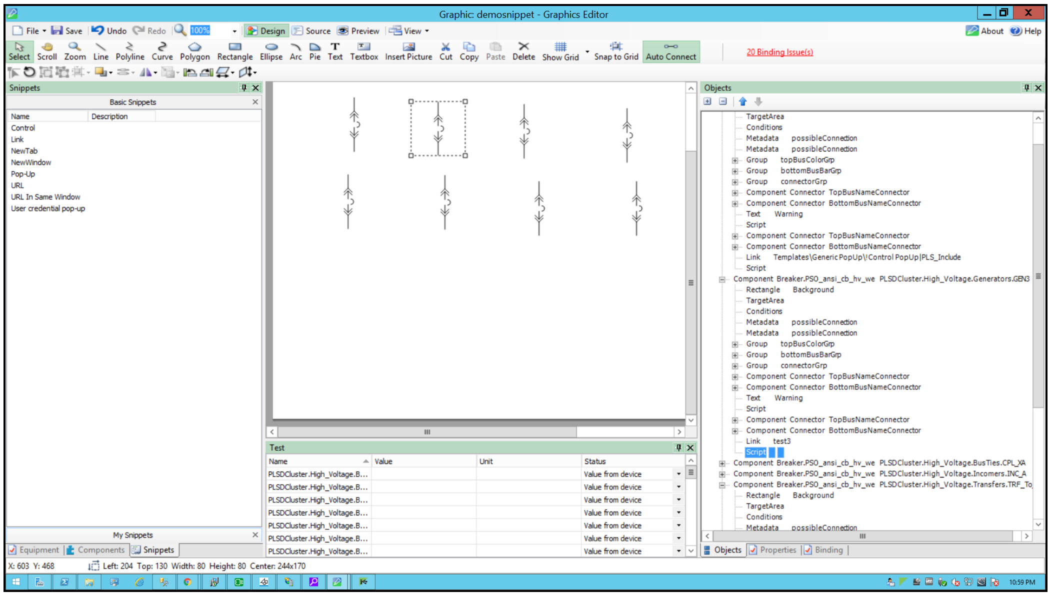
3. In the bottom right corner, click **Objects**, and then expand the TGML node.

   Two additional properties appear: **Link** and **Script**.

4. Update the link with the tgml file to be opened. For example: test3

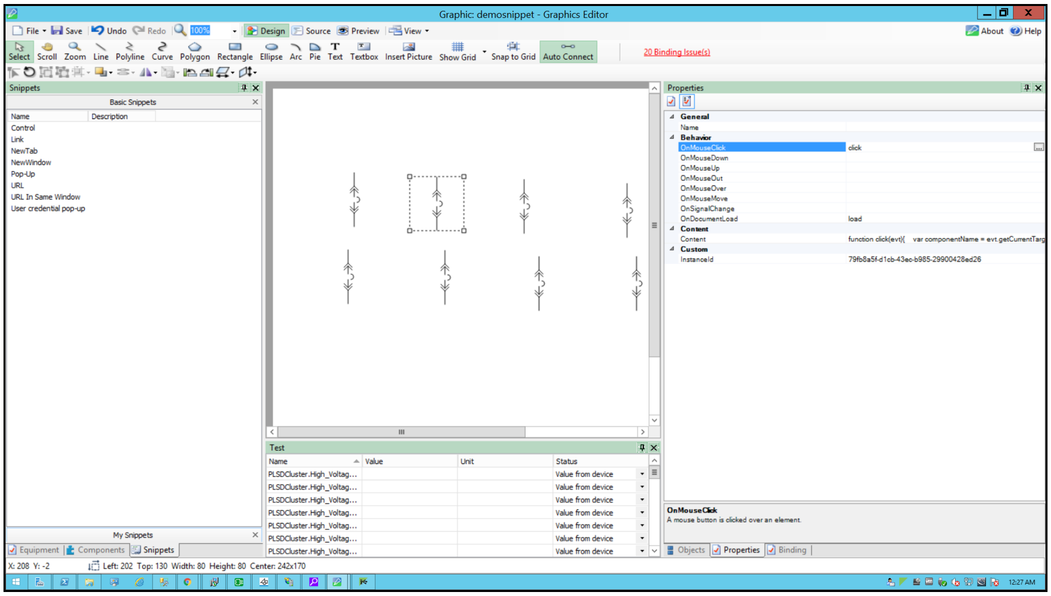

Graphic: demosnippet - Graphics Editor
File Save Undo Redo 100% Design Source Preview View About Help
Select Scroll Zoom Line Polyline Curve Polygon Rectangle Ellipse Arc Pie Text Textbox Insert Picture Cut Copy Paste Delete Show Grid Snap to Grid Auto Connect
20 Binding Issue(s)
Snippets
Basic Snippets
Name Description
Control
Link
NewTab
NewWindow
Pop-Up
URL
URL In Same Window
User credential pop-up
My Snippets
Equipment Components Snippets
Objects
Tgml
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Generators.GEN1
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Generators.GEN3
Rectangle Background
TargetArea
Conditions
Metadata possibleConnection
Metadata possibleConnection
Group topBusColorGrp
Group bottomBusBarGrp
Group connectorGrp
Component Connector TopBusNameConnector
Component Connector BottomBusNameConnector
Text Warning
Script
Component Connector TopBusNameConnector
Component Connector BottomBusNameConnector
Link test3
Script
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.BusTies.CPL_XA
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Incomers.INC_A
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Transfers.TRF_To_A
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.BusTies.CPL_AB
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Incomers.INC_B
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Transfers.TRF_To_B
Test
Name Value Unit Status
PLSDCluster.High_Voltag... Value from device
Objects Properties Binding
X: 208 Y: -2 Left: 202 Top: 130 Width: 80 Height: 80 Center: 242x170
12:25 AM

5. Click **Script**:



6. In the bottom right corner, click **Properties**, and then expand **Behavior**.

7. Click the ellipsis button in **OnMouseClick**:



Properties

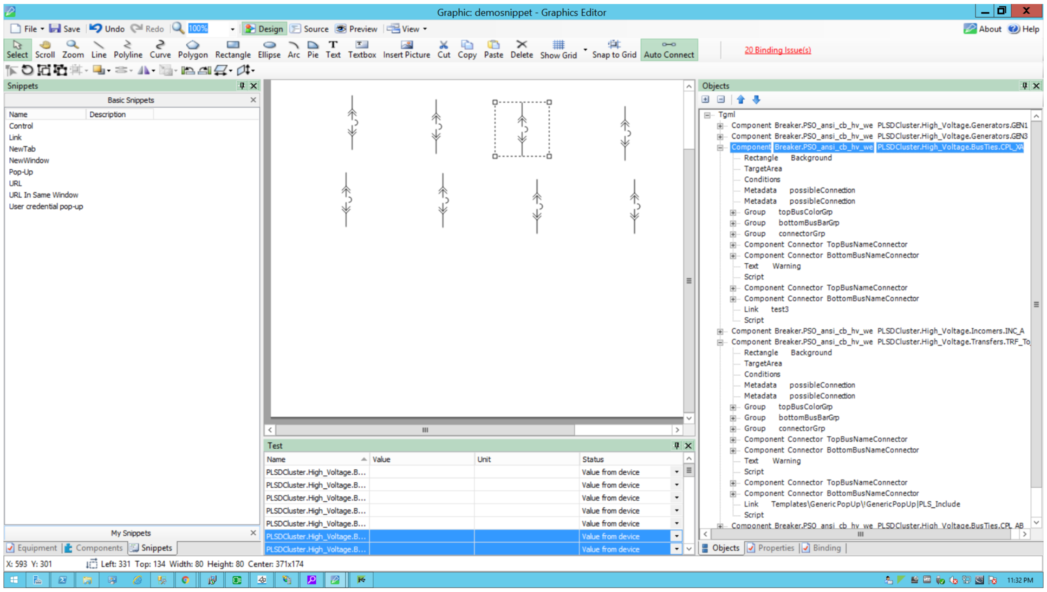
General	
Name	
Behavior	
OnMouseClick	click
OnMouseDown	
OnMouseUp	
OnMouseOut	
OnMouseOver	
OnMouseMove	
OnSignalChange	
OnDocumentLoad	load
Content	
Content	function click(evt){ var componentName = evt.getCurrentTarg
Custom	
InstanceId	79fb8a5f-d1cb-43ec-b985-29900428ed26

OnMouseClick
A mouse button is clicked over an element.

8. Use the following script to set the TGML snippet's click behavior, and then close the script window:

```
function click(evt)
{
//componentName is name of the component based on the component selection we
will fetch the component name
var componentName = evt.getCurrentTarget().getAttribute("Name");
//Collecting the links from the Component
var Link = evt.getCurrentTarget().getElementsByTagName("Link");

//customExpose-If two breakers are internally connected (means multi
equipment);
var customExpose = evt.getCurrentTarget().getAttribute("SubstituteNames");

for (var i=0;i< Link.length;i++) {
//LinkFileName : Extracting the file name from the Link
var LinkFileName = Link.item(i).getAttribute("Name");

//With invoke function you can configure the graphic component in Graphics
Editor to open a linked target object in a target location when you perform
a action(link) on the component
invoke(LinkFileName, "Type = Link | ComponentName=" + componentName + " |
CustomExpose=" + customExpose);
}
}

function load(evt)
{
}
```

9. Open the TGML graphic and click on the **Component** in the TGML:



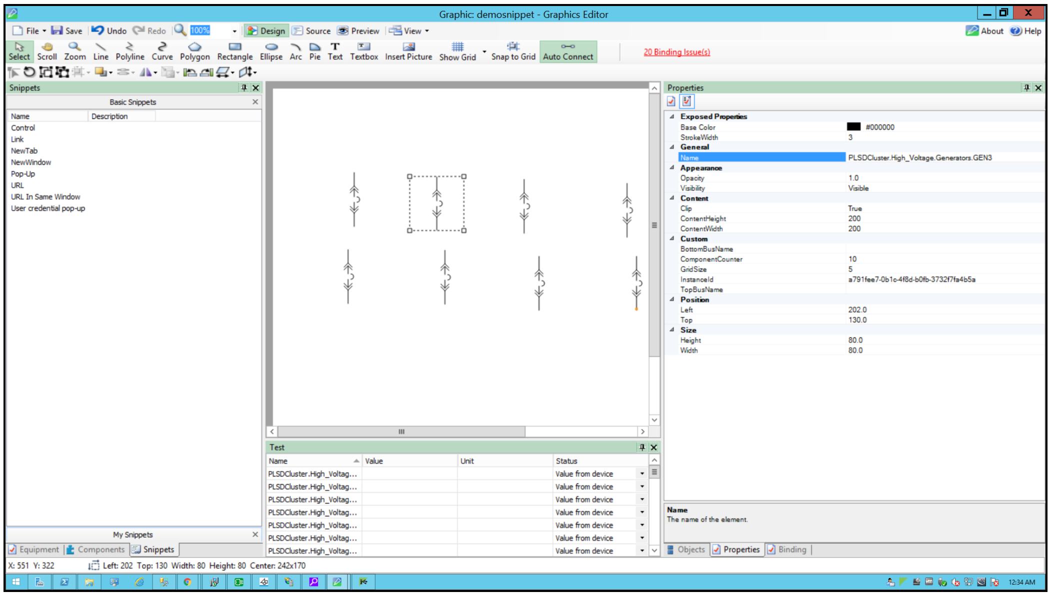
```
Tgml
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.GEN1
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.GEN3
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.BusTies.CPL_XA
    Rectangle   Background
    TargetArea
    Conditions
    Metadata   possibleConnection
    Metadata   possibleConnection
    Group   topBusColorGrp
    Group   bottomBusBarGrp
    Group   connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Text   Warning
    Script
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Link   test3
    Script
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Incomers.INC_A
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Transfers.TRF_To
    Rectangle   Background
    TargetArea
    Conditions
    Metadata   possibleConnection
    Metadata   possibleConnection
    Group   topBusColorGrp
    Group   bottomBusBarGrp
    Group   connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Text   Warning
    Script
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Link   Templates\GenericPopUp\!GenericPopUp|PLS_Include
    Script
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.BusTies.CPL_AB
```

Name	Value	Unit	Status
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device

X: 593 Y: 301 Left: 331 Top: 134 Width: 80 Height: 80 Center: 371x174

10. Click on **Properties** tab in the bottom right corner.

Click on **General > Name**, the name which is displaying in **Name** is a component name.



Properties	
Exposed Properties	
Base Color	#000000
StrokeWidth	3
General	
Name	PLSDCluster.High_Voltage.Generators.GEN3
Appearance	
Opacity	1.0
Visibility	Visible
Content	
Clip	True
ContentHeight	200
ContentWidth	200
Custom	
BottomBusName	
ComponentCounter	10
GridSize	5
InstanceId	a791fee7-0b1c-4f8d-b0fb-3732f7fa4b5a
TopBusName	
Position	
Left	202.0
Top	130.0
Size	
Height	80.0
Width	80.0

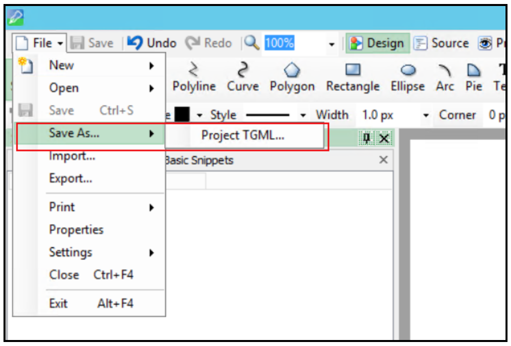
Name
The name of the element.

Test

Name	Value	Unit	Status
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device

X: 551 Y: 322 Left: 202 Top: 130 Width: 80 Height: 80 Center: 242x170

11. Go to **File** > **Save As** > **Project TGML**.

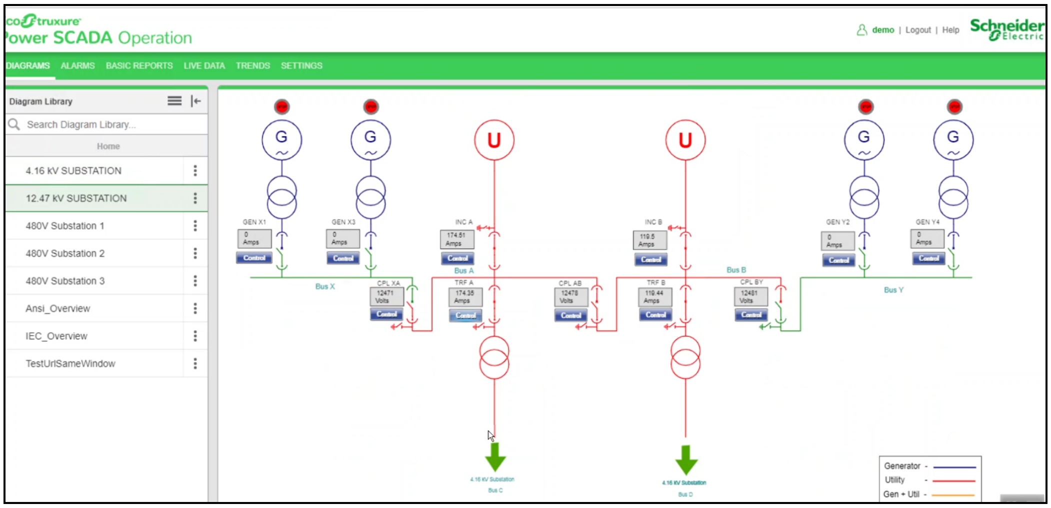


12. Enter a file name, and then click **Save**.

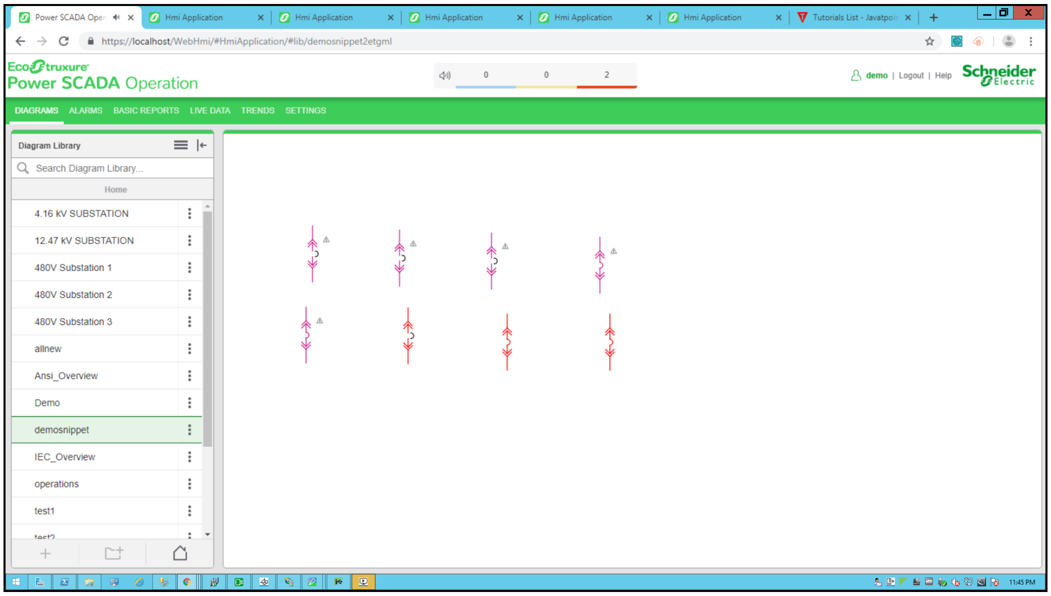
To view the snippet behavior:

1. In a web browser, log in to PSOWeb Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)

The Power SCADA Web Applications Home page appears:

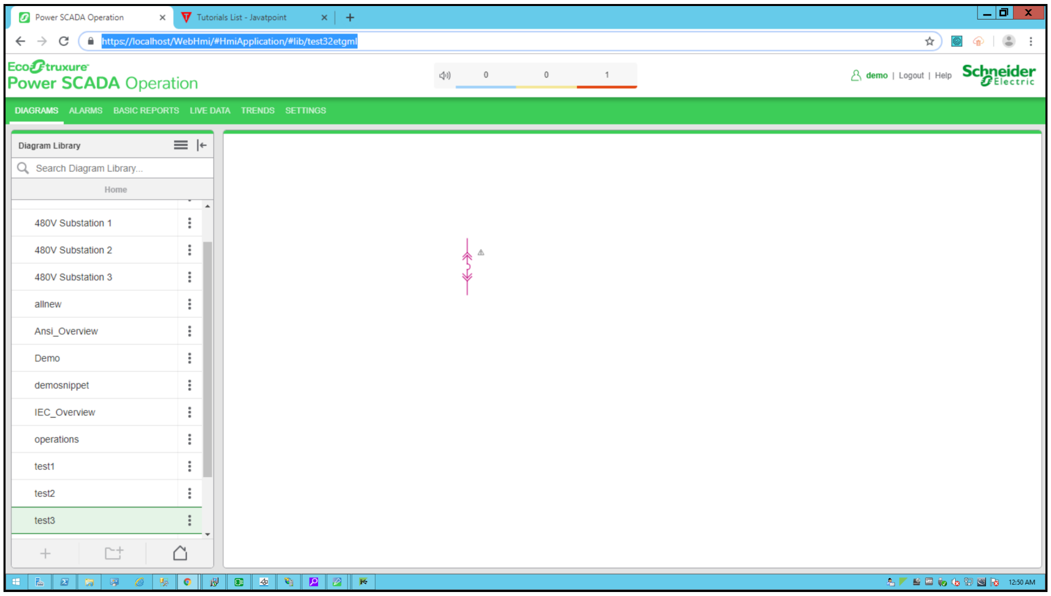


2. Select the new TGML file from the left panel **Diagram Library** as shown below.



3. Click on the circuit breaker.

The new link is opened:



NewTab snippet example

When you click a TGML graphic that has a configured URL In Same Window snippet, another site or web application page opens in the same window.

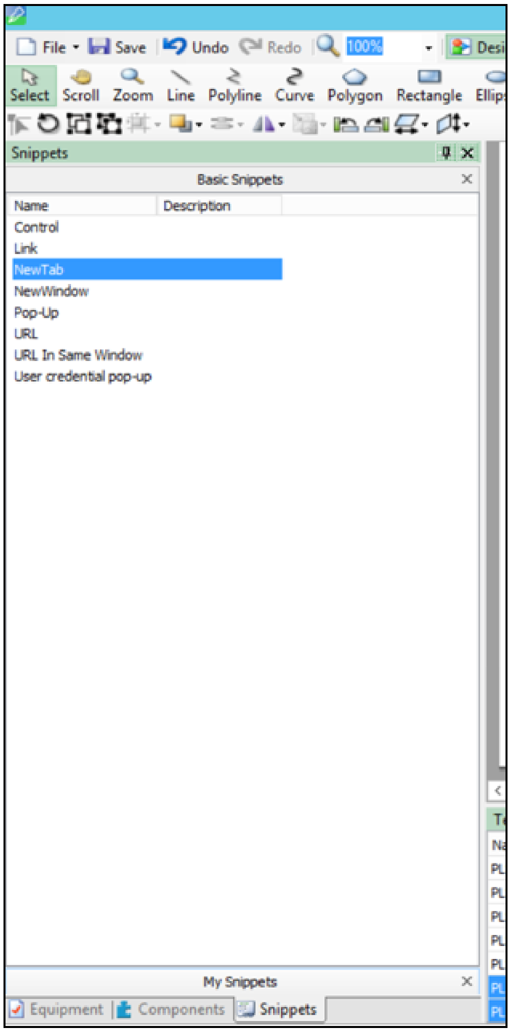
This topic uses an example to illustrate how to configure a NewTab snippet.

Prerequisites

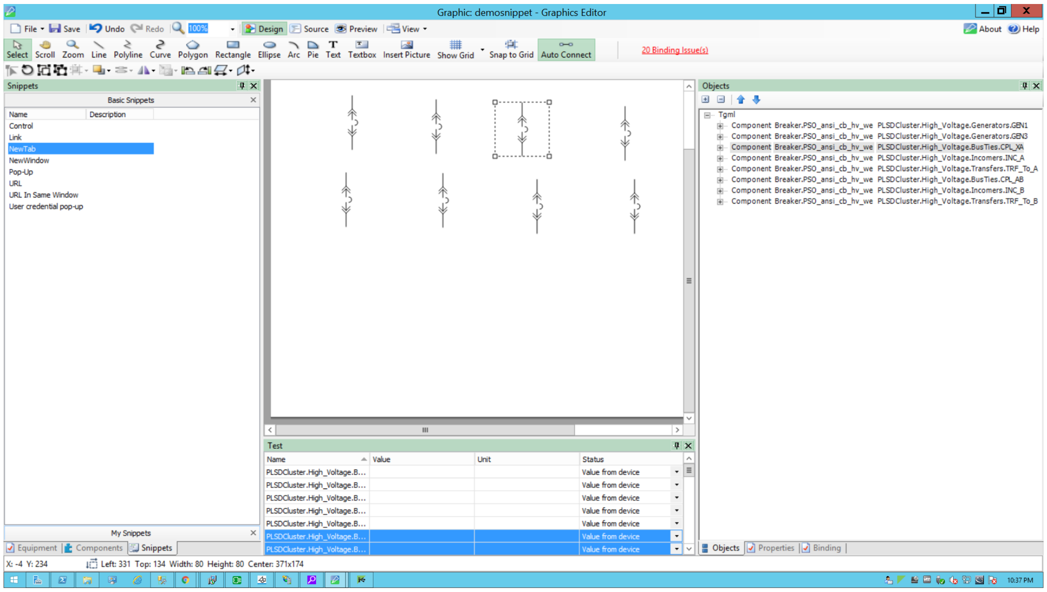
This example uses a graphic file that already has a binded component or equipment in the workspace. For more information on how to prepare the TGML graphic snippet examples, see "Prerequisites" on page 504.

To create a NewTab snippet:

1. Click **Snippets** pane in the bottom left corner and click on **NewTab** from the list of snippets.



2. Drag and drop the **NewTab** snippet over the selected component in the workspace and save it.



Graphic: demosnippet - Graphics Editor

20 Binding Issue(s)

Snippets

Basic Snippets

Name	Description
Control	
Link	
NewTab	
NewWindow	
Pop-Up	
URL	
URL In Same Window	
User credential pop-up	

My Snippets

Test

Name	Value	Unit	Status
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device

Objects

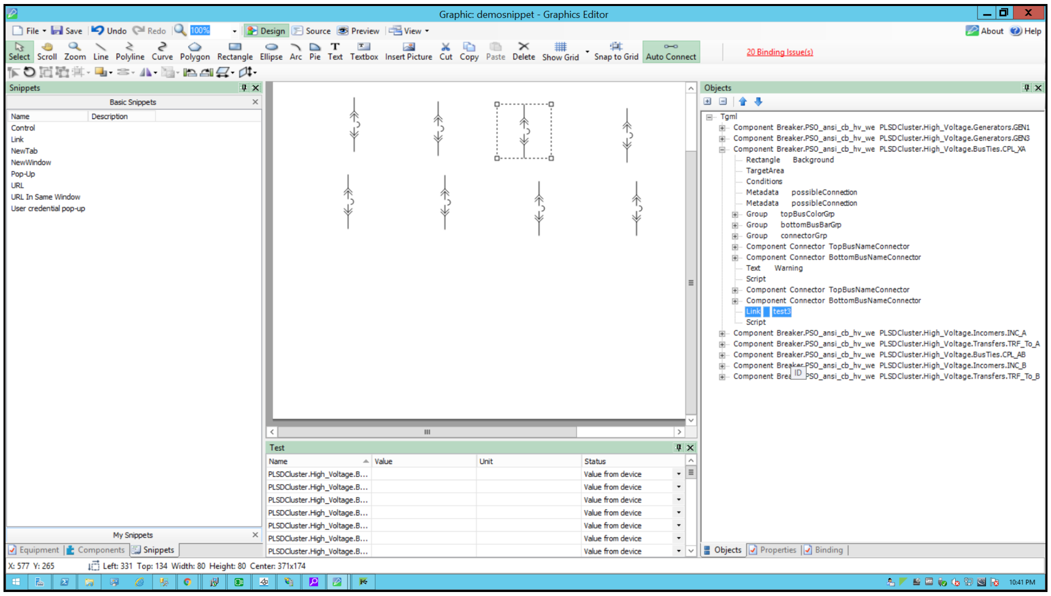
```
Tgml
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Generators.GEN1
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Generators.GEN3
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.BusTies.CPL_XA
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Incomers.INC_A
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Transfers.TRF_To_A
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.BusTies.CPL_AB
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Incomers.INC_B
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Transfers.TRF_To_B
```

X: -4 Y: 234 Left: 331 Top: 134 Width: 80 Height: 80 Center: 371x174

3. Click on **Objects** pane in the bottom right corner and click on **+** to open the TGML.

   Two additional properties appears: **Link** and **Script**.

4. Update the link with the TGML file to be opened (for example, test3).



Graphic: demosnippet - Graphics Editor

20 Binding Issue(s)

Snippets

Basic Snippets

Name	Description
Control	
Link	
NewTab	
NewWindow	
Pop-Up	
URL	
URL In Same Window	
User credential pop-up	

My Snippets

Test

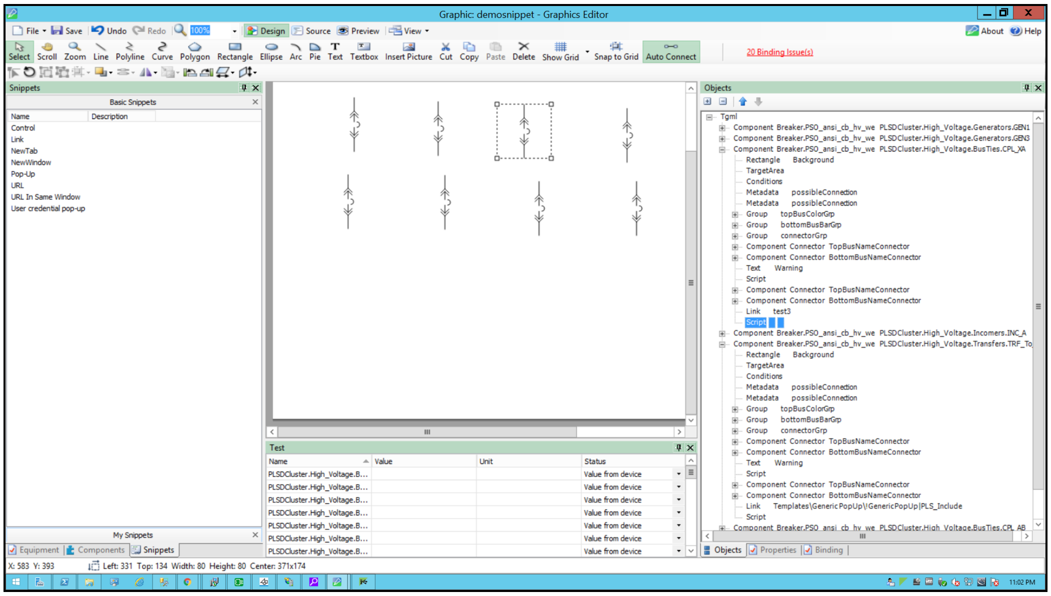
Name	Value	Unit	Status
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device

Objects

```
Tgml
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Generators.GEN1
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Generators.GEN3
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.BusTies.CPL_XA
    Rectangle Background
    TargetArea
    Conditions
    Metadata possibleConnection
    Metadata possibleConnection
    Group topBusColorGrp
    Group bottomBusBarGrp
    Group connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Text Warning
    Script
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Link test3
    Script
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Incomers.INC_A
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Transfers.TRF_To_A
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.BusTies.CPL_AB
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Incomers.INC_B
  Component Brea  PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Transfers.TRF_To_B
```

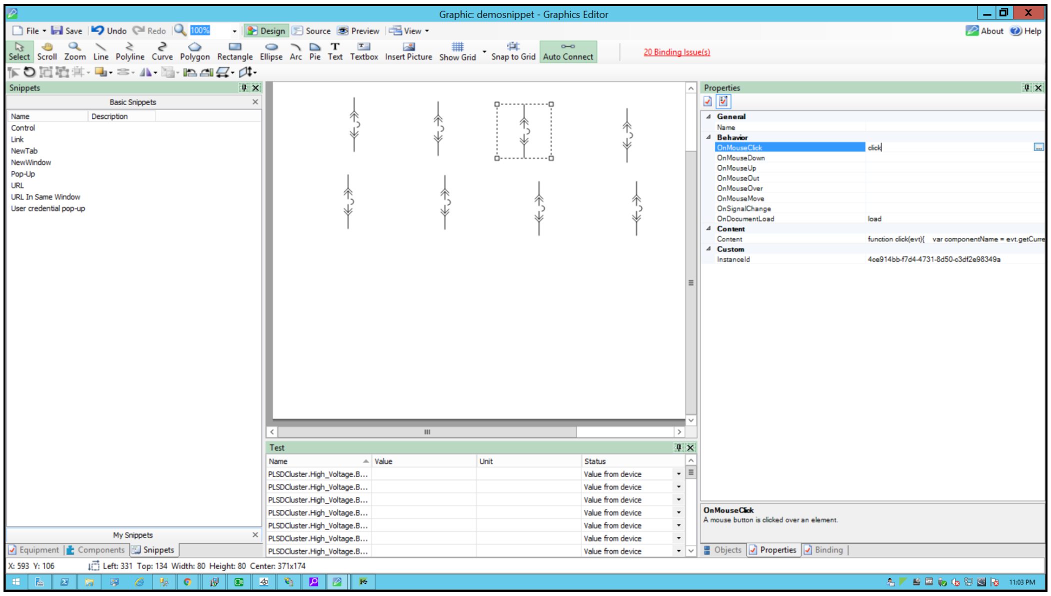
X: 577 Y: 265 Left: 331 Top: 134 Width: 80 Height: 80 Center: 371x174

5. Click on **Script** as shown below.



6. Click on **Properties** pane in the bottom right corner, and then expand **Behavior**.

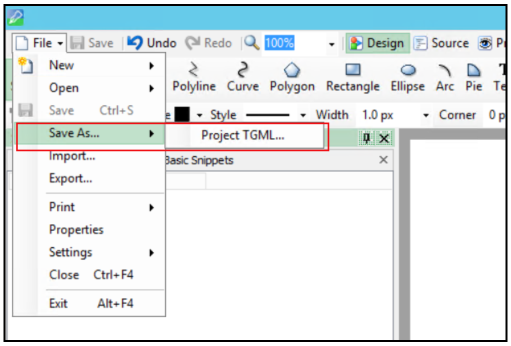
7. Click the ellipsis in **OnMouseClick**:



8. Use the following script to configure the NewTab:

```
function click(evt)
{
//componentName is name of the component based on the component selection we
will fetch the component name
var componentName = evt.getCurrentTarget().getAttribute("Name");

//Collecting the links from the Component
var Link = evt.getCurrentTarget().getElementsByTagName("Link");
//title is component name use for showing the title
var title = componentName;

//customExpose-If two breakers are internally connected (means multi
equipment);
var customExpose = evt.getCurrentTarget().getAttribute("SubstituteNames");
for (var i=0;i< Link.length;i++) {
    //LinkFileName : Extracting the file name from the Link
    var LinkFileName = Link.item(i).getAttribute("Name");
    //With invoke function you can configure the graphic component in
Graphics Editor to open a linked target object in a target location when you
perform a action(NewTab) on the component
    invoke(LinkFileName, "Type = NewTab | ComponentName=" + componentName +
" | Title=" + title + " | CustomExpose=" + customExpose);
    }
}

function load(evt)
{
}
```

9. Go to **File** > **Save As** > **Project TGML**.



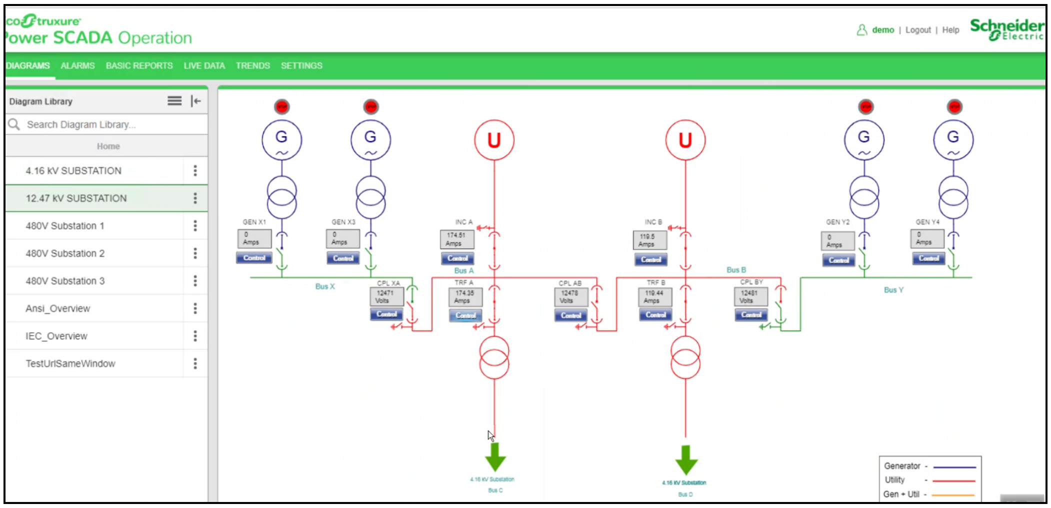
10. Type the file name in the **File name** field.

11. Click **Save**.

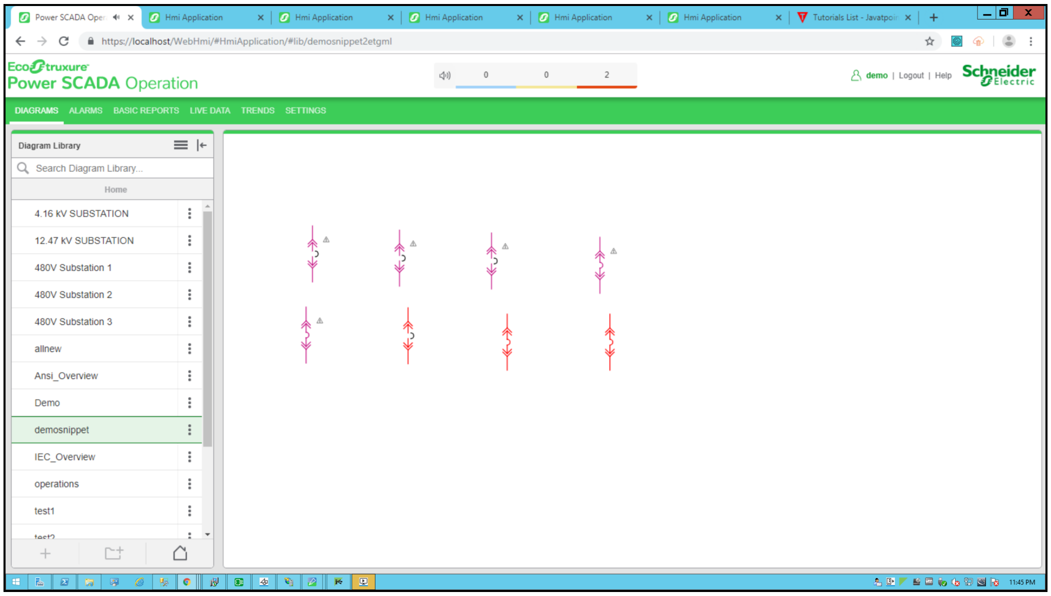
To view the snippet behavior:

1. Log in to PSO Web Applications(**https://localhost/webhmi or https://ipaddress/webhmi**).

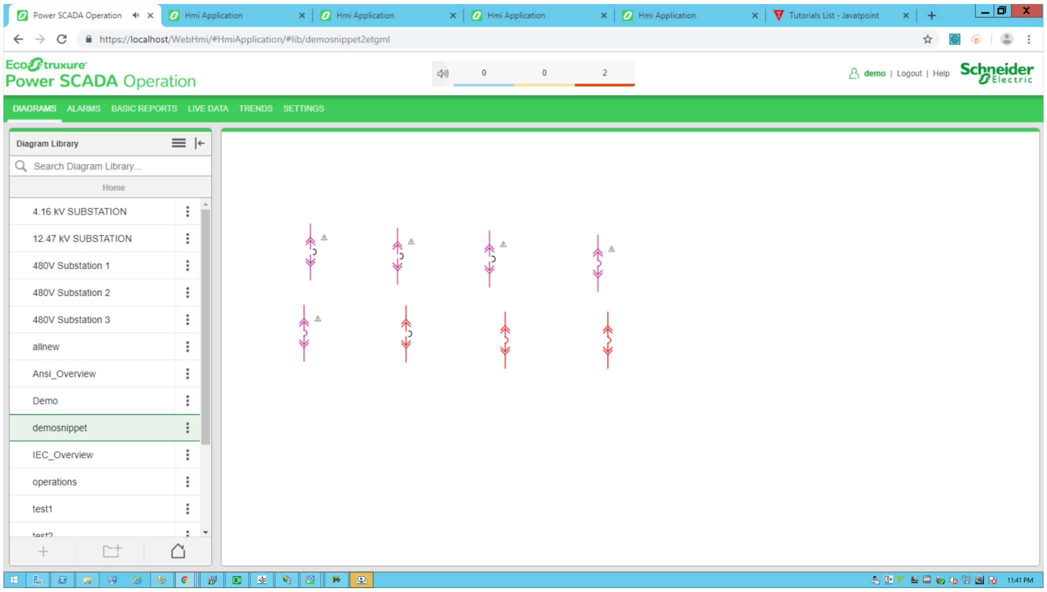
The Power SCADA Web Applications home page appears.



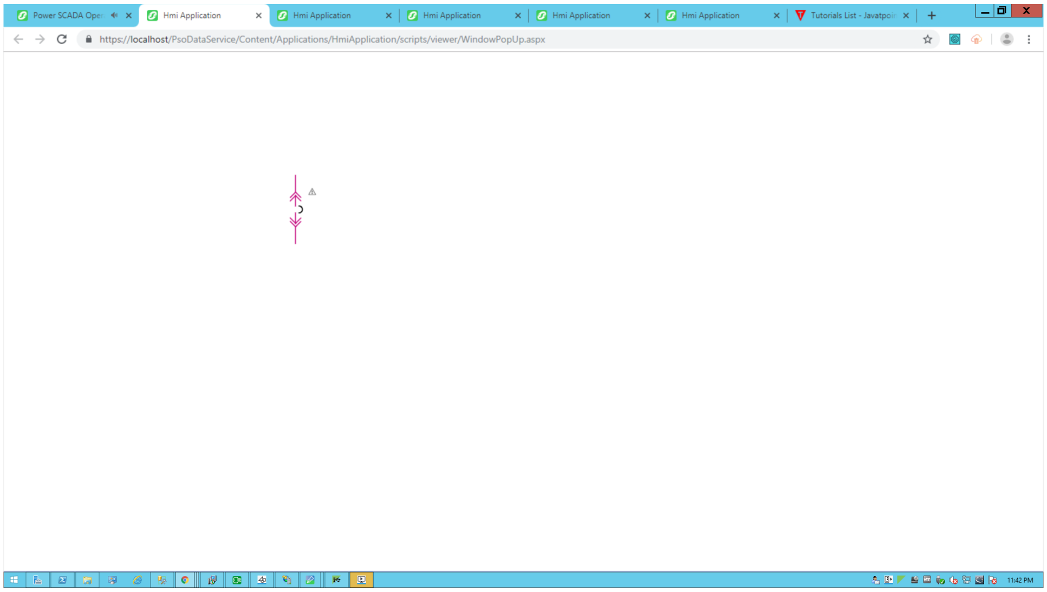
2. Select the new TGML file from the left panel **Diagram Library** as shown below.



3. Click on the breaker to open a **NewTab** from the component as shown below.



4. The **NewTab** opened screen is shown below.


Power SCADA Oper
Hmi Application
Tutorials List - Javatpoi
https://localhost/PsoDataService/Content/Applications/HmiApplication/scripts/viewer/WindowPopUp.aspx
11:42 PM

NewWindow Snippet

When you click a TGML graphic that has a configured URL In Same Window snippet, another site or web application page opens in the same window.

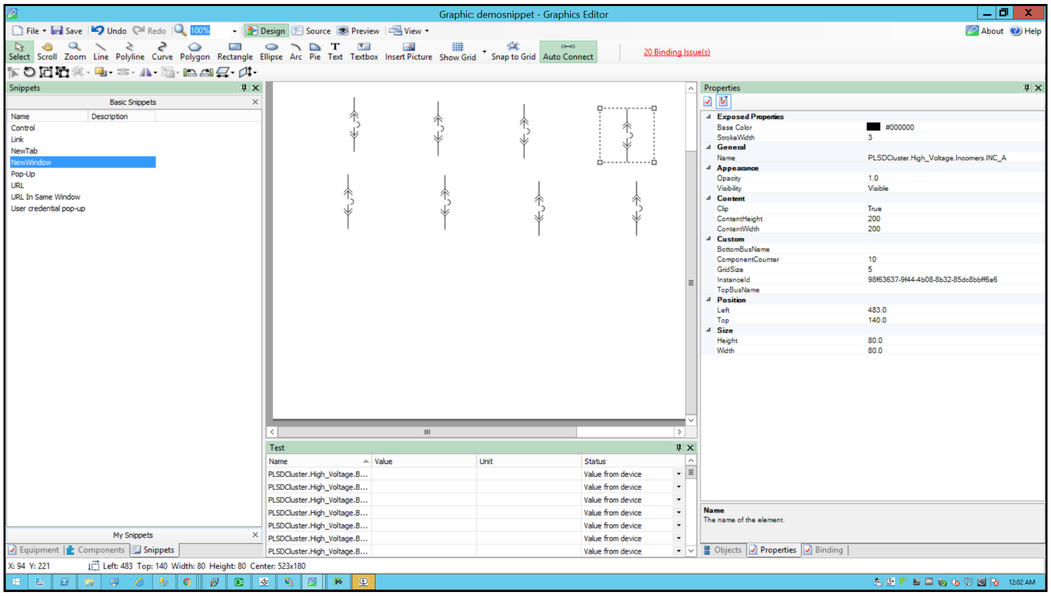
This topic uses an example to illustrate how to configure a NewWindow snippet.

Prerequisites

This example uses a graphic file that already has a binded component or equipment in the workspace. For more information on how to prepare the TGML graphic snippet examples, see "Prerequisites" on page 504.

To create a NewWindow snippet:

1. Click **Snippets** pane in the bottom left corner and click on **NewWindow** from the list of snippets.
2. Drag and drop the **NewWindow** snippet over the selected component in the workspace and save it.


Graphic: demosnippet - Graphics Editor
File Save Undo Redo 100% Design Source Preview View About Help
Select Scroll Zoom Line Polyline Curve Polygon Rectangle Ellipse Arc Pie Text Textbox Insert Picture Show Grid Snap to Grid Auto Connect
20 Binding Issue(s)
Snippets
Basic Snippets
Name Description
Control
Link
NewTab
NewWindow
Pop-Up
URL
URL In Same Window
User credential pop-up
My Snippets
Equipment Components Snippets
Test
Name Value Unit Status
PLSDCluster.High_Voltage.B... Value from device
Objects Properties Binding
X: 94 Y: 221 Left: 483 Top: 140 Width: 80 Height: 80 Center: 523x180
12:02 AM

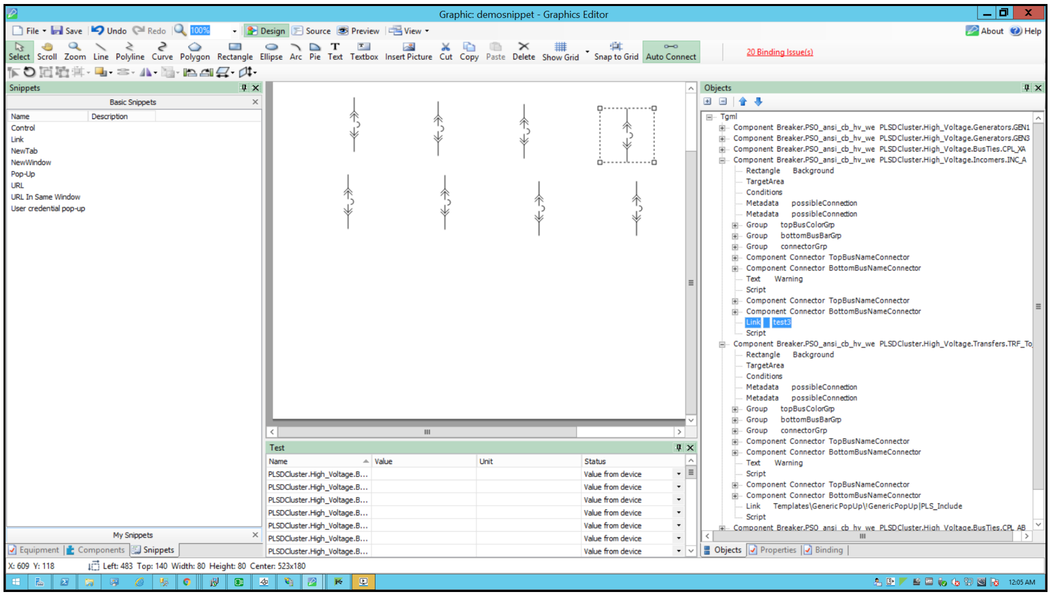
Properties

Exposed Properties	
Base Color	■ #000000
StrokeWidth	3
General	
Name	PLSDCluster.High_Voltage.Incomers.INC_A
Appearance	
Opacity	1.0
Visibility	Visible
Content	
Clip	True
ContentHeight	200
ContentWidth	200
Custom	
BottomBusName	
ComponentCounter	10
GridSize	5
InstanceId	98f63637-9f44-4b08-8b32-85dc8bbff6a6
TopBusName	
Position	
Left	483.0
Top	140.0
Size	
Height	80.0
Width	80.0

Name
The name of the element.

3. Click **Objects** pane in the bottom right corner and click on **+** to open the TGML.

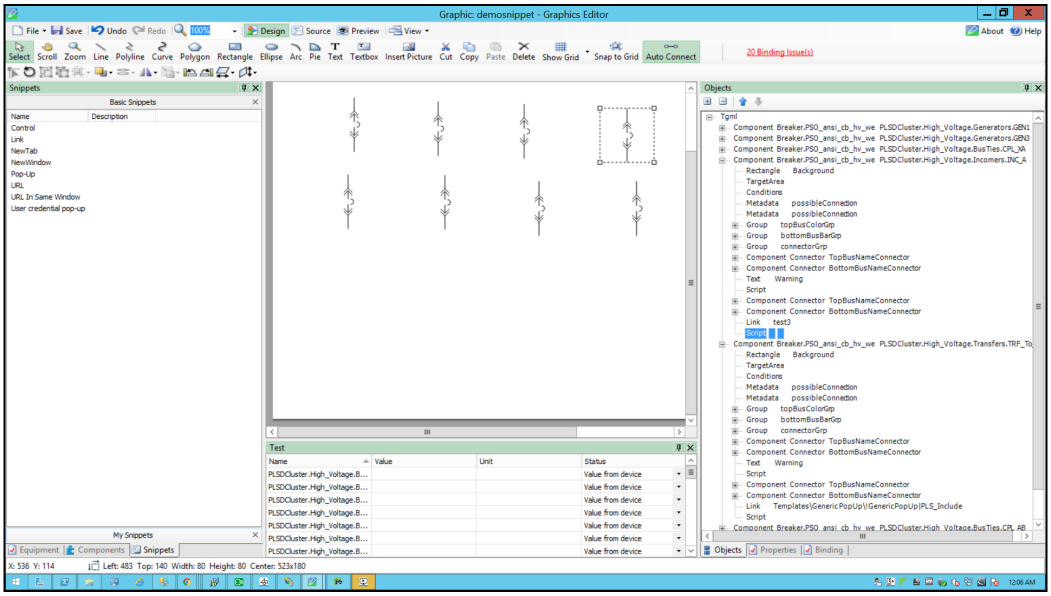
   Two additional properties appears: **Link** and **Script**.
4. Update the link with the TGML file to be opened (for example, test3).


Graphic: demosnippet - Graphics Editor
File Save Undo Redo 100% Design Source Preview View About Help
Select Scroll Zoom Line Polyline Curve Polygon Rectangle Ellipse Arc Pie Text Textbox Insert Picture Cut Copy Paste Delete Show Grid Snap to Grid Auto Connect
20 Binding Issue(s)
Snippets
Basic Snippets
Name Description
Control
Link
NewTab
NewWindow
Pop-Up
URL
URL In Same Window
User credential pop-up
My Snippets
Equipment Components Snippets
Test
Name Value Unit Status
PLSDCluster.High_Voltage.B... Value from device
Objects Properties Binding
X: 609 Y: 118 Left: 483 Top: 140 Width: 80 Height: 80 Center: 523x180
12:05 AM

Objects

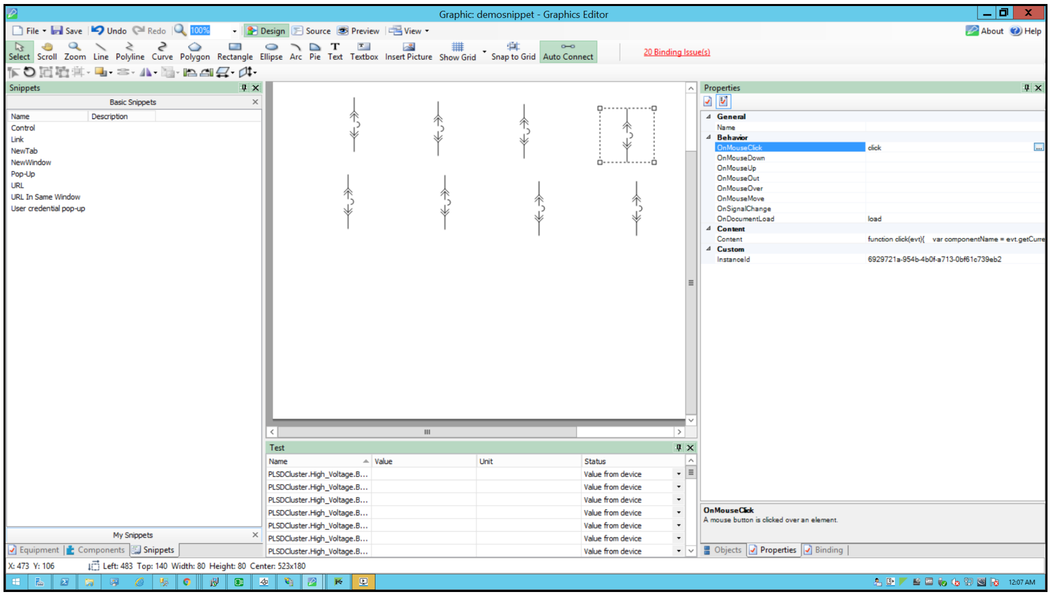
```
Tgml
    Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.GEN1
    Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.GEN3
    Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.BusTies.CPL_XA
    Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Incomers.INC_A
        Rectangle    Background
        TargetArea
        Conditions
        Metadata     possibleConnection
        Metadata     possibleConnection
        Group     topBusColorGrp
        Group     bottomBusBarGrp
        Group     connectorGrp
        Component Connector TopBusNameConnector
        Component Connector BottomBusNameConnector
        Text     Warning
        Script
        Component Connector TopBusNameConnector
        Component Connector BottomBusNameConnector
        Link     test3
        Script
    Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Transfers.TRF_To
        Rectangle    Background
        TargetArea
        Conditions
        Metadata     possibleConnection
        Metadata     possibleConnection
        Group     topBusColorGrp
        Group     bottomBusBarGrp
        Group     connectorGrp
        Component Connector TopBusNameConnector
        Component Connector BottomBusNameConnector
        Text     Warning
        Script
        Component Connector TopBusNameConnector
        Component Connector BottomBusNameConnector
        Link     Templates\GenericPopUp\!GenericPopUp|PLS_Include
        Script
    Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.BusTies.CPL_AB
```

5. Click on **Script** as shown below.


Graphic: demosnippet - Graphics Editor
File
Save
Undo
Redo
100%
Design
Source
Preview
View
About
Help
Select
Scroll
Zoom
Line
Polyline
Curve
Polygon
Rectangle
Ellipse
Arc
Pie
Text
Textbox
Insert Picture
Cut
Copy
Paste
Delete
Show Grid
Snap to Grid
Auto Connect
20 Binding Issue(s)
Snippets
Basic Snippets
Name
Description
Control
Link
NewTab
NewWindow
Pop-Up
URL
URL In Same Window
User credential pop-up
My Snippets
Equipment
Components
Snippets
Objects
Tgml
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Generators.GEN1
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Generators.GEN3
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.BusTies.CPL_XA
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Incomers.INC_A
Rectangle Background
TargetArea
Conditions
Metadata possibleConnection
Metadata possibleConnection
Group topBusColorGrp
Group bottomBusBarGrp
Group connectorGrp
Component Connector TopBusNameConnector
Component Connector BottomBusNameConnector
Text Warning
Script
Component Connector TopBusNameConnector
Component Connector BottomBusNameConnector
Link test3
Script
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Transfers.TRF_To
Rectangle Background
TargetArea
Conditions
Metadata possibleConnection
Metadata possibleConnection
Group topBusColorGrp
Group bottomBusBarGrp
Group connectorGrp
Component Connector TopBusNameConnector
Component Connector BottomBusNameConnector
Text Warning
Script
Component Connector TopBusNameConnector
Component Connector BottomBusNameConnector
Link Templates\GenericPopUp\GenericPopUp|PLS_Include
Script
Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.BusTies.CPL_AB
Objects
Properties
Binding
Test
Name
Value
Unit
Status
PLSDCluster.High_Voltage.B...
Value from device
X: 536 Y: 114
Left: 483 Top: 140 Width: 80 Height: 80 Center: 523x180
12:06 AM

6. Click on **Properties** pane in the bottom right corner and click on **Behaviour list**.

7. Click on three dots blue color box in **OnMouseClick** as shown below.



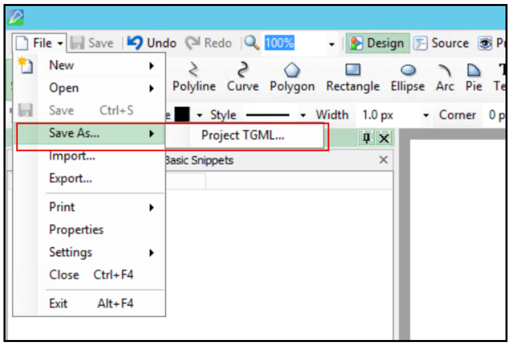
General
Name
Behavior
OnMouseClick click
OnMouseDown
OnMouseUp
OnMouseOut
OnMouseOver
OnMouseMove
OnSignalChange
OnDocumentLoad load
Content
Content function click(evt){ var componentName = evt.getCurre
Custom
InstanceId 6929721a-954b-4b0f-a713-0bf61c739eb2

OnMouseClick
A mouse button is clicked over an element.

8. Use the following script to configure the NewWindow snippet:

```
function click(evt)
{
//componentName is name of the component based on the component selection we
will fetch the component name
var componentName = evt.getCurrentTarget().getAttribute("Name");
//Collecting the links from the Component
var Link = evt.getCurrentTarget().getElementsByTagName("Link");

//title is component name use for showing the title
var title = componentName;

//customExpose-If two breakers are internally connected (means multi
equipment);
var customExpose = evt.getCurrentTarget().getAttribute("SubstituteNames");

//Sets the width of the window
var width = screen.width * 0.45;

//Sets the height of the window
var height = screen.width * 0.4;

for (var i=0;i< Link.length;i++) {
    //LinkFileName : Extracting the file name from the Link
    var LinkFileName = Link.item(i).getAttribute("Name");
    //With invoke function you can configure the graphic component in
Graphics Editor to open a linked target object in a target location when you
perform a action(NewWindow)on the component
    invoke(LinkFileName, "Type = NewWindow | ComponentName=" + componentName
+ " | Title=" + title + " | Width=" + width + " | Height=" + height + " |
CustomExpose=" + customExpose);
    }
}

function load(evt)
{
}
```

9. Go to **File** > **Save As** > **Project TGML**.



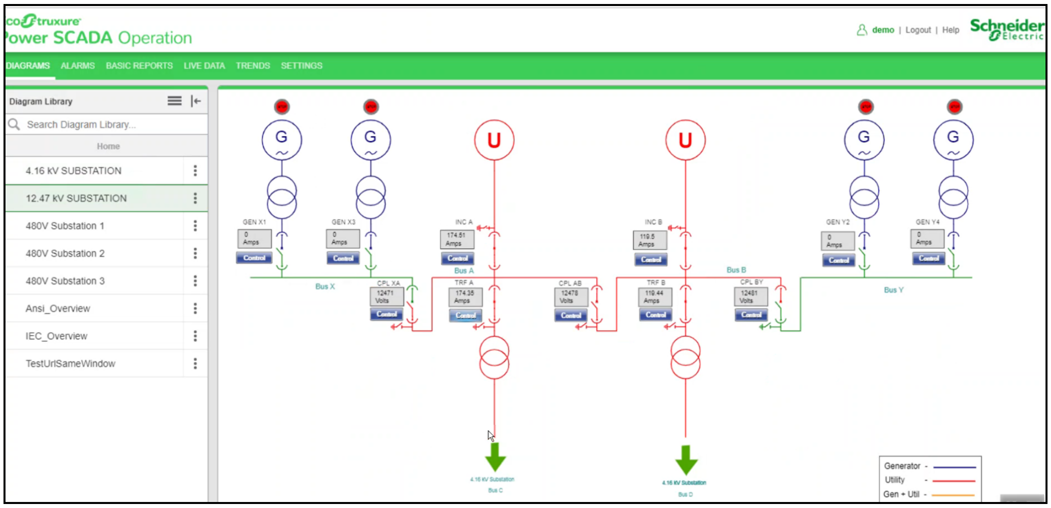
10. Enter the file name in the **File name** field.

11. Click **Save**.

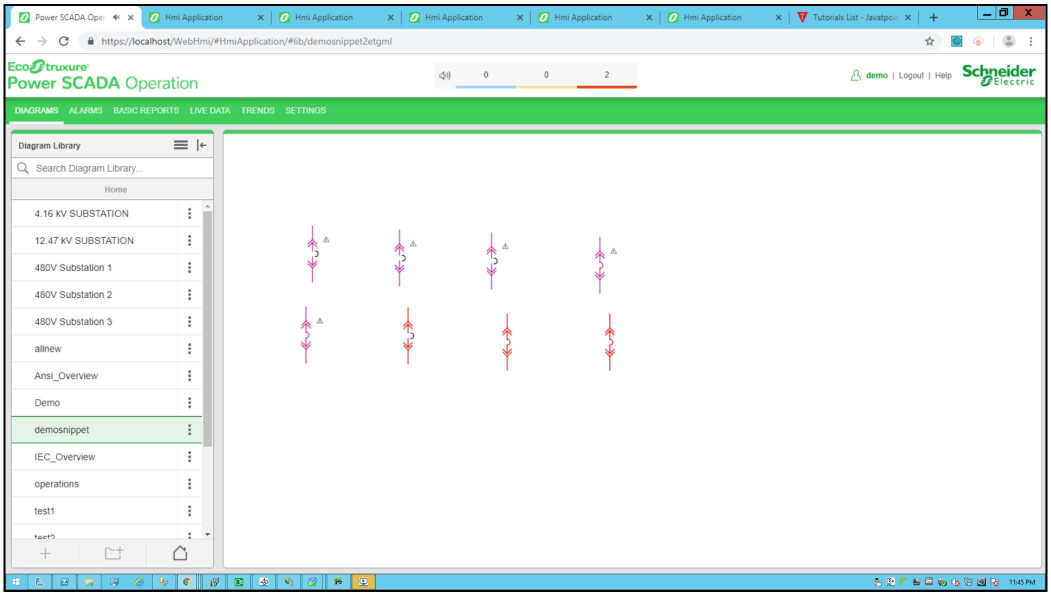
To view the snippet behavior:

1. In a web browser, log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**).

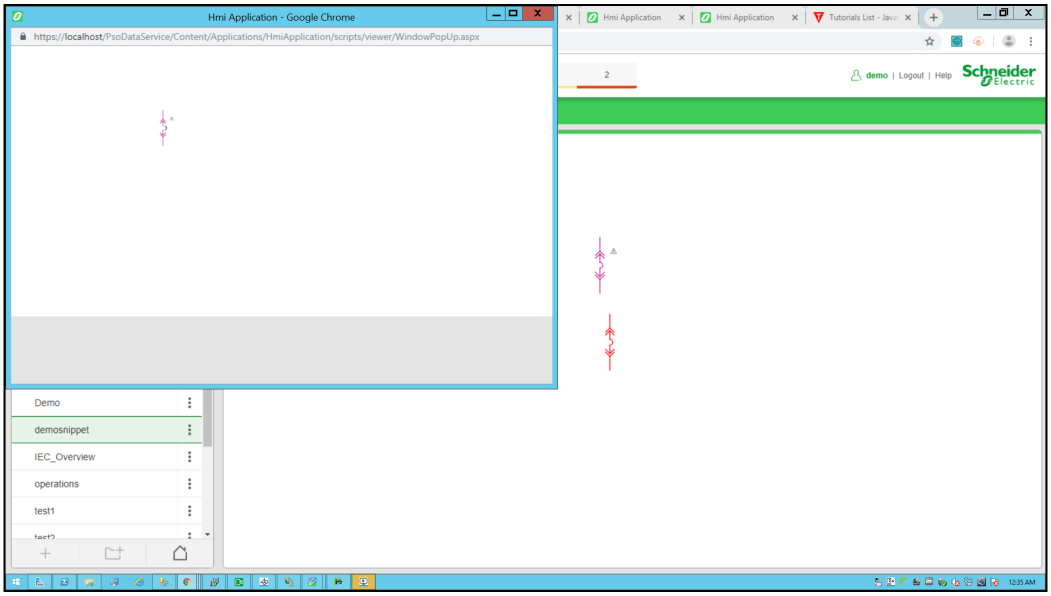
The Power SCADA Web Applications Home page appears.



2. Select the new TGML file from the left panel **Diagram Library** as shown below.



3. Click on the breaker to open in a **NewWindow** as shown below.



PopUp snippet example

When you click a TGML graphic that has a configured PopUp snippet, another TGML graphic opens.

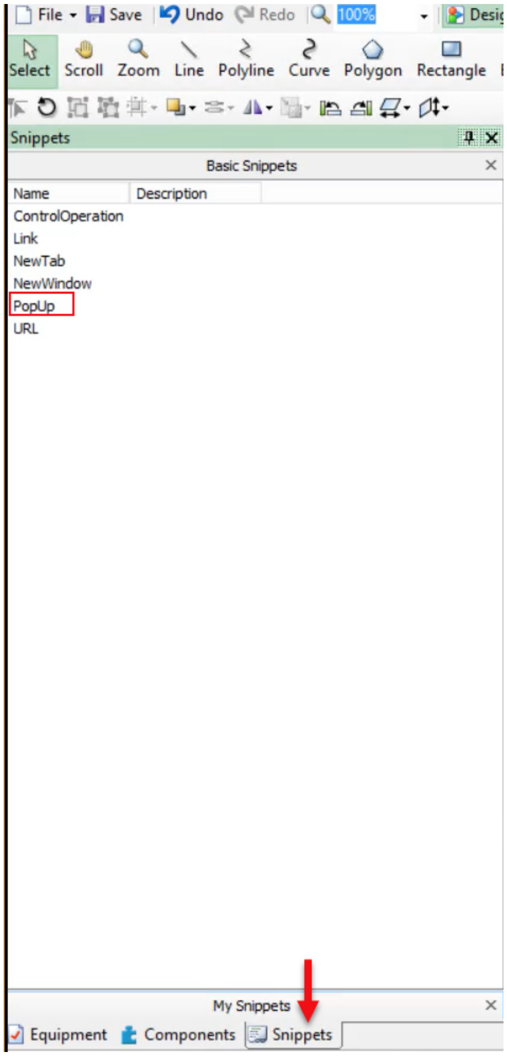
This topic uses an example to illustrate how to configure a PopUp snippet.

Prerequisites

This example uses a graphic file that already has a binded component or equipment in the workspace. For more information on how to prepare the TGML graphic snippet examples, see "Prerequisites" on page 504.

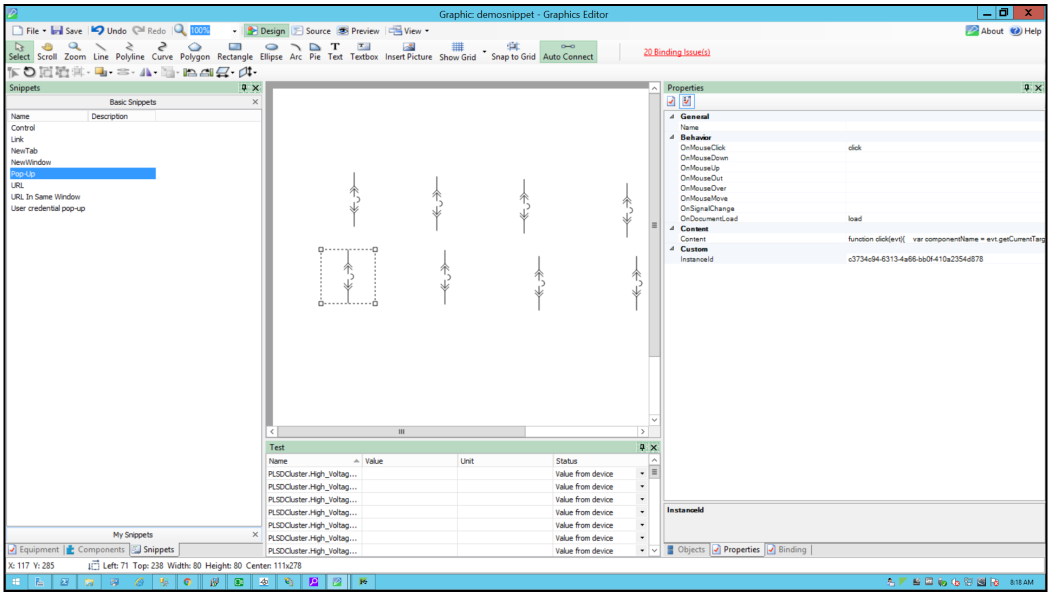
To create a PopUp snippet:

1. At the bottom left corner, click **Snippets**, and then click **PopUp**.



Name	Description
ControlOperation	
Link	
NewTab	
NewWindow	
PopUp	
URL	

2. Drag and drop the **PopUp** snippet over the selected component in the workspace.



Name	Description
Control	
Link	
NewTab	
NewWindow	
Pop-Up	
URL	
URL In Same Window	
User credential pop-up	

General	
Name	
Behavior	
OnMouseClick	click
OnMouseDown	
OnMouseUp	
OnMouseOut	
OnMouseOver	
OnMouseMove	
OnSignalChange	
OnDocumentLoad	load
Content	
Content	function click(evt){ var componentName = evt.getCurrentTarg
Custom	
InstanceId	c3734c94-6313-4a66-bb0f-410a2354d878

InstanceId

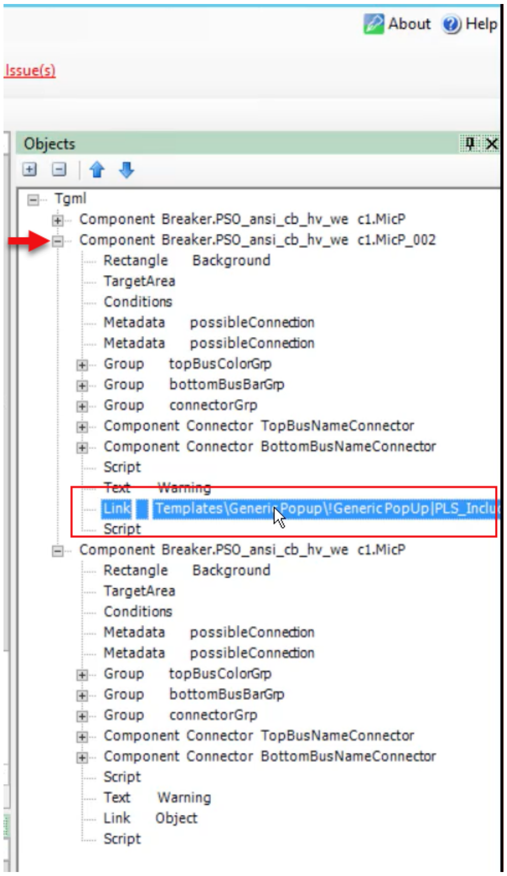
Name	Value	Unit	Status
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device

X: 117 Y: 285 Left: 71 Top: 238 Width: 80 Height: 80 Center: 111x278

3. At the bottom right corner, click **Objects**, and then expand the TGML > Component node.

   2 additional properties appear:

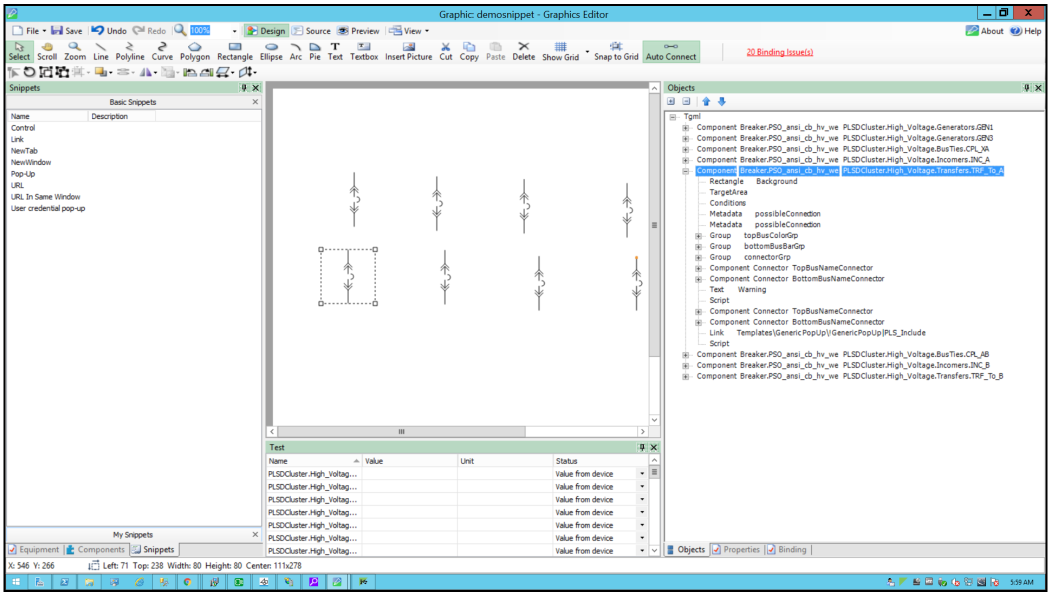
   - **Link**: Enter the page to be opened.
   - **Script**: Enter the display type.



```
Tgml
  Component Breaker.PSO_ansi_cb_hv_we c1.MicP
  Component Breaker.PSO_ansi_cb_hv_we c1.MicP_002
    Rectangle   Background
    TargetArea
    Conditions
    Metadata   possibleConnection
    Metadata   possibleConnection
    Group   topBusColorGrp
    Group   bottomBusBarGrp
    Group   connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Script
    Text   Warning
    Link   Templates\GenericPopup\!GenericPopUp|PLS_Inclu
    Script
  Component Breaker.PSO_ansi_cb_hv_we c1.MicP
    Rectangle   Background
    TargetArea
    Conditions
    Metadata   possibleConnection
    Metadata   possibleConnection
    Group   topBusColorGrp
    Group   bottomBusBarGrp
    Group   connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Script
    Text   Warning
    Link   Object
    Script
```

NOTE: You can add any popup TGML file name in the link. By default, we are using the generic popup tgml file as an example.

4. At the bottom right corner, click **Binding**, and then select the component and the device you want to bind together.


Graphic: demosnippet - Graphics Editor
File Save Undo Redo 100% Design Source Preview View About Help
Select Scroll Zoom Line Polyline Curve Polygon Rectangle Ellipse Arc Pie Text Textbox Insert Picture Cut Copy Paste Delete Show Grid Snap to Grid Auto Connect
20 Binding Issue(s)
Snippets
Basic Snippets
Name Description
Control
Link
NewTab
NewWindow
Pop-Up
URL
URL In Same Window
User credential pop-up
My Snippets
Equipment Components Snippets
Objects Properties Binding

Objects

```
Tgml
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.GEN1
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.GEN3
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.BusTies.CPL_XA
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Incomers.INC_A
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Transfers.TRF_To_A
    Rectangle   Background
    TargetArea
    Conditions
    Metadata   possibleConnection
    Metadata   possibleConnection
    Group   topBusColorGrp
    Group   bottomBusBarGrp
    Group   connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Text   Warning
    Script
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Link   Templates\GenericPopUp\!GenericPopUp|PLS_Include
    Script
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.BusTies.CPL_AB
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Incomers.INC_B
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Transfers.TRF_To_B
```

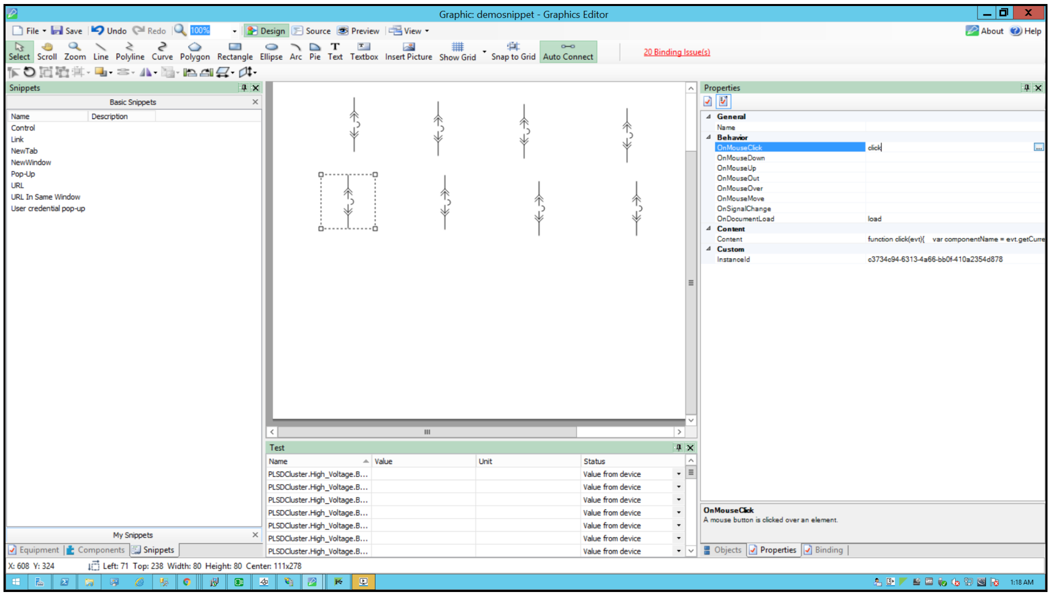
Test

Name	Value	Unit	Status
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device
PLSDCluster.High_Voltag...			Value from device

X: 546 Y: 266 Left: 71 Top: 238 Width: 80 Height: 80 Center: 111x278

5. At the bottom right corner, click **Properties**.

6. Expand **Behavior**, and then click the ellipsis in the **OnMouseClick** value:


Graphic: demosnippet - Graphics Editor
File Save Undo Redo 100% Design Source Preview View About Help
Select Scroll Zoom Line Polyline Curve Polygon Rectangle Ellipse Arc Pie Text Textbox Insert Picture Show Grid Snap to Grid Auto Connect
20 Binding Issue(s)
Snippets
Basic Snippets
Name Description
Control
Link
NewTab
NewWindow
Pop-Up
URL
URL In Same Window
User credential pop-up
My Snippets
Equipment Components Snippets
Objects Properties Binding

Properties

General	
Name	
Behavior	
OnMouseClick	click
OnMouseDown	
OnMouseUp	
OnMouseOut	
OnMouseOver	
OnMouseMove	
OnSignalChange	
OnDocumentLoad	load
Content	
Content	function click(evt){ var componentName = evt.getCurre
Custom	
InstanceId	c3734c94-6313-4a66-bb0f-410a2354d878

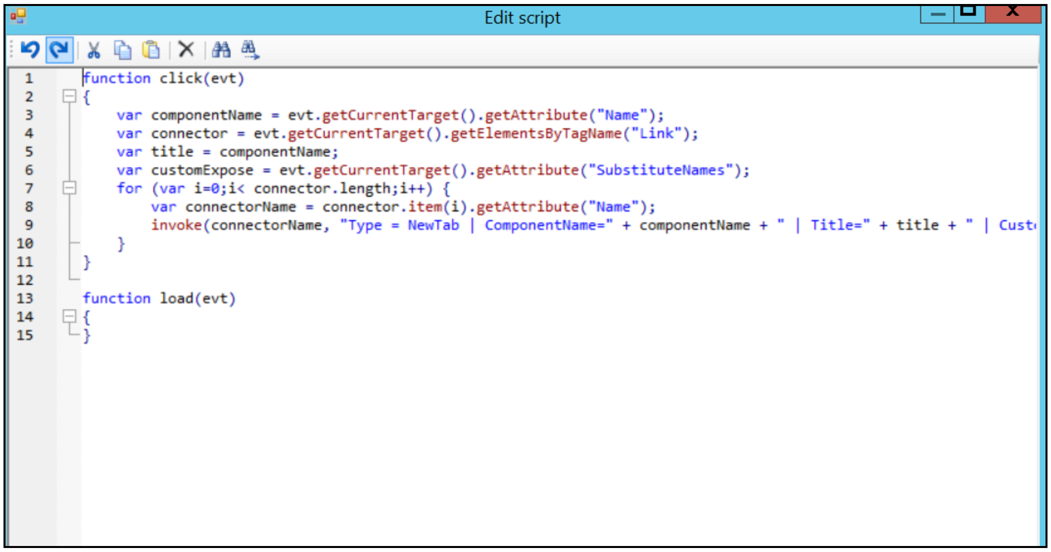
OnMouseClick
A mouse button is clicked over an element.

Test

Name	Value	Unit	Status
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device

X: 608 Y: 324 Left: 71 Top: 238 Width: 80 Height: 80 Center: 111x278

The following screen is displayed.

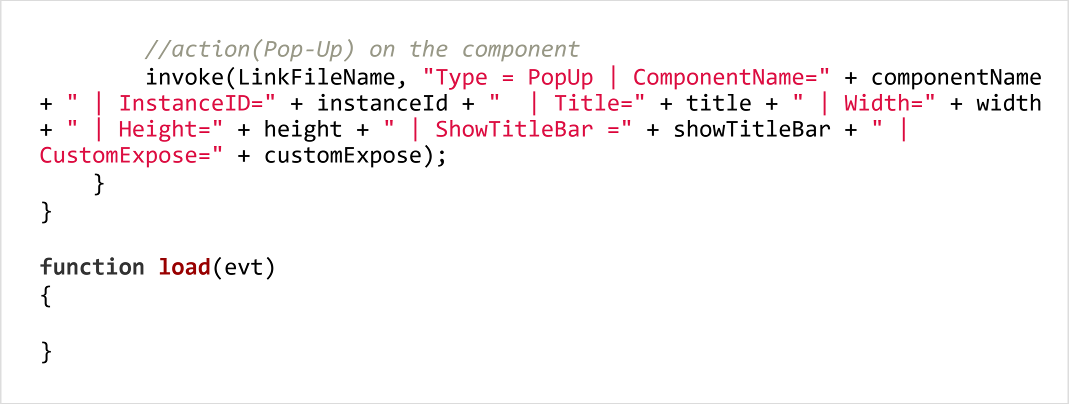


```
function click(evt)
{
    var componentName = evt.getCurrentTarget().getAttribute("Name");
    var connector = evt.getCurrentTarget().getElementsByTagName("Link");
    var title = componentName;
    var customExpose = evt.getCurrentTarget().getAttribute("SubstituteNames");
    for (var i=0;i< connector.length;i++) {
        var connectorName = connector.item(i).getAttribute("Name");
        invoke(connectorName, "Type = NewTab | ComponentName=" + componentName + " | Title=" + title + " | Cust
    }
}

function load(evt)
{
}
```

7. Edit the script as per below and close the window.

   Use the following script to configure the PopUp:

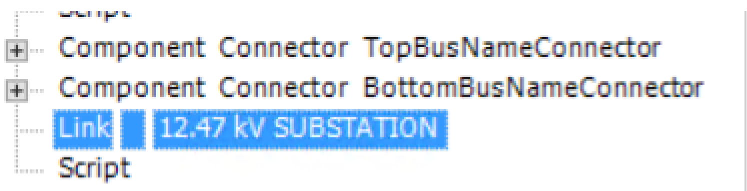
```
function click(evt)
{
    //componentName is name of the component based on the component
    // selection we will fetch the component name
    var componentName = evt.getCurrentTarget().getAttribute("Name");

    //Collecting the links from the Component
    var Link = evt.getCurrentTarget().getElementsByTagName("Link");

    //InstanceId-It is auto generating id each component pop up selection
    // it will create new instance id
    var instanceId = evt.getCurrentTarget().getAttribute("InstanceId");

    //title is component name used for showing the title
    var title = componentName;

    //customExpose-If two breakers are internally connected (means multi
equipment)
    var customExpose = evt.getCurrentTarget().getAttribute
("SubstituteNames");

    //Height & width can be configurable by the user
    var width = 370;
    var height = 370;

    //showTitleBar: Displays the Title Bar in the target pane when set to
Yes
    var showTitleBar = "Yes";

    for (var i=0;i< Link.length;i++){

        //LinkFileName : Extracting the file name from the Link
        var LinkFileName = Link.item(i).getAttribute("Name");

        //With invoke function you can configure the graphic component in
Graphics Editor
        //to open a linked target object in a target location when you
perform an
```



```
        //action(Pop-Up) on the component
        invoke(LinkFileName, "Type = PopUp | ComponentName=" + componentName
+ " | InstanceID=" + instanceId + "  | Title=" + title + " | Width=" + width
+ " | Height=" + height + " | ShowTitleBar =" + showTitleBar + " |
CustomExpose=" + customExpose);
    }
}

function load(evt)
{

}
```

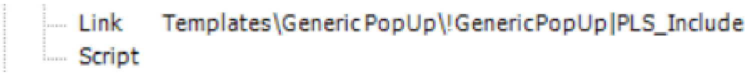
8. Go back to **Object** pane, and then click on the link to edit.

   If the TGML file is present in same project, enter the same TGML file name.



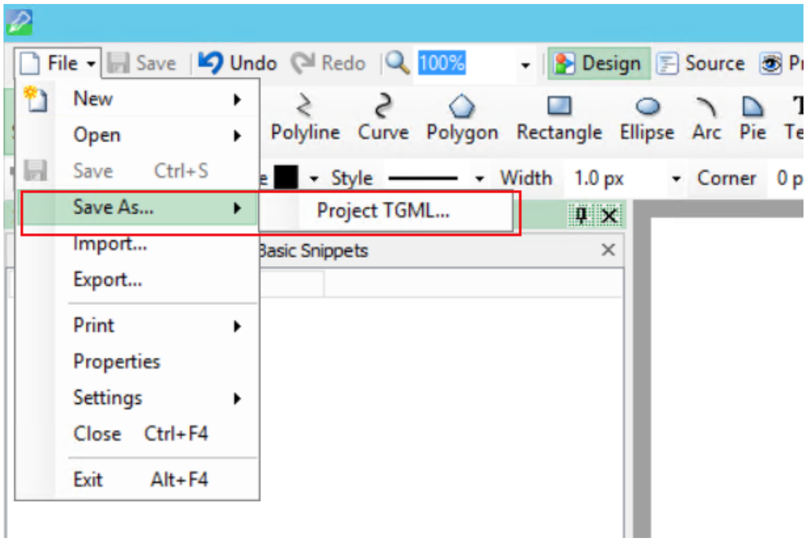
```
Component Connector TopBusNameConnector
Component Connector BottomBusNameConnector
Link      12.47 kV SUBSTATION
Script
```

   If the TGML file is present in another project, enter the TGML file with project name shown as example below.



```
Link     Templates\GenericPopUp\!GenericPopUp|PLS_Include
Script
```

9. Go to **File** > **Save As** > **Project TGML**.

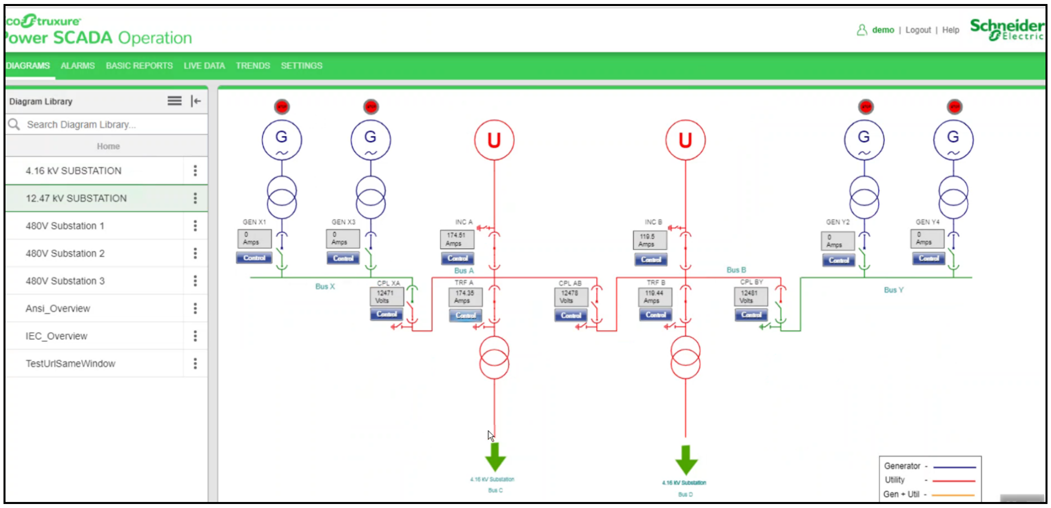


10. Enter a name, and then save the file.

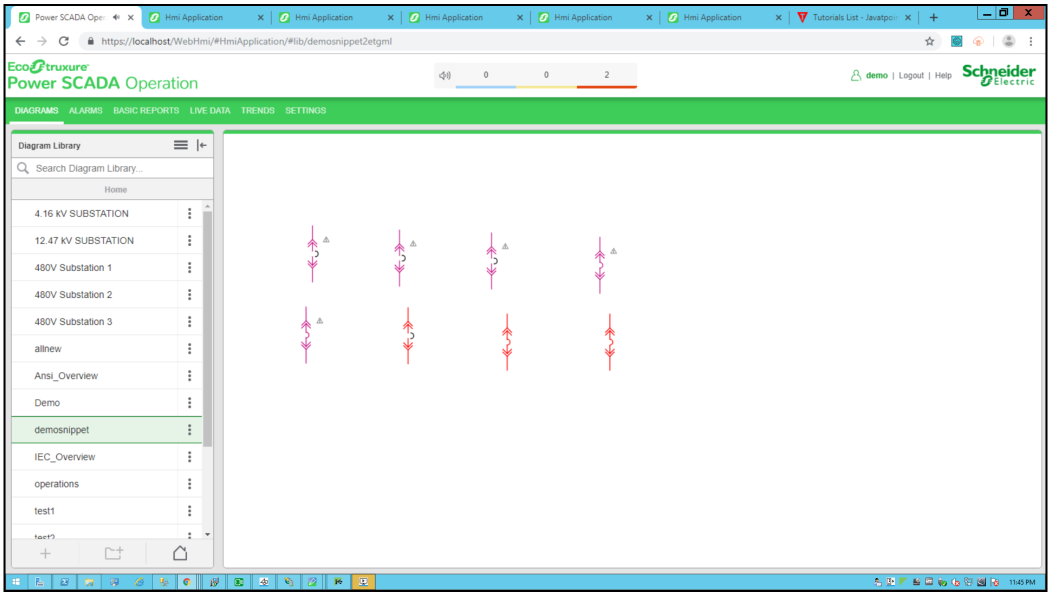
To view the snippet behavior:

1. In a web browser, log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**).

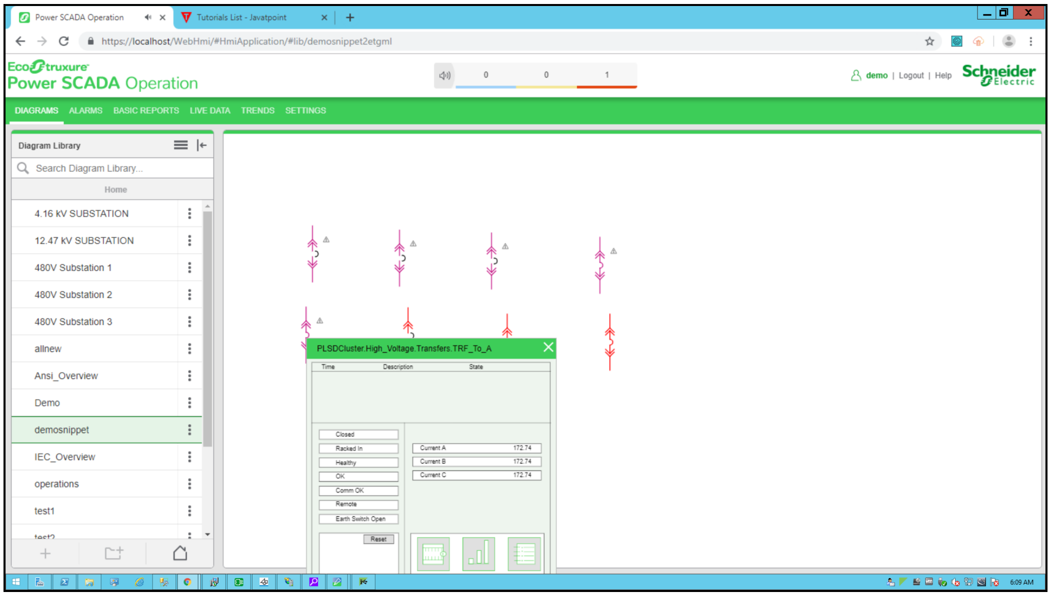
The Power SCADA Operation Web Applications Home page appears:



2. Select the new TGML file from the **Diagram Library** from the left hand panel as follows:



3. Click on the breaker to open a pop-up displaying real time readings from the component as shown below.



URL snippet example

When you click a TGML graphic that has a configured URL snippet, the URL opens in a browser window.

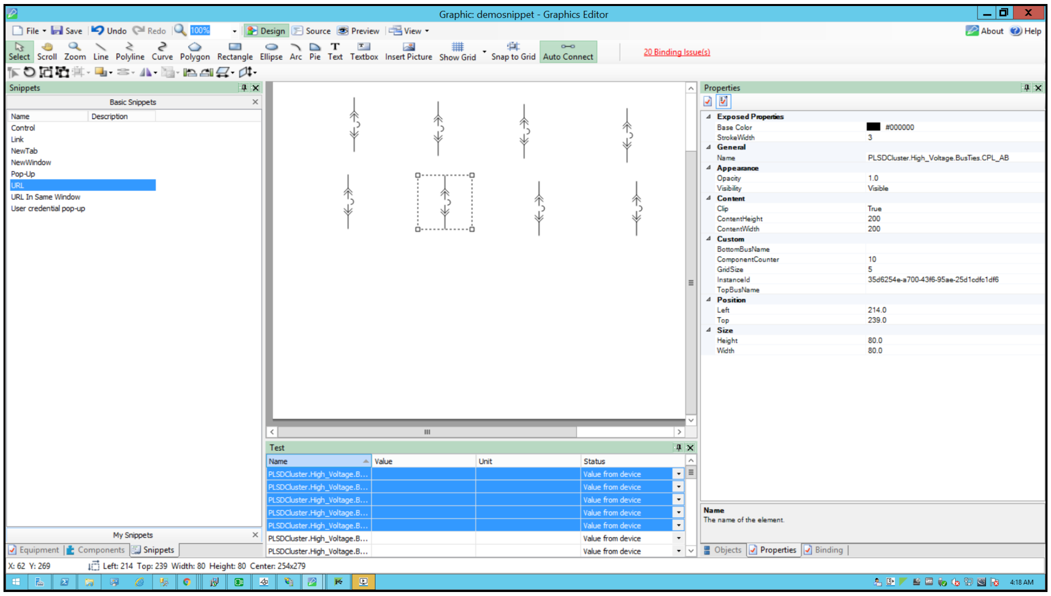
This topic uses an example to illustrate how to configure a URL snippet.

Prerequisites

This example uses a graphic file that already has a binded component or equipment in the workspace. For more information on how to prepare the TGML graphic snippet examples, see "Prerequisites" on page 504.

To configure the URL snippet:

1. In the bottom left corner, click **Snippets**, and then click **URL**.
2. Drag and drop the **URL** snippet over the selected component in the workspace, and then save it.


Graphic: demosnippet - Graphics Editor
File Save Undo Redo 100% Design Source Preview View About Help
Select Scroll Zoom Line Polyline Curve Polygon Rectangle Ellipse Arc Pie Text Textbox Insert Picture Show Grid Snap to Grid Auto Connect
20 Binding Issue(s)
Snippets
Basic Snippets
My Snippets
Equipment Components Snippets
Properties
Objects Properties Binding
X: 62 Y: 269 Left: 214 Top: 239 Width: 80 Height: 80 Center: 254x279
4:18 AM

Name	Description
Control	
Link	
NewTab	
NewWindow	
Pop-Up	
URL	
URL In Same Window	
User credential pop-up	

Name	Value	Unit	Status
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device

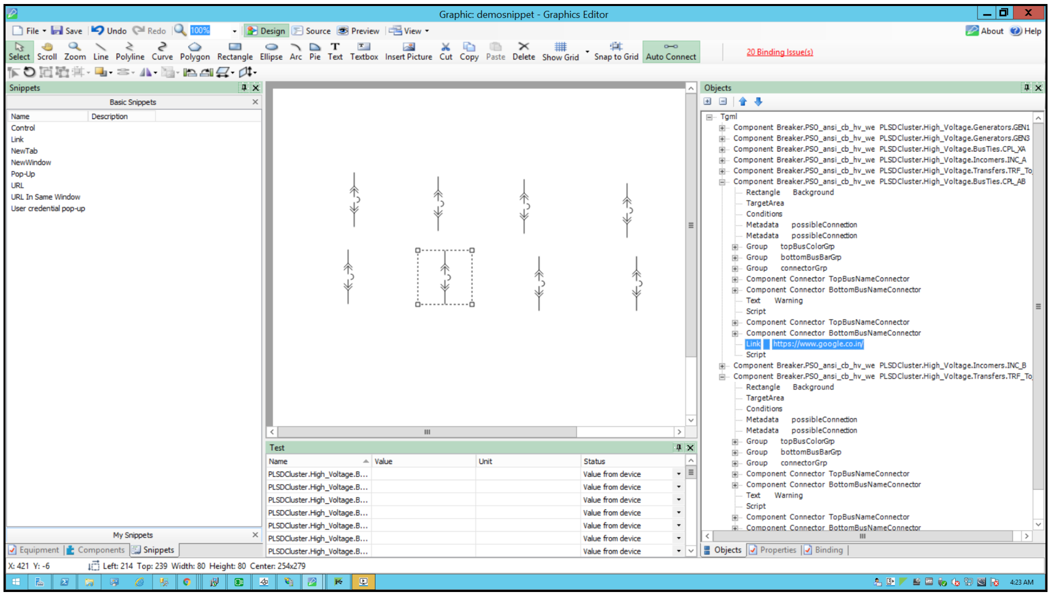
Properties	
Exposed Properties	
Base Color	■ #000000
StrokeWidth	3
General	
Name	PLSDCluster.High_Voltage.BusTies.CPL_AB
Appearance	
Opacity	1.0
Visibility	Visible
Content	
Clip	True
ContentHeight	200
ContentWidth	200
Custom	
BottomBusName	
ComponentCounter	10
GridSize	5
InstanceId	35d6254e-a700-43f6-95ae-25d1cdfc1df6
TopBusName	
Position	
Left	214.0
Top	239.0
Size	
Height	80.0
Width	80.0

Name
The name of the element.

3. In the bottom right corner, click **Objects**, and then expand the TGML > Component node.

   Two additional properties appear: **Link** and **Script**.

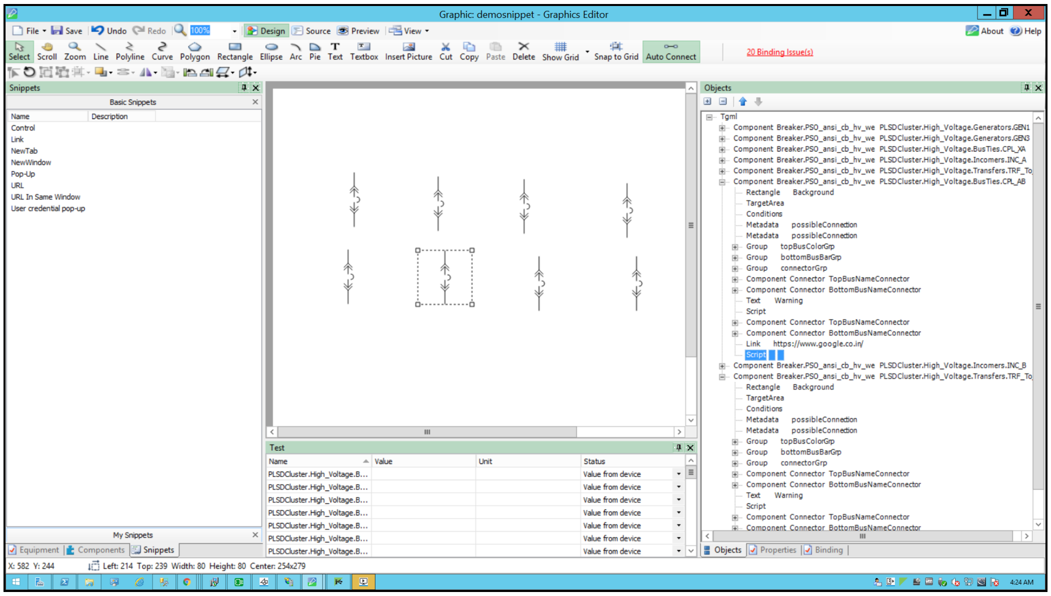
4. Update the link with the URL to be opened (for example, https://www.google.co.in/).


Graphic: demosnippet - Graphics Editor
File Save Undo Redo 100% Design Source Preview View About Help
Select Scroll Zoom Line Polyline Curve Polygon Rectangle Ellipse Arc Pie Text Textbox Insert Picture Cut Copy Paste Delete Show Grid Snap to Grid Auto Connect
20 Binding Issue(s)
Snippets
Basic Snippets
My Snippets
Equipment Components Snippets
Test
Objects
Objects Properties Binding
X: 421 Y: -6 Left: 214 Top: 239 Width: 80 Height: 80 Center: 254x279
4:23 AM

```
Tgml
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.GEN1
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.GEN3
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.BusTies.CPL_XA
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Incomers.INC_A
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Transfers.TRF_To
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.BusTies.CPL_AB
    Rectangle    Background
    TargetArea
    Conditions
    Metadata    possibleConnection
    Metadata    possibleConnection
    Group    topBusColorGrp
    Group    bottomBusBarGrp
    Group    connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Text    Warning
    Script
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Link    https://www.google.co.in/
    Script
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Incomers.INC_B
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Transfers.TRF_To
    Rectangle    Background
    TargetArea
    Conditions
    Metadata    possibleConnection
    Metadata    possibleConnection
    Group    topBusColorGrp
    Group    bottomBusBarGrp
    Group    connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Text    Warning
    Script
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
```

Name	Value	Unit	Status
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device

5. Click **Script**:



Name	Description
Control	
Link	
NewTab	
NewWindow	
Pop-Up	
URL	
URL In Same Window	
User credential pop-up	

```
Tgml
  Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.GEN1
  Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.GEN3
  Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.BusTies.CPL_XA
  Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Incomers.INC_A
  Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Transfers.TRF_To
  Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.BusTies.CPL_AB
    Rectangle    Background
    TargetArea
    Conditions
    Metadata    possibleConnection
    Metadata    possibleConnection
    Group    topBusColorGrp
    Group    bottomBusBarGrp
    Group    connectorGrp
    Component  Connector  TopBusNameConnector
    Component  Connector  BottomBusNameConnector
    Text    Warning
    Script
    Component  Connector  TopBusNameConnector
    Component  Connector  BottomBusNameConnector
    Link    https://www.google.co.in/
    Script
  Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Incomers.INC_B
  Component  Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Transfers.TRF_To
    Rectangle    Background
    TargetArea
    Conditions
    Metadata    possibleConnection
    Metadata    possibleConnection
    Group    topBusColorGrp
    Group    bottomBusBarGrp
    Group    connectorGrp
    Component  Connector  TopBusNameConnector
    Component  Connector  BottomBusNameConnector
    Text    Warning
    Script
    Component  Connector  TopBusNameConnector
    Component  Connector  BottomBusNameConnector
```

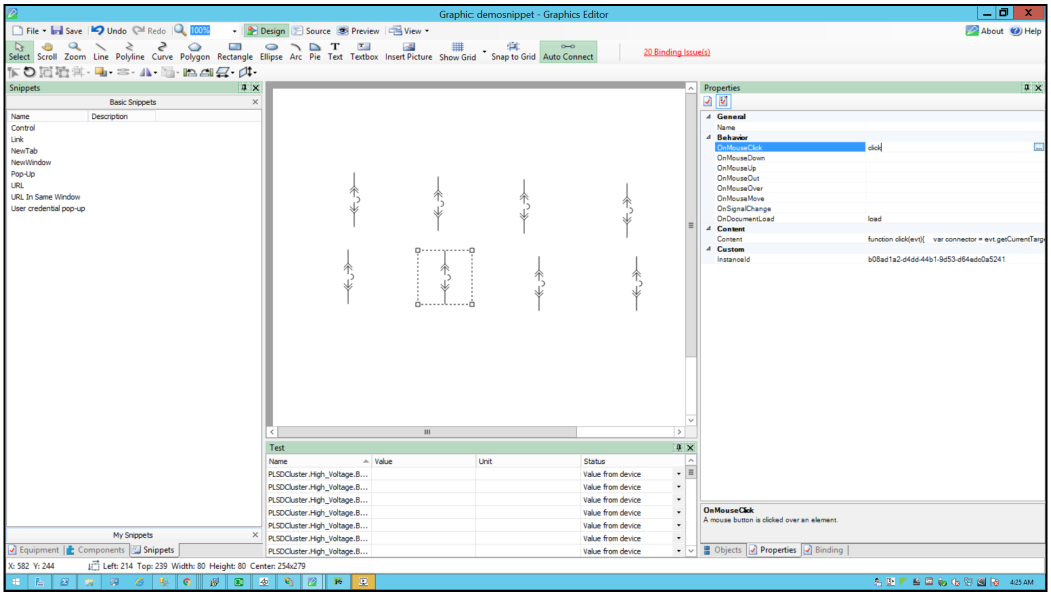
Name	Value	Unit	Status
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device

X: 582 Y: 244 Left: 214 Top: 239 Width: 80 Height: 80 Center: 254x279

6. In the bottom right corner, click **Properties**.

7. Expand **Behavior**.

8. In **OnMouseClick**, click the ellipsis button that appears:



Name	Description
Control	
Link	
NewTab	
NewWindow	
Pop-Up	
URL	
URL In Same Window	
User credential pop-up	

General	
Name	
Behavior	
OnMouseClick	click
OnMouseDown	
OnMouseUp	
OnMouseOut	
OnMouseOver	
OnMouseMove	
OnSignalChange	
OnDocumentLoad	load
Content	
Content	function click(evt){ var connector = evt.getCurrentTarg
Custom	
InstanceId	b08ad1a2-d4dd-44b1-9d53-d64edc0a5241

OnMouseClick
A mouse button is clicked over an element.

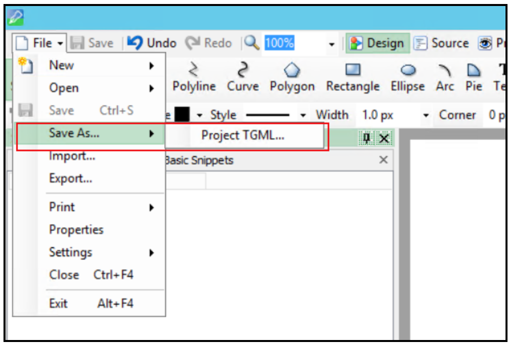
Name	Value	Unit	Status
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device

X: 582 Y: 244 Left: 214 Top: 239 Width: 80 Height: 80 Center: 254x279

9. Use the following script to set the TGML snippet's click behavior (on click, open a URL in the same window), and then close the script window:

```
function click(evt)
{
    //Collecting the links from the Component
    var Link = evt.getCurrentTarget().getElementsByTagName("Link");

    //if You Want to open the given URL In Same Window or not, here same
window is NO means it will open the Url in different window
    var sameWindow = "No";

    for (var i=0;i< Link.length;i++) {
        //LinkFileName : Extracting the file name from the Link
        var LinkFileName = Link.item(i).getAttribute("Name");
        //With invoke function you can configure the graphic component in
Graphics Editor to open a linked target object in a target location when you
perform an action(URL) on the component
        invoke(LinkFileName, "Type = Href | HrefSameWindow = "+sameWindow);
    }
}

function load(evt)
{
}
```

10. Go to **File** > **Save As** > **Project TGML**.

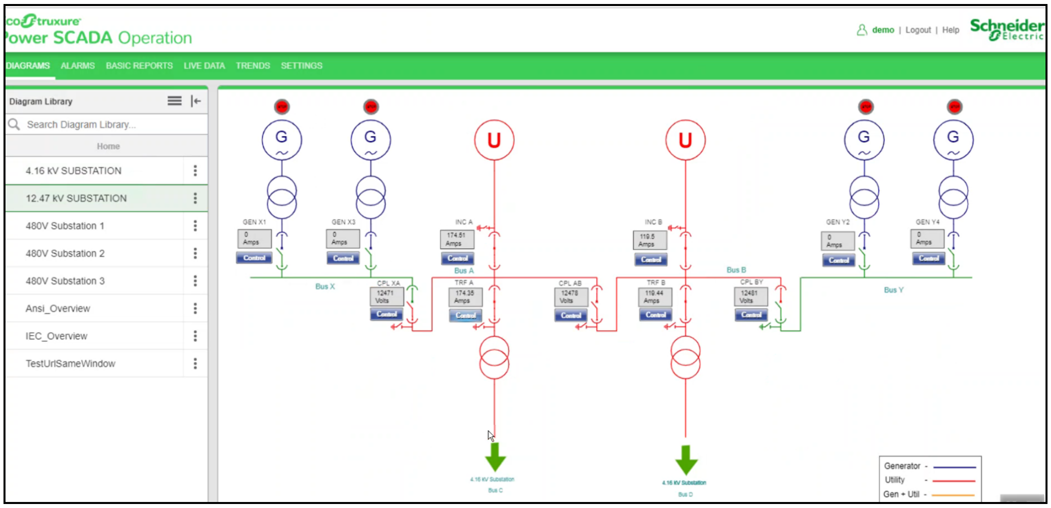


11. Enter a file name, and then click **Save**.

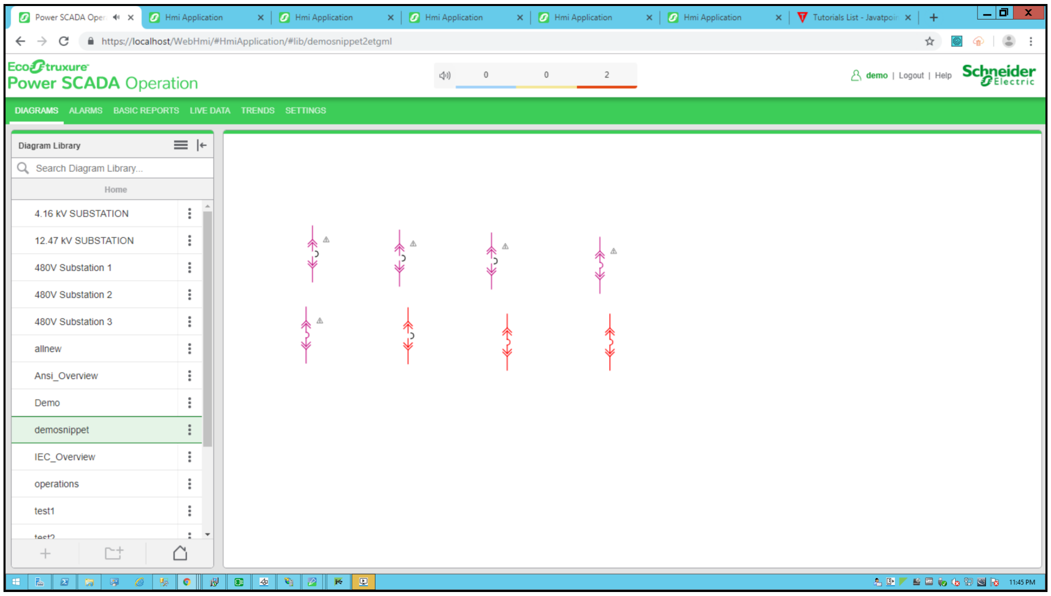
To view the snippet behavior:

1. In a web browser, log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)

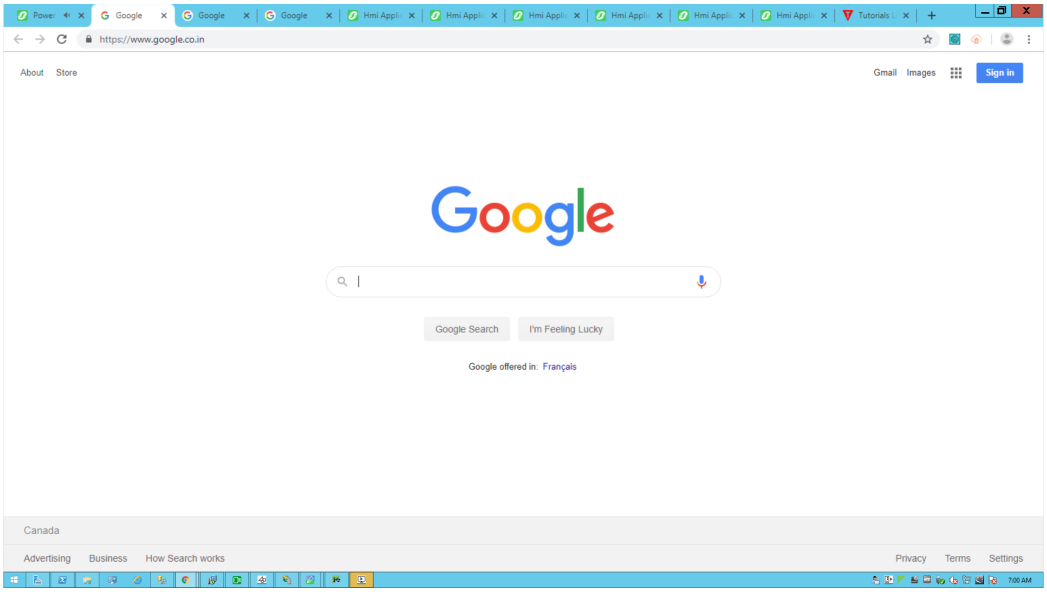
The Power SCADA Web Applications Home page appears:



2. Select the new TGML file from the left panel **Diagram Library**:



3. Click on the breaker to open the **URL** in a different window:



URL in Same Window

When you click a TGML graphic that has a configured URL In Same Window snippet, another site or web application page opens in the same window.

This topic uses an example to illustrate how to configure a URL In Same Window snippet.

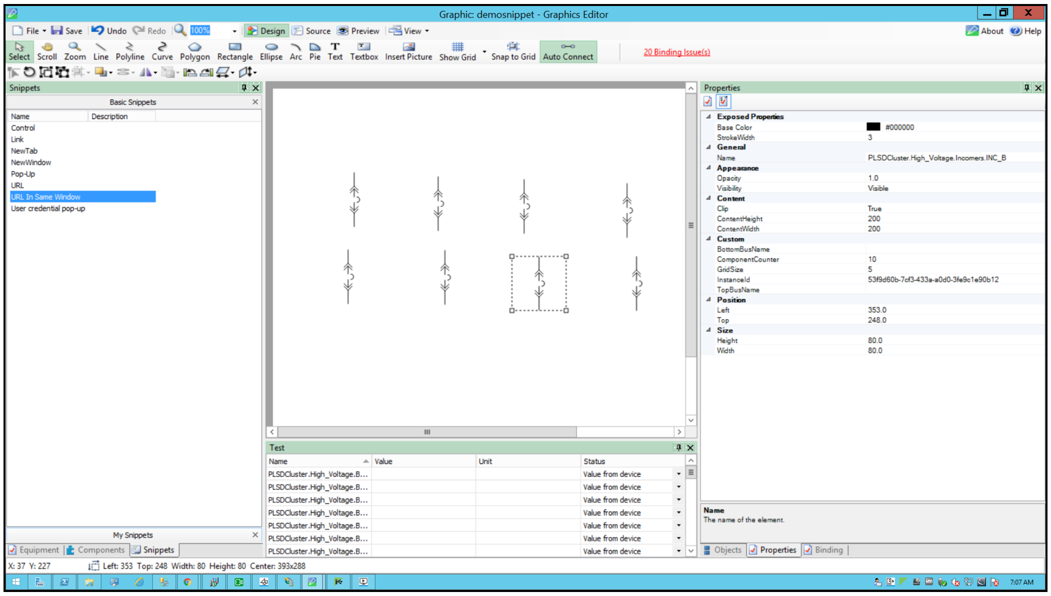
Prerequisites

This example uses a graphic file that already has a binded component or equipment in the workspace. For more information on how to prepare the TGML graphic snippet examples, see "Prerequisites" on page 504.

To configure the URL In Same Window snippet:

1. In the bottom left corner, click **Snippets**, and then click **URL In Same Window**.

   – Drag and drop the **URL In Same Window** snippet over the selected component in the workspace, and then save it.



Graphic: demosnippet - Graphics Editor
File Save Undo Redo 100% Design Source Preview View
About Help
Select Scroll Zoom Line Polyline Curve Polygon Rectangle Ellipse Arc Pie Text Textbox Insert Picture Show Grid Snap to Grid Auto Connect
20 Binding Issue(s)
Snippets
Basic Snippets
Name Description
Control
Link
NewTab
NewWindow
Pop-Up
URL
URL In Same Window
User credential pop-up
My Snippets
Equipment Components Snippets
Test
Objects Properties Binding
X: 37 Y: 227 Left: 353 Top: 248 Width: 80 Height: 80 Center: 393x288

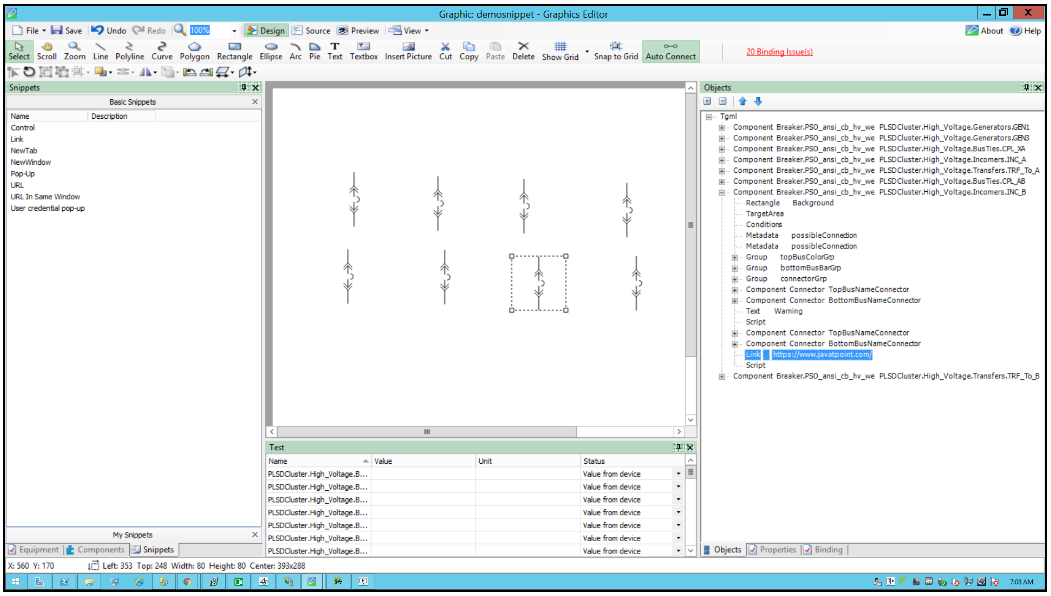
Properties

Exposed Properties	
Base Color	#000000
StrokeWidth	3
General	
Name	PLSDCluster.High_Voltage.Incomers.INC_B
Appearance	
Opacity	1.0
Visibility	Visible
Content	
Clip	True
ContentHeight	200
ContentWidth	200
Custom	
BottomBusName	
ComponentCounter	10
GridSize	5
InstanceId	53f9d60b-7cf3-433a-a0d0-3fe9c1e90b12
TopBusName	
Position	
Left	353.0
Top	248.0
Size	
Height	80.0
Width	80.0

Name
The name of the element.

Name	Value	Unit	Status
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device

2. In the bottom right corner, click **Objects**, and then expand the TGML > Component node.
   - Two additional properties appear: **Link** and **Script**.
   - Update the link with the URL to be opened in the same window (for example, https://www.javatpoint.com/).



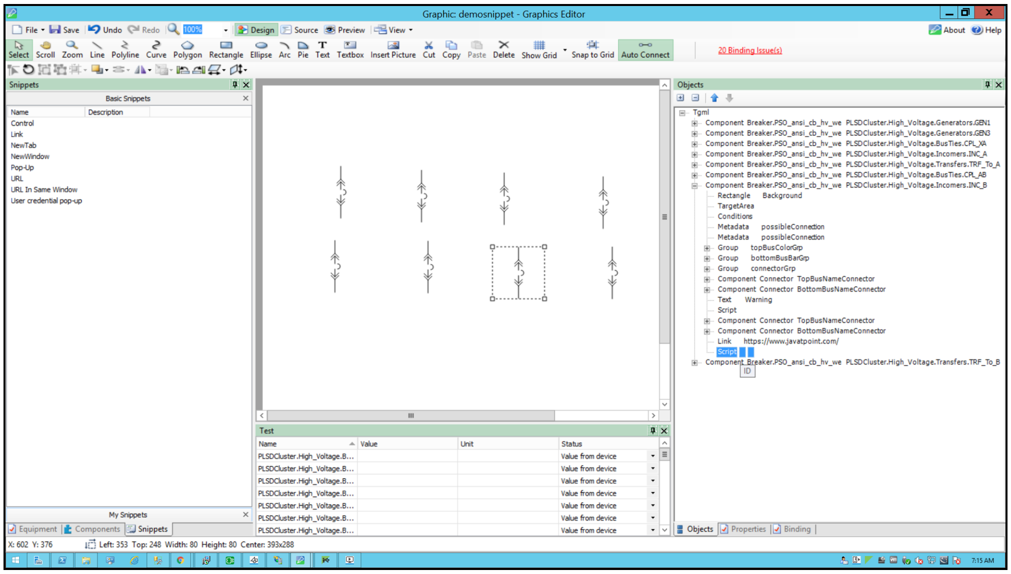
Graphic: demosnippet - Graphics Editor
File Save Undo Redo 100% Design Source Preview View
About Help
Select Scroll Zoom Line Polyline Curve Polygon Rectangle Ellipse Arc Pie Text Textbox Insert Picture Cut Copy Paste Delete Show Grid Snap to Grid Auto Connect
20 Binding Issue(s)
Snippets
Basic Snippets
Name Description
Control
Link
NewTab
NewWindow
Pop-Up
URL
URL In Same Window
User credential pop-up
My Snippets
Equipment Components Snippets
Test
Objects Properties Binding
X: 560 Y: 170 Left: 353 Top: 248 Width: 80 Height: 80 Center: 393x288

Objects

```
Tgml
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Generators.GEN1
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Generators.GEN3
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.BusTies.CPL_XA
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Incomers.INC_A
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Transfers.TRF_To_A
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.BusTies.CPL_AB
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Incomers.INC_B
    Rectangle Background
    TargetArea
    Conditions
    Metadata possibleConnection
    Metadata possibleConnection
    Group topBusColorGrp
    Group bottomBusBarGrp
    Group connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Text Warning
    Script
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Link https://www.javatpoint.com/
    Script
  Component Breaker.PSO_ansi_cb_hv_we PLSDCluster.High_Voltage.Transfers.TRF_To_B
```

Name	Value	Unit	Status
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device

3. Click **Script**:



Name	Description
Control	
Link	
NewTab	
NewWindow	
Pop-Up	
URL	
URL In Same Window	
User credential pop-up	

```
Tgml
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.GEN1
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Generators.GEN3
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.BusTies.CPL_XA
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Incomers.INC_A
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Transfers.TRF_To_A
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.BusTies.CPL_AB
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Incomers.INC_B
    Rectangle    Background
    TargetArea
    Conditions
    Metadata    possibleConnection
    Metadata    possibleConnection
    Group    topBusColorGrp
    Group    bottomBusBarGrp
    Group    connectorGrp
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Text    Warning
    Script
    Component Connector TopBusNameConnector
    Component Connector BottomBusNameConnector
    Link    https://www.javatpoint.com/
    Script
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Transfers.TRF_To_B
```

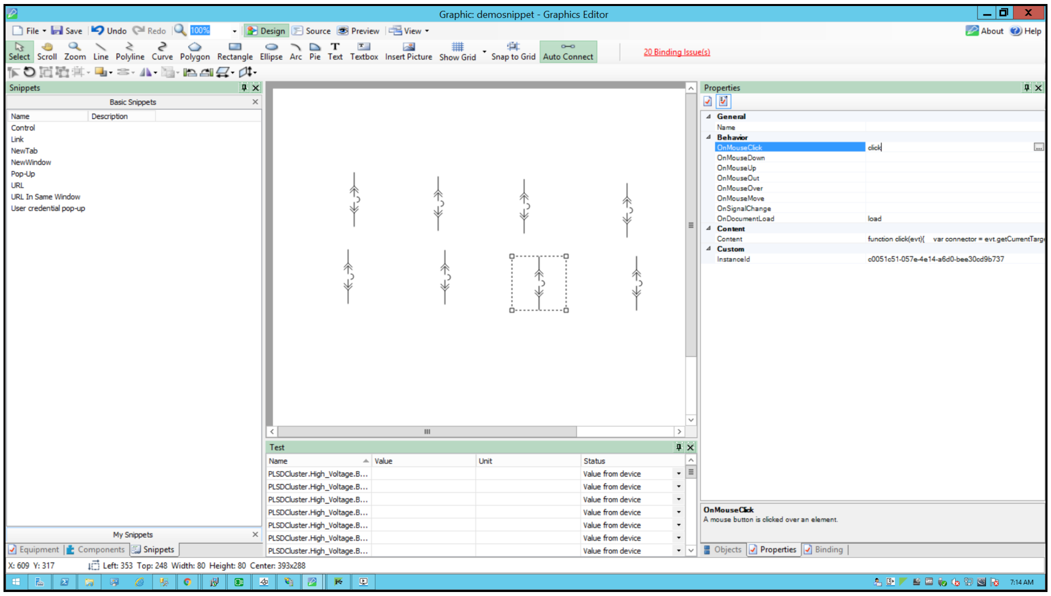
Name	Value	Unit	Status
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device

X: 602 Y: 376 Left: 353 Top: 248 Width: 80 Height: 80 Center: 393x288

4. In the bottom right corner, click **Properties**.

5. Expand **Behavior**.

6. In **OnMouseClick**, click the ellipsis button that appears:



Name	Description
Control	
Link	
NewTab	
NewWindow	
Pop-Up	
URL	
URL In Same Window	
User credential pop-up	

Name	Value	Unit	Status
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device
PLSDCluster.High_Voltage.B...			Value from device

General	
Name	
Behavior	
OnMouseClick	click
OnMouseDown	
OnMouseUp	
OnMouseOut	
OnMouseOver	
OnMouseMove	
OnSignalChange	
OnDocumentLoad	load
Content	
Content	function click(evt){ var connector = evt.getCurrentTarg
Custom	
InstanceId	c0051c51-057e-4e14-a6d0-bee30cd9b737

OnMouseClick
A mouse button is clicked over an element.

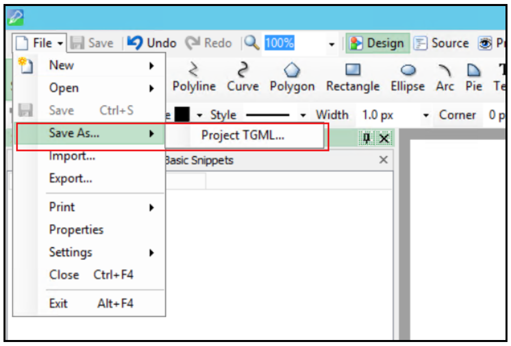
X: 609 Y: 317 Left: 353 Top: 248 Width: 80 Height: 80 Center: 393x288

7. Use the following script to set the TGML snippet's click behavior (on click, open a URL in the same window), and then close the script window:

function click(evt)

```
function click(evt)
{
//Collecting the links from the Component
var Link = evt.getCurrentTarget().getElementsByTagName("Link");

//if You Want to open the given  Url in same window or not ,here same window
is Yes meaning it will open the Url in same window
var sameWindow = "Yes";

    for (var i=0;i< Link.length;i++) {

    //LinkFileName : Extracting the file name from the Link
    var LinkFileName = Link.item(i).getAttribute("Name");

    //With invoke function you can configure the graphic component in
    //Graphics Editor to open a linked target object in a target location
    //when you perform a action(URL In Same Window) on the component
    invoke(LinkFileName, "Type = Href | HrefSameWindow = "+sameWindow);
    }
}

function load(evt)
{
}
```

8. Go to **File** > **Save As** > **Project TGML**.

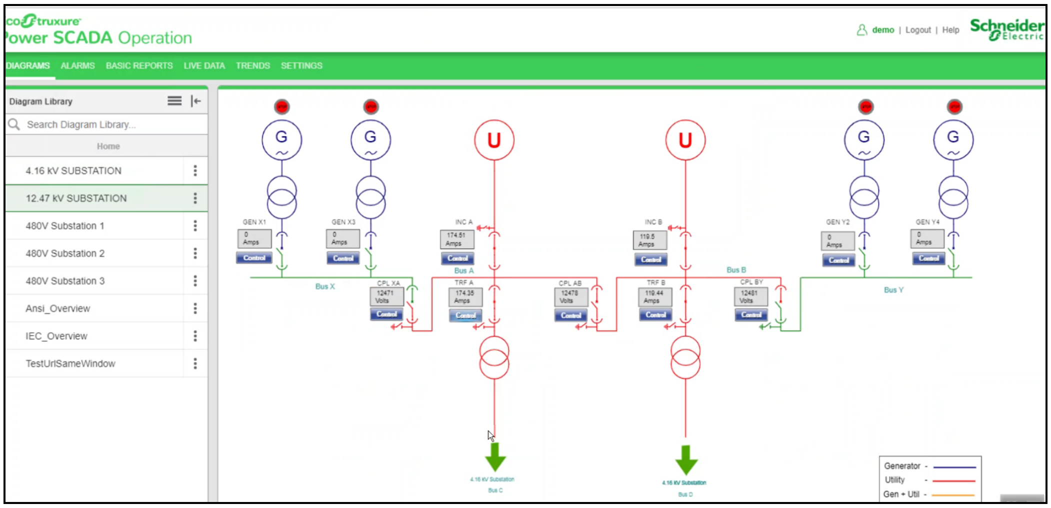


9. Type the file name in the **File name** field, and then click **Save**.

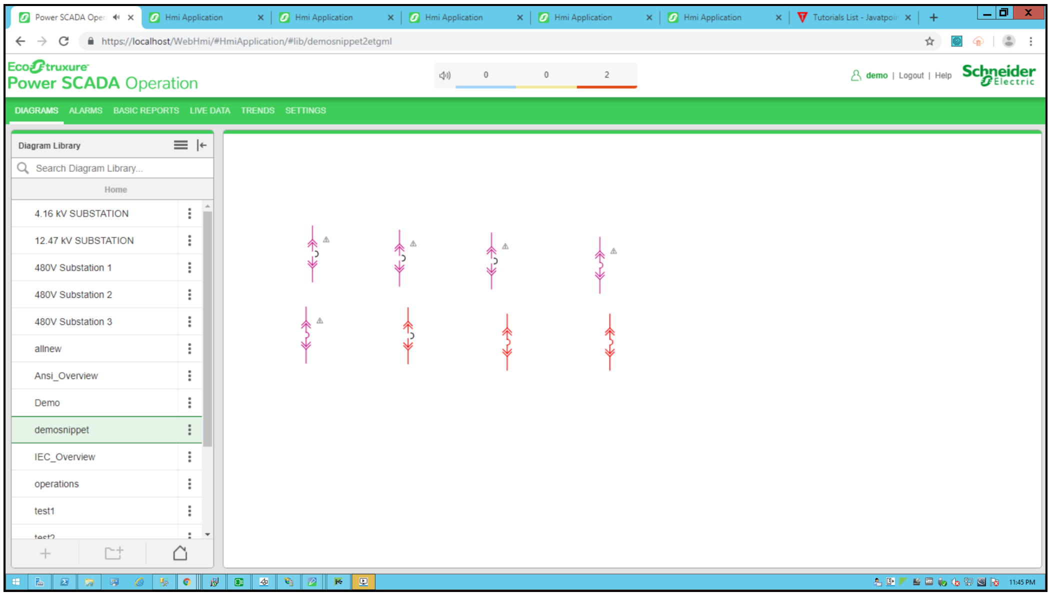
To view the snippet behavior:

1. In a web browser, log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)

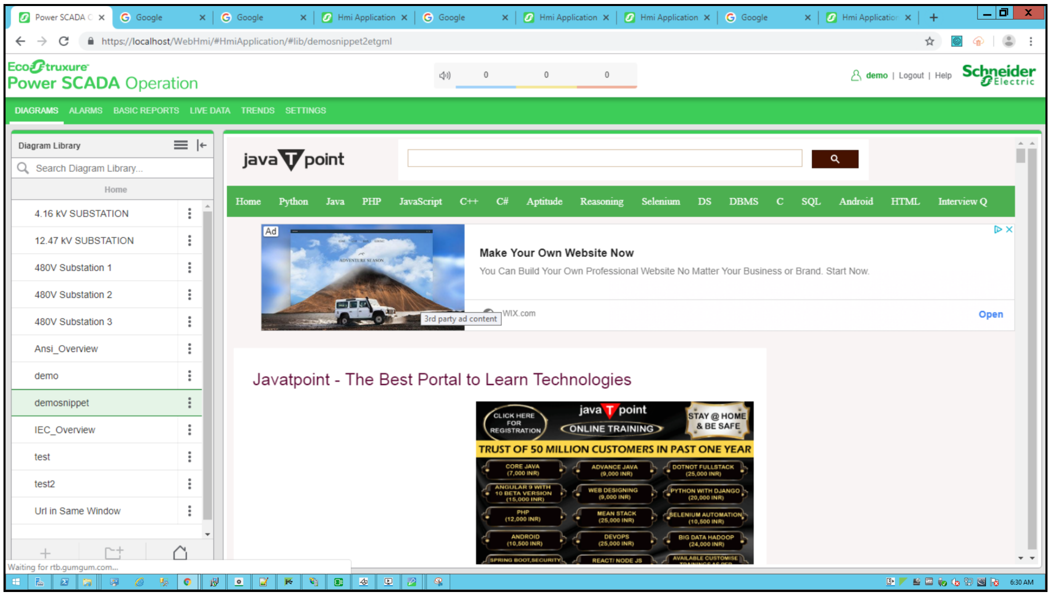
The Power SCADA Web Applications Home page appears:



2. Select the new TGML file from the left panel **Diagram Library**:



3. Click on the breaker to open the URL in the same window:


https://localhost/WebHmi/#HmiApplication/#lib/demosnippet2etgml
EcoStruxure Power SCADA Operation
demo | Logout | Help
Schneider Electric
DIAGRAMS ALARMS BASIC REPORTS LIVE DATA TRENDS SETTINGS
Diagram Library
Search Diagram Library...
Home
4.16 kV SUBSTATION
12.47 kV SUBSTATION
480V Substation 1
480V Substation 2
480V Substation 3
Ansi_Overview
demo
demosnippet
IEC_Overview
test
test2
Url in Same Window
javaTpoint
Home Python Java PHP JavaScript C++ C# Aptitude Reasoning Selenium DS DBMS C SQL Android HTML Interview Q
Make Your Own Website Now
Javatpoint - The Best Portal to Learn Technologies
Waiting for rtb.gumgum.com...

Advanced Tag Debugger

The Advanced Tag Debugger is a TGML graphic component that lets you diagnose and troubleshoot devices from the PSO Web Applications. You can use it to check device quality, status, read values from, and write values into the registers of configured devices.

⚠ 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, or equipment damage.

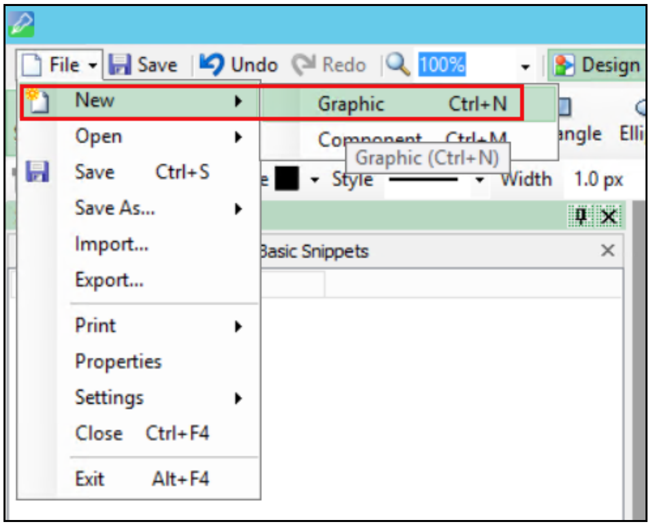
Before you use the Advanced Tag Debugger, add the component to a diagram, and then configure it. The topics in this section outline how to configure and use the Advanced Tag Debugger, and includes the following topics:

- "Configuring the Advanced Tag Debugger" on page 553
- "Opening the Advanced Tag Debugger" on page 555
- "Using the Advanced Tag Debugger" on page 557

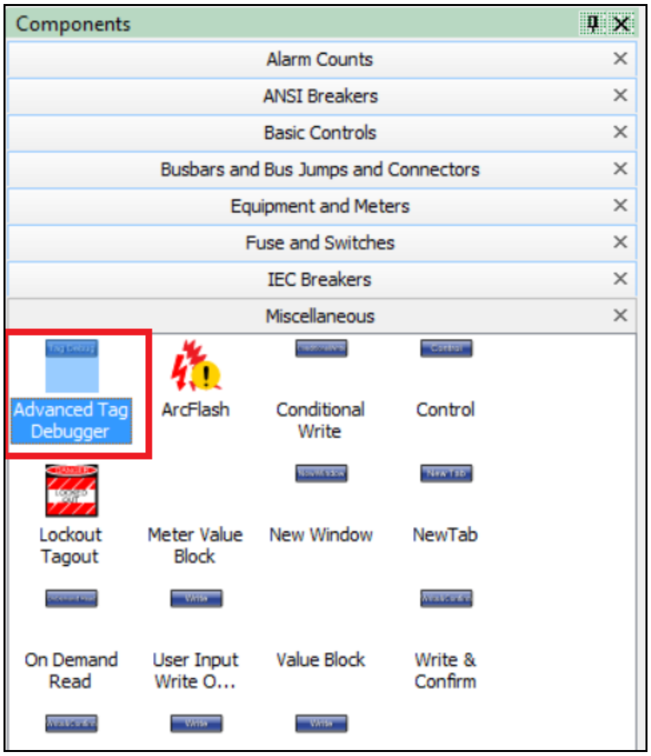
Configuring the Advanced Tag Debugger

To configure the Advanced Tag Debugger:

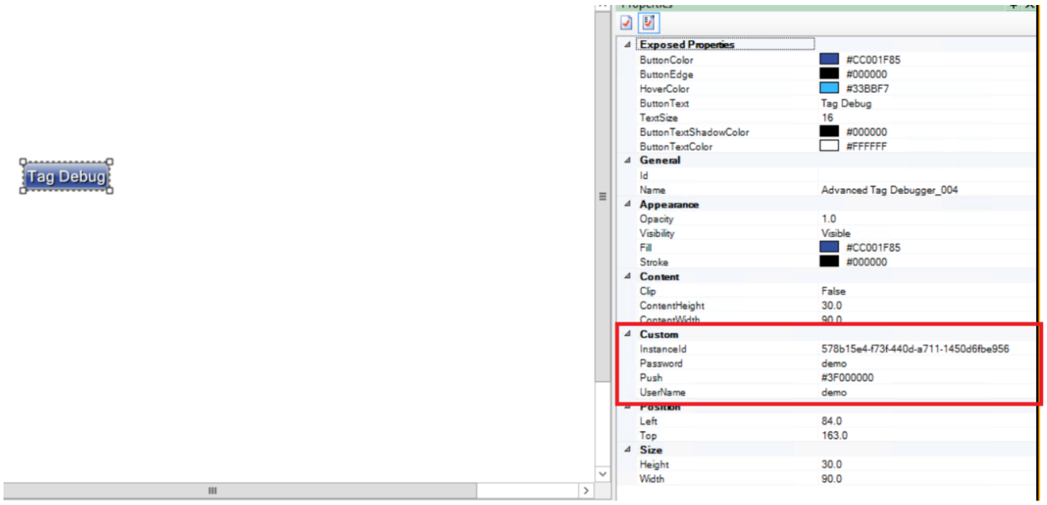
1. In the Graphics Editor, click **File > New > Graphic**.



2. In the bottom left corner, click **Components**, expand **Miscellaneous**, and then drag and drop **Advanced Tag Debugger** to the workspace.

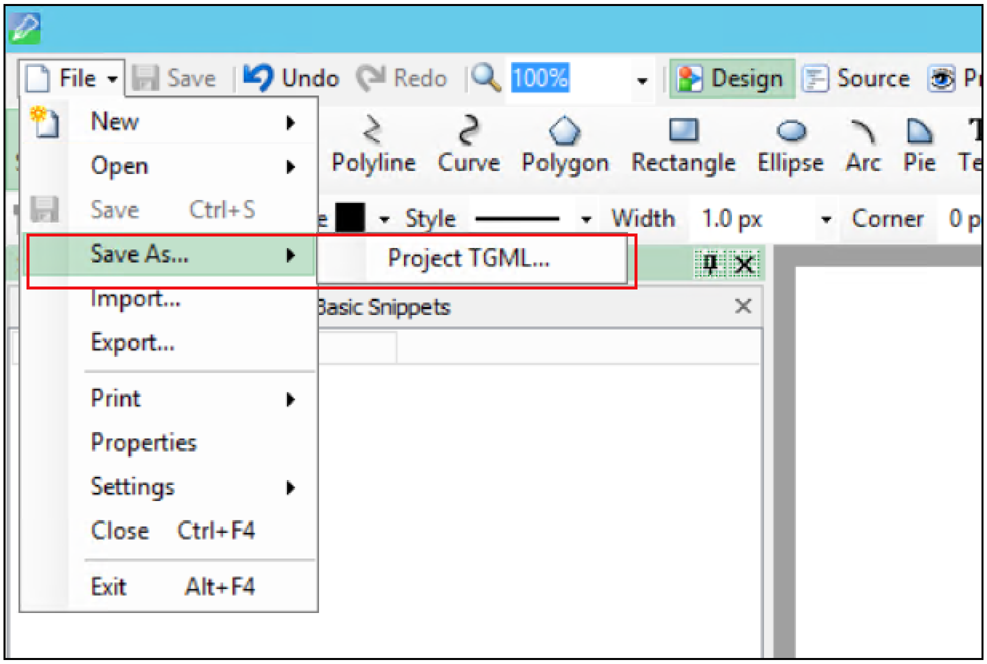


3. In the bottom right corner, click **Properties**, and then in the **Custom** section enter the User-Name and Password. For example:



Exposed Properties	
ButtonColor	#CC001F85
ButtonEdge	#000000
HoverColor	#33BBF7
ButtonText	Tag Debug
TextSize	16
ButtonTextShadowColor	#000000
ButtonTextColor	#FFFFFF
General	
Id	
Name	Advanced Tag Debugger_004
Appearance	
Opacity	1.0
Visibility	Visible
Fill	#CC001F85
Stroke	#000000
Content	
Clip	False
ContentHeight	30.0
ContentWidth	90.0
Custom	
InstanceId	578b15e4-f73f-440d-a711-1450d6fbe956
Password	demo
Push	#3F000000
UserName	demo
Position	
Left	84.0
Top	163.0
Size	
Height	30.0
Width	90.0

4. Go to **File** > **Save As** > **Project TGML**.



5. Enter the file name in the **File name** field.

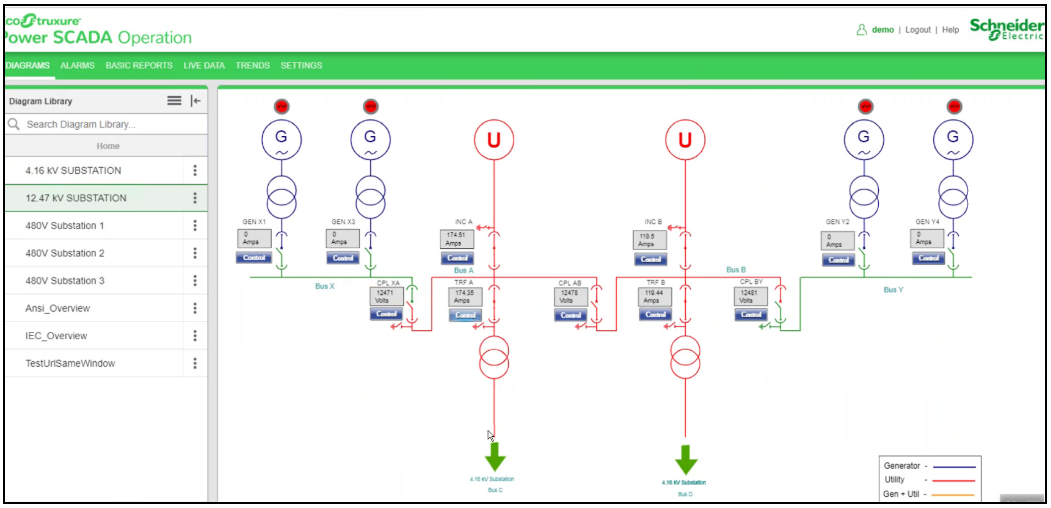
6. Click **Save**.

Opening the Advanced Tag Debugger

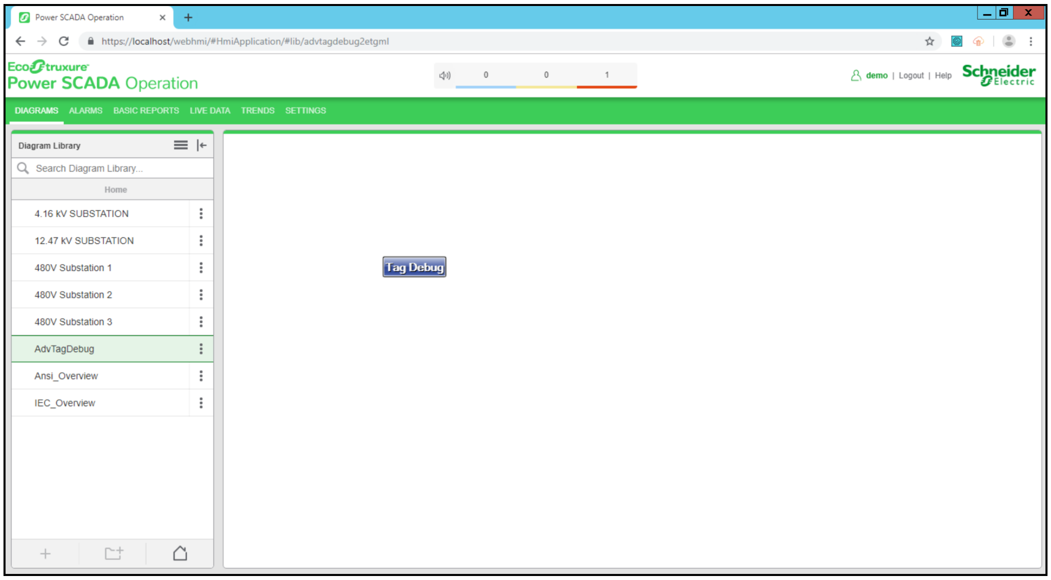
To open the Advanced Tag Debugger:

1. In a web browser, log in to the PSO Web Applications (https://localhost/webhmi or https://ipaddress/webhmi)

   The Power SCADA Web Applications Home page appears:

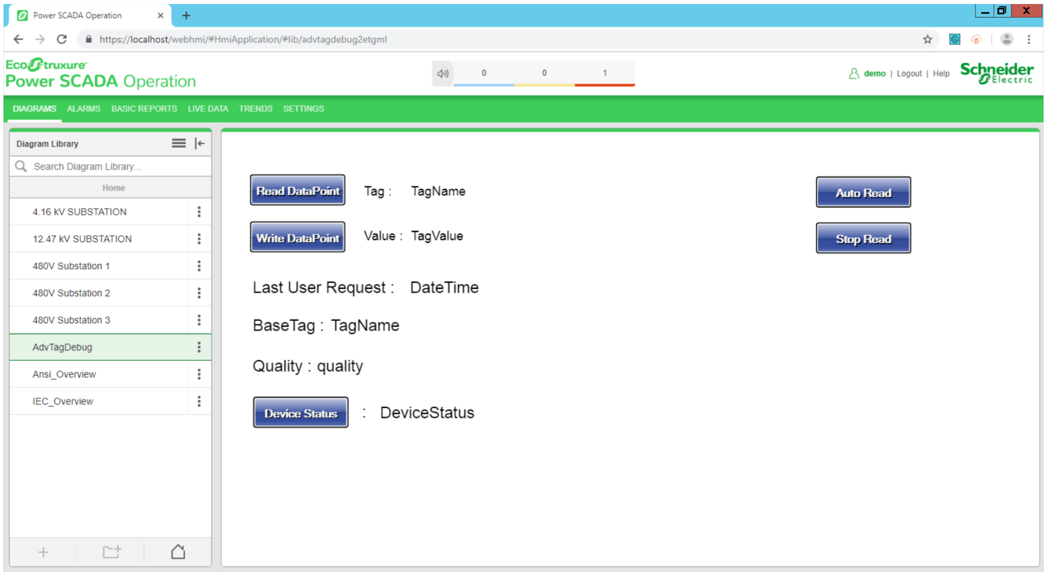


2. Select the new TGML file from the **Diagram Library**:



3. Click **Tag Debug**.

The Advanced Tag Debugger is displayed:



Tag : TagName

Value : TagValue

Last User Request : DateTime

BaseTag : TagName

Quality : quality

Device Status : DeviceStatus

Using the Advanced Tag Debugger

Use the Advanced Tag Debugger to read, and to continuously read the data point of a tag, check device quality and status, and write data point values to a tag.

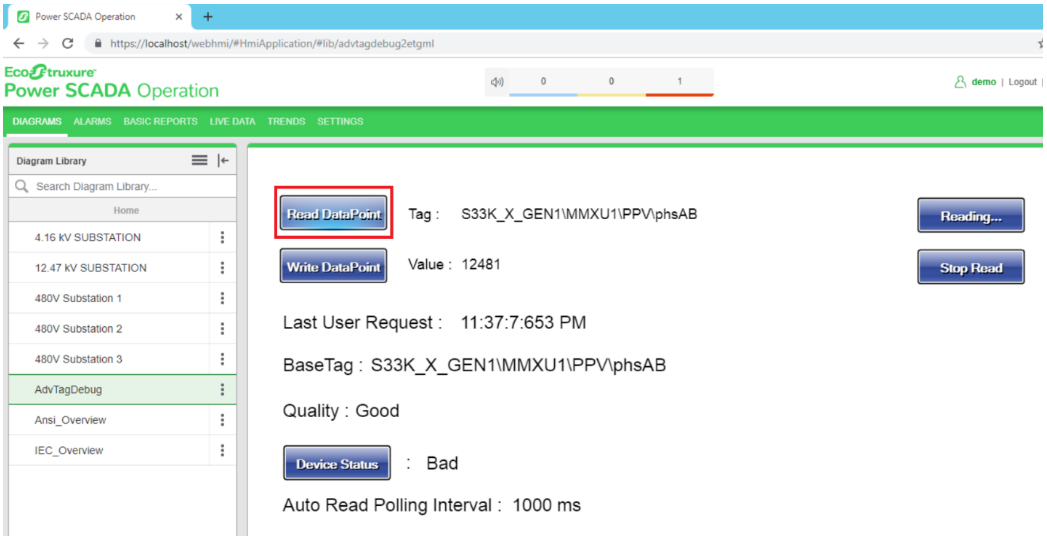
Prerequisites:

Configure and save a diagram that includes the Advanced Tag Debugger component.

Reading a data point

To read a data point:

1. In the Advanced Tag Debugger, click **Read DataPoint**.



Tag : S33K_X_GEN1\MMXU1\PPV\phsAB

Value : 12481

Last User Request : 11:37:7:653 PM

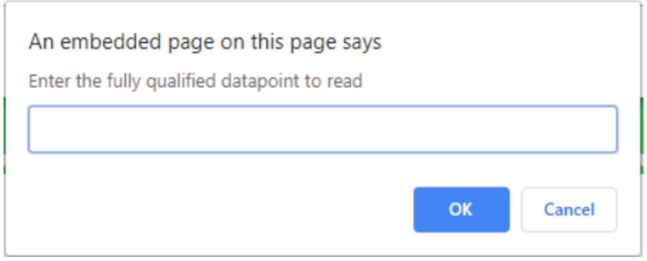
BaseTag : S33K_X_GEN1\MMXU1\PPV\phsAB

Quality : Good

Device Status : Bad

Auto Read Polling Interval : 1000 ms

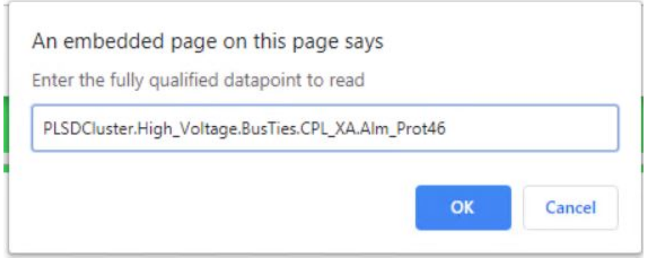
The following dialog appears:



An embedded page on this page says

Enter the fully qualified datapoint to read

2. Enter the specific item name in specific format to read. For example: PLSDCluster.High_Voltage.BusTies.CPL_XA.Alm_Prot46



An embedded page on this page says

Enter the fully qualified datapoint to read

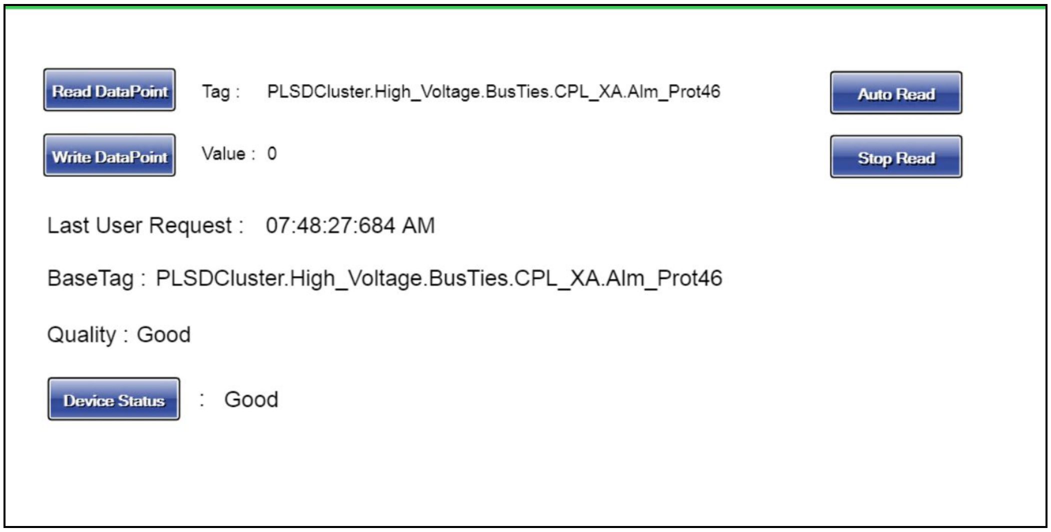
PLSDCluster.High_Voltage.BusTies.CPL_XA.Alm_Prot46

3. Click **OK**.

   The following data point information is displayed:

   - **Value:** Displays the real-time data point value
   - **Last User Request:** Displays the timestamp of the read request
   - **Base Tag:** Displays the Tag Name
   - **Quality:** Displays the quality of the tag (1 = Good, 2 = NA, 3 = Bad)

   For example:



Tag : PLSDCluster.High_Voltage.BusTies.CPL_XA.Alm_Prot46

Value : 0

Last User Request : 07:48:27:684 AM

BaseTag : PLSDCluster.High_Voltage.BusTies.CPL_XA.Alm_Prot46

Quality : Good

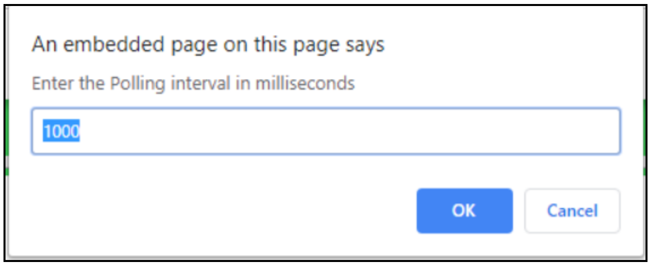
Device Status : Good

Continuously reading data points

To continuously read the data point:

1. Click **Auto Read**.

   The following dialog appears:

   An embedded page on this page says

   Enter the Polling interval in milliseconds

   1000

2. (Optional) Edit the default polling interval.

   NOTE: Editing the polling interval may delay the response time.

3. Click **OK**.

   The value is displays and is automatically refreshed at the polling rate.

4. Click **Stop Read** to stop the continuous read request polling.

Device Status

1. Use **Device Status** to check the device status online.

When you click **Device Status**, Power SCADA Operation checks the health of the device. If the response is 1, then the device status display is Good, otherwise the device status display is Bad.

Writing data points

Use **Write DataPoint** to write a value into specific parameter which has write permission.

⚠ WARNING

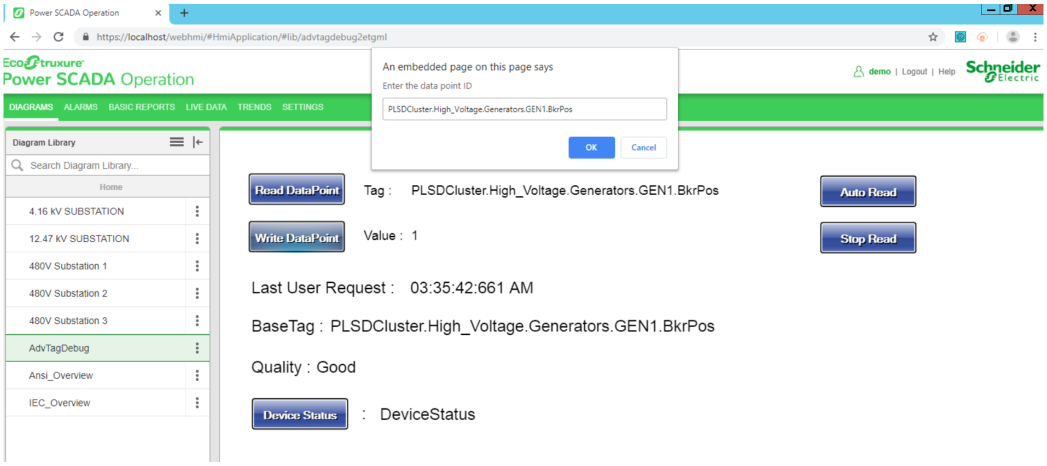
UNINTENDED EQUIPMENT OPERATION

- 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, or equipment damage.

To write a data point:

1. Click **Write DataPoint**. The following dialog appears:



An embedded page on this page says

Enter the data point ID

PLSDCluster.High_Voltage.Generators.GEN1.BkrPos

Tag : PLSDCluster.High_Voltage.Generators.GEN1.BkrPos

Value : 1

Last User Request : 03:35:42:661 AM

BaseTag : PLSDCluster.High_Voltage.Generators.GEN1.BkrPos

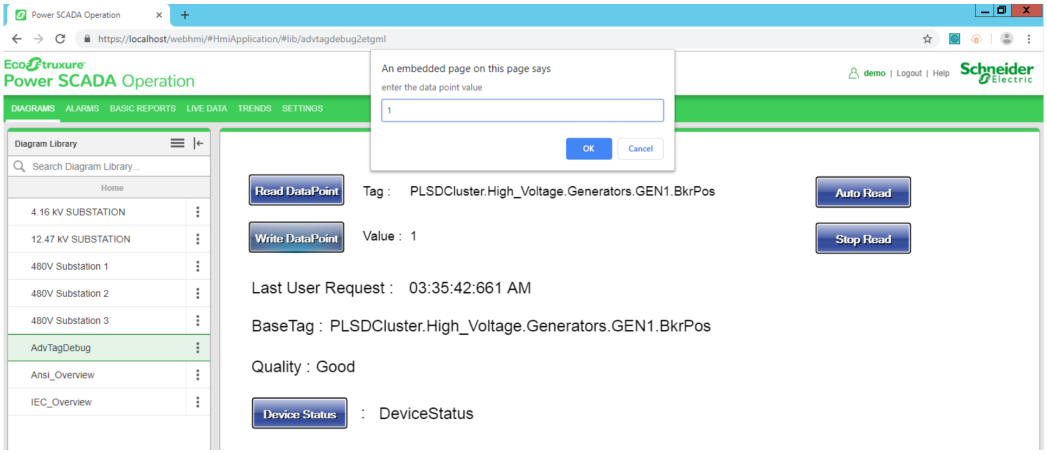
Quality : Good

: DeviceStatus

2. Enter the fully qualified item name, and then click **OK**.

   NOTE: Use the fully qualified item name Tag Name is not supported. The fully qualified item name format is **Cluster Name.Equipment.Item Name**.

3. Enter the values to be written into the data point. For example:



An embedded page on this page says

enter the data point value

1

Tag : PLSDCluster.High_Voltage.Generators.GEN1.BkrPos

Value : 1

Last User Request : 03:35:42:661 AM

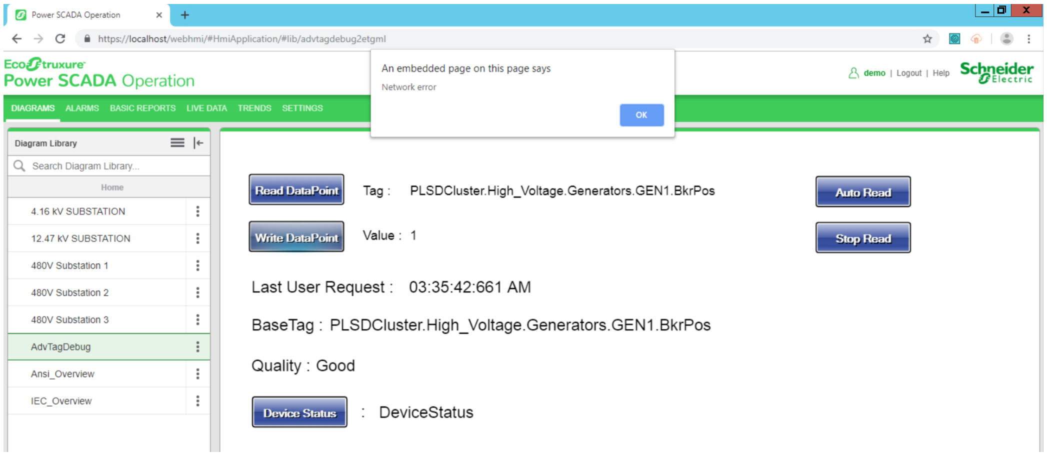
BaseTag : PLSDCluster.High_Voltage.Generators.GEN1.BkrPos

Quality : Good

: DeviceStatus

4. Click **OK** to perform write operation.

   If there are any issue entering the data point value, an error message appears. For example:



An embedded page on this page says

Network error

Tag : PLSDCluster.High_Voltage.Generators.GEN1.BkrPos

Value : 1

Last User Request : 03:35:42:661 AM

BaseTag : PLSDCluster.High_Voltage.Generators.GEN1.BkrPos

Quality : Good

: DeviceStatus

Advanced one-line diagrams

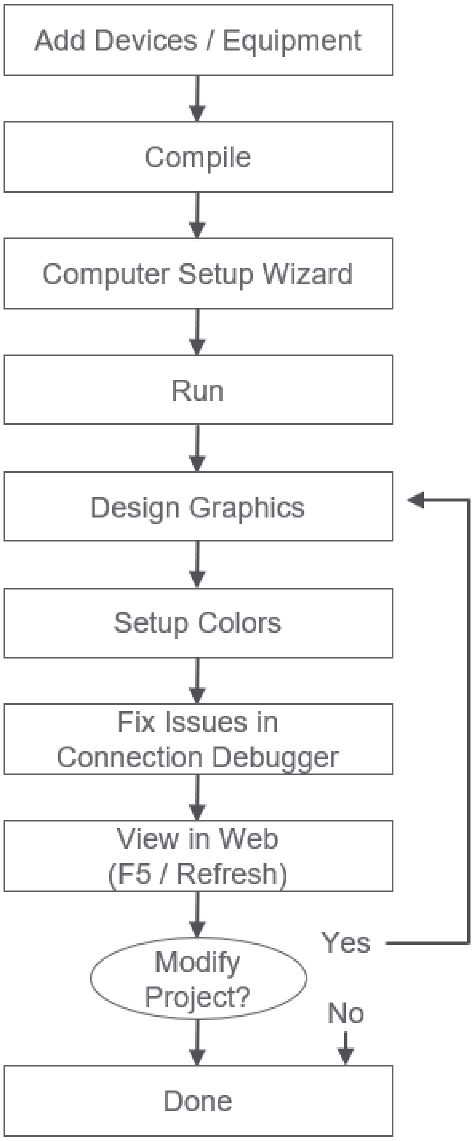
Animated advanced one-line diagrams provide built-in support for power flow diagrams. One-line colors animate based on the source(s) that feed the circuit.

For more information, see the following sections:

- Advanced one-line flow chart
- Creating a one-line on a graphics page
- Enable lockout/tagout
- Assigning one-line colors

Advanced 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 more information, see the following sections:

- Adding a graphics page in the Graphics Editor
- "Creating a one-line on a graphics page" on page 562
- Assigning one-line colors

Creating a one-line on a graphics page

You will build a one-line by adding components to the new page. Each component has properties and conditions that control the display. For more information about components, see the Components Overview section. For more information on binding properties and more, see the Diagrams reference Library Components section.

To begin creating a one-line:

1. In the Power SCADA Operation folder, open the Graphics Editor.
2. From the left pane, drag and drop a component in the **Design** workspace.
3. Set the various properties to animate the one-line diagram. These properties include such information as conditions, attached bus names, colors, and the component names. The following properties are required:
   a. All components must have a background rectangle with an opacity of .01
   b. All IDs must be unique, and the ID must have a type prefix if it interacts with the one-line engine (for example: ‘ATS.’, ‘Breaker.’, ‘Meter.’, ‘Motor.’, ‘Source.’, ‘Switch.’, ‘Transformer.’)
   c. Busbar name properties (for example: **TopBusName**) must all have an assigned name.
4. Each component type requires specific properties and conditions to be set. For more information see the following sections:
   - Configuring a meter
   - Configuring a source
   - Configuring circuit breaker or switch
   - Configuring an automatic transfer switch (ATS)
   - Configuring a transformer
   - Configuring a motor

Before you can view your one-line, you must edit the menu structure, which controls the appearance of the graphics page. For more information, see the Defining the Diagrams menu structure section.

Configuring a meter

To configure a meter you have added to a one-line in the Graphics Editor:

1. Edit the following properties:
   a. Custom Properties:
      BusName
      Set to the busbar component name that represents the bus the meter is sensing.
   b. Binding Conditions:
      ActiveCondition

Using a variable tag, create a Boolean condition for whether or not the meter sees that the bus is energized.

The advanced one-line will color all of the busbars according to energized sources and connectivity, then it will check all of the meters on the one-line to ensure there are no incorrectly de-energized buses based on the active conditions set above.

If a meter is configured incorrectly, use the Connection Debugger to troubleshoot the following:

- Ensure the **BusName** is set.
- Ensure the **ActiveCondition** has a valid Boolean condition. If the condition uses a variable tag, it must be a fully qualified variable tag that is configured in the system (e.g. **Cluster-1.EquipmentName.TagName** > 0).

Configuring a source

To configure a source you have added to a one-line in the Graphics Editor:

1. Edit the following properties:
   a. Custom Properties:
      ActiveColor
      Set to the color you want all connected busbars to display when the source is energized.
      BusName
      Set to the busbar component name that represents the bus the source is feeding.
   b. Binding Conditions:
      ActiveCondition
      Using a variable tag, create a Boolean condition for whether or not the source is energized.

The advanced one-line will color all of the busbars connected to this source based on the active condition and active color set above.

If a source is configured incorrectly, use the Connection Debugger to troubleshoot the following:

- Ensure the **BusName** is set.
- Ensure the **ActiveCondition** has a valid Boolean condition. If the condition uses a variable tag, it must be a fully qualified variable tag that is configured in the system (e.g. **Cluster-1.EquipmentName.TagName** > 0).

Configuring a circuit breaker or switch

To configure a circuit breaker or switch you have added to a one-line in the Graphics Editor:

1. Edit the following properties:
   a. Custom Properties:
      BottomBusName
      Set to the busbar component name that represents the bottom bus to which the breaker or switch is connected.
      TopBusName
      Set to the busbar component name that represents the top bus to which the breaker or

switch is connected.

b. Binding Conditions:
ActiveCondition, and for a circuit breaker **EarthSwitchCond**, **RkdPosCond**, and **TripCond**
You can leave these empty for most breakers and switches, which will result in the Advanced One-Line Engine using standard variable tags. If you need to change a condition, then using a variable tag, create Boolean conditions for the breaker or switch statuses.

These components establish the connectivity by which the advanced one-line will energize buses based on connected sources.

If a circuit breaker or switch is configured incorrectly, use the Connection Debugger to troubleshoot the following:

- Ensure the **BottomBusName** and **TopBusName** are set.
- Ensure the conditions are either empty or have valid Boolean conditions. If a condition uses a variable tag, it must be a fully qualified variable tag that is configured in the system (e.g. **Cluster1.EquipmentName.TagName** > 0).

Configuring an automatic transfer switch (ATS)

To configure an automatic transfer switch (ATS) you have added to a one-line in the Graphics Editor:

1. Edit the following properties:
   a. Custom Properties:
   CommonBusName, **MainBusName**, and **AuxBusName**
   Set to the busbar component name that represents the buses to which the ATS is connected.

   b. Binding Conditions:
   PosOnEmergencyCond and **PosOnUtilityCond**
   You can leave these empty if your equipment has an item name (**AtsSwMainCls** or **AtsSwAuxCls**) set for the variable tag. You can also set the condition using a variable tag by creating a Boolean condition for the position of the ATS.

This component helps establish the connectivity by which the advanced one-line will energize buses based on connected sources.

If an ATS is configured incorrectly, use the Connection Debugger to troubleshoot the following:

- Ensure the **CommonBusName**, **MainBusName**, and **AuxBusName** are set.
- Ensure the conditions have valid Boolean conditions. If a condition uses a variable tag, it must be a fully qualified variable tag that is configured in the system (e.g. **AtsSwMainCls** or **AtsSwAuxCls**).

Configuring a transformer

To configure a transformer that you have added to a one-line in the Graphics Editor:

1. Edit the following properties:
    a. Custom Properties:
    BottomBusName
    Set to the busbar component name that represents the bottom bus to which the transformer is connected.
    TopBusName
    Set to the busbar component name that represents the top bus to which the transformer is connected.
    b. Binding Conditions:
    ActiveCondition
    Leave this condition empty.

If a transformer is configured incorrectly, use the Connection Debugger to troubleshoot the following:

- Ensure the **BottomBusName** and **TopBusName** are set.

Configuring a motor

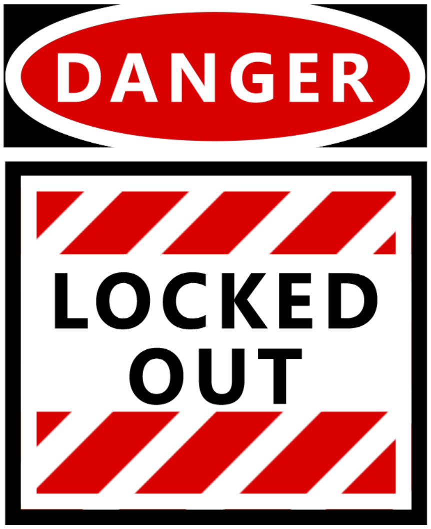
To configure a motor that you have added to a one-line in the Graphics Editor:

1. Edit the following properties:
    a. Custom Properties:
    BottomBusName
    Set to the busbar component name that represents the bottom bus to which the motor is connected.
    TopBusName
    Set to the busbar component name that represents the top bus to which the motor is connected.
    b. Binding Conditions: **ActiveCondition** and **MotorTripCondition**
    Leave these conditions empty.

If a motor is configured incorrectly, use the Connection Debugger to troubleshoot the following:

- Ensure the **BottomBusName** and **TopBusName** are set.

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 Diagrams page. 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; and 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 Components library, add the lockout/tagout component to the graphics page. Position it beside the equipment that is being monitored.
2. Bind to the fully qualified equipment and item name that indicates lockout/tagout status.

   NOTE: Bind to a specific item name, not to a device.

3. By default the status checks the value bound above for 2 values: 0 = lockout/tagout and 1 =

normal operation.

4. If you need a different value check, modify the JavaScript within the component.

Assigning one-line colors

Line coloring is based on the source and meter line active states. Sources dictate the colors for each component. 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.

To assign a color to a source:

1. Select the Source on the graphics page and in the Properties pane under Custom, set the **ActiveColor**.

To edit the default coloring:

1. In the Connection Debugger, click **Set Colors**. The default colors for the different states are selected.
2. Edit the desired color states.
3. Click the **X** to close the Set Colors window.

Multi-Source Coloring

When a component or bus is powered or energized by more than one source color, the advanced one-line uses the default multi-source color for all the components connected to the sources.

To set specific multi-source coloring:

1. In the Connection Debugger, click **Set Colors**, then click **Advanced**. A data grid appears displaying all the selected Active Colors for each source in the project.
2. Click **Add**, and then select the new multi-source color.
3. Check the boxes below the sources you want to combine. This associates those sources with the new multi-source color you added.

   > **NOTE**: Hover over a source color to see a list of all the sources using that Active Color.

4. To delete a previously configured multi-source color, select it in the data grid, and then click **Delete**.

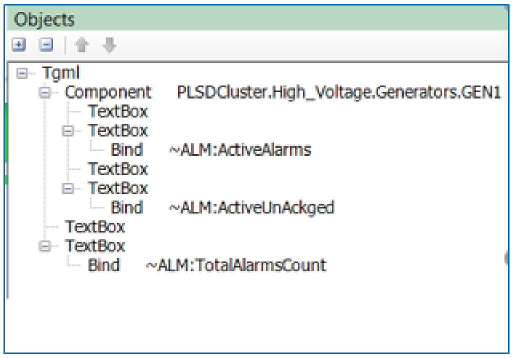
Alarm integration

Alarm integration helps show different types of alarm counts on TGML graphics. You can have **Active Alarms Count**, **UnAcknowledged Alarms Count** and **Total Alarm Count with GroupBy** for different equipment or clusters rendered on the TGML graphics. It is also possible to group these counts based on priority, type of alarm, and incidents.

Designing an Alarm TGML

To design an Alarm TGML:

1. In the Graphics Editor, create a TGML file with some text boxes to display alarm counts with specific labels. For example, **Active Alarms Count**, or **Total Alarm Count with GroupBy**.
2. Create binds for counts that should be displayed on the TGML. Binds can be for overall counts or specific to equipment.

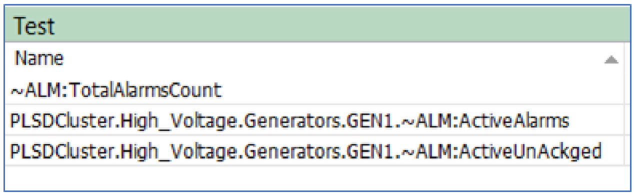


Objects

```
Tgml
  Component   PLSDCluster.High_Voltage.Generators.GEN1
    TextBox
    TextBox
      Bind    ~ALM:ActiveAlarms
    TextBox
    TextBox
      Bind    ~ALM:ActiveUnAckged
  TextBox
  TextBox
    Bind    ~ALM:TotalAlarmsCount
```

NOTE: Make sure that the name of the Alarms bind starts with ~ALM: and the specific type of alarm count. For example, ~ALM:Active Alarms Count or ~ALM:Total Alarm Count with GroupBy. You can do the bind manually or select the bind from the binding window.

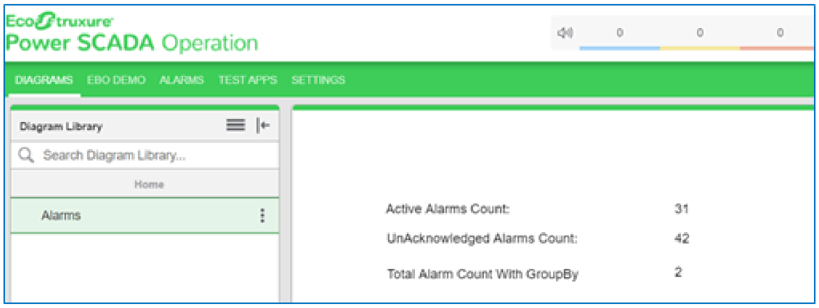
Example:



Test

Name
~ALM:TotalAlarmsCount
PLSDCluster.High_Voltage.Generators.GEN1.~ALM:ActiveAlarms
PLSDCluster.High_Voltage.Generators.GEN1.~ALM:ActiveUnAckged

3. Save the TGML file.
4. Confirm the Alarm TGML in Diagrams.



Active Alarms Count: 31

UnAcknowledged Alarms Count: 42

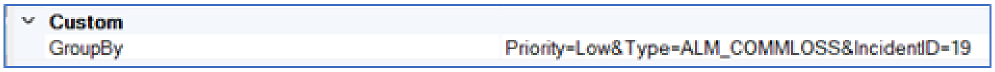
Total Alarm Count With GroupBy 2

Alarm count grouping

The custom property **GroupBy** is used to filter alarm counts for specific equipment or all components.

You can modify the custom property: **GroupBy** for the following attributes:

- Priority
- Alarm Type (Type)
- IncidentID



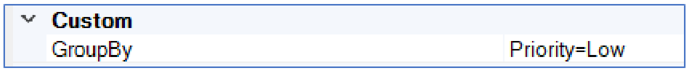
Custom

GroupBy Priority=Low&Type=ALM_COMMLOSS&IncidentID=19

Each parameter must be separated by an ampersand (&) and the value must be followed by '='.

Use Case 1:

If you want the information on total alarms count with **Priority** set to **Low**, then you can use the custom property: **GroupBy** as shown below, but the bind remains same as **~ALM:TotalAlarmsCount**.

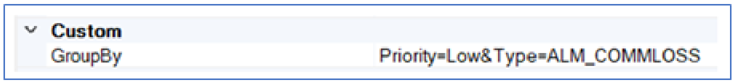


Custom

GroupBy Priority=Low

Use Case 2:

If you want the information on total alarms count with **Priority** set to **Low** and **Type** set to **ALM_COMMLOSS**, then you can use the custom property: **GroupBy** as shown below, but the bind remains same as **~ALM:TotalAlarmsCount**.

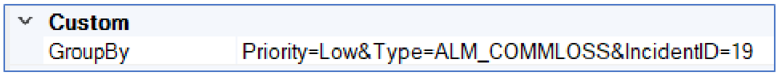


Custom

GroupBy Priority=Low&Type=ALM_COMMLOSS

Use Case 3:

If you want the information on alarms count by equipment name with **Priority** set to **Low**, **Type** set to **ALM_COMMLOSS**, and **IncidentID** set to **IN_OVER_VOLTAGE_1234**, then you can use the custom property: **GroupBy** as shown below, and the bind looks like **PLSDCluster.High_Voltage.Generators.GEN1.~ALM: TotalAlarmsCount**.



Custom

GroupBy Priority=Low&Type=ALM_COMMLOSS&IncidentID=19

NOTE: If the component binds with the equipment name, it summarizes the counts based on equipment name accordingly.

For more information, see the "Binding and filtering alarm counts" on page 1206 workflow.

Trends configuration

Use the Trends application to view trends for real-time data. The information in the Trends application is accessed through trend graphs that are saved in the library. Power SCADA Operation does not provide any pre-configured trends. Configure your own trends to meet your needs.

⚠ 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

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, or equipment damage.

Open the Trends application from the **TRENDS** link in the Web Applications banner.

For information on how to configure the Trends application, see:

- Adding a new trend
- Editing a trend
- Sharing a trend
- Moving a trend
- Deleting a trend

For reference information see:

- Trends UI

For information on how to use Trends, see Trends.

Adding a new trend

Add new trends to monitor real-time data in graphical format.

To add a completely new trend to the library:

1. In Trends, open the Trend Library and navigate to the folder where you want to create the trend.
   (Optional) Add a new folder by clicking **Add Folder**

at the bottom of the library panel, or by clicking **Add Folder** in the **Options** menu ≡ at the top of the library.

2. In the Trend Library, at the bottom of the panel, click **Add Trend** +. This creates a new trend and opens the Add Trend dialog.
3. In Add Trend, enter the configuration information on the **General**, **Axes**, **Chart**, and **Data** tabs. See Trends configuration for details on the configuration options.

   NOTE: A public item is visible to all users in your user group. A private item is visible to you and any user in your user group with Edit permissions on this item type. See "Assign and control user privileges" on page 686 for details.

4. **Save** the trend.

To add a copy of an existing trend to the library:

1. In Trends, open the Trend Library and navigate to the trend you want to copy.

   (Optional) Add a new folder by clicking **Add Folder** at the bottom of the library panel, or by clicking **Add Folder** in the **Options** menu ≡ at the top of the library.

2. Right-click the trend name or click **Options** ⋮ for this trend, and select **Duplicate** to create a copy in the same folder. Select **Copy To** to create a copy in a different folder.
3. (Optional) In the Trend Library, select the new trend, right-click the trend name or click **Options** ⋮ for this trend, and select **Edit** to open the trend settings. Change the trend name and other relevant settings.

   NOTE: A public item is visible to all users in your user group. A private item is visible to you and any user in your user group with Edit permissions on this item type. See "Assign and control user privileges" on page 686 for details.

4. **Save** the modified trend settings.

Related topics:

- Adding a new trend
- Editing a trend
- Sharing a trend
- Moving a trend
- Deleting a trend

For reference information see:

- Trends UI

For information on how to use Trends, see Trends.

Editing a trend

Edit a trend to change the trend name, add a data series, remove a data series or change the trend settings.

To edit a trend:

1. Open the Trend Setup dialog by:
   - Clicking **Edit** ⚙ on the top right of the trend in the trend display pane.
   - Right-clicking a trend name in the Trend Library and selecting the **Edit** menu item.
   - Clicking **Options** ⋮ for this trend in the Trend Library, and selecting the **Edit** menu item.
2. Change the **General**, **Axes**, **Chart**, and **Data** settings for the trend in the Trend Setup dialog. See Trends configuration for details on the configuration options.
3. **Save** the modified settings.

Related topics:

- Adding a new trend
- Editing a trend
- Sharing a trend
- Moving a trend
- Deleting a trend

For reference information see:

- Trends UI

For information on how to use Trends, see Trends.

Sharing a trend

Share trends with other user groups.

NOTE: For Sharing to be enabled, at least one user group, in addition to the Global group, must be configured. To share an item with another user group, you must be a member of that group. The item to be shared must be marked as Public, not Private.

To share a trend:

1. In Trends, open the Trend Library and navigate to the trend you want to share.
2. Right-click the trend name or click **Options** ⋮ for this trend, and select **Share**. This opens the Share Trend window.
3. In Share Trend, select the user groups you want to share this trend with.

   (Optional) Specify a name for the shared trend. The groups you are sharing this trend with will see this name. The name of the original trend remains unchanged.
4. Click **OK** to share this trend.

> **NOTE**: When you share an item with another user group, it appears in the **Shared** folder of this group. You cannot share a shared item.

Related topics:

- Adding a new trend
- Editing a trend
- Sharing a trend
- Moving a trend
- Deleting a trend

For reference information see:

- Trends UI

For information on how to use Trends, see Trends.

Moving a trend

Move trends to a different location in the Library to make them easier to find or easier to manage.

To move a trend:

1. In Trends, open the Trend Library and navigate to the trend you want to move.

   (Optional) Add a new folder by clicking **Add Folder** 🗀 at the bottom of the library panel, or by clicking **Add Folder** in the **Options** menu ≡ at the top of the library.

2. Right-click the trend name or click **Options** ⋮ for this trend, and select **Move To**. This opens the Select Location window.
3. In Select Location, select the location you want to move this trend to.
4. Click **OK** to move the trend.

Related topics:

- Adding a new trend
- Editing a trend
- Sharing a trend
- Moving a trend
- Deleting a trend

For reference information see:

- Trends UI

For information on how to use Trends, see Trends.

Deleting a trend

Delete trends that are no longer needed.

To delete a trend:

1. In Trends, open the Trend Library and navigate to the trend you want to delete.
2. Right-click the trend name or click **Options** ⋮ for this trend, and select **Delete**
3. In Delete Content, click **Yes**, to delete the trend from the Trend Library.

NOTE: Access to this application or function is controlled by user privileges. See "Assign and control user privileges" on page 686 for details.

Related topics:

- Adding a new trend
- Editing a trend
- Sharing a trend
- Moving a trend
- Deleting a trend

For reference information see:

- Trends UI

For information on how to use Trends, see Trends.

Web Applications settings

TIP: You can open the Settings page from the **SETTINGS** link in the Web Applications banner.

Use the Settings page to access Web Applications settings and configuration tools.

NOTE: Access to this application or function is controlled by user privileges. See "Assign and control user privileges" on page 686 for details.

TIP: Use Search, in the Settings Library, to find the settings and tools you are looking for based on keywords.

The Settings page consists of a Settings Library and a configuration area. The Settings Library provides access to the following settings and tools:

Category	Settings/Tools
Alarms	"Alarm Views" on page 575
Integrations	"Authorized Hosts" on page 579

Category	Settings/Tools
Personalization	"Personal Preferences" on page 576 "System localization" on page 577 "System Theme" on page 577
Security	"Session timeout" on page 579

Alarm Views

Use the alarm view settings to:

- Change the number of items that are displayed in the alarms display.
- Change the priority classifications for alarms and incidents.
- Customize the behavior of the alarm annunciator with these settings.
- Customize the display of Load Impact events in Alarm and Incident views.

To change the number of Incidents, Alarms, and Events displayed in the Alarm Viewer:

1. Open **Power Operation**.
2. **SETTINGS** > **Alarms** > **Alarm Views**.
3. Enter values for:
   - Maximum Number of Incidents Displayed: changes the maximum items displayed in the Recent Incidents view.
   - Maximum Number of Alarms Displayed: changes the maximum items displayed in All Alarms and Recent Alarms views.
   - Maximum Number of Events Displayed: changes the maximum items displayed in the Recent Events view.
4. Click **Save**.

To change the Alarm Viewer update interval:

1. Under **Display Settings**, select the **Update Interval**.
2. Click **Save** to apply the changed settings.

To turn the Alarm Annunciator on or off:

1. Under **Annunciator**, turn **Enable** on or off.

   When the Annunciator is turned off, it is not visible in the Web Applications banner.
2. Click **Save** to apply the changed settings.

To change what type of state counts are shown in the Alarm Annunciator:

1. Under **Annunciator**, select the state type for **Show counts for**.
2. Click **Save** to apply the changed settings.

To change the Alarm priorities that are shown in the Alarm Annunciator:

1. Under **Priority Classification**, select or clear the **Visible in Annunciator** check boxes for the Alarm priorities you want to include or exclude from the Annunciator.
2. Click **Save** to apply the changed settings.

To change for which Alarm priorities an Alarm notification sound is played:

1. Under **Priority Classification**, select or clear the **Audible in Annunciator** check boxes for the Alarm priorities you want a notification sound to be played for or not.
2. Click **Save** to apply the changed settings.

To change the sound that is played for Alarm notification:

1. Under **Annunciator**, click **Select Sound File**.
2. In Select Audio File, select the sound you want, or if the sound is not in the Media Library,
   a. Click **Upload Audio File** and either choose a sound file available on your system by clicking **Choose Files**, or drag a sound file into the application area.
   b. Click **Finish** to add it to the Media Library.
3. Click **OK** to complete your sound selection.
4. Click **Save** to apply the changed settings.

To change the Alarm Annunciator update interval:

1. Under **Annunciator**, select the **Update Interval**.
2. Click **Save** to apply the changed settings.

To change the display color and Alarm priority ranges for the Alarm Viewer:

1. Under **Priority Classification**, set the **Color** and **Start** values for the different alarm priorities. The **End** values are adjusted automatically.
2. Click **Save** to apply the changed settings.

To change the display of Load Impact events in Alarm and Incident views:

1. Under **Load Impact Display**, select or clear the check boxes for the options you want or not.
2. Click **Save** to apply the changed settings.

Personal Preferences

Use the personal preferences settings to set your personal localization preferences and choose your personal theme color.

NOTE: Your personal localization settings overrule the system localization settings for your user account. By default, your personal localization settings are the same as the system localization settings. See "System and personal localization settings" on page 1002 for details on the behavior of these settings.

To change any of the personal preferences:

1. Edit the fields or select the options you want from the drop-down lists.
2. Click **Save** to apply the changed settings.

System localization

Use system localization settings to select the language, region, and currency symbol. The setting for **Region** determines date, time, and currency formats.

NOTE: Your personal localization settings overrule the system localization settings for your user account. By default, your personal localization settings are the same as the system localization settings. See "System and personal localization settings" on page 1002 for details on the behavior of these settings.

To change any of the system localization settings:

1. Select the options you want from the drop-down lists.
2. Click **Save** to apply the changed settings.

System Theme

Use the system theme settings to:

- Choose the Default theme or a User Defined theme
- Specify if you want to display the vendor logo in the top right corner of the Web Applications window.
- Change the image and text that is displayed in the top left corner of the Web Applications window.
- Choose a theme color for the borders and other elements of the user interface. You can enable high contrast mode which uses a dark background color for the application.
- Choose the location of the library panel to be on the right or left side of the user interface.
- Specify if you want to use compact mode navigation.

  NOTE: Compact navigation replaces the main navigation bar at the top of the Web Applications user interface with an options button ☰. The options button is displayed at the top left corner of the banner. When you click the button, the navigation links to the different Web applications are shown. Compact mode is used for small displays, such as on mobile devices. The Web Applications user interfaces switches to compact mode automatically when the browser size is reduced below a certain size. Turning on the **Always use compact mode for Navigation** setting forces this mode regardless of browser size.

- Set the colors for the waveform and bust data plots.
- Reset the theme to system defaults.

To select the theme to be default or user defined:

1. Under **General Theme**, click **Default Theme** or click **User Defined**.

   NOTE: With the Default Theme all color, image, and logo options are set to the factory defaults. You can change the location of the navigation panel, choose to always use compact mode, and you can customize the colors for the waveform and burst data plots.

2. Click **Save** to apply the changed settings.

To specify the display of the vendor logo:

1. Under **General Theme**, click **User Defined**.
2. Turn on **Show Vendor logo** to display the logo or turn off **Show Vendor logo** to hide the logo, in the top right corner of the Web Applications window.
3. Click **Save** to apply the changed settings.

To change the top left logo and text:

1. Under **General Theme**, click **User Defined**.
2. Under **Image**, click **Select**.
3. In Select Image, select the image you want, or if the image is not in the Image Library,
   - Click **Upload Image** and either choose an image file available on your system by clicking **Choose Files** or drag an image file into the application area.
   - Click **Finish** to add it to the Image Library.
4. Click **OK** to complete your image selection.

   The image file name is shown under **Image**. The image is updated on the banner when you save your settings. You can use GIF, JPG, JPEG, or PNG image formats. The maximum file size is 2MB. Images are automatically resized to fit the logo area on the banner.

5. Use the **Text** field to change the text beside the logo in the banner. The text is updated when you save your settings.
6. Click **Save** to apply the changed settings.

To change the theme color:

1. Under **General Theme**, click **User Defined**.
2. Under **Theme Color**, select from several preset color themes or create your own using the color selector that opens when you click the color theme icon on the right. When you click a preset color, it is temporarily applied to the interface to show you the effect of the change.

   TIP: Enable high contrast mode to create a dark mode type theme with dark backgrounds.

3. Click **Save** to apply the changed settings.

To choose the location of the library panel:

1. Under **Navigation**, select **Left** or **Right**.
2. Click **Save** to apply the changed settings.

To specify the use of compact mode navigation:

1. Under **Navigation**, turn on **Always use compact mode for Navigation**.
2. Click **Save** to apply the changed settings.

To change the color settings for Waveform and Burst Data:

1. Under **Waveform and Burst Data**, set the color that is used to display the different measurement types.

   NOTE: Click **Reset to Default** to set the colors to the system default.

2. Click **Save** to apply the changed settings.

To reset the theme to the system defaults:

1. Click **Default Theme**.
2. Click **Save** to apply the changed settings.

Authorized Hosts

Use the authorized hosts settings to define third-party web resources that are allowed to either embed (frame) the Power SCADA Operation web applications, or to which the Power SCADA Operation web applications can redirect requests.

To define a third-party web resource as a **Hosts That Can Frame**, add the Uniform Resource Locator (URL) of that resource to the list, for example `https://localhost:446`.

NOTE: Add all the names (URLs) that might be used for a host, for example the server name, "localhost", the IP address, and so on.

To define a third-party web resource as **Hosts That Can Be Redirected To**, add the hostname (no protocol, no port number) of that resource to the list, for example `localhost`.

NOTE: Reset Internet Information Services (IIS) on the Power SCADA Operation server after updating the Authorized Hosts settings.

An example for an application that requires an entry in the **Hosts That Can Frame** list is the integration of Power SCADA Operation with EcoStruxure Building Operation . As part of that integration, Power SCADA Operation Web Applications are embedded in EcoStruxure Building Operation. For this to work, the EcoStruxure Building Operation server URL must be added to the list of hosts that can frame.

Session timeout

Use the session timeout settings to define the timeout behavior of the software web applications.

NOTE: You can enter a timeout value from 1 minute to 1440 minutes (1 day)

When a session timeout is configured, web application clients are logged out after a period of inactivity. The default timeout is 20 minutes. To restart or unlock the session you must enter the login credentials.

A session is considered inactive when none of the following actions are detected for the duration of the timeout period:

- Mouse movement
- Mouse clicks
- Keyboard activity
- Touch screen activity

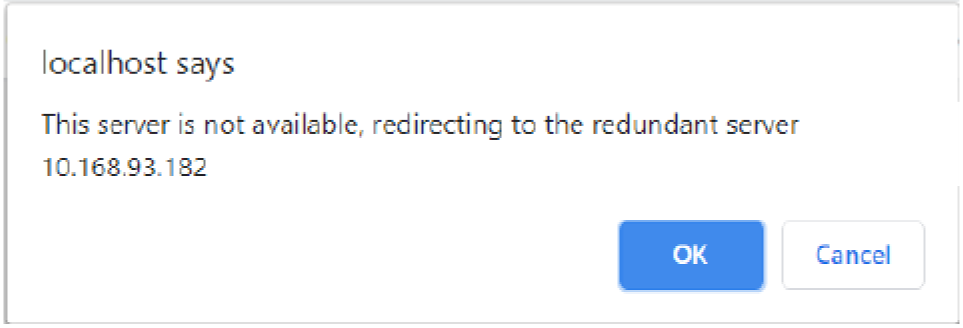
NOTE: If you are integrating PSO with EcoStruxure Building Operation, set the inactivity timeout in EBO to be higher than the value in PSO.

Web redundancy

Web redundancy monitors the availability of the Power SCADA web server and provides notifications about the redundant server if there is a network issue or if the web pages are not available.

Power SCADA Operation can be deployed in a distributed architecture where multiple servers are part of a single system.

If the primary server is not available during operation of Power SCADA Operation WebHMI, a notification will open from the browser. See Setting up and testing a redundant web server for details on allowing notifications.



localhost says

This server is not available, redirecting to the redundant server 10.168.93.182

Configuring browser encryption

1. Navigate to **Administrative Tools** and open **Internet Information Services (IIS)**.
2. In the left Connections pane, select **[Machine Name**] > **Sites** > **Default Web Site**.
3. In the right Actions pane, click **Bindings**.
4. In the Site Bindings window, select **https** and click **Edit**.

5. In the Edit Site Binding window, in the SSL certificate drop-down, select **[Machine Name] Grpc Certificate**.

6. Click **OK** to close the Edit Site Binding window, then click **Close** to close the Site Bindings window.

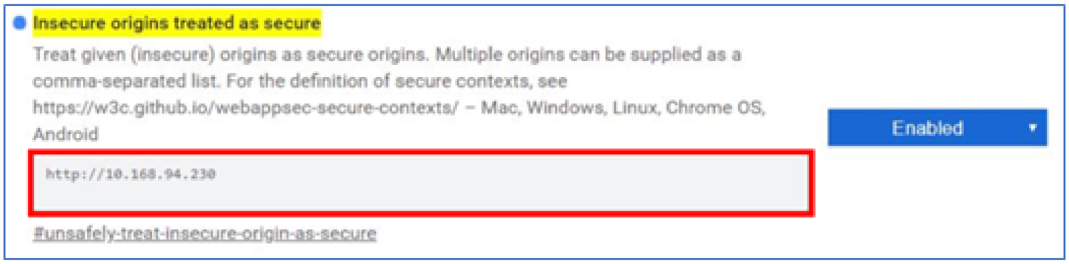
7. Restart IIS.

Configuring without encryption (not recommended)

Setup browser configuration to work in Google Chrome.

1. Type `chrome://flags` in the URL area of Google Chrome.

2. Search for **Insecure origins treated as secure** and enter the IP address.



Insecure origins treated as secure

Treat given (insecure) origins as secure origins. Multiple origins can be supplied as a comma-separated list. For the definition of secure contexts, see https://w3c.github.io/webappsec-secure-contexts/ – Mac, Windows, Linux, Chrome OS, Android

http://10.168.94.230

#unsafely-treat-insecure-origin-as-secure

Enabled ▾

NOTE: Use the Power SCADA web server IP address.

3. Set **Insecure origins treated as secure** to **Enabled**.

NOTE: You will not receive the pop-up message if **Enabled** is not selected for this setting.

Notifications

Notifications are not available in the following scenarios:

- Browsing the pages for the first time.
- After clearing the browsing history.
- Navigating between TGML diagrams and menu items.

For these scenarios, return to the website using the primary network server to receive notifications.

Setting up and testing a redundant web server

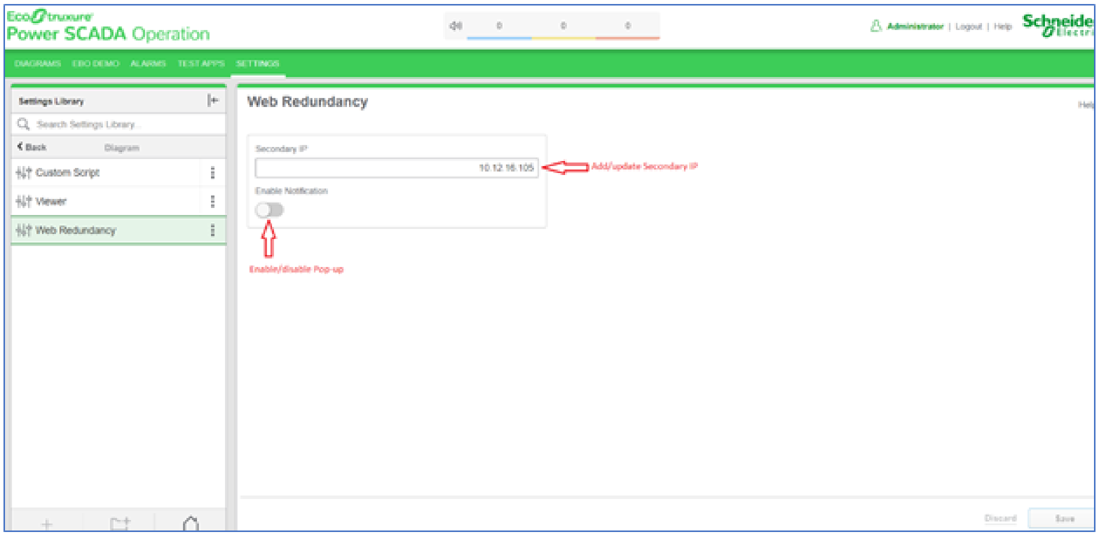
Use this procedure to configure a redundant secondary web server for failover, allowing browser notifications, and test that it is working.

1. Open Power SCADA Operation.

   Example path: https://[localhost]/WebHmi/Login.

2. Select **SETTINGS** tab > **Web Redundancy**.

3. Enter the IP address for the secondary Power SCADA Operation server in the **Secondary IP** text box:



Web Redundancy

Secondary IP: 10.12.16.105

Enable Notification

4. Turn on the **Enable Notification** toggle. This allows the web browser to display notifications.

5. Click **Save**.

6. Refresh the web browser.

7. Close Power SCADA Operation WebHMI.

8. Clear your web browser history and browsing data.

9. Reopen Power SCADA Operation WebHMI.

   If the IIS server is running on a virtual machine, stop it from the left Connections pane in your operating system. Then close and reopen Power SCADA Operation.

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 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\v2020 R2\bin`).

Web Applications Access and Privilege Levels

Power SCADA Operation 2020 Web Applications use the following access and privilege levels:

Access Levels	Privilege Levels					
	None = 0	Observer = 1	User = 2	Controller =3	Operator = 4	Administrator = 5
AlarmViewer.AcknowledgeAlarm				X	X	X
AlarmViewer.DeleteAny						X
AlarmViewer.EditAny						X
AlarmViewer.Owner				X	X	X
AlarmViewer.SetSystemDefaultItem						X
AlarmViewer.ViewIncidents			X	X	X	X
ApplicationAccess.AlarmViewer			X	X	X	X
ApplicationAccess.HmiApplication		X	X	X	X	X
ApplicationAccess.Event			X	X	X	X
ApplicationAccess.RealtimeData		X	X	X	X	X
ApplicationAccess.RealtimeTrend		X	X	X	X	X
ApplicationAccess.Tgml		X	X	X	X	X
ApplicationAccess.WebConfig		X	X	X	X	X
ConfigurationAccess.Alarms						X
ConfigurationAccess.CustomScripting						X
ConfigurationAccess.MyPreferences		X	X	X	X	X
ConfigurationAccess.Localization						X
ConfigurationAccess.Theme						X
ConfigurationAccess.Security						X
ConfigurationAccess.Tgml				X		X
Diagrams.Owner			X	X	X	X
Diagrams.EditAny						X
Diagrams.DeleteAny						X
Diagrams.SetSystemDefaultItem						X
Diagrams.ControlActions				X	X	X
RealtimeTrend.DeleteAny						X
RealtimeTrend.EditAny						X
RealtimeTrend.Owner				X	X	X
No Access	X					

These access and privilege levels are not the same as the roles and privileges in Citect, Windows, and Active Directory.

- For Citect users, privilege levels are mapped to the Web Applications access levels.
- For local Windows users, the local Windows groups are mapped to the Web Applications access levels.
- For Active Directory Windows users, the Active Directory groups are mapped to the Web Applications access levels.

The Schneider Electric Core Services configuration file is used to configure user access levels. The default installation path is: `C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020 R2\Applications\AppServices\bin\`

`configuration.xml` contains sections for Windows Groups and Citect privilege levels.

Citect Privilege Level Mapping to Web Applications Privilege Levels

Citect privilege levels map to the following Web Applications privilege (and access) levels:

Web Applications Privilege Levels	Citect Privilege Levels
None = 0	• `<ConfigurationItem Key="Priv0" Category="Security" Application="CitectPlatform"> <Value>0</Value> </ConfigurationItem>`
Observer = 1	• `<ConfigurationItem Key="Priv1" Category="Security" Application="CitectPlatform"> <Value>1</Value> </ConfigurationItem>`
User = 2	• `<ConfigurationItem Key="Priv2" Category="Security" Application="CitectPlatform"> <Value>2</Value> </ConfigurationItem>`
Controller = 3	• `<ConfigurationItem Key="Priv3" Category="Security" Application="CitectPlatform"> <Value>3</Value> </ConfigurationItem>` • `<ConfigurationItem Key="Priv4" Category="Security" Application="CitectPlatform"> <Value>3</Value> </ConfigurationItem>`

Web Applications Privilege Levels	Citect Privilege Levels
Operator = 4	• </ConfigurationItem> <ConfigurationItem Key="Priv5" Category="Security" Application="CitectPlatform"> <Value>4</Value> • </ConfigurationItem> <ConfigurationItem Key="Priv6" Category="Security" Application="CitectPlatform"> <Value>4</Value> </ConfigurationItem>
Administrator = 5	• <ConfigurationItem Key="Priv7" Category="Security" Application="CitectPlatform"> <Value>5</Value> </ConfigurationItem> • <ConfigurationItem Key="Priv8" Category="Security" Application="CitectPlatform"> <Value>5</Value> </ConfigurationItem>

For example, Citect privilege level 3 ("Priv3") maps to access level 3, which is a Controller. To make Citect privilege 3 be a ‘User’ access level, change the “Value” element for Priv3 to 2:

```
<ConfigurationItem Key="Priv3" Category="Security" Application="CitectPlatform">
<Value>2</Value>
</ConfigurationItem>
```

Windows Groups Privilege Level Mapping to Web Applications Privilege Levels

Default local Windows Groups are created during the install. Adding local Windows Users to these groups will grant them the following mapped Web Applications privilege (and access) levels:

Web Applications Privilege Levels	Windows Group Privilege Levels
None = 0	• NA
Observer = 1	• <ConfigurationItem Key="OsObservers" Category="Security" Application="CitectPlatform"> <Value>PSO_Observers</Value> </ConfigurationItem>

Web Applications Privilege Levels	Windows Group Privilege Levels
User = 2	• `<ConfigurationItem Key="OsUsers" Category="Security" Application="CitectPlatform"> <Value>PSO_Users</Value> </ConfigurationItem>`
Controller = 3	• `<ConfigurationItem Key="OsControllers" Category="Security" Application="CitectPlatform"> <Value>PSO_Controllers</Value> </ConfigurationItem>`
Operator = 4	• `<ConfigurationItem Key="OsOperators" Category="Security" Application="CitectPlatform"> <Value>PSO_Operators</Value> </ConfigurationItem>`
Administrator = 5	• `<ConfigurationItem Key="OsAdministrators" Category="Security" Application="CitectPlatform"> <Value>PSO_Administrators</Value> </ConfigurationItem>`

For example, all local Windows Users added to the PSO_Controllers group will be granted the Web Application Controller = 3 access level. The values are a semicolon delimited list.

Active Directory Privilege Levels

Active Directory (AD) Windows Users added to local Windows Groups are not supported. AD Users will be authenticated against AD Windows Groups.

For example, if an AD User is in the AD Windows Group “Web_Controllers”, add that group to the **OsControllers** section:

```
<ConfigurationItem Key="OsControllers" Category="Security" Application="CitectPlatform">
<Value>PSO_Controllers;Web_Controllers</Value>
</ConfigurationItem>
```

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	Operator1	Operator2	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.

Changing 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 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.

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 625

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 587 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 overwrite 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 624 for details.

Cybersecurity

This section provides information on how to help secure your system during the Configure phase.

NOTE: It is recommended that you protect your system from unauthorized access by leveraging the Access Control for objects in Windows Active Directory and/or Windows Authentication.

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

Topic	Content
McAfee Application Control	Configuring McAfee Application Control to prevent unauthorized applications from running on your systems.
Securing the Network and Servers	Configuring firewalls to restrict and control traffic between IT, OT, and Internet network zones.
Two-Factor Authentication	Configuring a one-time password (OTP) to accomplish two-factor authentication.
Configuring Power SCADA Operation for Network Segmentation	Configuring Power SCADA Operation to communicate with multiple network adapters in a network segmentation architecture.
Use Single Sign-On	Associating a Citect user with a Power SCADA Operation or a Power Monitoring Expert username and password.
Configuring 3rd Party Certificates	Configuring 3rd party Certificates for use with Power SCADA Operation.

McAfee Application Control

McAfee Application Control (Application Control) is a dynamic whitelisting program that is used to help prevent 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 can prevent a legitimate application from executing if the required rules are not defined. Application Control tracks all unsuccessful attempts made by authorized applications to modify protected files or run other executable files.

You can review information for unsuccessful 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, 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 help 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.

Two-Factor Authentication

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 679 for details.

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.
- 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.

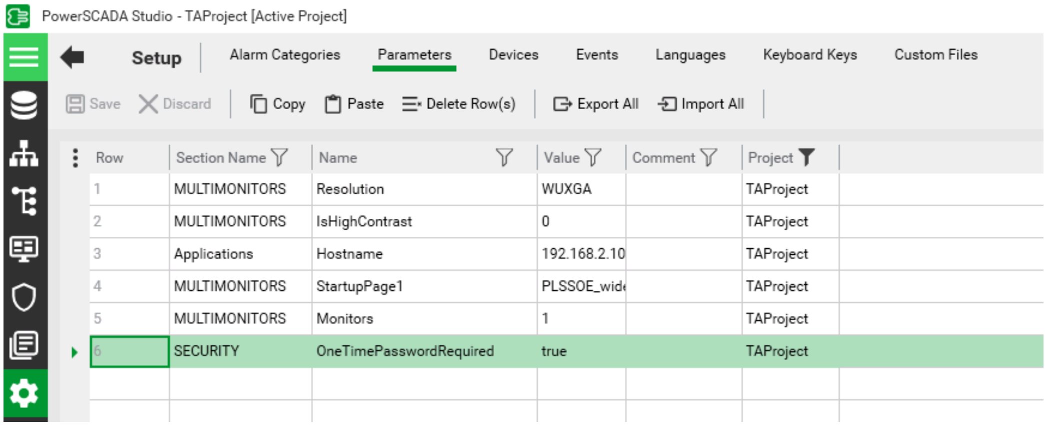
Add the Citect parameter

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



Row	Section Name	Name	Value	Comment	Project
1	MULTIMONITORS	Resolution	WUXGA		TAProject
2	MULTIMONITORS	IsHighContrast	0		TAProject
3	Applications	Hostname	192.168.2.10		TAProject
4	MULTIMONITORS	StartupPage1	PLSSOE_wid		TAProject
5	MULTIMONITORS	Monitors	1		TAProject
6	SECURITY	OneTimePasswordRequired	true		TAProject

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 2020 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:

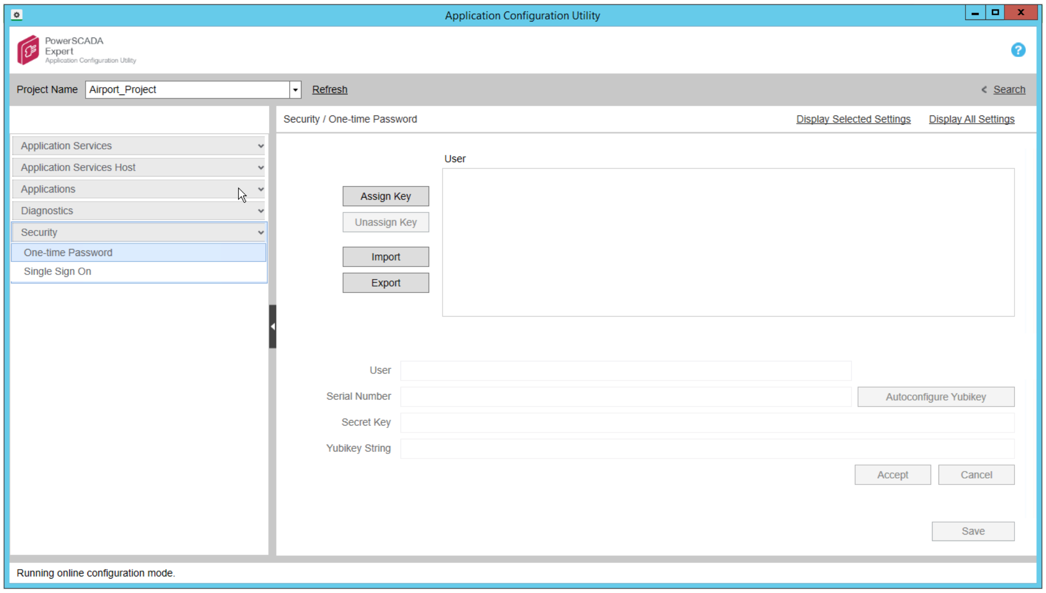
- Autoconfigure requires that you have a USB port available on your computer.
- If you do not have a USB port available on the server – because it is in a virtual machine or you do not have physical access– program the key on a remote machine (see "Manually configure the YubiKey" on page 597, below), and then transfer the configuration to the server (see "Two-Factor Authentication" on page 593, below).

- Autoconfigure will not work on virtual machines.
- 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 597 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**.



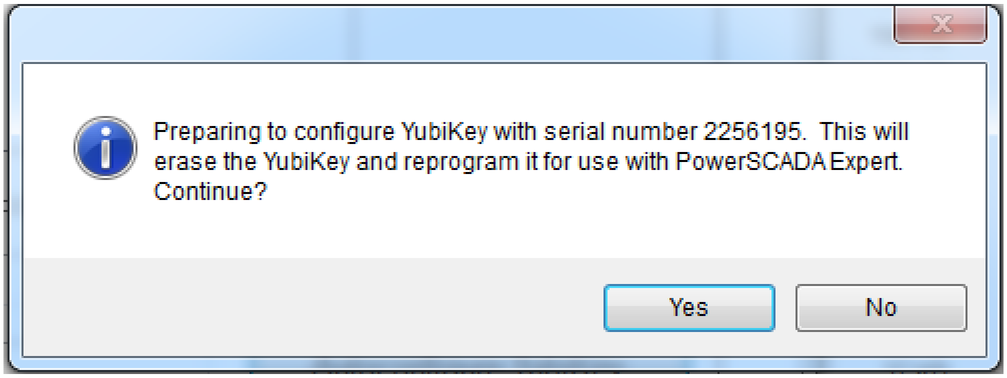
Project Name: Airport_Project

Security / One-time Password

Running online configuration mode.

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:



Preparing to configure YubiKey with serial number 2256195. This will erase the YubiKey and reprogram it for use with PowerSCADA Expert. Continue?

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

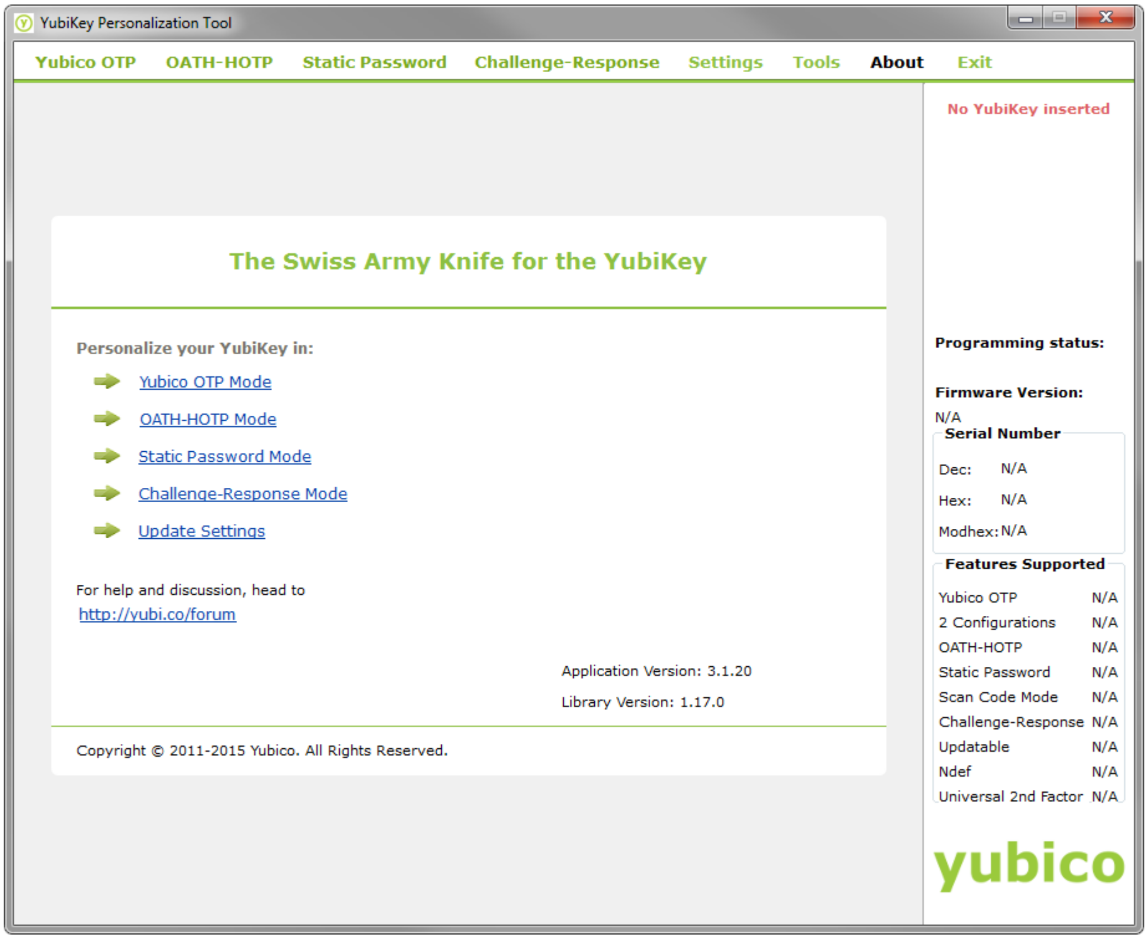
If you cannot auto-configure the YubiKey, program and configure it manually.

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:



No YubiKey inserted

The Swiss Army Knife for the YubiKey

Personalize your YubiKey in:

- Yubico OTP Mode
- OATH-HOTP Mode
- Static Password Mode
- Challenge-Response Mode
- Update Settings

For help and discussion, head to
http://yubi.co/forum

Application Version: 3.1.20

Library Version: 1.17.0

Copyright © 2011-2015 Yubico. All Rights Reserved.

Programming status:

Firmware Version:
N/A

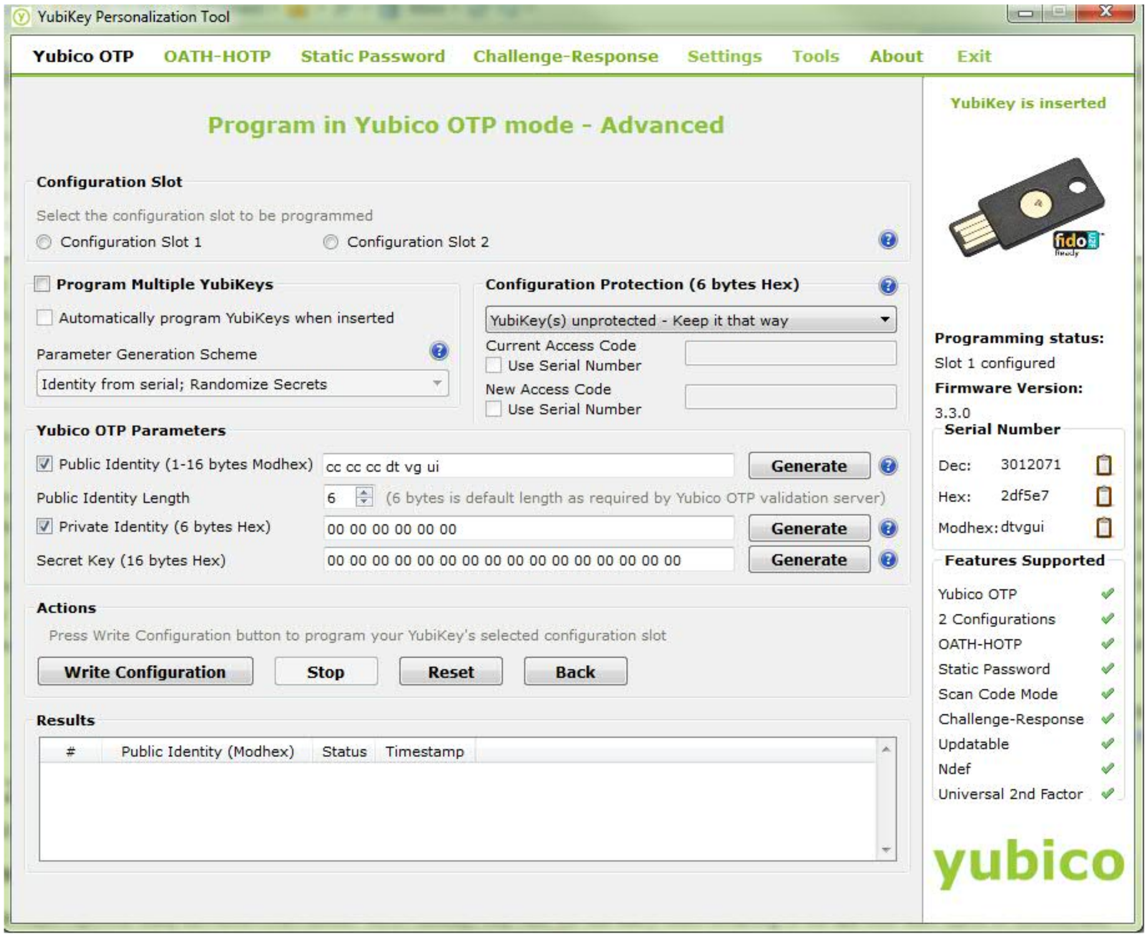
Serial Number

Dec:	N/A
Hex:	N/A
Modhex:	N/A

Features Supported

Yubico OTP	N/A
2 Configurations	N/A
OATH-HOTP	N/A
Static Password	N/A
Scan Code Mode	N/A
Challenge-Response	N/A
Updatable	N/A
Ndef	N/A
Universal 2nd Factor	N/A

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:



Program in Yubico OTP mode - Advanced

Configuration Slot

Select the configuration slot to be programmed

○ Configuration Slot 1 ○ Configuration Slot 2

□ Program Multiple YubiKeys

□ Automatically program YubiKeys when inserted

Parameter Generation Scheme: Identity from serial; Randomize Secrets

Configuration Protection (6 bytes Hex): YubiKey(s) unprotected - Keep it that way

Current Access Code: □ Use Serial Number

New Access Code: □ Use Serial Number

Yubico OTP Parameters

☑ Public Identity (1-16 bytes Modhex): cc cc cc dt vg ui

Public Identity Length: 6 (6 bytes is default length as required by Yubico OTP validation server)

☑ Private Identity (6 bytes Hex): 00 00 00 00 00 00

Secret Key (16 bytes Hex): 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00

Actions

Press Write Configuration button to program your YubiKey's selected configuration slot

Results

#	Public Identity (Modhex)	Status	Timestamp

Programming status: Slot 1 configured

Firmware Version: 3.3.0

Serial Number

Dec:	3012071
Hex:	2df5e7
Modhex:	dtvgui

Features Supported

Yubico OTP	✓
2 Configurations	✓
OATH-HOTP	✓
Static Password	✓
Scan Code Mode	✓
Challenge-Response	✓
Updatable	✓
Ndef	✓
Universal 2nd Factor	✓

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**.

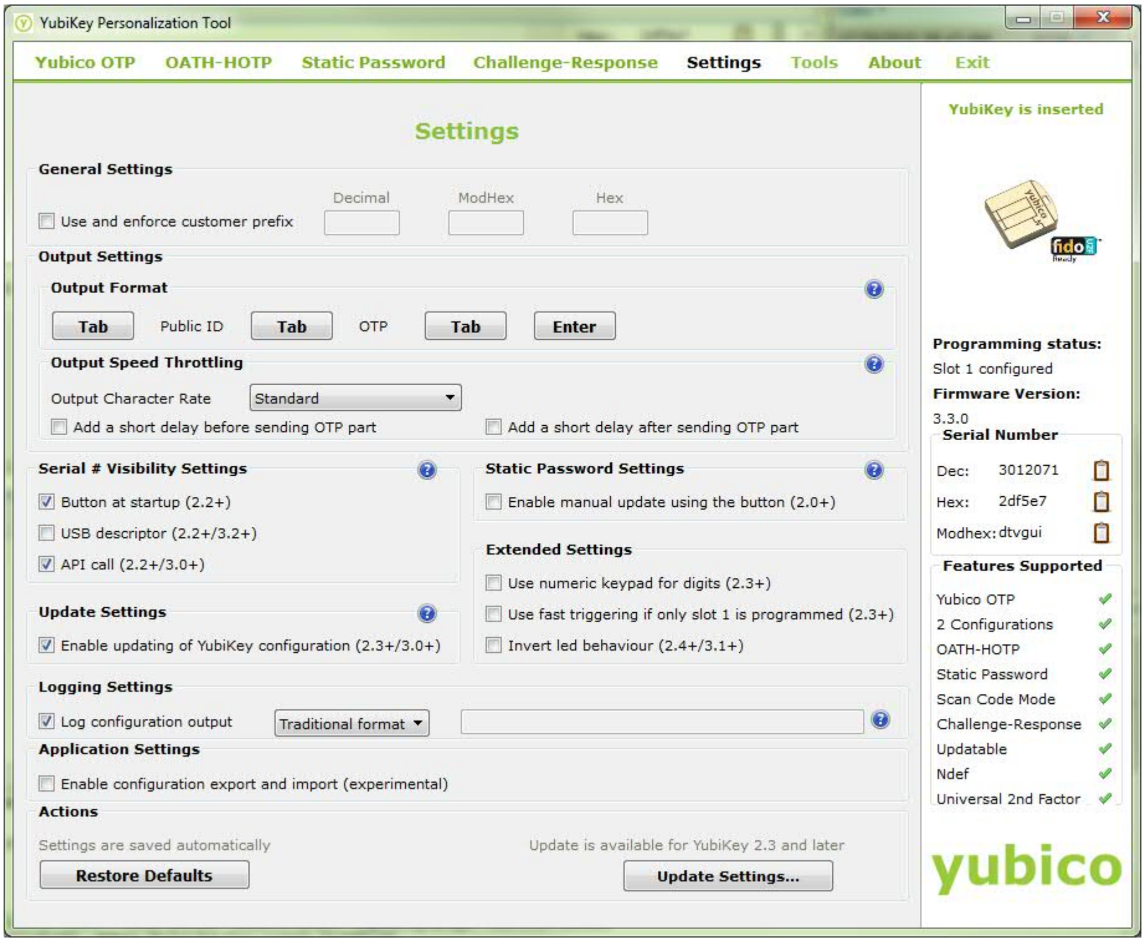
   c. Click **Private Identity** and then click **Generate**.

   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:



Settings

General Settings

☐ Use and enforce customer prefix Decimal ModHex Hex

Output Settings

Output Format

Tab Public ID Tab OTP Tab Enter

Output Speed Throttling

Output Character Rate: Standard

☐ Add a short delay before sending OTP part ☐ Add a short delay after sending OTP part

Serial # Visibility Settings

☑ Button at startup (2.2+)

☐ USB descriptor (2.2+/3.2+)

☑ API call (2.2+/3.0+)

Static Password Settings

☐ Enable manual update using the button (2.0+)

Extended Settings

☐ Use numeric keypad for digits (2.3+)

☐ Use fast triggering if only slot 1 is programmed (2.3+)

☐ Invert led behaviour (2.4+/3.1+)

Update Settings

☑ Enable updating of YubiKey configuration (2.3+/3.0+)

Logging Settings

☑ Log configuration output Traditional format

Application Settings

☐ Enable configuration export and import (experimental)

Actions

Settings are saved automatically Update is available for YubiKey 2.3 and later

Restore Defaults Update Settings...

Programming status: Slot 1 configured

Firmware Version: 3.3.0

Serial Number

Dec: 3012071

Hex: 2df5e7

Modhex: dtvgui

Features Supported

Yubico OTP	✓
2 Configurations	✓
OATH-HOTP	✓
Static Password	✓
Scan Code Mode	✓
Challenge-Response	✓
Updatable	✓
Ndef	✓
Universal 2nd Factor	✓

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.

Logging 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.

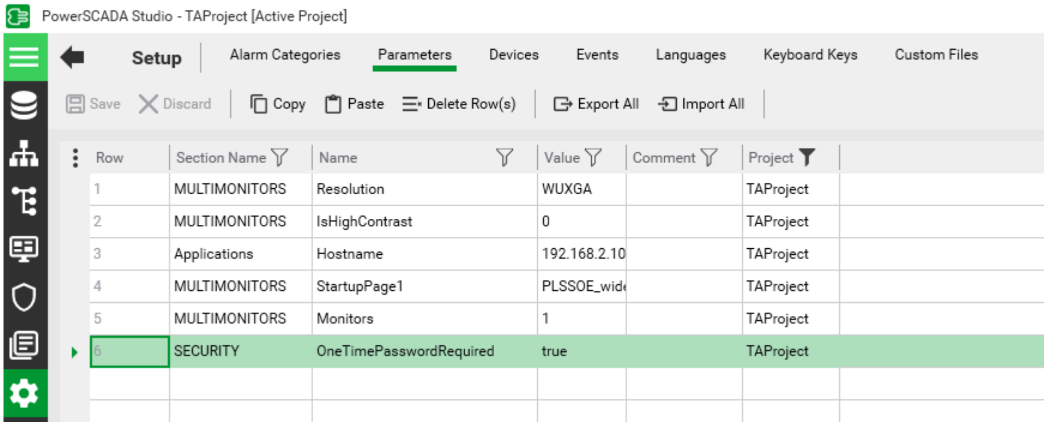
   The one-time password is generated. The key and software communicate behind the scenes to verify the uniqueness of the one-time password and to click OK.

   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.



Row	Section Name	Name	Value	Comment	Project
1	MULTIMONITORS	Resolution	WUXGA		TAProject
2	MULTIMONITORS	IsHighContrast	0		TAProject
3	Applications	Hostname	192.168.2.10		TAProject
4	MULTIMONITORS	StartupPage1	PLSSOE_wid		TAProject
5	MULTIMONITORS	Monitors	1		TAProject
6	SECURITY	OneTimePasswordRequired	true		TAProject

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.

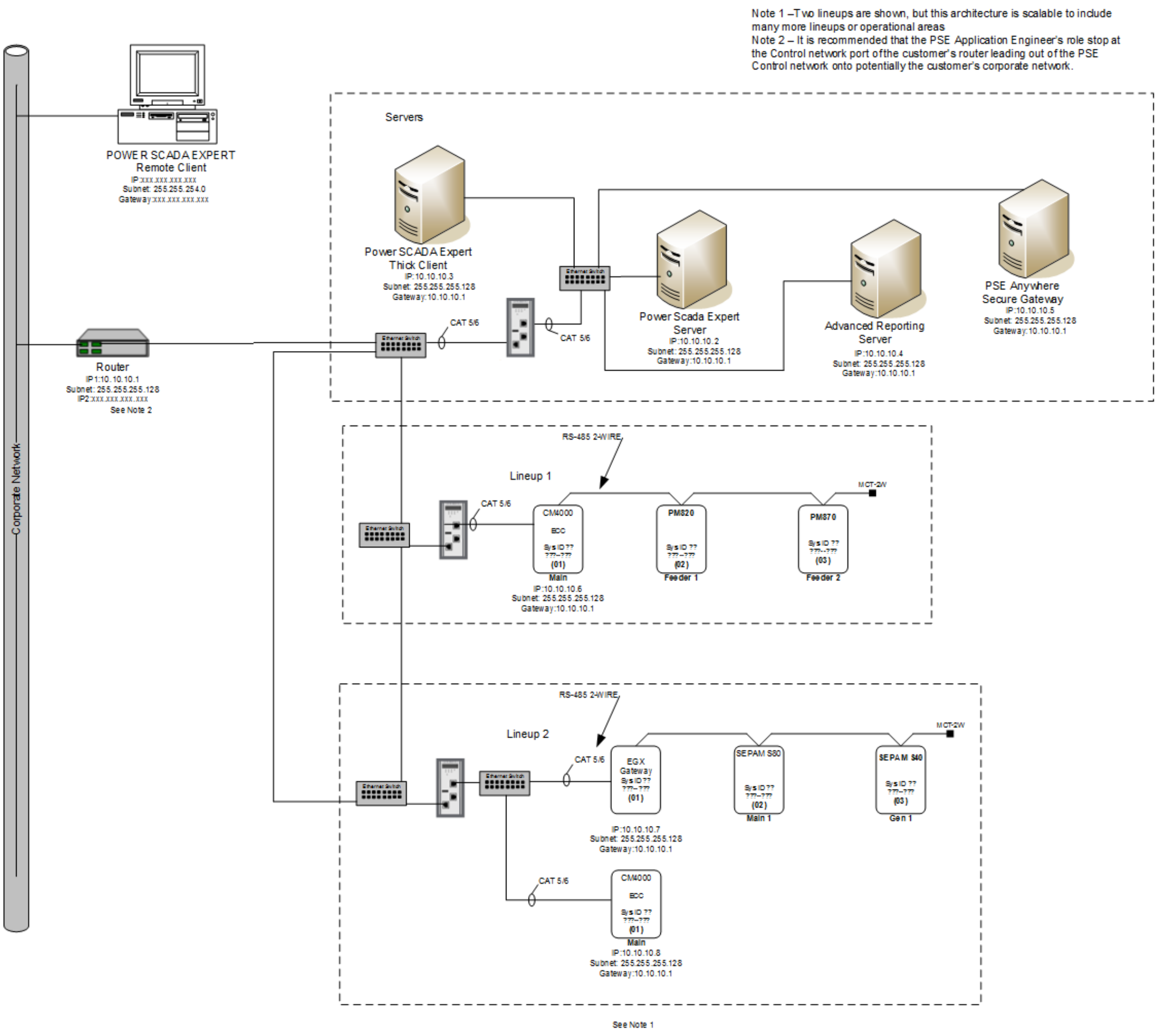
Power SCADA now supports electronic software keys to allow IT departments to lock-down USB ports on server computers.

The configuration setup steps are:

1. Create a project.
2. Add all the devices on the network.
3. 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.

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 ofthe customer's router leading out of the PSE Control network onto potentially the customer's corporate network.

Configuring Power SCADA Operation for Network Segmentation

Power SCADA Operation can be configured to communicate with multiple network adapters in a network segmentation architecture. For security reasons, consider network segmentation for the following scenarios:

- Multiple Power SCADA Operation servers or clients are configured to run over a WAN or the Internet.
  - Confirm that appropriate security precautions (such as a VPN) are used when connecting networks over a potentially public link (such as the Internet).
- An untrusted corporate network is connected to the control system network.

To configure a Power SCADA Operation project for network segmentation, follow these guidelines:

1. Citect SCADA Help: Add Network Addresses
2. Vijeo Citect 2015 Web Client Guide: Port-Forwarding / Address Forwarding (see Power SCADA Operation 2020Installation disc)
3. Firewall Ports – SCADA Server Configuration

Firewall ports—SCADA Server configuration

Each server component has a unique default port assigned to it. This default port may only be used with that type of server. However, application engineers may choose ports other than the defaults, depending on the design of the project. Non-default ports need to also be added to the firewall exceptions.

For more information about the ports for Advanced Reporting and Dashboards, see "Ports" in the *Power Monitoring Expert 2020 – IT Guide*. Which ports are required for a specific installation depends on the Power Monitoring Expert system configuration and the monitoring devices used.

The following table lists the default port numbers and their associated server type:

Default port	Server type	Server role
80	Web Server	Project files for web client
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
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
23103	Application Server	Hosts web service endpoints and data processing for Basic Reports and LiveView
23104	Event Notification Services	Allows redundant event notification services to synchronize
23200	Web Service	Hosts Web Service endpoints and data processing for WebHmi

NOTE: If Power SCADA Operation alarm, trend, report and I/O servers are created using non-default ports, you will also need to create those ports exceptions.

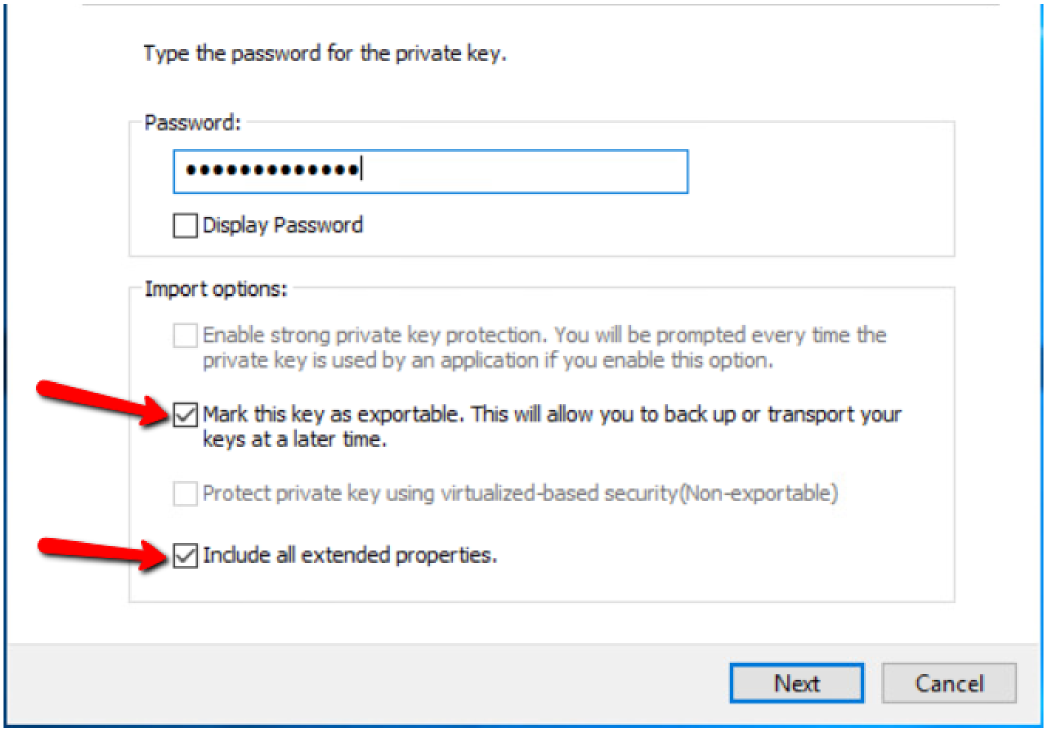
Configure 3rd party certificates

To configure 3rd party certificates for use with Power SCADA Operation, you must configure the service layer, edit the certificate, and then update the registry.

NOTE: The 3rd party certificate you wish to use must be in the Personal Information Exchange (PFX) file format.

Configuring the Service Layer

1. Navigate to and double-click the PFX file you want to import. The Certificate Import Wizard appears.
2. Select **Local Machine** and click **Next**.
3. In the File name field, verify the name of the file you are importing, then click **Next.**
4. If a password exists for the private key, enter it in the **Password** field.
5. Select the **Mark this key as exportable. This will allow you to back up or transport your keys at a later time.** and **Include all extended properties.** check boxes.



Type the password for the private key.

Password:

•••••••••••••

□ Display Password

Import options:

□ Enable strong private key protection. You will be prompted every time the private key is used by an application if you enable this option.

☑ Mark this key as exportable. This will allow you to back up or transport your keys at a later time.

□ Protect private key using virtualized-based security(Non-exportable)

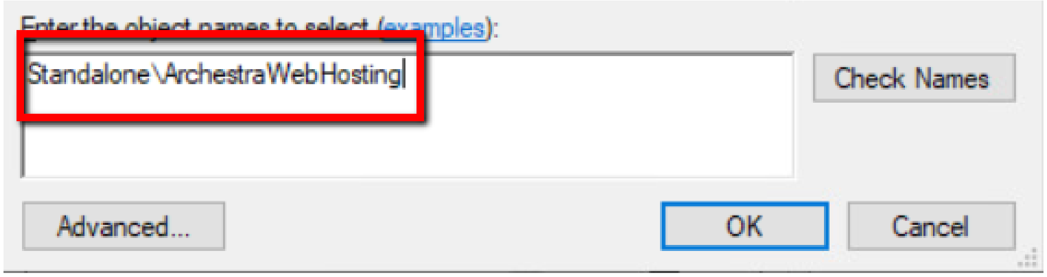
☑ Include all extended properties.

6. Click **Next**.
7. On the Certificate Store page, choose the default option **(Automatically select the certificate store based on the type of certificate)**, then click **Next**.
8. Click **Finish**.
9. Click **OK**.

Setting permissions on the certificate

1. Press **Window + R** to open the Run window.
2. In the Open field, type **mmc** and click **OK**.
3. In the Console window, select **File** > **Add/remove snap-in**.
4. In the left pane, select **Certificates**, then click **Add**.
5. In the Certificates snap-in window, select **Computer account**, then click **Next**.

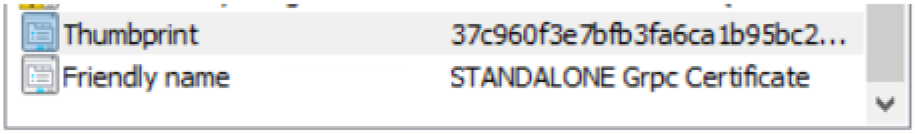
6. Select **Local computer**, then click **Finish**.
7. Click **OK** to close the Add or Remove Snap-ins window.
8. In the Console Root pane, expand **Certificates** > **Personal** > **Certificates**. The installed certificate appears in the right pane.
9. Right-Click the certificate and select **All Tasks** > **Manage Private Keys…**
10. Click **Add** and type **<ComputerName\ArchestraWebHosting** then click **OK**.



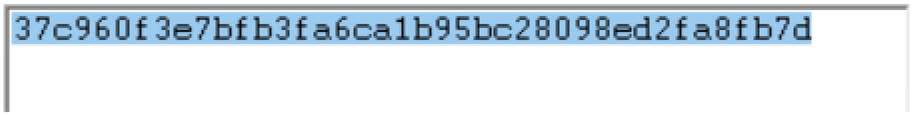
Enter the object names to select (examples):

Standalone\ArchestraWebHosting

11. Verify that Full Control and Read permissions are allotted to the ArchestraWebHosting group.
12. Click **OK**.
13. Double-Click the certificate to view it. Select the **Details** tab, then locate and click the Thumbprint field in the list.
14. Highlight the value, then press **Ctrl + C** to copy the value and press **Ctrl + V** to paste it to notepad or another text editor. You will need this value to update the registry.



Thumbprint	37c960f3e7bfb3fa6ca1b95bc2...
Friendly name	STANDALONE Grpc Certificate



37c960f3e7bfb3fa6ca1b95bc28098ed2fa8fb7d

NOTE: Some operating systems may store the Thumbprint with spaces, you may have to delete the spaces prior to updating the registry.

Updating the Registry

NOTICE

IRREVERSIBLE OPERATING SYSTEM DAMAGE OR DATA CORRUPTION

Before making any changes, back up your Windows Registry to a network folder or other remote location.

Failure to follow these instructions can result in permanent loss of 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.

1. Start a Windows command-prompt in Administrator mode.
2. Copy and paste the following command to create a backup of the registry key:
   Reg copy "HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\Schneider Electric\Power SCADA Operation\WebApplications\Default" "HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\Schneider Electric\Power SCADA Operation\WebApplications\Default_orig" /s /f
3. Copy and paste the following command to update the registry value:
   Reg Add "HKEY_LOCAL_MACHINE\SOFTWARE\WOW6432Node\Schneider Electric\Power SCADA Operation\WebApplications\Default" /t REG_SZ /v SslThumbprint /d “<PASTE THE THUMBPRINT HERE>” /f
4. Run the following commands for the changes to take effect in the services:
   %windir%\System32\inetsrv\Appcmd stop apppool /apppool.name:PsoWebserviceAppPool

   %windir%\System32\inetsrv\Appcmd start apppool /apppool.name:PsoWebserviceAppPool

   %windir%\System32\inetsrv\Appcmd stop apppool /apppool.name:PlatformServerAppPool

   %windir%\System32\inetsrv\Appcmd start apppool /apppool.name:PlatformServerAppPool
5. Close the command prompt.

NOTE: For more information on configuring the System Management Server, see the instructions included in the Encryption section.

Customize default behaviors

In this section, you will find these topics:

- "Customize a project using Cicode" on page 607
- "Localizing Power SCADA Operation" on page 610
- "Running Power SCADA Operation as a Windows Service" on page 613

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:

- "PLSProviderEngine.ci Module" on page 607
- "Clear cache and refresh platform" on page 609

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\v2020 R2\bin\Help\Citect SCADA`).

For information about driver-specific INI parameters that you can configure, see "Citect INI Parameters" on page 885.

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 607 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
  - Default Starter Project
- Power SCADA Applications
  - Basic Reports
  - LiveView

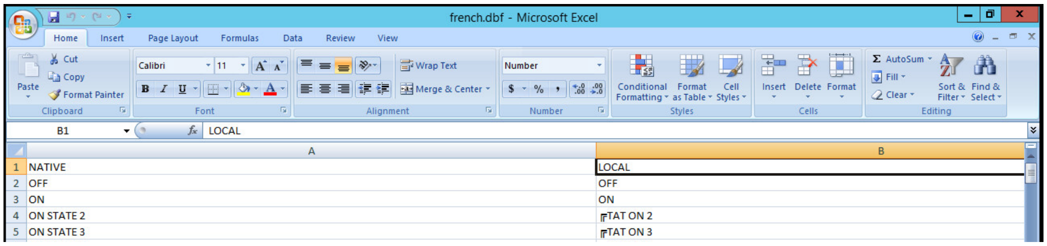
You must create all custom project content in the local language.

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\v2020 R2\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.

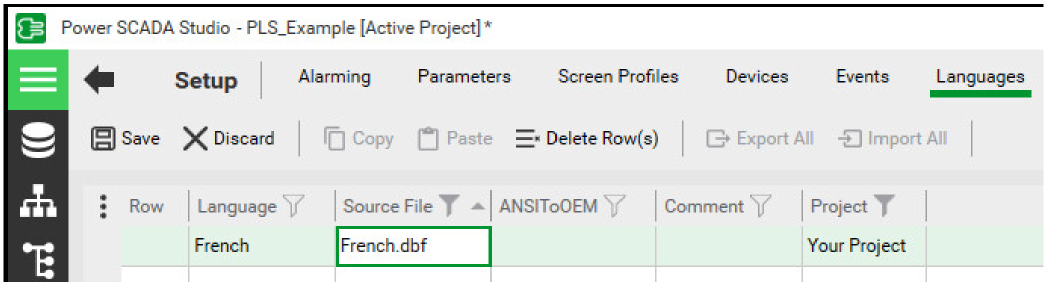


LOCAL

	A	B
1	NATIVE	LOCAL
2	OFF	OFF
3	ON	ON
4	ON STATE 2	╓TAT ON 2
5	ON STATE 3	╓TAT ON 3

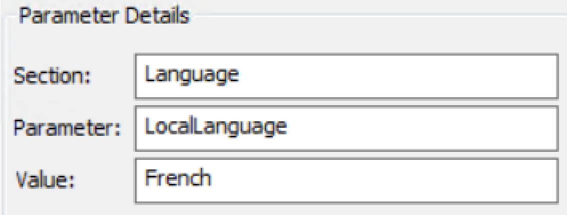
4. Click **Save As**, enter [Localized Language].dbf as the File Name.

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**.



Row	Language	Source File	ANSIToOEM	Comment	Project
	French	French.dbf			Your Project

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.


Parameter Details

Section: Language

Parameter: LocalLanguage

Value: French

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\v2020 R2 \Applications\AppServices\bin\Resources	CDMMetadataNameResources.resx CDMMetadataValueResources.resx CDMTopicDescriptiveNameResources.res CDMUnitResources.resx

Basic Reports	C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020 R2\Applications\AppServices\bin\Resources	ReportDefinitionResources.en-US.resx Reporting.RapidAccess.resx Reporting.StandardReports.resx Reporting.Utilities.en-US.resx * CDM files also required
LiveView	C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020 R2\Applications\LiveView\Viewer\App_GlobalResources	LiveViewViewer.resx ** CDM files also required

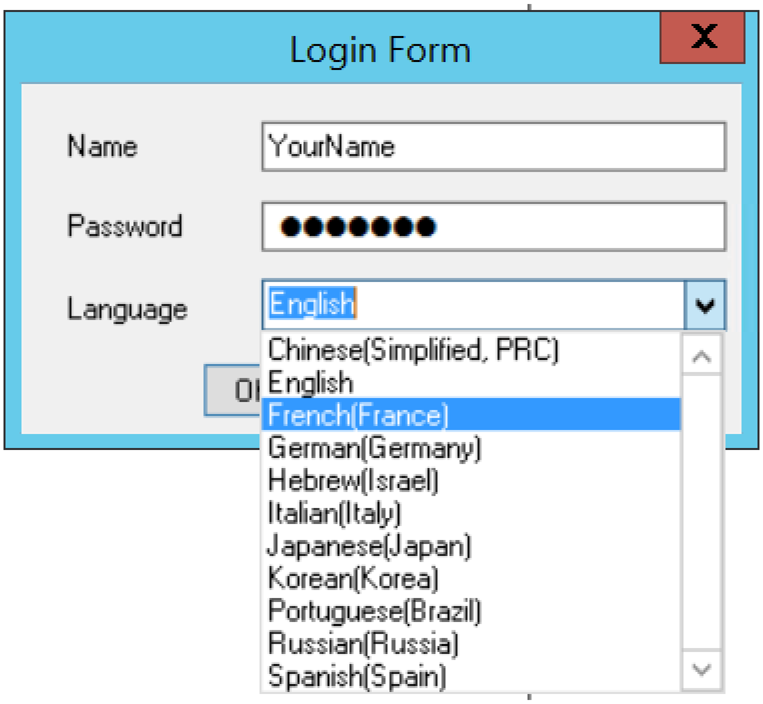
To localize a Power SCADA application:

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**.



Login Form

Name: YourName

Password: ●●●●●●●

Language: English

NOTE: To correctly display Basic Reports and LiveView, set your desired localized language in the browser.

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

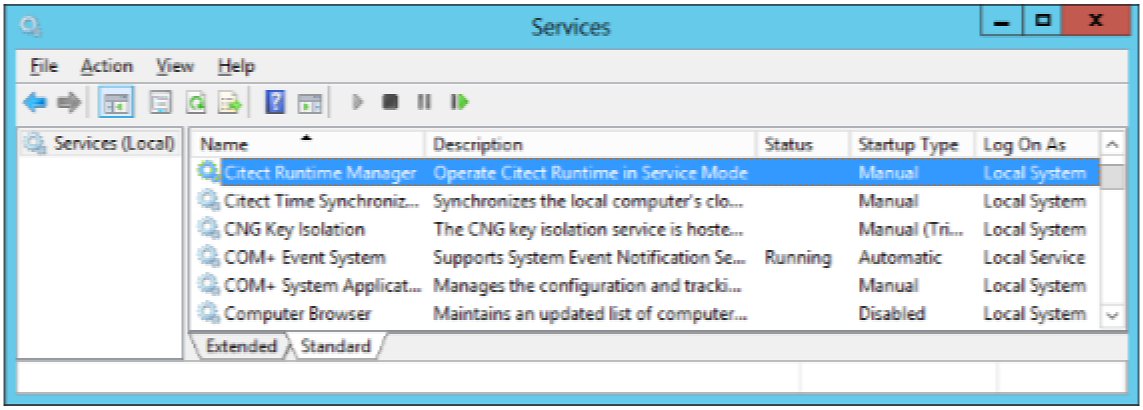
The text you enter in the Comment field is added to the default language file, named `English.DBF`. After the project is compiled, this file is located in ...\Documents and Settings\All Users\Application Data\Schneider Electric\Power SCADA Operation\v2020 R2\User\<*your project*>. `English.dbf` contains 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:



Name	Description	Status	Startup Type	Log On As
Citect Runtime Manager	Operate Citect Runtime in Service Mode		Manual	Local System
Citect Time Synchroniz...	Synchronizes the local computer's clo...		Manual	Local System
CNG Key Isolation	The CNG key isolation service is hoste...		Manual (Tri...	Local System
COM+ Event System	Supports System Event Notification Se...	Running	Automatic	Local Service
COM+ System Applicat...	Manages the configuration and tracki...		Manual	Local System
Computer Browser	Maintains an updated list of computer...		Disabled	Local System

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.

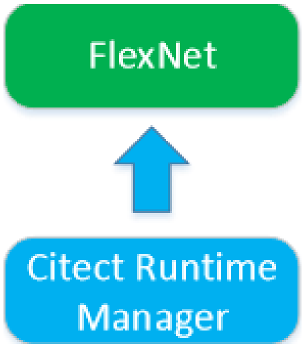
Running the Power SCADA Operation Windows service automatically provides the following benefits:

- 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”
```

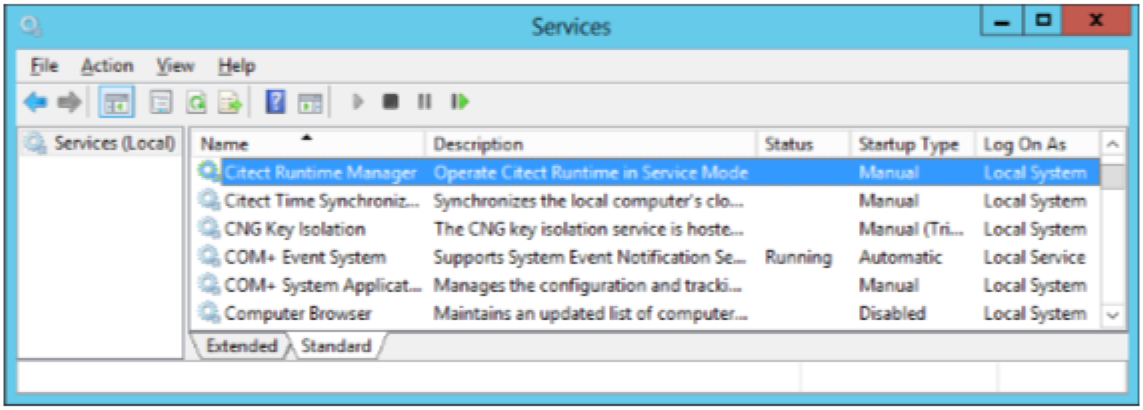
NOTE: You must insert a space character after `depend=`. You must also enclose service names in quotes, which includes spaces in their names.

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.



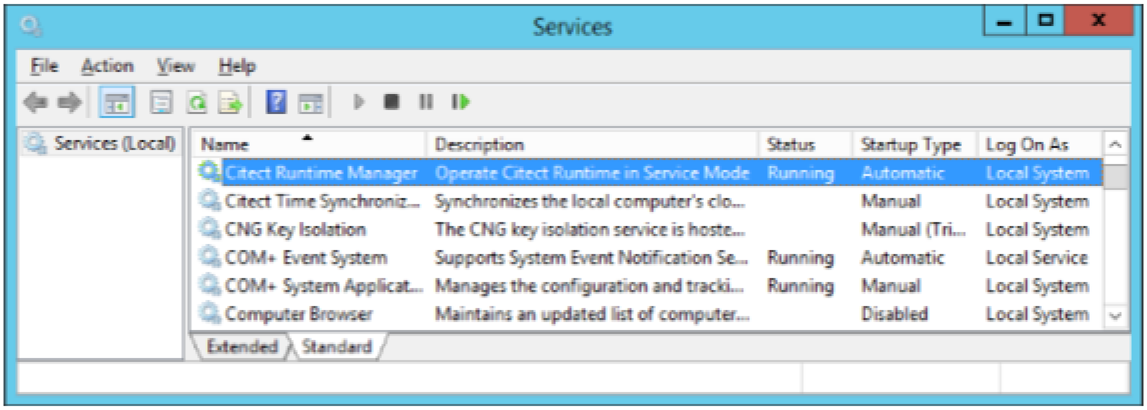
Name	Description	Status	Startup Type	Log On As
Citect Runtime Manager	Operate Citect Runtime in Service Mode		Manual	Local System
Citect Time Synchroniz...	Synchronizes the local computer's clo...		Manual	Local System
CNG Key Isolation	The CNG key isolation service is hoste...		Manual (Tri...	Local System
COM+ Event System	Supports System Event Notification Se...	Running	Automatic	Local Service
COM+ System Applicat...	Manages the configuration and tracki...		Manual	Local System
Computer Browser	Maintains an updated list of computer...		Disabled	Local System

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.



Name	Description	Status	Startup Type	Log On As
Citect Runtime Manager	Operate Citect Runtime in Service Mode	Running	Automatic	Local System
Citect Time Synchroniz...	Synchronizes the local computer's clo...		Manual	Local System
CNG Key Isolation	The CNG key isolation service is hoste...		Manual (Tri...	Local System
COM+ Event System	Supports System Event Notification Se...	Running	Automatic	Local Service
COM+ System Applicat...	Manages the configuration and tracki...	Running	Manual	Local System
Computer Browser	Maintains an updated list of computer...		Disabled	Local System

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\v2020 R2\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 693.
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 617

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

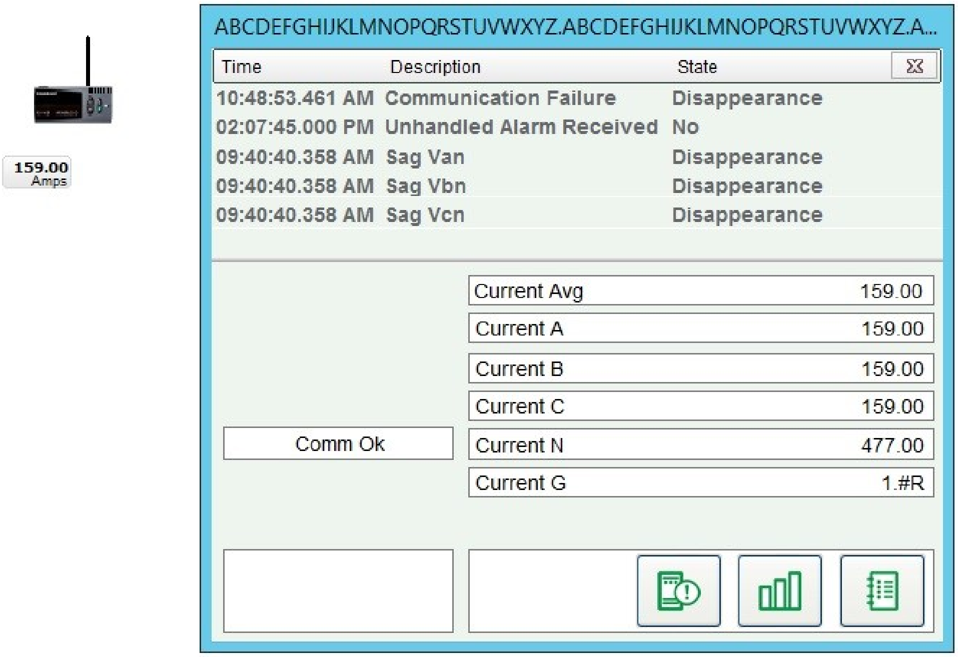
After the system is configured and communicating in runtime mode, verify that all devices are communicating correctly. All devices that are not communicating will trigger "Communication Failure" alarms, which can be seen in the active alarm log screen. For more information on how to add this screen to the project, see "Adding Alarm Pages" on page 315. 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:



ABCDEFGHIJKLMNOPQRSTUVWXYZ.ABCDEFGHIJKLMNOPQRSTUVWXYZ.A...

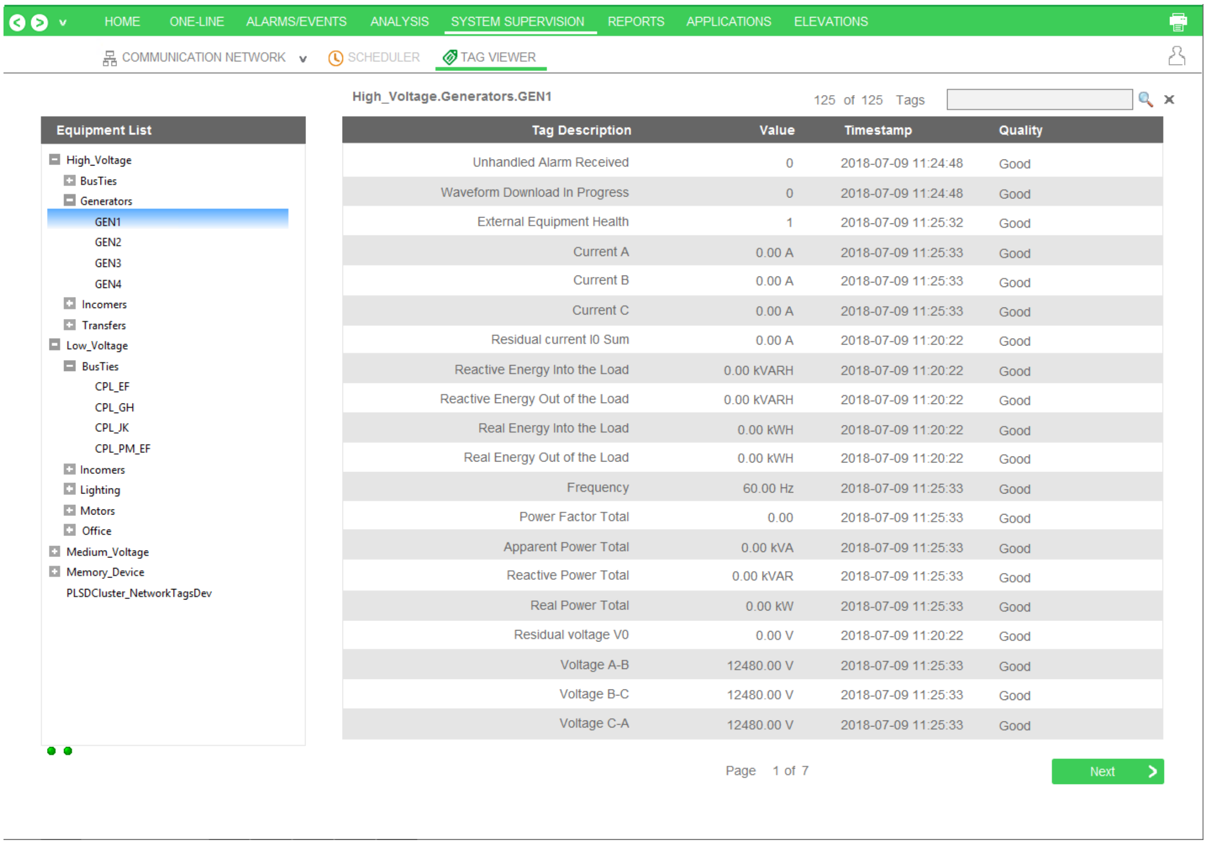
Time	Description	State
10:48:53.461 AM	Communication Failure	Disappearance
02:07:45.000 PM	Unhandled Alarm Received	No
09:40:40.358 AM	Sag Van	Disappearance
09:40:40.358 AM	Sag Vbn	Disappearance
09:40:40.358 AM	Sag Vcn	Disappearance

Comm Ok

Current Avg	159.00
Current A	159.00
Current B	159.00
Current C	159.00
Current N	477.00
Current G	1.#R

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.



High_Voltage.Generators.GEN1

125 of 125 Tags

Tag Description	Value	Timestamp	Quality
Unhandled Alarm Received	0	2018-07-09 11:24:48	Good
Waveform Download In Progress	0	2018-07-09 11:24:48	Good
External Equipment Health	1	2018-07-09 11:25:32	Good
Current A	0.00 A	2018-07-09 11:25:33	Good
Current B	0.00 A	2018-07-09 11:25:33	Good
Current C	0.00 A	2018-07-09 11:25:33	Good
Residual current I0 Sum	0.00 A	2018-07-09 11:20:22	Good
Reactive Energy Into the Load	0.00 kVARH	2018-07-09 11:20:22	Good
Reactive Energy Out of the Load	0.00 kVARH	2018-07-09 11:20:22	Good
Real Energy Into the Load	0.00 kWH	2018-07-09 11:20:22	Good
Real Energy Out of the Load	0.00 kWH	2018-07-09 11:20:22	Good
Frequency	60.00 Hz	2018-07-09 11:25:33	Good
Power Factor Total	0.00	2018-07-09 11:25:33	Good
Apparent Power Total	0.00 kVA	2018-07-09 11:25:33	Good
Reactive Power Total	0.00 kVAR	2018-07-09 11:25:33	Good
Real Power Total	0.00 kW	2018-07-09 11:25:33	Good
Residual voltage V0	0.00 V	2018-07-09 11:20:22	Good
Voltage A-B	12480.00 V	2018-07-09 11:25:33	Good
Voltage B-C	12480.00 V	2018-07-09 11:25:33	Good
Voltage C-A	12480.00 V	2018-07-09 11:25:33	Good

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

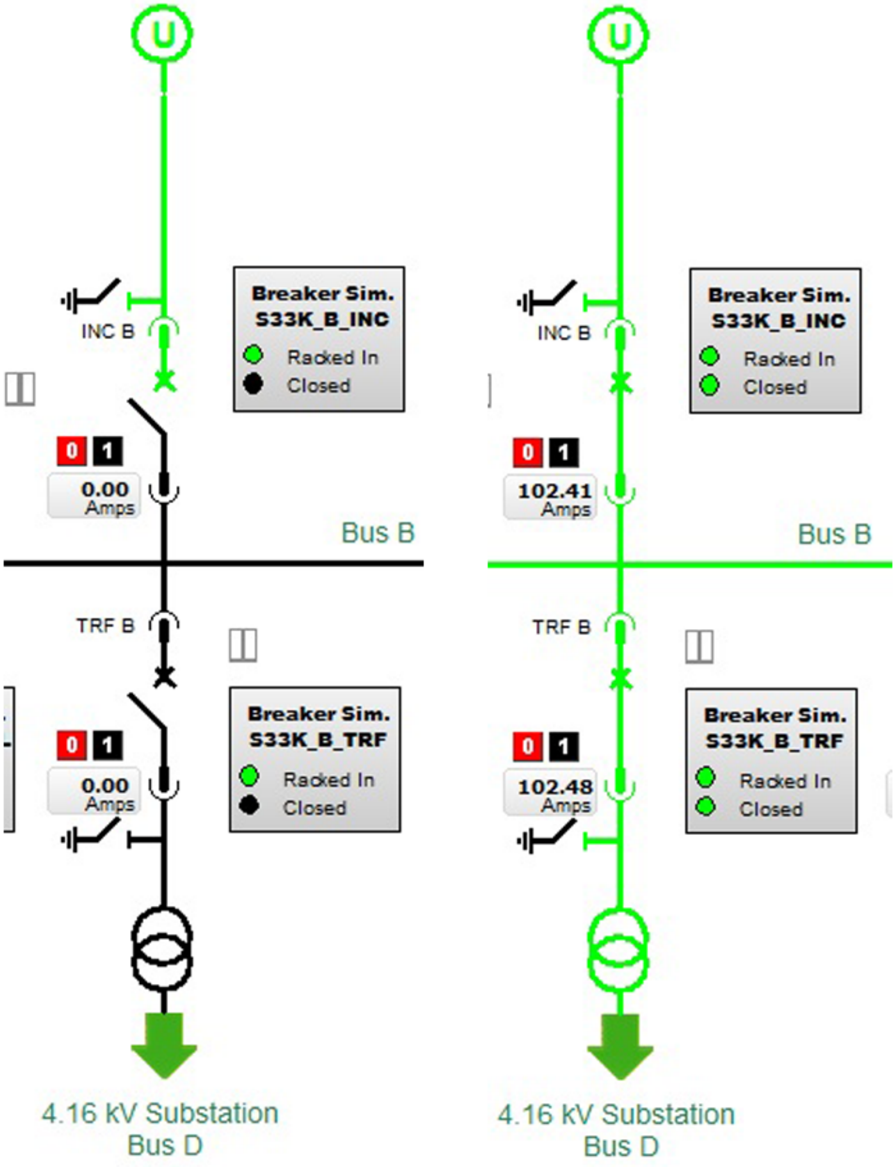
- Do not rely solely on the display of the graphic on the one-line.
- Use this procedure only during development, and not on a live deployed system.
- Before energizing or de-energizing any equipment from this software, verify that all personnel are a safe distance from all energized equipment.
- Before 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 808.

Distributed systems

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

Topic	Description
"Setting up more than two I/O Servers per cluster" on page 622	How to add multiple I/O Servers per cluster.
"Set up the Advanced Reporting and Dashboards Server" on page 627	How to configure the Advanced Reporting and Dashboards Module.
"Configure the Power SCADA Anywhere Server" on page 641	Information on how to configure Power SCADA Anywhere.
"EcoStruxure Web Services setup" on page 644	Information on how to configure EcoStruxure Web Services.
"Time synchronization" on page 645	Considerations for synchronizing time across a distributed system.
"Time zone settings" on page 646	Information on how distributed time zones are handled in Power SCADA Operation.
"OFS system time stamping" on page 647	How to configure OPC Factory Server (OFS) time stamping in Power SCADA Operation.
"Configure Power SCADA Operation as an OPC-DA Server" on page 669	How to configure an OPC-DA Server.
"Configure Power SCADA Operation as an OPC-DA Client" on page 670	How to configure an OPC-DA Client.
"Multi-site architectures" on page 671	Examples and configuration guidelines for multi-site (multi-clustered) systems.

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 and password or a Power Monitoring Expert username and password. This allows the Citect user to access external applications, such as Dashboards, using an SSO user password from Power Monitoring Expert.

> **NOTE**: SSO only works with Control Clients. From within a View-only Client, you must log in to PME to view Dashboards and Reports.

To set up single sign-on, see:

- "Add Single Sign-On Settings to Citect.ini" on page 624
- "Configure Single Sign-On (SSO)" on page 625

For information on using SSO with the Advanced Reporting and Dashboards Module, see "Add Advanced Dashboards and Reports into Web Applications" on page 628.

Add Single Sign-On Settings to Citect.ini

Open the Citect.ini file (typically in C:\ProgramData\Schneider Electric\Power SCADA Operation\v2020 R2\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 2020
- `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**.
3. Click **Topology > Edit > I/O Servers**, and change **Allow RPC** to **TRUE** for at least one I/O server per machine.

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 624

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 built 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 in this section if your Power SCADA system includes the Advanced Reporting and Dashboards Module.

To set up the Advanced Reporting and Dashboards Server:

1. "Install Advanced Reporting and Dashboards Module" on page 627
2. "Add Devices on Advanced Reporting and Dashboards server" on page 627
3. "Configure ETL" on page 628
4. (Optional) "Add Advanced Dashboards and Reports into Web Applications" on page 628
5. (Optional) "Add advanced dashboards and reports into Power SCADA Runtime" on page 632

The detailed steps are provided in the following topics.

Install Advanced Reporting and Dashboards Module

The installer 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.

Add Devices on Advanced Reporting and Dashboards server

At this stage of the configuration, 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 2020 – 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 625 for more information.

Configure ETL

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.

For more information on the ETL Administration Tool, see "ETL for Power SCADA Operation" on page 968.

Add Advanced Dashboards and Reports into Web Applications

NOTE: The following instructions are for adding Advanced Dashboards and Reports into the PSO Web Applications. To add Advanced Dashboards and Reports into the Power SCADA Runtime, see "Add advanced dashboards and reports into Power SCADA Runtime" on page 632.

To add Advanced Dashboards and Reports into Web Applications:

1. "Synchronize the PSO and Advanced Reporting and Dashboards Module users" on page 629
2. "Add PSO to Advanced Reporting and Dashboards Module whitelist" on page 629
3. "Specify the PSO Web Applications server location" on page 630
4. "Add a tab to PSO Web Applications" on page 630

The detailed steps are provided in the following topics.

TIP: For information on troubleshooting web errors in the Advanced Reporting and Dashboards Module, see "Web Applications" on page 817 in the Troubleshooting chapter.

Synchronize the PSO and Advanced Reporting and Dashboards Module users

To use Single Sign-On (SSO) with the Advanced Reporting and Dashboards Module, you must also create the users in the Advanced Reporting and Dashboards Module. For standard users that will use SSO, the login name and password must match in PSO and in the Advanced Reporting and Dashboards Module. For Active Directory users, the user or group must be added to the Advanced Reporting and Dashboards Module.

To synchronize the PSO and Advanced Reporting and Dashboards Module users:

1. On the Advanced Reporting Server, log into **Web Applications** with sufficient privileges.
2. Open **Settings > Users > User Manager**.
3. Add the users and/or groups.
4. For standard users, make sure the password matches the one used in PSO.

   NOTE: If the Advanced Reporting Server is set up with user groups, add the newly added users and/or groups to a user group.

   NOTE: If running a multilingual system, make sure the user's language is the same in both PSO and Advanced Reports.

For more information on the User Manager, see the Power Monitoring Expert System Guide.

Add PSO to Advanced Reporting and Dashboards Module whitelist

NOTE: This step is required only if the Power SCADA Server and the Advanced Reporting Server are hosted on different machines.

Add the PSO hostname to the whitelist in Advanced Reporting and Dashboards Module so that Advanced Reporting and Dashboards Module can be added to PSO Web Applications.

To add PSO to the Advanced Reporting and Dashboards Module whitelist:

1. On the Advanced Reporting Server, log into Web Applications with sufficient privileges.
2. Open **Settings -> Integrations -> Authorized Hosts**.
3. In **Hosts That Can Frame**, add the hostname(s) of the Power SCADA Server. For example, `https://pso.se.com` or `http://pso.se.com:8080`.
4. Click **Save**. The changes will take effect within a minute.

Specify the PSO Web Applications server location

Add a configuration setting in PSO Web Applications to specify the location of the Advanced Reporting server.

To specify the PSO Web Applications server location:

1. Locate the configuration file `HmiConfiguration.json` under Program Files. For example, `C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020 R2\Applications\Web\SystemDataService\App_Data\Configuration\HmiConfiguration.json`.

2. Add the following item (after confirming it isn't already there) to the `HmiConfiguration.json` file.

   ```
   {
    "ItemType": "Integration",
    "ItemIdentifier": "PME",
    "ItemKey": "HttpRoot",
    "OwnerIdentityId": "GlobalSetting",
    "Value": "ADVANCED-REPORTS-URL"
   }
   ```

   Where `ADVANCED-REPORTS-URL` is the protocol, host name, and port (if non-standard) of the Advanced Reporting Server as a user of PSO would see in their browser. For example: `https://pme.se.com`

   NOTE: The `Value` should not include a trailing slash or anything after the server name. For example: https://pme.se.com/web will result in a Permission Denied error.

3. Save and close `HmiConfiguration.json`.

Add a tab to PSO Web Applications

You can add any custom tab to PSO Web Applications. You can add tabs that show the entire Advanced Reporting and Dashboards Module. You can also add tabs that show a specific WebReach diagram or a specific report.

Prerequisites

- The Diagram URL for a specific device. See "Getting the device name and testing the WebReach Diagrams URL" on page 633 for details.
- The report ID. See "Get the Advanced Reports Report ID" on page 632 for details.

To add a tab to the PSO Web Applications:

1. Locate the configuration file `ApplicationMenuConfig.json` under Program Files. For example, `C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020 R2\Applications\Web\SystemDataService\App_Data\Configuration\ApplicationMenuConfig.json`.

2. Identify the relative URL for the application to show in the tab and convert it to an encoded URL.

   The following table lists example encoded URLs:

Advanced Reporting and Dashboards Module	Example Encoded URL
Dashboards	/psodataservice/pme/auth?returnUrl=%2fdashboards
Reports	/psodataservice/pme/auth?returnUrl=%2freporter
Diagram	/psodataservice/pme/auth?returnUrl=%2Fion%2Fdefault.aspx%3Fdgm%3DOPEN_TEMPLATE_DIAGRAM%26HideBackToNetwork%3D1%26displayNode%3DDemo.Vic7650
Specific report	/psodataservice/pme/auth?returnUrl=%2Freporter%2FDefault.aspx%23lib%2Faaabf223-d776-4919-b110-2f07abc14768

   For more information on encoded URLs, see https://www.urlencoder.org.

3. Add the following entry to the `ApplicationMenuConfig.json` file, replacing the example values highlighted in yellow with the relevant values.

   Advanced Reporting and Dashboards Module:

```
{
 "Id": "PmeDashboards",
 "Description": "PME Dashboards",
 "DisplayName": "Dashboards",
 "ResourceSet": null,
 "Enabled": true,
 "Target": "/psodataservice/pme/auth?returnUrl=%2fdashboards",
 "IsFactoryApplication": false,
 "RequiredPrivilege": null
}
```

   The following table describes the fields in the JSON file:

Field	Description
`Id`	A relevant and unique id for the tab. **NOTE**: The `Id` cannot contain spaces.
`Description`	A description of the tab.

Field	Description
`DisplayName`	The text that will display in the user interface for the new tab.
`Target`	The encoded URL.

4. Save and close `ApplicationMenuConfig.json`.

Add advanced dashboards and reports into Power SCADA Runtime

The following links provide instructions on how to add links to Advanced Dashboards and Reports into the Power SCADA Runtime.

Use the following links to find the content you are looking for:

Topic
"Add the WebReach Server Parameter" on page 632
"Get the Advanced Reports Report ID" on page 632
"Getting the device name and testing the WebReach Diagrams URL" on page 633
"Add the Advanced Reports Root Page Menu Item" on page 633
"Add Advanced Reports page menu items" on page 634
"Add the Dashboards Page Menu Item" on page 635
"Finish Advanced Reports Page Menu Items" on page 636
"Add a Menu Item to Launch a Web Diagram" on page 636
"Finish WebDiagram Page Menu Items" on page 637
"Add Web Diagrams to Equipment Popups" on page 637

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.

Getting the device name and testing the WebReach Diagrams URL

To display the diagram, determine the device name using SQL, and then test the URL in a browser.

To determine the device name:

1. In SQL Server Management Studio, select the ION_Network database.
2. Enter and run the following query:

   ```
   SELECT Name FROM dbo.device
   ```

3. Find the device name that you want to display.

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>
   ```

   Where <*servername*> is the name of the Power Monitoring Expert server, and <*devicename*> is the name you found in the previous step.

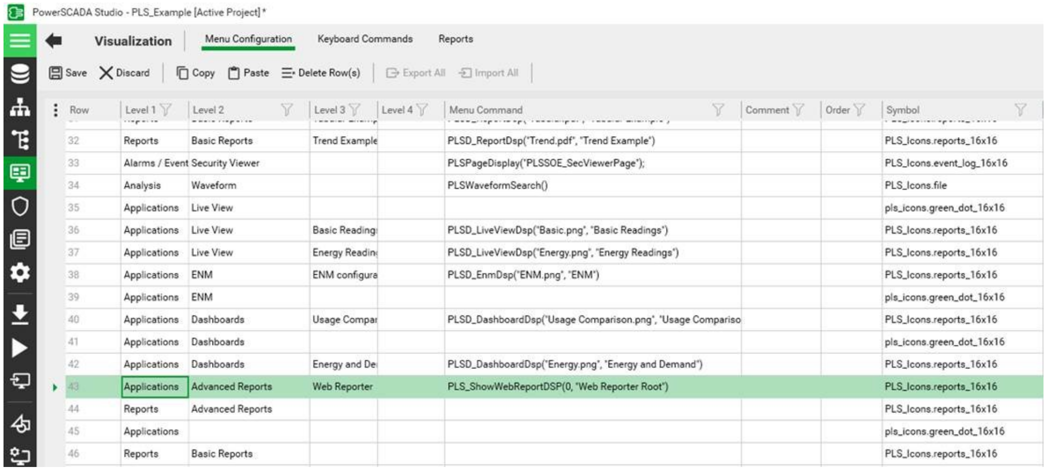
   For example, a real URL would look like:

   ```
    http://10.168.94.77/Ion/default.aspx?dgm=OPEN_TEMPLATE_
   DIAGRAM&node=TVD.7650
   ```

The device diagram displays in the browser window, and you can 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**.



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	Advanced Reports	Web Reporter		PLS_ShowWebReportDSP(0, "Web Reporter Root")			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

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

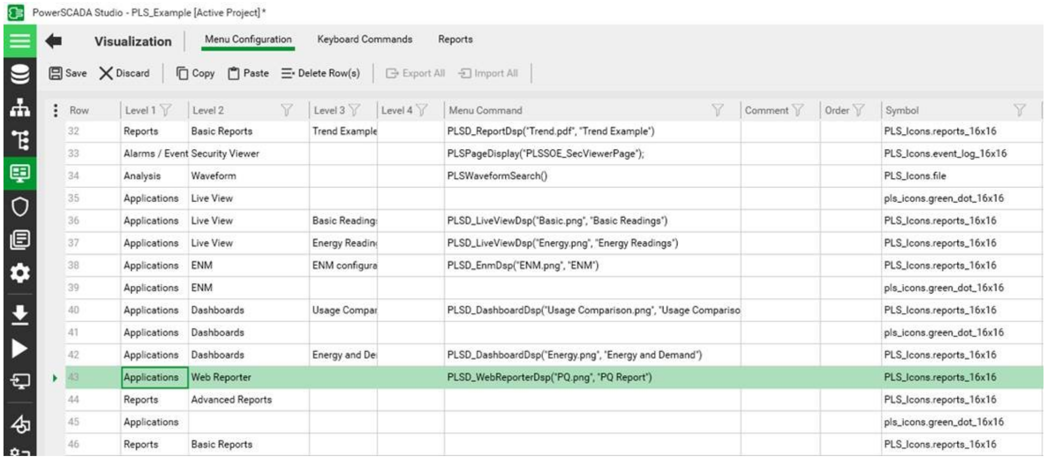
You can add menu items that navigate directly to a saved Advanced Report, such as a report for Energy Analysis over the last two months.

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**.



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

To determine the ReportID that you enter see "Get the Advanced Reports Report ID" on page 632. 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

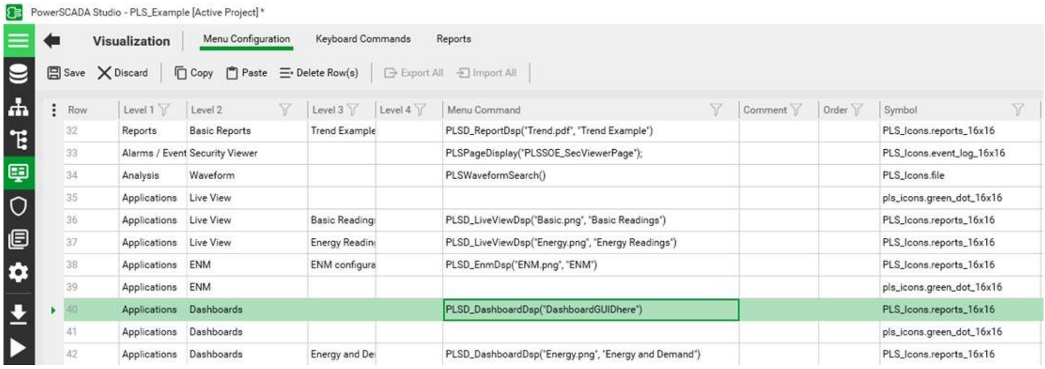
To add a menu item to launch a specific dashboard:

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:



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			PLSD_DashboardDsp("DashboardGUIDhere")			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

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:

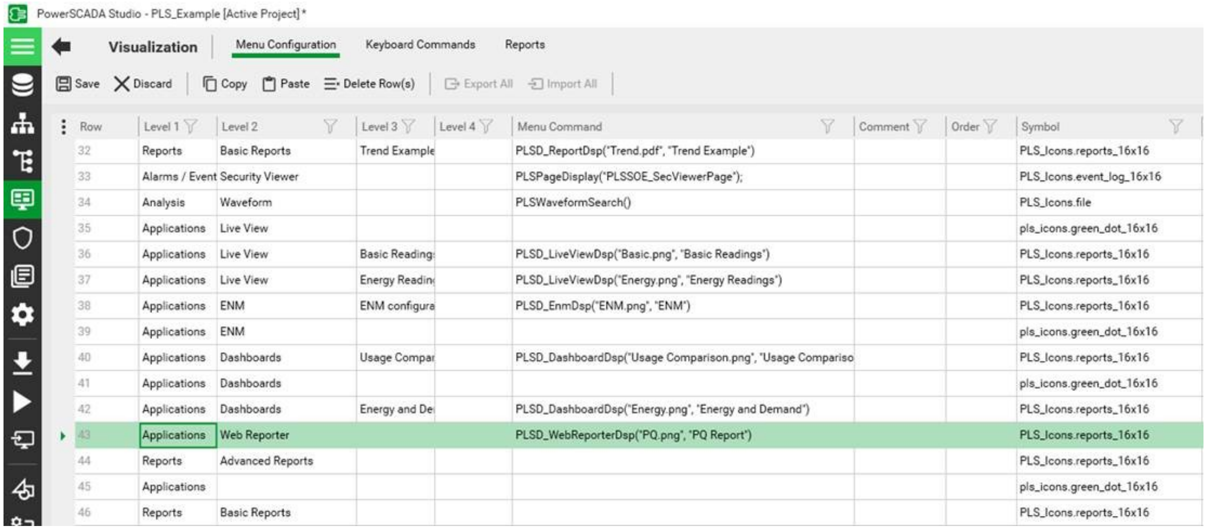
1. In Power SCADA Studio, click **Visualization** 🖥 **> Menu Configuration**.
2. Enter the call to the PLS_ShowDashboardDsp function (found in the PLS_Applications.ci file) with a custom page name and an empty dashboard ID: `PLS_ShowDashboardDsp("", "CustomPageName")`, or with no custom parameters to 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 632.

For example:



PowerSCADA Studio - PLS_Example [Active Project] *

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 637

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 "Getting the device name and testing the WebReach Diagrams URL" on page 633.

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.

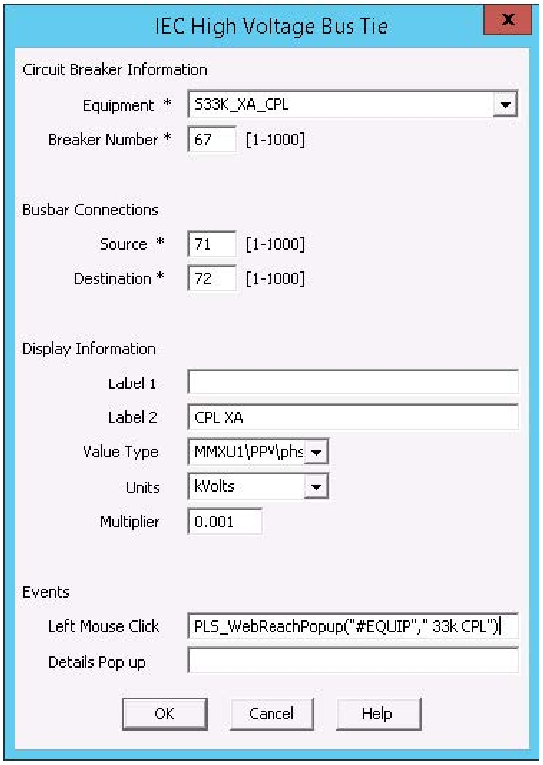
To launch the diagram from a meter genie equipment page:

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 following:



IEC High Voltage Bus Tie

Circuit Breaker Information

Equipment * S33K_XA_CPL

Breaker Number * 67 [1-1000]

Busbar Connections

Source * 71 [1-1000]

Destination * 72 [1-1000]

Display Information

Label 1

Label 2 CPL XA

Value Type MMXU1\PPV\phs

Units kVolts

Multiplier 0.001

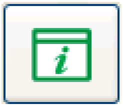
Events

Left Mouse Click PLS_WebReachPopup("#EQUIP"," 33k CPL")

Details Pop up

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.

Add EcoStruxure Building Operation in Web Applications

To display EcoStruxure Building Operation in the Web Applications, you must complete the following tasks:

- "Step 1: Synchronize PSO and EBO users" on page 639
- "Step 2: Allow EcoStruxure Building Operation to be embedded in Web Applications" on page 639
- "Step 3: Specify the EcoStruxure Building Operation server location" on page 639
- "Step 4: Add a tab to PSO Web Applications" on page 640

NOTE: When integrating EcoStruxure Building Operation in Web Applications, set the inactivity timeout in EBO to be higher than the timeout value in PSO. For more information on inactivity timeout, see "Session timeout" on page 579 (for PSO), and search for 'Automatic Logoff' in EcoStruxure Building Operation help.

Step 1: Synchronize PSO and EBO users

For standard users who will use SSO, the login name and password must match in PSO and in the EcoStruxure Building Operation.

NOTE: MD5 digest authentication is supported; LDAP is not supported.

To synchronize PSO and EBO users:

- In EcoStruxure Building Operation Security Settings Control Panel, select **Allow authentication with MD5 hash**.

NOTE: If running a multilingual system, make sure the user's language is the same in both PSO and EcoStruxure Building Operation.

Step 2: Allow EcoStruxure Building Operation to be embedded in Web Applications

To allow EcoStruxure Building Operation to be embedded in Web Applications:

1. In EcoStruxure Building Operation Security Settings, select **Enable WebStation to be embedded into another website**.
2. For improved security, enter the Power SCADA Server in **Website to allow access to WebStation when embedded**.

Step 3: Specify the EcoStruxure Building Operation server location

Add a configuration setting in PSO Web Applications to specify the location of the EcoStruxure Building Operation server.

To add a configuration setting in PSO Web Applications:

1. Locate the configuration file `HmiConfiguration.json` under Program Files. For example:
   ```
   C:\Program Files (x86)\Schneider Electric\Power
   SCADA Operation\v2020 R2\Applications\Web\SystemDataService\App_
   ```

Data\Configuration\HmiConfiguration.json.

2. Add the following item (after confirming it is not already there) to the HmiConfiguration.json file:

```
{
 "ItemType": "Integration",
 "ItemIdentifier": "EBO",
 "ItemKey": "HttpRoot",
 "OwnerIdentityId": "GlobalSetting",
 "Value": "EBO-SERVER-URL"
}
```

   Where EBO-SERVER-URL is the protocol, host name, and port (if non-standard) of the EcoStruxure Building Operation Server as a user of PSO would see in their browser. It should not include a trailing slash. For example: https://ebo.se.com

3. Save and close HmiConfiguration.json.

Step 4: Add a tab to PSO Web Applications

To add a tab to the PSO Web Applications that displays EcoStruxure Building Operation:

1. Locate the configuration file ApplicationMenuConfig.json under Program Files. For example: C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020 R2\Applications\Web\SystemDataService\App_Data\Configuration\ApplicationMenuConfig.json.

2. Add the following entry to the ApplicationMenuConfig.json file, replacing the example values highlighted in yellow with the relevant values:

```
{
 "Id": "EBO",
 "Description": "EBO Server 1 System",
 "DisplayName": "EBO Server 1 System",
 "ResourceSet": null,
 "Enabled": true,
 "Target": "
/psodataservice/ebo/auth?returnUrl=%2F%3fkiosk#%2FServer%201%2FSystem",
 "IsFactoryApplication": false,
 "RequiredPrivilege": null
}
```

   The following table describes the fields in the JSON file:

Field	Description
Id	A relevant and unique id for the tab.
Description	A description of the tab.

Field	Description
`DisplayName`	The text that will display in the user interface for the new tab. The value `EBO_Application_Title` allows for the translation of 'EBO’
`ResourceSet`	Possible values for EcoStruxure Building Operation : - `HmiApplication` if using `EBO_Application_Title` - The name of the RESX file for translation - `null` to use `DisplayName` as the title
`Target`	The encoded URL.

3. Save and close `ApplicationMenuConfig.json`.

TIP: For information on troubleshooting web errors in the Advanced Reporting and Dashboards Module, see "Web Applications" on page 817 in the Troubleshooting chapter.

Configure the Power SCADA Anywhere Server

NOTICE

INOPERABLE SYSTEM

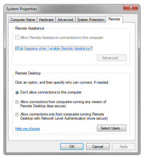
Ensure that you have received Power SCADA training and understand the importance of the Power SCADA Operation productivity tools and workflows.

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 built 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:



□ Allow Remote Assistance connections to this computer

Click an option, and then specify who can connect, if needed.

● Don't allow connections to this computer

○ Allow connections from computers running any version of Remote Desktop (less secure)

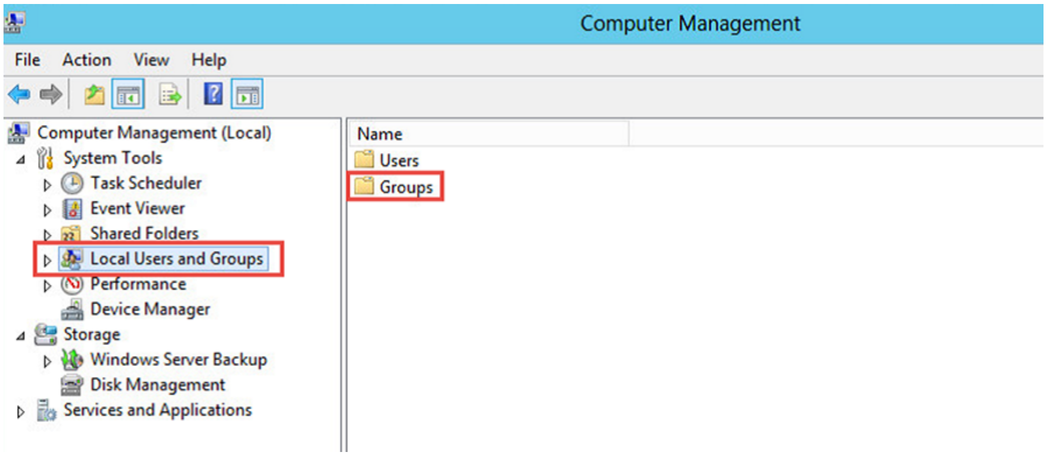
○ Allow connections only from computers running Remote Desktop with Network Level Authentication (more secure)

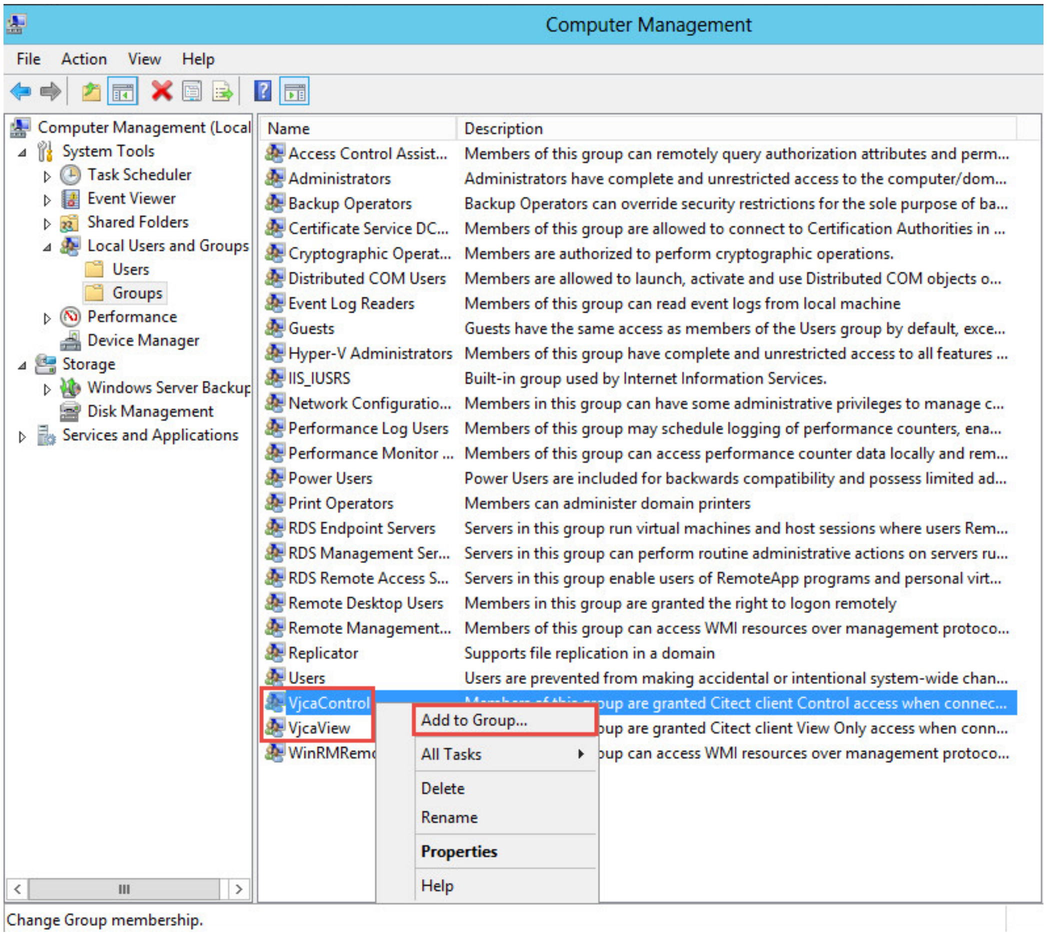
   b. Click **Allow Remote Assistance connections to this computer**.

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:

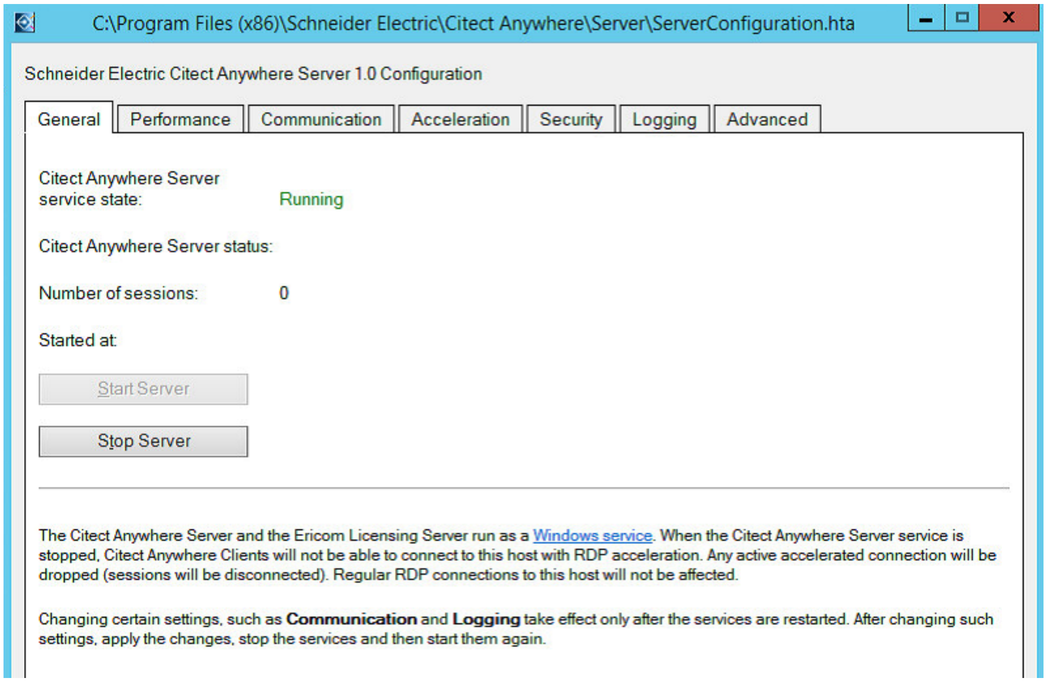




Name	Description
Access Control Assist...	Members of this group can remotely query authorization attributes and perm...
Administrators	Administrators have complete and unrestricted access to the computer/dom...
Backup Operators	Backup Operators can override security restrictions for the sole purpose of ba...
Certificate Service DC...	Members of this group are allowed to connect to Certification Authorities in ...
Cryptographic Operat...	Members are authorized to perform cryptographic operations.
Distributed COM Users	Members are allowed to launch, activate and use Distributed COM objects o...
Event Log Readers	Members of this group can read event logs from local machine
Guests	Guests have the same access as members of the Users group by default, exce...
Hyper-V Administrators	Members of this group have complete and unrestricted access to all features ...
IIS_IUSRS	Built-in group used by Internet Information Services.
Network Configuratio...	Members in this group can have some administrative privileges to manage c...
Performance Log Users	Members of this group may schedule logging of performance counters, ena...
Performance Monitor ...	Members of this group can access performance counter data locally and rem...
Power Users	Power Users are included for backwards compatibility and possess limited ad...
Print Operators	Members can administer domain printers
RDS Endpoint Servers	Servers in this group run virtual machines and host sessions where users Rem...
RDS Management Ser...	Servers in this group can perform routine administrative actions on servers ru...
RDS Remote Access S...	Servers in this group enable users of RemoteApp programs and personal virt...
Remote Desktop Users	Members in this group are granted the right to logon remotely
Remote Management...	Members of this group can access WMI resources over management protoco...
Replicator	Supports file replication in a domain
Users	Users are prevented from making accidental or intentional system-wide chan...
VjcaControl	Members of this group are granted Citect client Control access when connec...
VjcaView	oup are granted Citect client View Only access when conn...
WinRMRem	oup can access WMI resources over management protoco...

Change Group membership.

3. Ensure that the Citect Anywhere service is started. To confirm this, use the Server-Configuration for Citect Anywhere:



C:\Program Files (x86)\Schneider Electric\Citect Anywhere\Server\ServerConfiguration.hta

Schneider Electric Citect Anywhere Server 1.0 Configuration

Citect Anywhere Server service state: Running

Citect Anywhere Server status:

Number of sessions: 0

Started at:

The Citect Anywhere Server and the Ericom Licensing Server run as a Windows service. When the Citect Anywhere Server service is stopped, Citect Anywhere Clients will not be able to connect to this host with RDP acceleration. Any active accelerated connection will be dropped (sessions will be disconnected). Regular RDP connections to this host will not be affected.

Changing certain settings, such as **Communication** and **Logging** take effect only after the services are restarted. After changing such settings, apply the changes, stop the services and then start them again.

If the server is stopped, click **Start Server**.

Connect to Power SCADA Anywhere

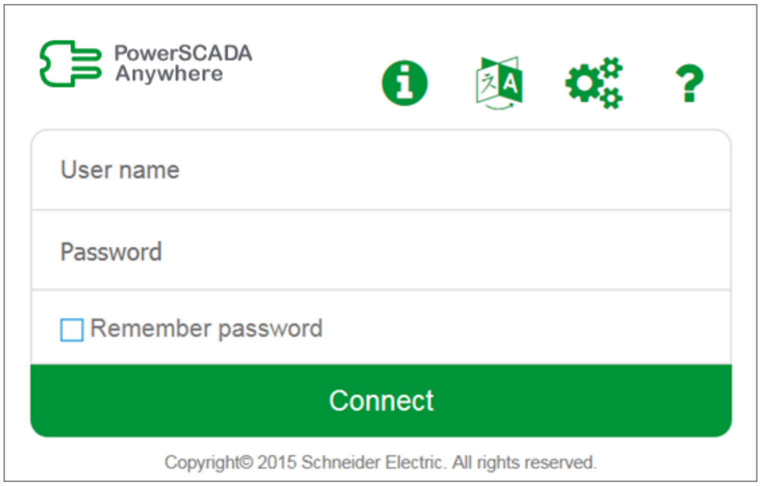
The following browsers are supported by Power SCADA Anywhere:

- 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 built 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 100 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 username/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\v2020 R2\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.

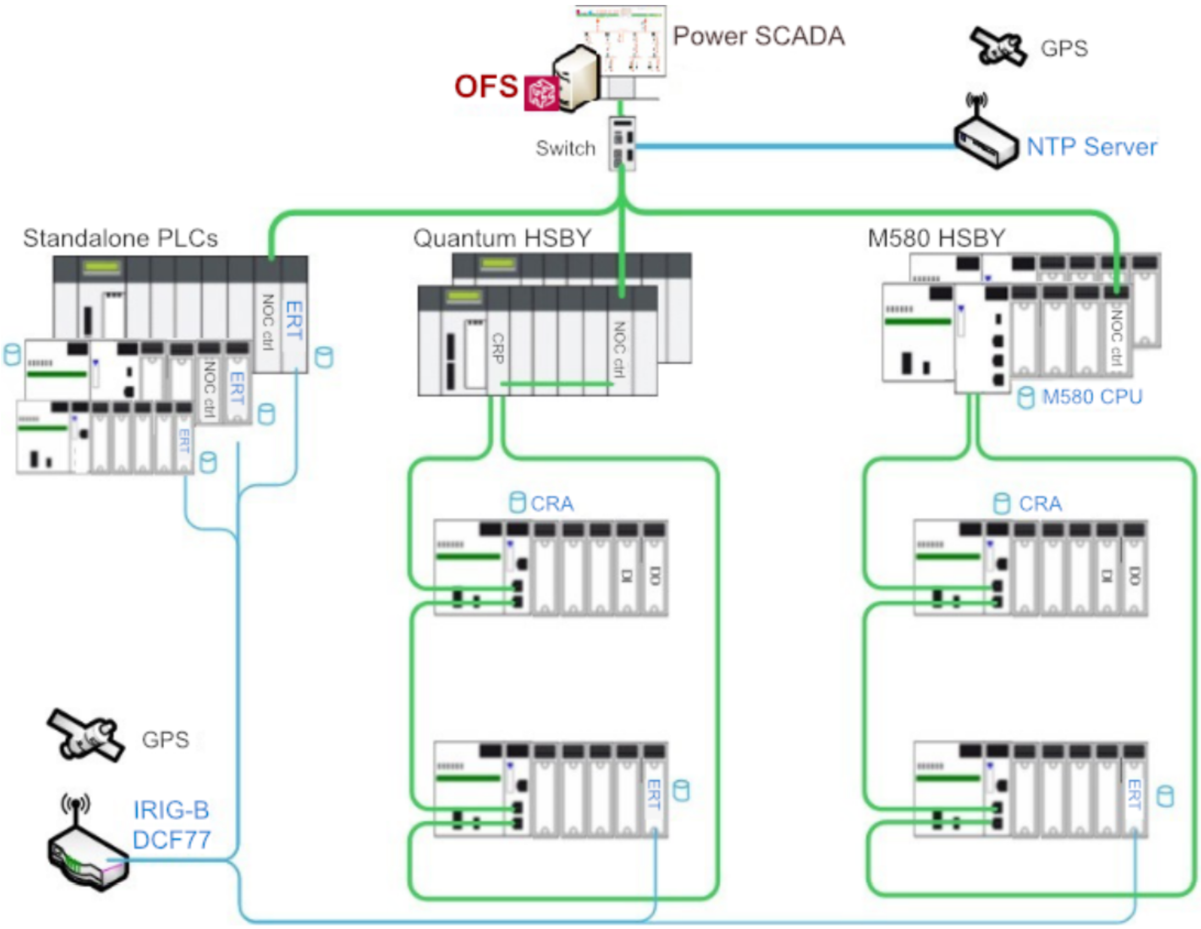
System Time Stamping helps the user analyze the source of abnormal behaviors in an automation system.

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

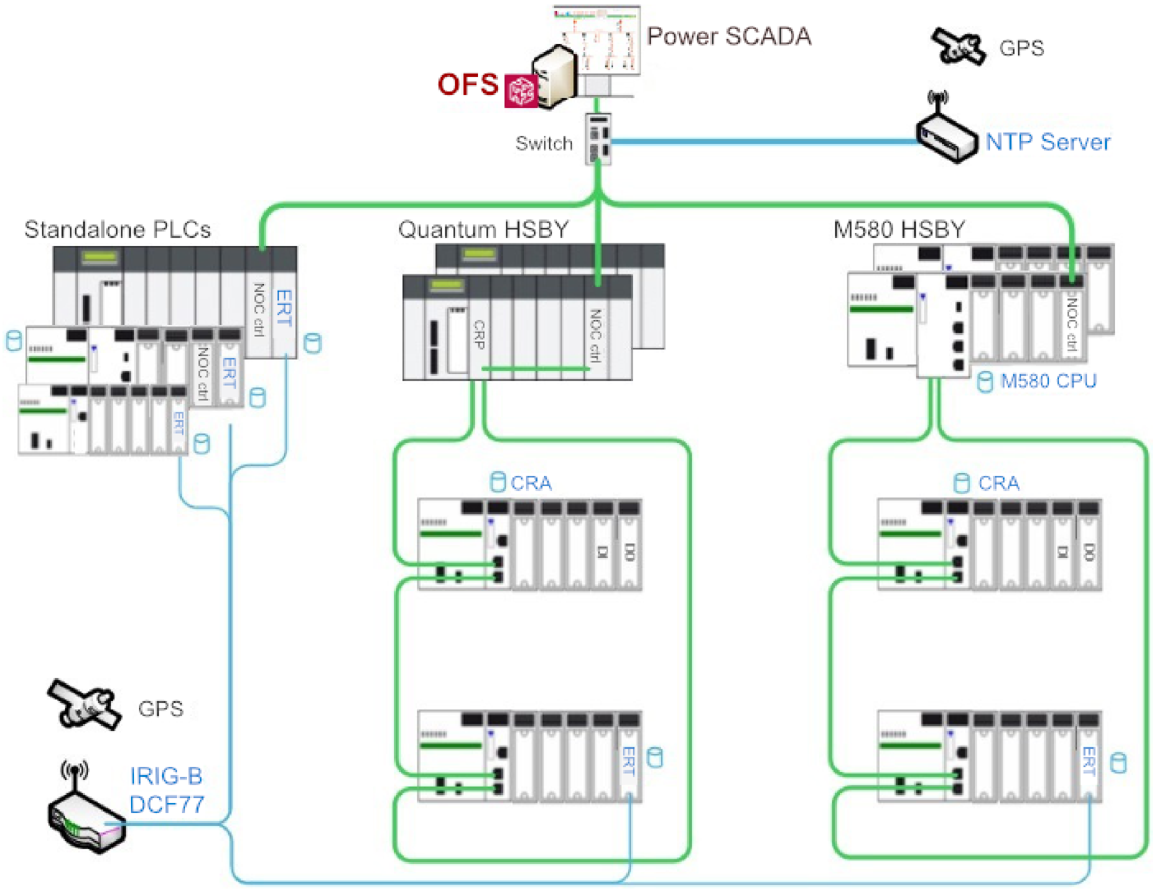
Hardware:

- 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

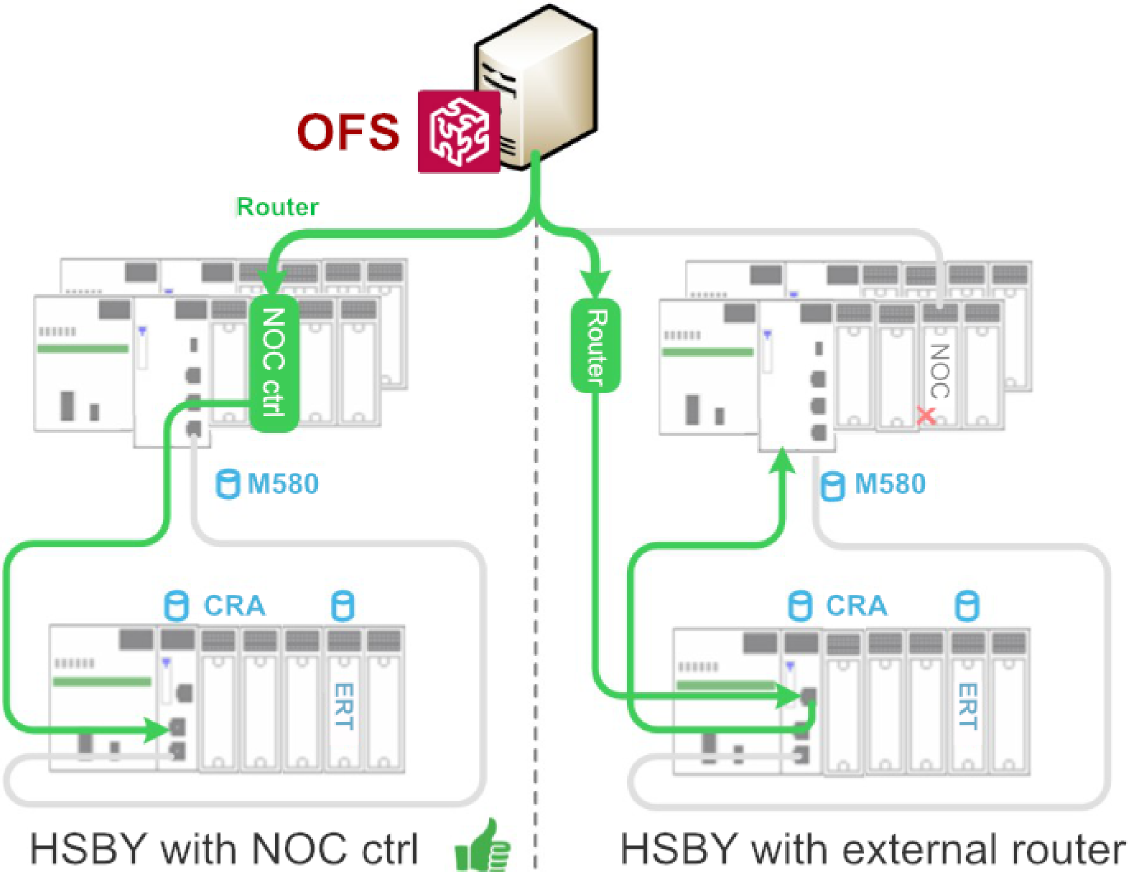
This chapter discusses how to select the architecture for the system time stamping application. We also introduce the method to synchronize the time clock between the multiple time sources and the time stamping modules, and list the time resolution with the different time stamping solutions.

Architecture selection

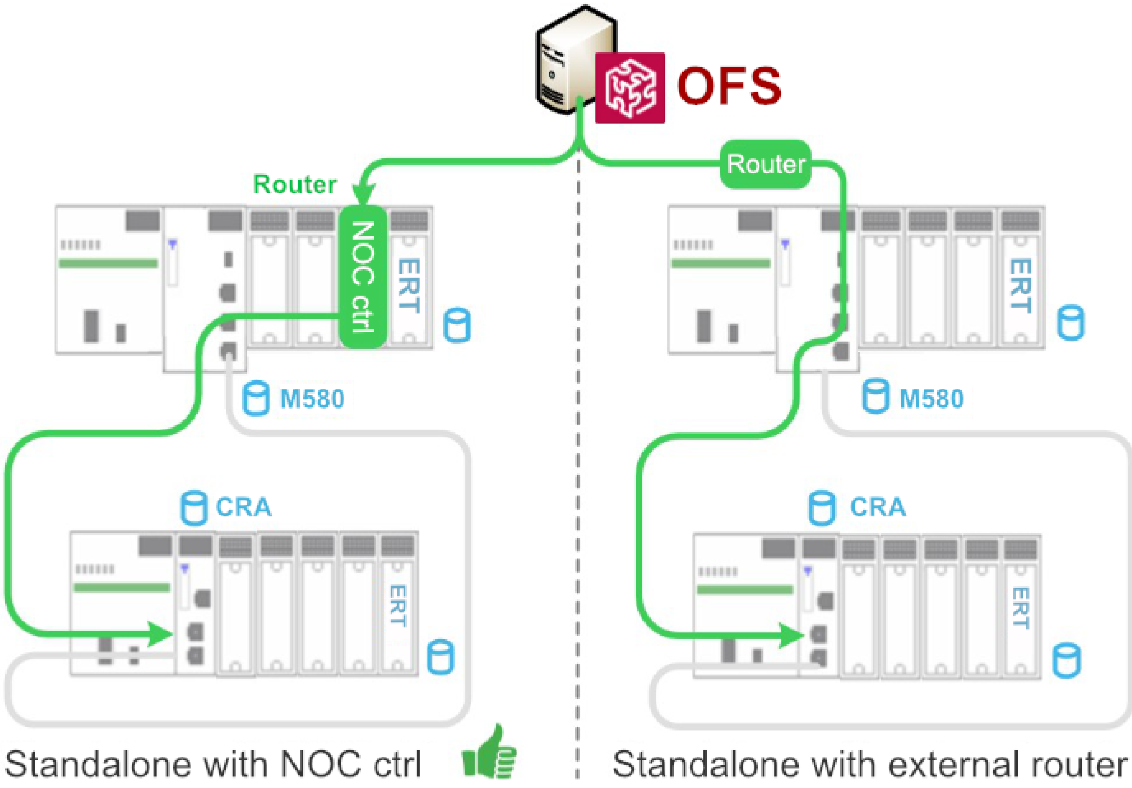


There are three types of modules which are supported by the system time stamping solution, including the M340/eX80ERT, eX80CRA, and M580 CPU. In the system time stamping architecture, OFS is used to automatically transfer the events from the time stamping module to the SCADA. As the time stamping module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

- In the standalone architecture, 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 to set up the connection between OFS and the time stamping module.

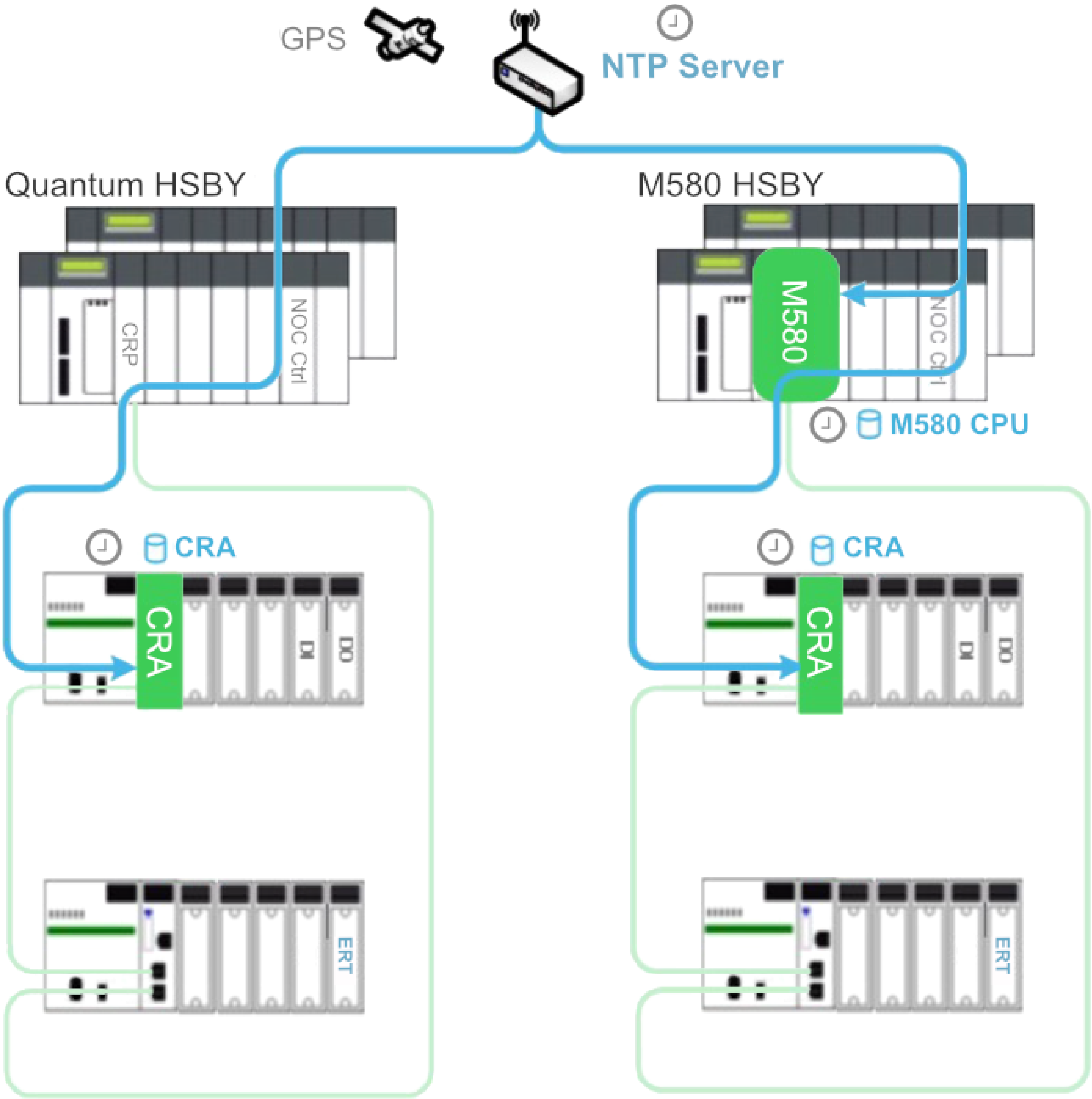


- In the HSBY architecture, either select the NOC control module as a router, or select a third-party router directly connected to the RIO network to set up the connection between OFS and the time stamping module.

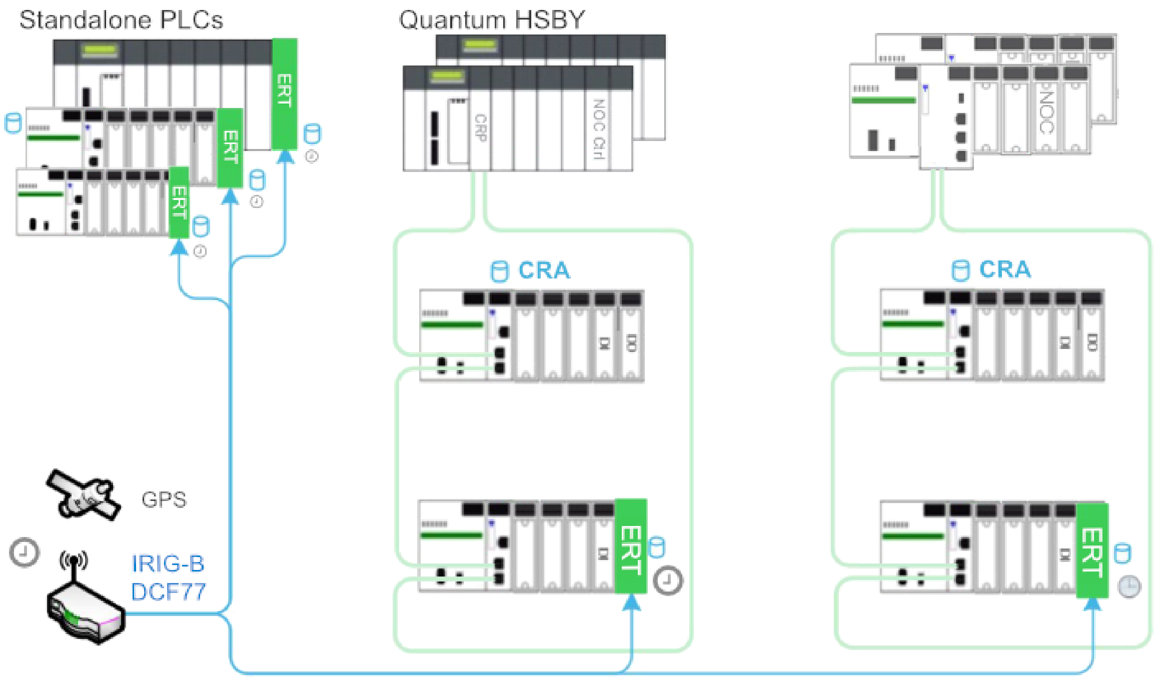


Time synchronization

- The external NTP server provides the time clock for the CPUs and CRAs. 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.

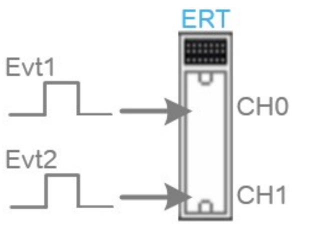
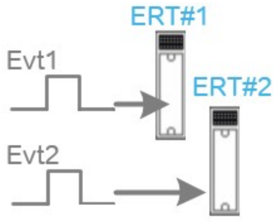
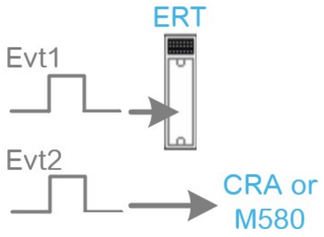
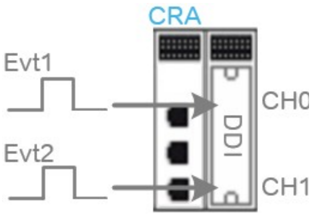
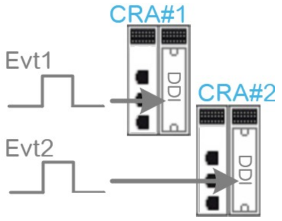
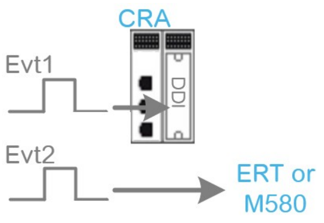
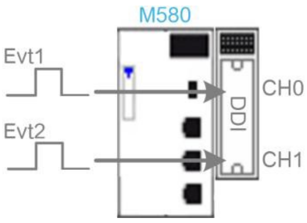
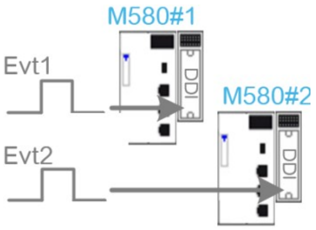
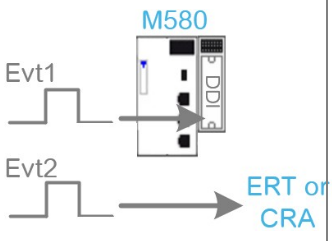


- The IRIG-B 004/5/6/7 or DCF77 signals generated by the GPS receiver are used to synchronize the ERT module's time clock.


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

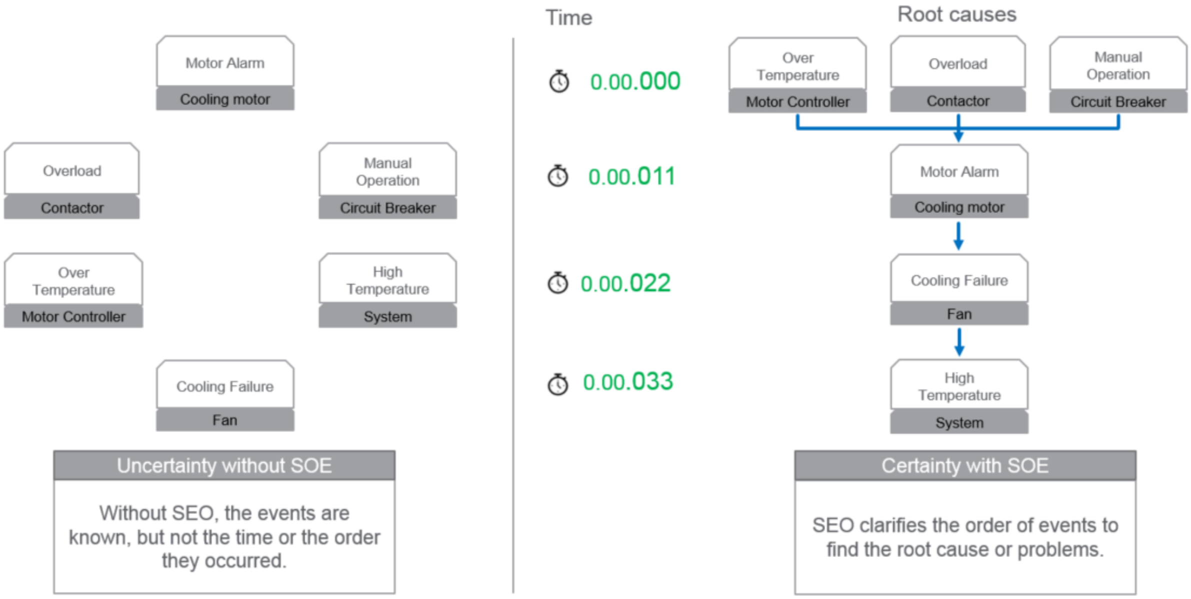
Event resolution

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

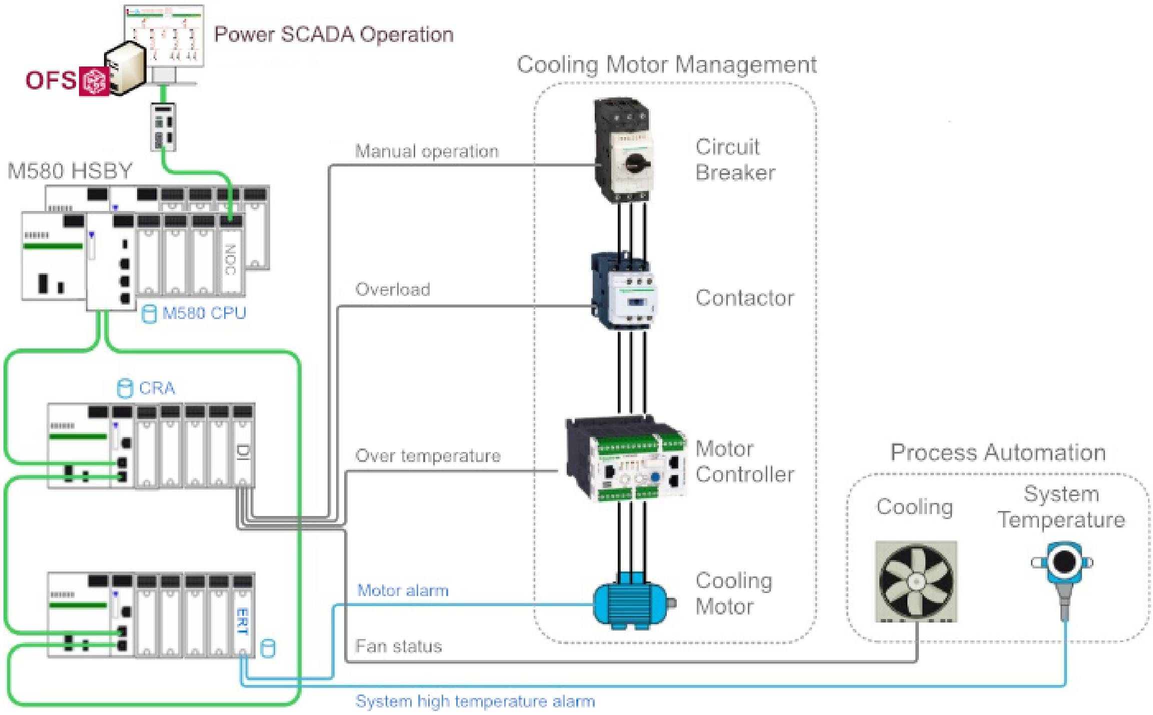
Design

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.

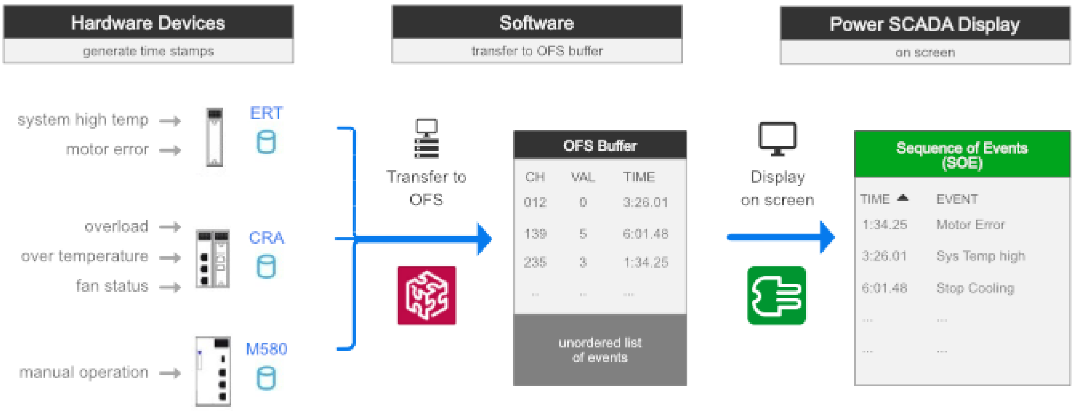


In the above diagram, a cooling control system includes a circuit breaker, a contactor, a motor controller, a motor, and a fan. The fan is used to cool down the system temperature when the temperature is higher than the pre-set value. For the process automation monitoring, some device statuses and process 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.

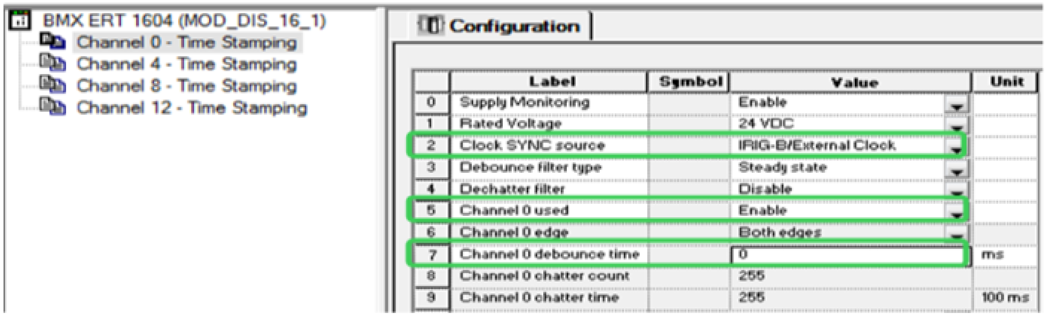
> **NOTE**: Its minimum value = 4 * number of events configured (including SOE_UNCERTAIN). If this configured value is too small, Unity Pro will show a build error and indicate the minimum events number in the message window.

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.

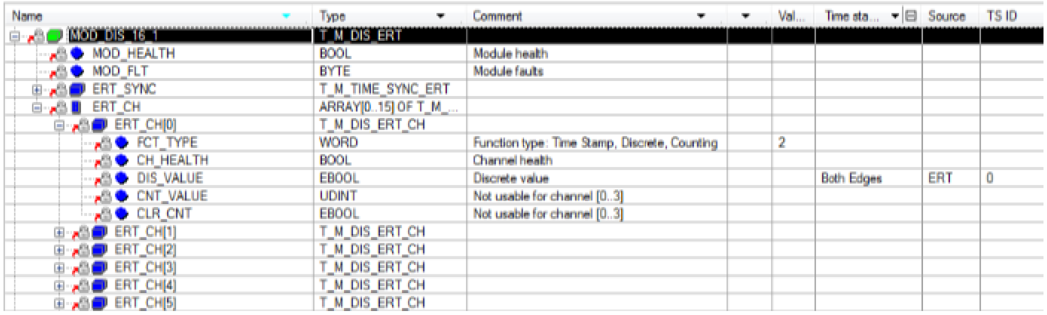
   For example:



BMX ERT 1604 (MOD_DIS_16_1)
- Channel 0 - Time Stamping
- Channel 4 - Time Stamping
- Channel 8 - Time Stamping
- Channel 12 - Time Stamping

	Label	Symbol	Value	Unit
0	Supply Monitoring		Enable	
1	Rated Voltage		24 VDC	
2	Clock SYNC source		IRIG-B/External Clock	
3	Debounce filter type		Steady state	
4	Dechatter filter		Disable	
5	Channel 0 used		Enable	
6	Channel 0 edge		Both edges	
7	Channel 0 debounce time		0	ms
8	Channel 0 chatter count		255	
9	Channel 0 chatter time		255	100 ms

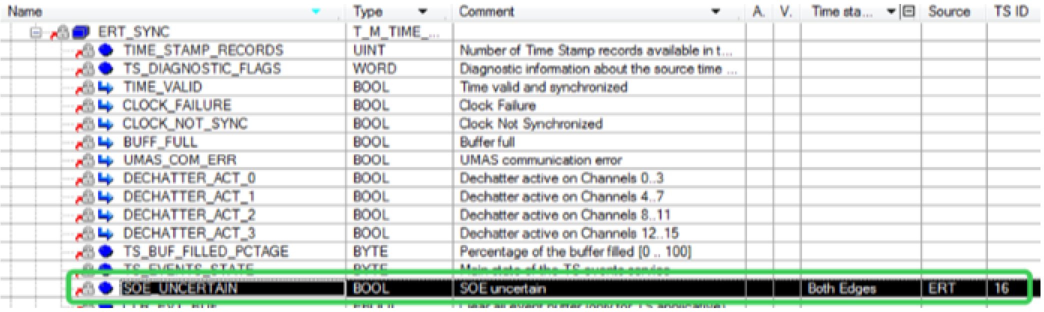
2. Open the module’s ‘Device DDT’ tab and then click **Goto detail**. All the elements within this Device DDT are shown in the Data Editor.



Name	Type	Comment		Val...	Time sta...	Source	TS ID
MOD_DIS_16_1	T_M_DIS_ERT						
MOD_HEALTH	BOOL	Module health					
MOD_FLT	BYTE	Module faults					
ERT_SYNC	T_M_TIME_SYNC_ERT						
ERT_CH	ARRAY[0..15] OF T_M_...						
ERT_CH[0]	T_M_DIS_ERT_CH						
FCT_TYPE	WORD	Function type: Time Stamp, Discrete, Counting		2			
CH_HEALTH	BOOL	Channel health					
DIS_VALUE	EBOOL	Discrete value			Both Edges	ERT	0
CNT_VALUE	UDINT	Not usable for channel [0..3]					
CLR_CNT	EBOOL	Not usable for channel [0..3]					
ERT_CH[1]	T_M_DIS_ERT_CH						
ERT_CH[2]	T_M_DIS_ERT_CH						
ERT_CH[3]	T_M_DIS_ERT_CH						
ERT_CH[4]	T_M_DIS_ERT_CH						
ERT_CH[5]	T_M_DIS_ERT_CH						

3. The parameter, SOE_UNCERTAIN, is activated by default, and is time stamped by both

edges.

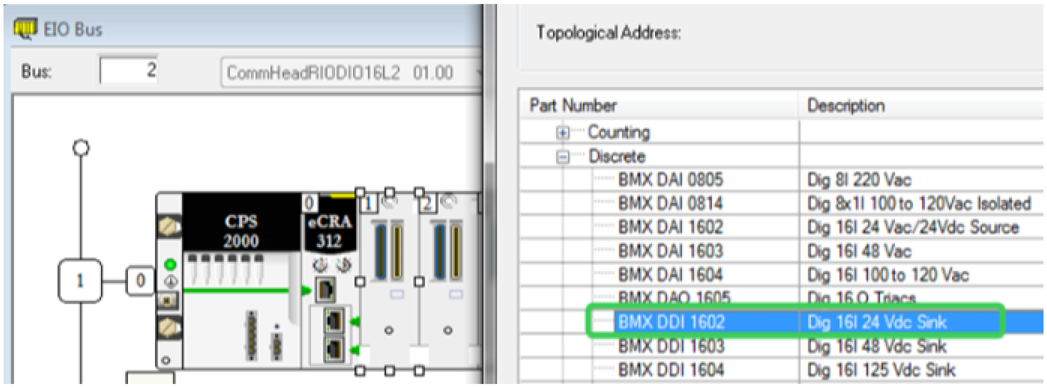


Name	Type	Comment	A.	V.	Time sta...	Source	TS ID
ERT_SYNC	T_M_TIME_...						
TIME_STAMP_RECORDS	UINT	Number of Time Stamp records available in t...					
TS_DIAGNOSTIC_FLAGS	WORD	Diagnostic information about the source time ...					
TIME_VALID	BOOL	Time valid and synchronized					
CLOCK_FAILURE	BOOL	Clock Failure					
CLOCK_NOT_SYNC	BOOL	Clock Not Synchronized					
BUFF_FULL	BOOL	Buffer full					
UMAS_COM_ERR	BOOL	UMAS communication error					
DECHATTER_ACT_0	BOOL	Dechatter active on Channels 0..3					
DECHATTER_ACT_1	BOOL	Dechatter active on Channels 4..7					
DECHATTER_ACT_2	BOOL	Dechatter active on Channels 8..11					
DECHATTER_ACT_3	BOOL	Dechatter active on Channels 12..15					
TS_BUF_FILLED_PCTAGE	BYTE	Percentage of the buffer filled [0 .. 100]					
TS_EVENTS_STATE	BYTE	Main state of the TS events service					
SOE_UNCERTAIN	BOOL	SOE uncertain			Both Edges	ERT	16

x80 CRA

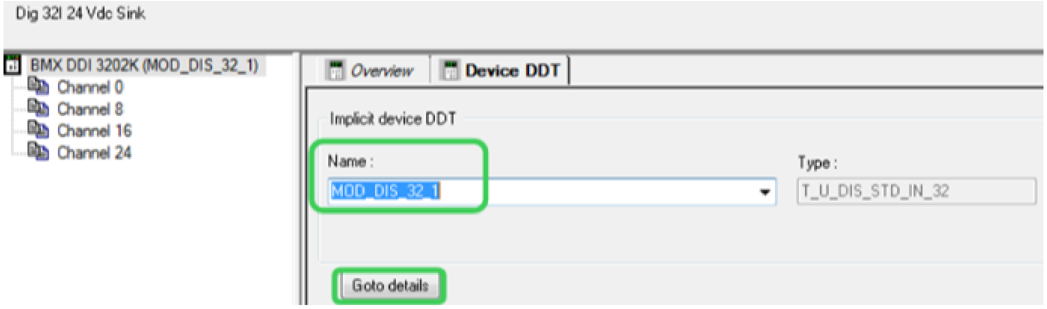
The x80 CRA module can be installed in the x80 remote I/O drops.

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:



Part Number	Description
Counting	
Discrete	
BMX DAI 0805	Dig 8I 220 Vac
BMX DAI 0814	Dig 8x1I 100 to 120Vac Isolated
BMX DAI 1602	Dig 16I 24 Vac/24Vdc Source
BMX DAI 1603	Dig 16I 48 Vac
BMX DAI 1604	Dig 16I 100 to 120 Vac
BMX DAO 1605	Dig 16 O Triacs
BMX DDI 1602	Dig 16I 24 Vdc Sink
BMX DDI 1603	Dig 16I 48 Vdc Sink
BMX DDI 1604	Dig 16I 125 Vdc Sink

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.



Dig 32I 24 Vdc Sink

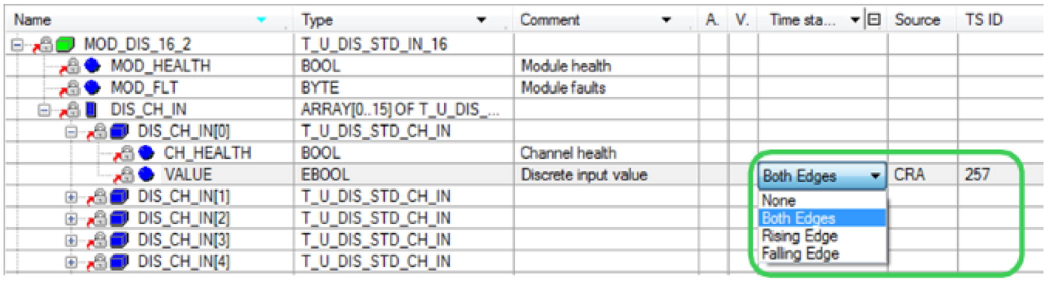
BMX DDI 3202K (MOD_DIS_32_1)
- Channel 0
- Channel 8
- Channel 16
- Channel 24

Implicit device DDT

Name : MOD_DIS_32_1

Type : T_U_DIS_STD_IN_32

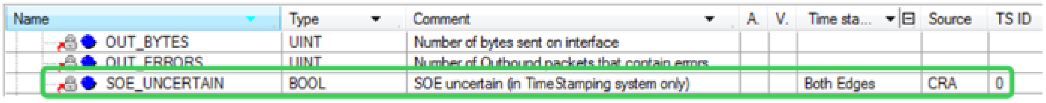
3. Expand the elements under the implicit device DDT name of the BMX discrete I/O module. Expand the elements under ‘DIS_CH_IN’ of the input module, or ‘DIS_CH_OUT’ of the output module. Expand the elements under the required time stamping channel, and enable the channel by selecting the proper event in the ‘Time stamping’ cell.



Name	Type	Comment	A.	V.	Time sta...	Source	TS ID
MOD_DIS_16_2	T_U_DIS_STD_IN_16						
MOD_HEALTH	BOOL	Module health					
MOD_FLT	BYTE	Module faults					
DIS_CH_IN	ARRAY[0..15] OF T_U_DIS_...						
DIS_CH_IN[0]	T_U_DIS_STD_CH_IN						
CH_HEALTH	BOOL	Channel health					
VALUE	EBOOL	Discrete input value			Both Edges	CRA	257
DIS_CH_IN[1]	T_U_DIS_STD_CH_IN						
DIS_CH_IN[2]	T_U_DIS_STD_CH_IN						
DIS_CH_IN[3]	T_U_DIS_STD_CH_IN						
DIS_CH_IN[4]	T_U_DIS_STD_CH_IN						

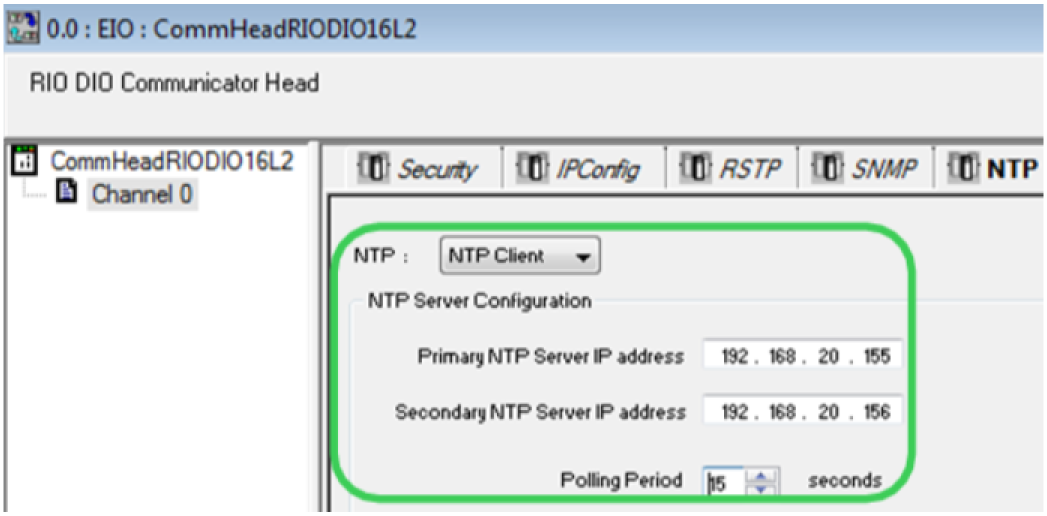
NOTE: For the M580, this attribute can be ‘None,’ ‘Both Edges,’ ‘Rising Edge,’ or ‘Falling Edge.’ For Quantum, however, the only options are ‘None’ or ‘Both Edges.’

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.



Name	Type	Comment	A.	V.	Time sta...	Source	TS ID
OUT_BYTES	UINT	Number of bytes sent on interface					
OUT_ERRORS	UINT	Number of Outbound packets that contain errors					
SOE_UNCERTAIN	BOOL	SOE uncertain (in TimeStamping system only)			Both Edges	CRA	0

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:



0.0 : EIO : CommHeadRIODIO16L2

RIO DIO Communicator Head

NTP : NTP Client

NTP Server Configuration

Primary NTP Server IP address 192 . 168 . 20 . 155

Secondary NTP Server IP address 192 . 168 . 20 . 156

Polling Period 15 seconds

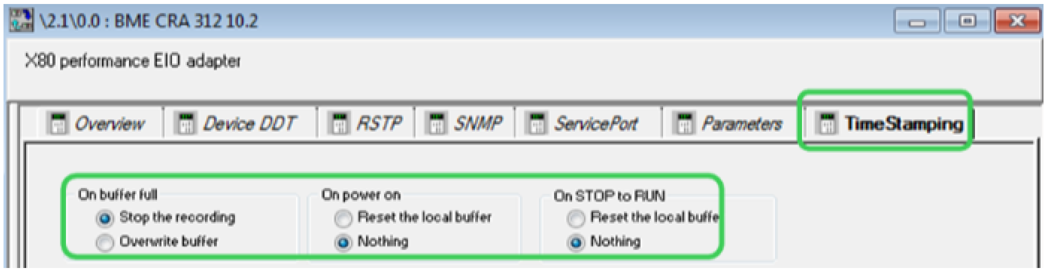
NOTE: It is recommended that the polling period be set to lower than 20s in order to get a time stamp resolution of 10ms between two events on different CRA modules.

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.



\2.1\0.0 : BME CRA 312 10.2

X80 performance EIO adapter

On buffer full

● Stop the recording

○ Overwrite buffer

On power on

○ Reset the local buffer

● Nothing

On STOP to RUN

○ Reset the local buffe

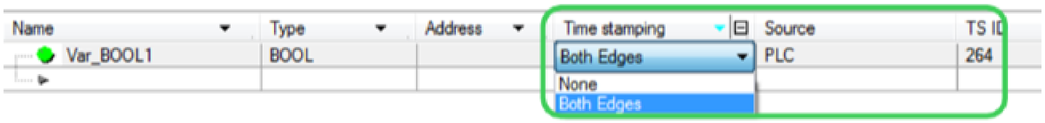
● Nothing

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.

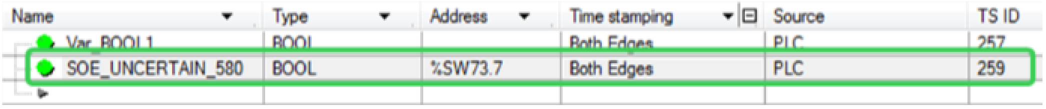
1. In the 'Data Editor,' select a BOOL type internal variable which can trigger a time stamping event; then select the trigger condition. Unity Pro will generate a TS ID.



Name	Type	Address	Time stamping	Source	TS ID
Var_BOOL1	BOOL		Both Edges	PLC	264

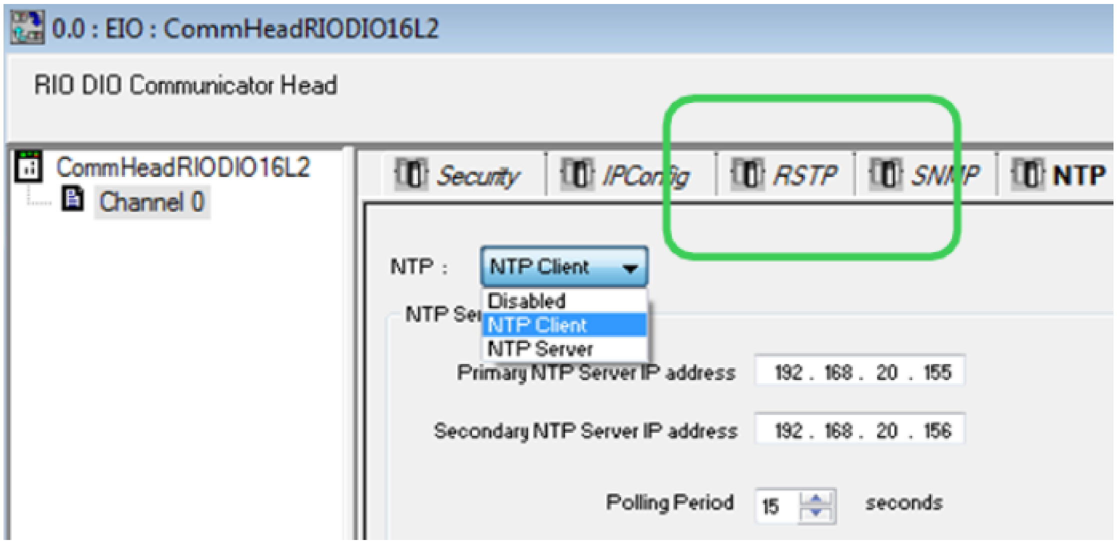
NOTE: The internal buffer of the M580 CPU's time stamped events will behave as follows: The CPU stops recording new events when the buffer is full.

2. Manually create the SOE_UNCERTAIN variable for the M580 CPU, and locate this BOOL at %SW73.7. Enable its time stamping selection.



Name	Type	Address	Time stamping	Source	TS ID
Var_BOOL1	BOOL		Both Edges	PLC	257
SOE_UNCERTAIN_580	BOOL	%SW73.7	Both Edges	PLC	259

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.



0.0 : EIO : CommHeadRIODIO16L2

RIO DIO Communicator Head

CommHeadRIODIO16L2
Channel 0

NTP : NTP Client

NTP Se

Primary NTP Server IP address 192 . 168 . 20 . 155

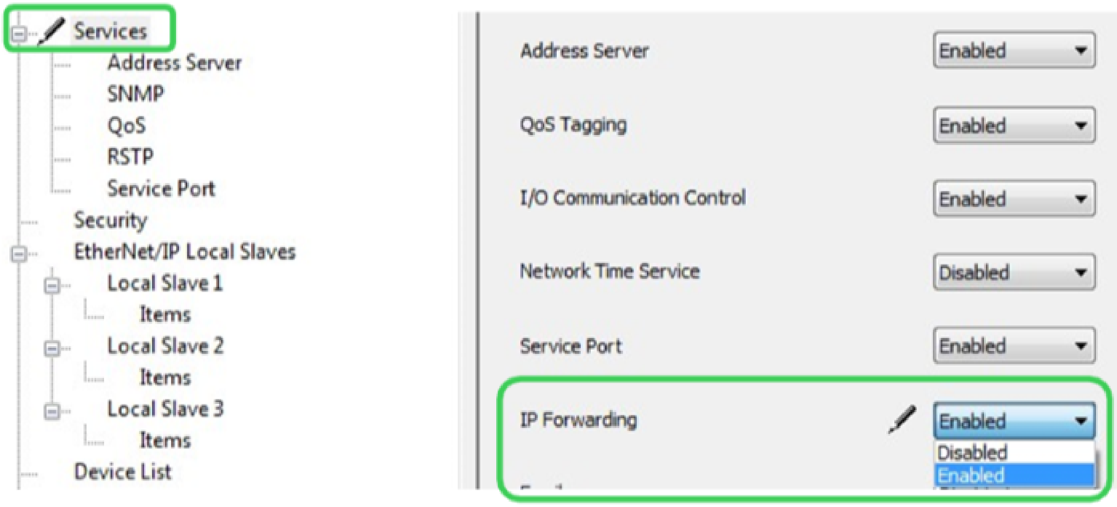
Secondary NTP Server IP address 192 . 168 . 20 . 156

Polling Period 15 seconds

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.



Address Server Enabled

QoS Tagging Enabled

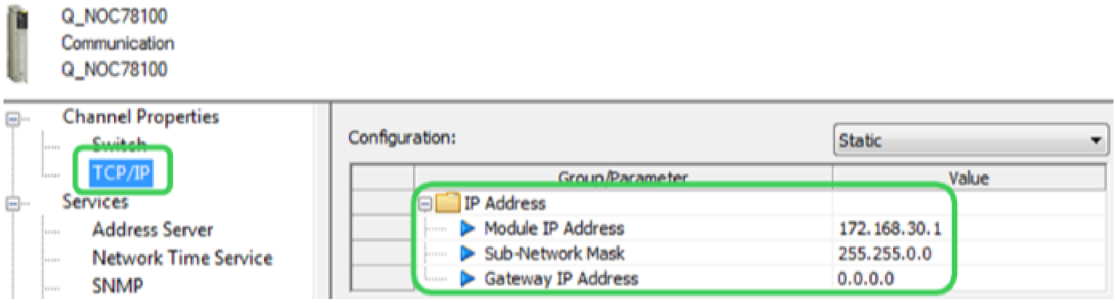
I/O Communication Control Enabled

Network Time Service Disabled

Service Port Enabled

IP Forwarding Enabled

2. Configure its IP address for the control network port (Eth port 3&4) on the 'TCP/IP' page.

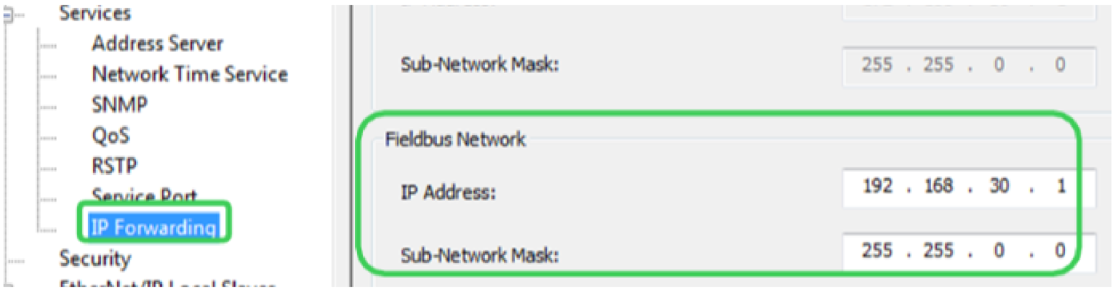


Q_NOC78100
Communication
Q_NOC78100

Configuration: Static

Group/Parameter	Value
IP Address	
Module IP Address	172.168.30.1
Sub-Network Mask	255.255.0.0
Gateway IP Address	0.0.0.0

Configure its IP address for the device network port (Eth port 2) on the 'IP Forwarding' page.



Sub-Network Mask: 255 . 255 . 0 . 0

Fieldbus Network

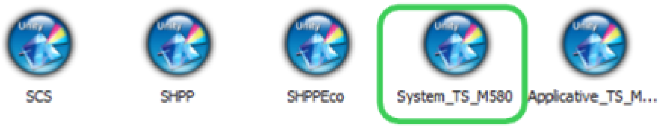
IP Address: 192 . 168 . 30 . 1

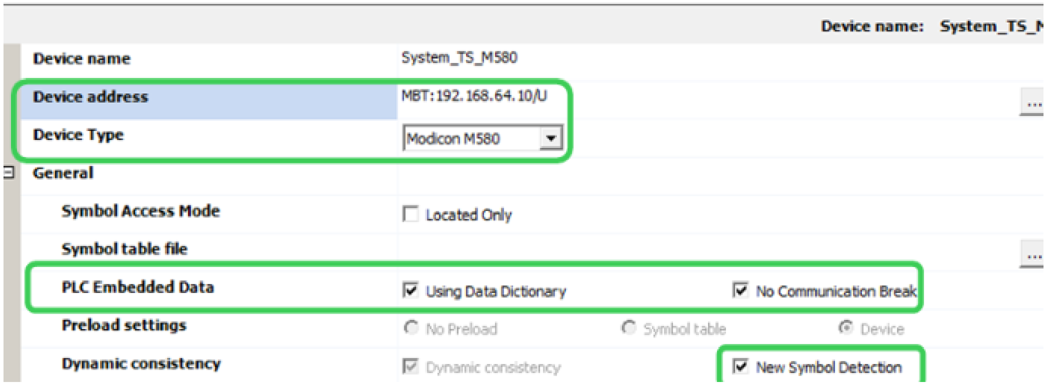
Sub-Network Mask: 255 . 255 . 0 . 0

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**.

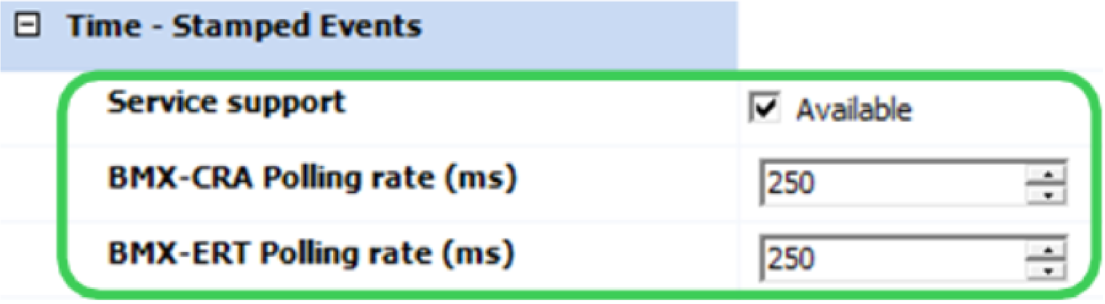




Device name: System_TS_M

Device name	System_TS_M580	
Device address	MBT:192.168.64.10/U	...
Device Type	Modicon M580	
General		
Symbol Access Mode	□ Located Only	
Symbol table file		...
PLC Embedded Data	☑ Using Data Dictionary	☑ No Communication Break
Preload settings	○ No Preload ○ Symbol table	● Device
Dynamic consistency	☑ Dynamic consistency	☑ 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.



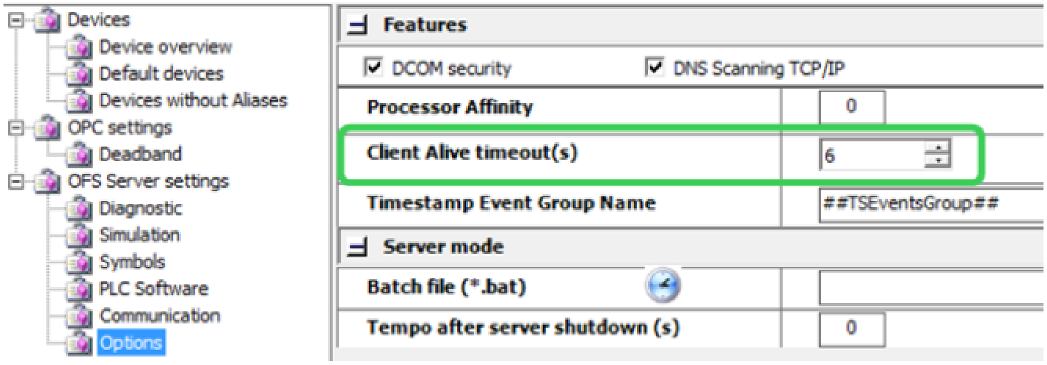
Time - Stamped Events

Service support	☑ Available
BMX-CRA Polling rate (ms)	250
BMX-ERT Polling rate (ms)	250

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.



☑ DCOM security ☑ DNS Scanning TCP/IP

Processor Affinity 0

Client Alive timeout(s) 6

Timestamp Event Group Name ##TSEventsGroup##

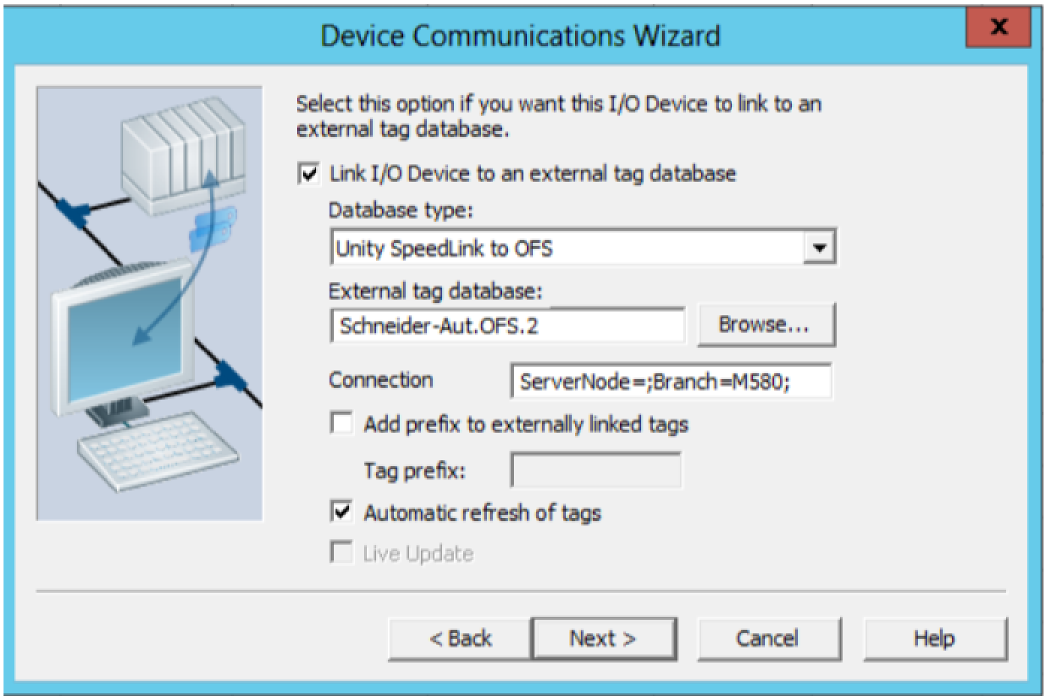
Batch file (*.bat)

Tempo after server shutdown (s) 0

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:



Device Communications Wizard

Select this option if you want this I/O Device to link to an external tag database.

☑ Link I/O Device to an external tag database

Database type: Unity SpeedLink to OFS

External tag database: Schneider-Aut.OFS.2

Connection: ServerNode=;Branch=M580;

□ Add prefix to externally linked tags

Tag prefix:

☑ Automatic refresh of tags

□ Live Update

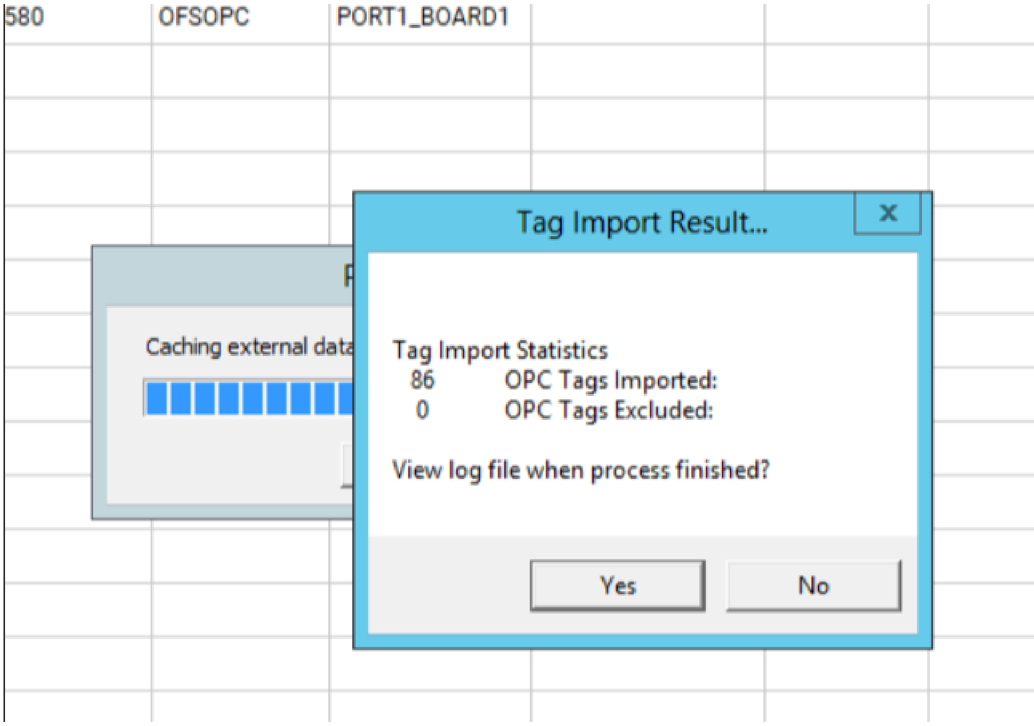
h. Click **Next**.

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.



Tag Import Result...

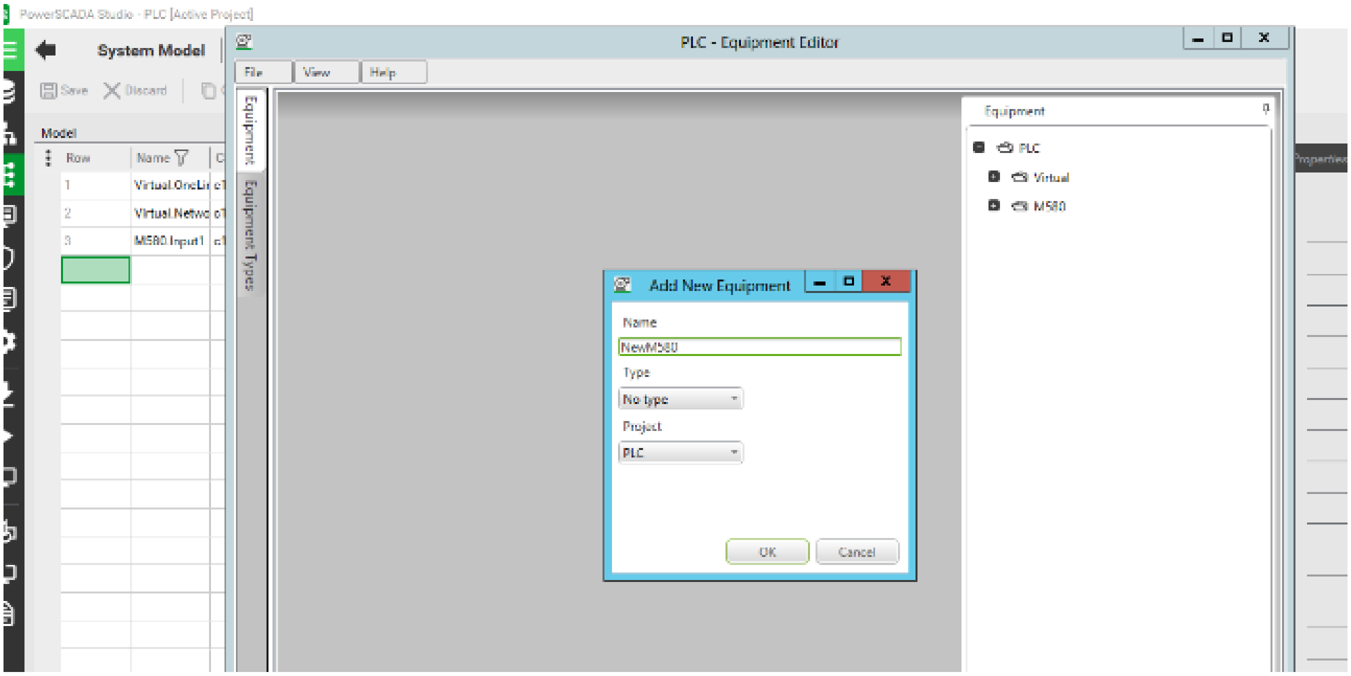
Tag Import Statistics
86 OPC Tags Imported:
0 OPC Tags Excluded:

View log file when process finished?

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:



Row	Name	C
1	Virtual.OneLi	c1
2	Virtual.Netwo	c1
3	M580.Input1	c1

Name: NewM580

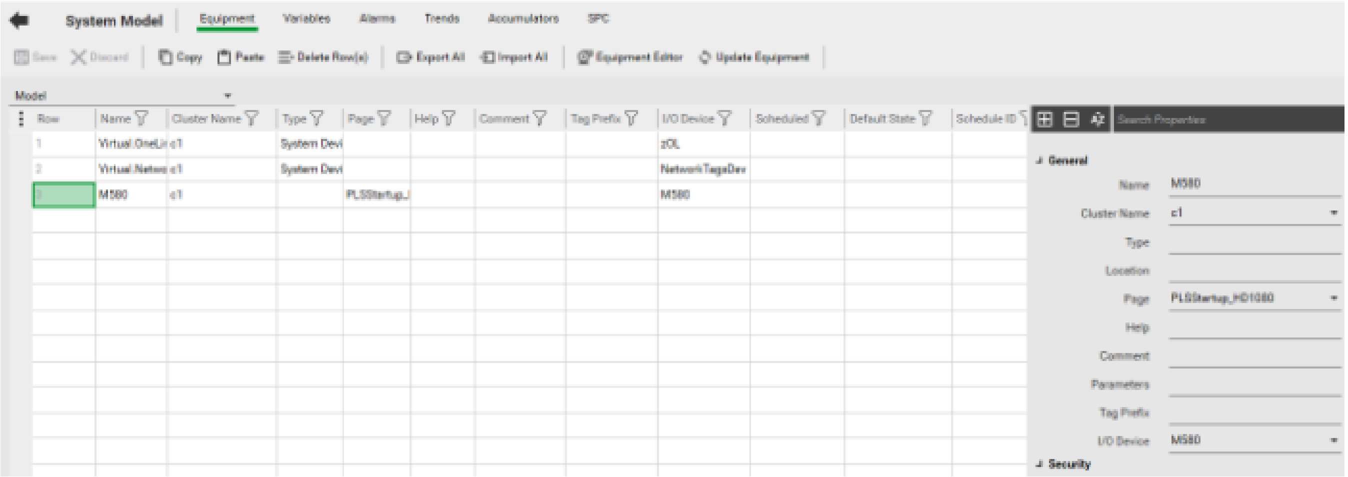
Type: No type

Project: PLC

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:



Row	Name	Cluster Name	Type	Page	Help	Comment	Tag Prefix	I/O Device	Scheduled	Default State	Schedule ID
1	Virtual.OneLi	c1	System Devi					zOL			
2	Virtual.Netwo	c1	System Devi					NetworkTagsDev			
3	M580	c1		PLSStartup_				M580			

Name: M580

Cluster Name: c1

Type:

Location:

Page: PLSStartup_HD1080

Help:

Comment:

Parameters:

Tag Prefix:

I/O Device: M580

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**.

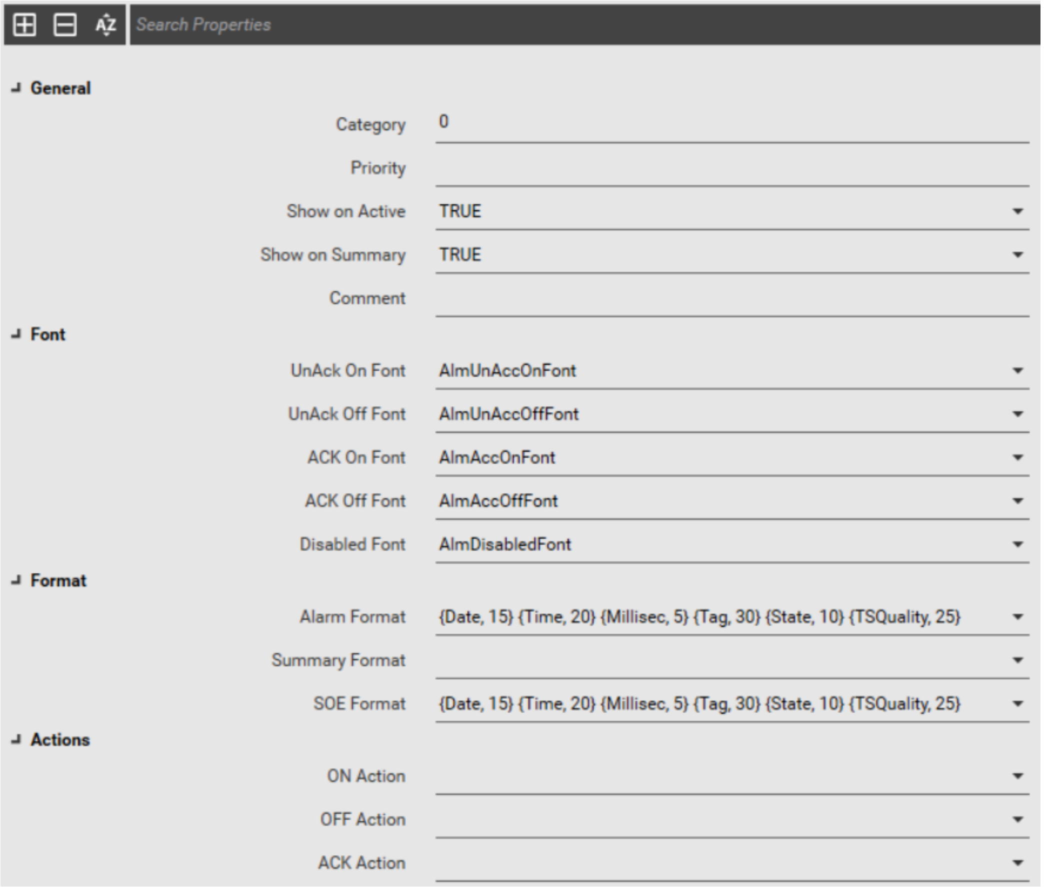
b. In the grid, enter the **Category** number.

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:



General	
Category	0
Priority	
Show on Active	TRUE
Show on Summary	TRUE
Comment	
Font	
UnAck On Font	AlmUnAccOnFont
UnAck Off Font	AlmUnAccOffFont
ACK On Font	AlmAccOnFont
ACK Off Font	AlmAccOffFont
Disabled Font	AlmDisabledFont
Format	
Alarm Format	{Date, 15} {Time, 20} {Millisec, 5} {Tag, 30} {State, 10} {TSQuality, 25}
Summary Format	
SOE Format	{Date, 15} {Time, 20} {Millisec, 5} {Tag, 30} {State, 10} {TSQuality, 25}
Actions	
ON Action	
OFF Action	
ACK Action	

6. Create the system time stamping alarms:

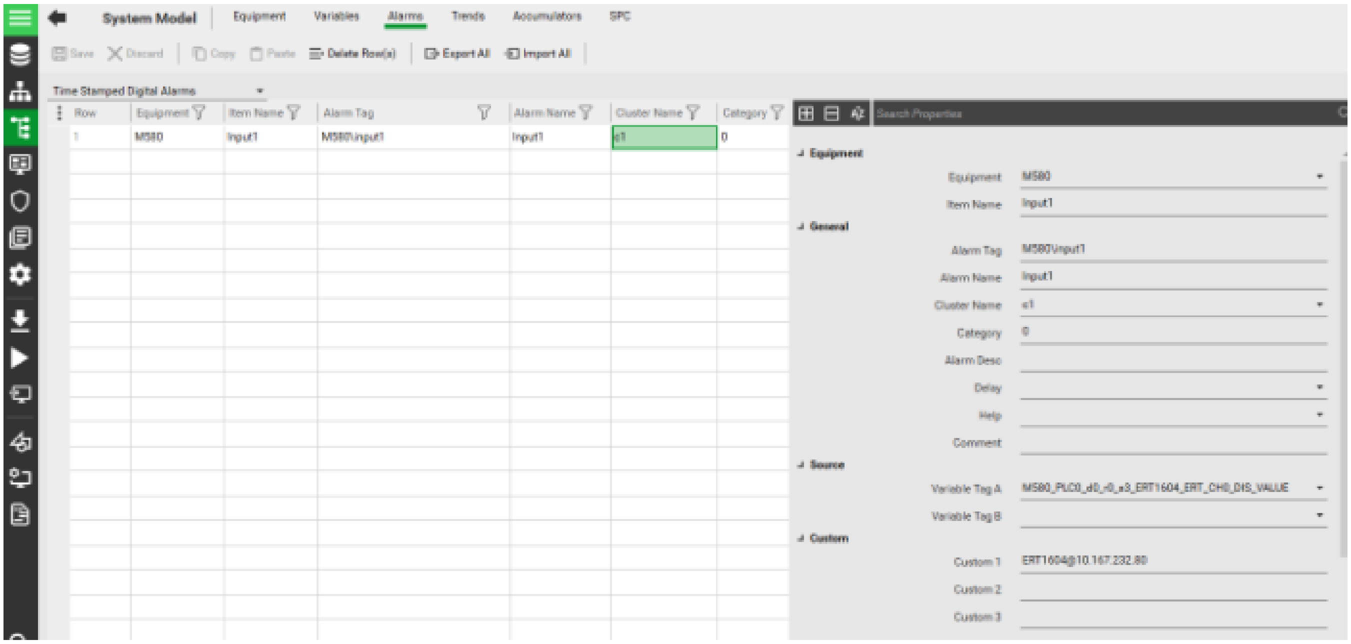
a. Click **System Model > Alarms**.

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:



Time Stamped Digital Alarms

Row	Equipment	Item Name	Alarm Tag	Alarm Name	Cluster Name	Category
1	M580	Input1	M580\Input1	Input1	c1	0

Equipment

Equipment: M580

Item Name: Input1

General

Alarm Tag: M580\Input1

Alarm Name: Input1

Cluster Name: c1

Category: 0

Alarm Desc:

Delay:

Help:

Comment:

Source

Variable Tag A: M580_PLC0_d0_r0_s3_ERT1604_ERT_CH0_DIS_VALUE

Variable Tag B:

Custom

Custom 1: ERT1604@10.167.232.80

Custom 2:

Custom 3:

⚠ WARNING

LOSS OF ALARMS

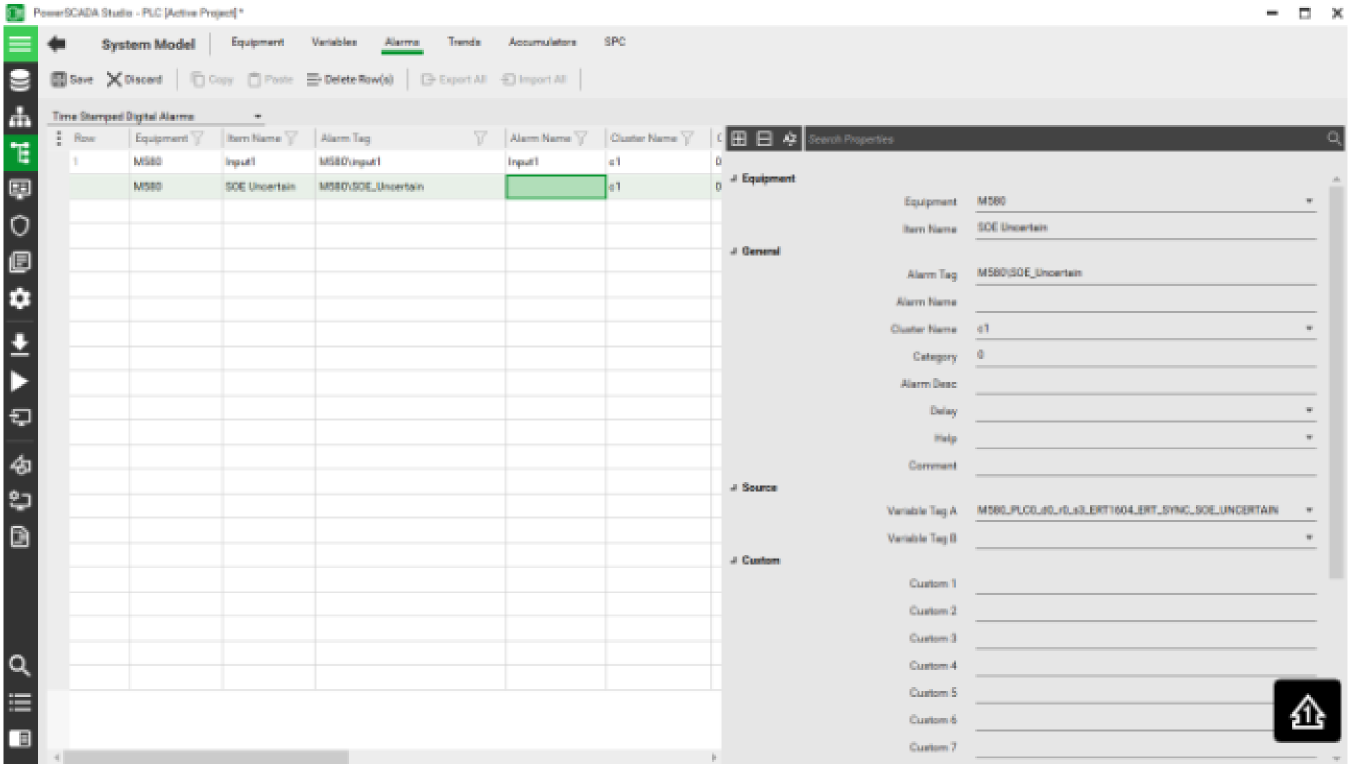
- To be able to detect that the event buffer is full, configure a tag and an alarm tag associated with the SOE_Uncertain parameter in UnityPro.
- Respond quickly to a buffer full alarm if it appears, as this will avoid a situation where the buffer becomes inoperable.

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:



Row	Equipment	Item Name	Alarm Tag	Alarm Name	Cluster Name	C
1	M580	Input1	M580\input1	Input1	c1	0
	M580	SOE Uncertain	M580\SOE_Uncertain		c1	0

Equipment

Equipment: M580

Item Name: SOE Uncertain

General

Alarm Tag: M580\SOE_Uncertain

Alarm Name:

Cluster Name: c1

Category: 0

Alarm Desc:

Delay:

Help:

Comment:

Source

Variable Tag A: M580_PLC0_d0_r0_s3_ERT1604_ERT_SYNC_SOE_UNCERTAIN

Variable Tag B:

Custom

Custom 1:

Custom 2:

Custom 3:

Custom 4:

Custom 5:

Custom 6:

Custom 7:

Implementation

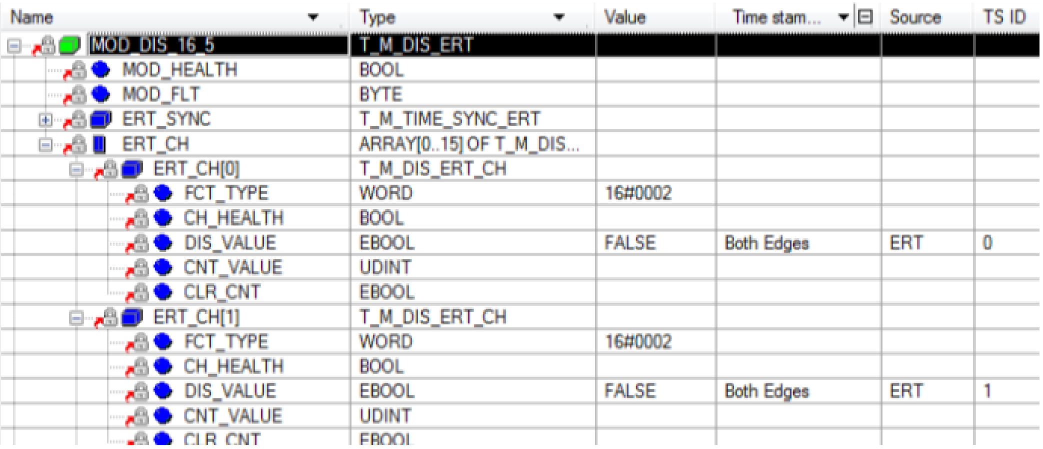
This chapter presents the detailed steps for the engineering implementation.

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:

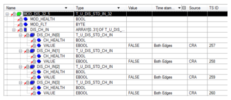
   1) ERT_CH[0] → High temperature alarm; 2) ERT_CH[1] → Motor alarm



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:

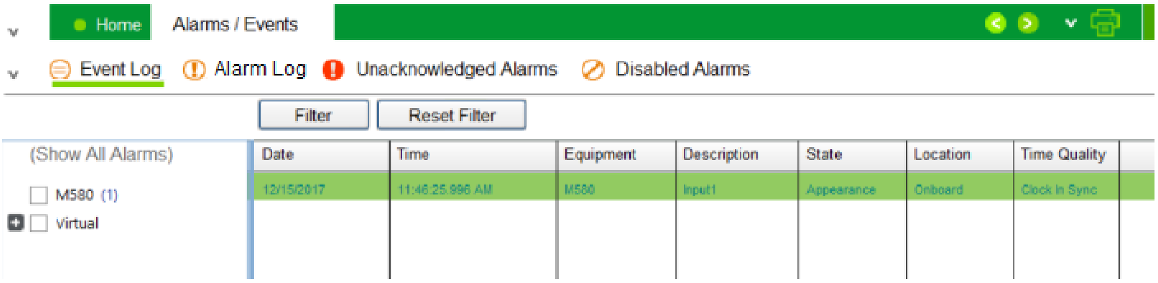
1) DIS_CH_IN[0] → Overload; 2) DIS_CH_IN[1] → Fan status; 3) DIS_CH_IN[2] → Over temperature; 4) DIS_CH_IN[3] → Manual operation



Name	Type	Value	Time stam...	Source	TS ID
MOD_DIS_32_5	T_U_DIS_STD_IN_32				
MOD_HEALTH	BOOL				
MOD_FLT	BYTE				
DIS_CH_IN	ARRAY[0..31] OF T_U_DIS_...				
DIS_CH_IN[0]	T_U_DIS_STD_CH_IN				
CH_HEALTH	BOOL				
VALUE	EBOOL	FALSE	Both Edges	CRA	257
DIS_CH_IN[1]	T_U_DIS_STD_CH_IN				
CH_HEALTH	BOOL				
VALUE	EBOOL	FALSE	Both Edges	CRA	258
DIS_CH_IN[2]	T_U_DIS_STD_CH_IN				
CH_HEALTH	BOOL				
VALUE	EBOOL	FALSE	Both Edges	CRA	259
DIS_CH_IN[3]	T_U_DIS_STD_CH_IN				
CH_HEALTH	BOOL				
VALUE	EBOOL	FALSE	Both Edges	CRA	260

Operation

You can view the SOE history in the Power SCADA Operation event log:



Date	Time	Equipment	Description	State	Location	Time Quality
12/15/2017	11:46:25.996 AM	M580	Input1	Appearance	Onboard	Clock In Sync

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.

Multi-site architectures

A multi-site architecture (or multi-clustered system) allows you to scale your system as your needs evolve.

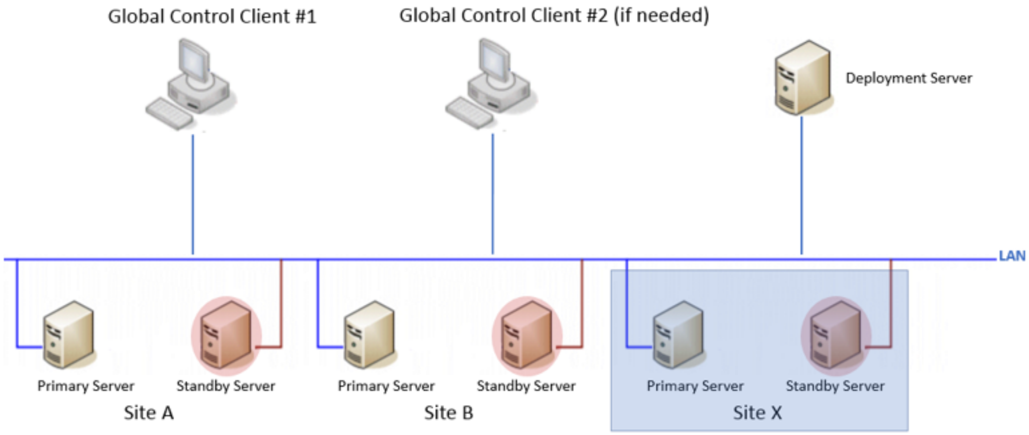
Use the information in the tables below to find the content you are looking for.

Topic	Description
"Server architecture" on page 672	Example multi-clustered system server architecture.

Topic	Description
" SCADA project structure" on page 672	SCADA project structure for a multi-clustered system.
"Project development structure" on page 673	Example project development structure for a multi-clustered system.
"Configuration guidelines" on page 675	Guidelines for configuring multi-clustered systems.

Server architecture

The following image shows an example multi-clustered system server architecture. Each site consists of a primary and standby server. Each site on the Local Area Network (LAN) is connected to the global control client servers. There can be more than one global control client on the network.



NOTE: The global control clients can connect to different sites or the same sites.

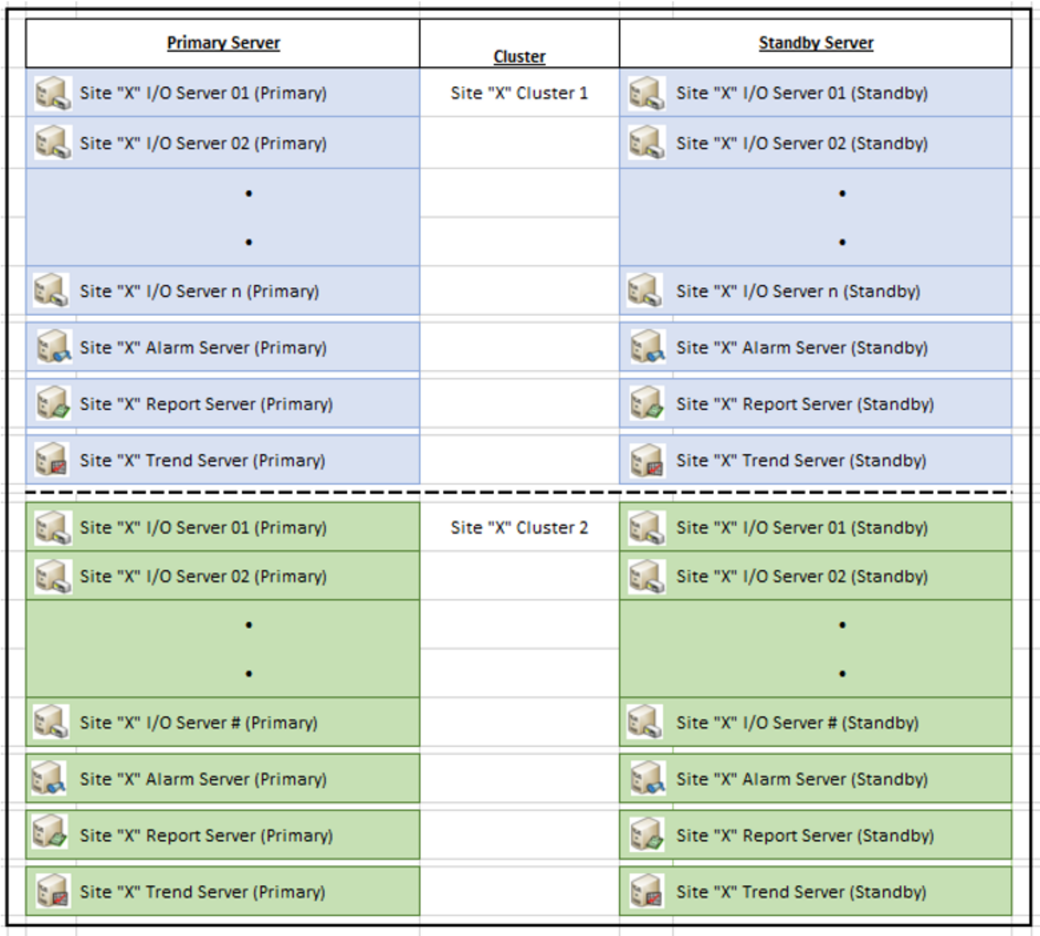
As part of workflow and version control on-site, a deployment server is used to deploy Power SCADA Operation projects to the individual sites and global control client servers. This server contains the different software projects.

The deployment server:

- Stores multiple versions of a project's runtime files in a central network location.
- Deploys a specific version of a project to a runtime computer, or a pre-defined group of computers.
- Rolls back a computer to a previous version of a project.
- Manages the restart options on the destination computer when a new version of a project is received.

SCADA project structure

The following diagram shows the structure of the SCADA project for each site. Each site has two or more clusters with their own specific servers.



Primary Server	Cluster	Standby Server
Site "X" I/O Server 01 (Primary)	Site "X" Cluster 1	Site "X" I/O Server 01 (Standby)
Site "X" I/O Server 02 (Primary)		Site "X" I/O Server 02 (Standby)
•		•
•		•
Site "X" I/O Server n (Primary)		Site "X" I/O Server n (Standby)
Site "X" Alarm Server (Primary)		Site "X" Alarm Server (Standby)
Site "X" Report Server (Primary)		Site "X" Report Server (Standby)
Site "X" Trend Server (Primary)		Site "X" Trend Server (Standby)
Site "X" I/O Server 01 (Primary)	Site "X" Cluster 2	Site "X" I/O Server 01 (Standby)
Site "X" I/O Server 02 (Primary)		Site "X" I/O Server 02 (Standby)
•		•
•		•
Site "X" I/O Server # (Primary)		Site "X" I/O Server # (Standby)
Site "X" Alarm Server (Primary)		Site "X" Alarm Server (Standby)
Site "X" Report Server (Primary)		Site "X" Report Server (Standby)
Site "X" Trend Server (Primary)		Site "X" Trend Server (Standby)

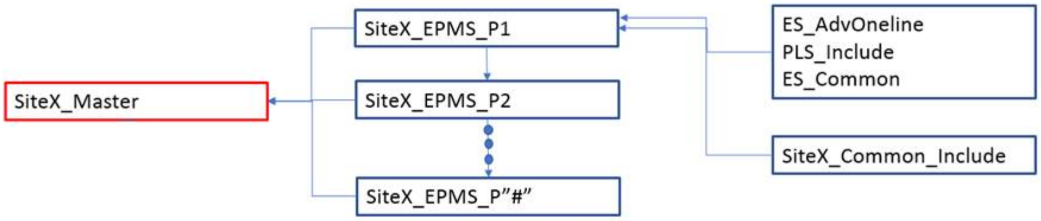
NOTE: The "X" indicates the site name or site identifier. Each server name must be unique in each cluster and have its own specific ports.

Each cluster has the following internal SCADA servers (Primary and Standby):

1. I/O Server
   a. The number of I/O servers will depend on one of two limits: 200 devices per I/O server or 50,000 tags, whichever comes first. Once one of the limits has been reached, a new I/O server must be created.
2. Alarm Server
3. Reports Server
4. Trend Server

Project development structure

The following image shows an example project development structure for a multi-clustered system.



NOTE: Arrows point from the include project to the project that includes them. For example, The SiteX_Common_Include project is an include project of the Site_X_EPMS_P1 project.

Follow the preceding example project development structure for each site. In the example, two clusters are created in SiteX_EPMS_P1 phase. The clusters can also be created on any individual phases.

SCADA Project	Description
Site_X_Master	Master project used to combine all individual site phase projects (SiteX_EPMS_P1,P2, P3, etc.). This project has the site menus, one-line menus, and other page links that are available only on that site. The SiteX_Master project does not have any Servers or I/O devices.
SiteX_EPMS_P1	This project works without the need for any other phases. In the example project, the initial project will have two clusters. Additional clusters can be created on this initial project or on subsequent phase projects. Each cluster has its own I/O, Alarm, Trend and Report Servers. The SiteX_EPMS_P1 project must have the computer and network information that will be used for the site. Other projects do not need computer and network information.
SiteX_EPMS_P2	This project is an example additional phase project include that has the original SiteX_EPMS_P1 project as an include. This means it receives all the site information (Clusters, Computer, Network, etc.). It can create its own cluster and unique servers if necessary.
SiteX_Common_Include	This is an include project that will contain all the unique pop-ups, pages, genies, etc. that will be specific to the particular sites.
ES_AdvOneline	Default project for Advanced One-line bus animations and genies. This project will be the same throughout all the sites.
ES_Common	Default project for common templates, genies, utilities, etc. This project will be the same throughout all the sites.
PLS_Include	Default Power SCADA include project. This is a default project that is required on any project. This project will be the same throughout all the sites.

Configuration guidelines

Take note of the following guidelines when you configure a multi-clustered system.

Project structure

- All popups, genies, utilities, graphic pages that will be shared through all projects must be in a common include project. For example, ES_Common.
- All other popups, genies, utilities, graphic pages that are unique to the site must have unique names. The cluster must be defined in the page properties in the “Cluster context” section. If the pages will not be used on any global control clients, save the unique items in an include project that is specific to the site. For example, SiteA_Common_Include.
- Put all labels, users, roles, and groups that will be shared or are the same in each project in a common project. For example, ES_Common.
- Default alarm pages automatically call in all clusters. The default Engineering Solutions alarm pages are in the ES_Common project and the default SCADA alarm pages are in the PLS_ Include. No changes are needed.

Real-time readings

- Real-time readings from devices of different clusters can be viewed on graphic pages. The two clusters must be selected on the Client component when setting up the Cluster Connections Setup in the Computer Setup Wizard. This applies to the Advanced One-line as well.
- Real-time readings from devices on different clusters will not work when used on a template page, since the template pages only retrieve data from one cluster.

Naming

- Cluster names, Computer names, networks names, alarm server names, report server names, trend server names, and I/O server names must all be unique. Add site specific prefixes to be able to easily distinguish the names.

  NOTE: Only add the computer name and the network name to one project per site.

- I/O device names and port names can be the same on different clusters. If possible, use site specific prefixes to be able to easily distinguish the names in Project Studio user interface.
- Equipment names should be dot (“.”) based. The first should be the site specific, then any other information. For example, SiteA.First_Floor.MVGear.PLC2.
- If the unique item will be brought to the global client for use later when multiple sites are combined, the name of the popup, genie, utility, graphic page, etc. must be unique. It is recommended that the site name be used as a prefix. For example, if a unique graphic page is named, it should be named SiteA_First_Floor.
- Variable tags, alarm tags, etc. from I/O devices cannot be split across different clusters.
- Cicode files that will be used per site need a unique name. Like with graphics, it is recommended that the Cicode files be named with the site prefix.

- All Menu setup for the project must be in the site master project. If multiple sites are included in one project, it is recommended to label the menus appropriately.

I/O devices

- Create the NetworkTagsDev I/O device for each cluster. It does not matter which I/O server it is created on for the cluster.
- The zOL, zES devices must be created on only one cluster.

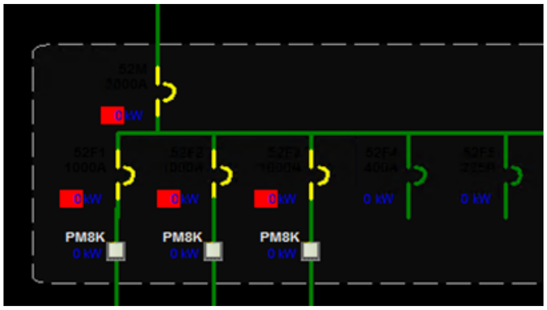
> **NOTE**: The zOL and zES must be on the network that is locally run.

Advanced one-line

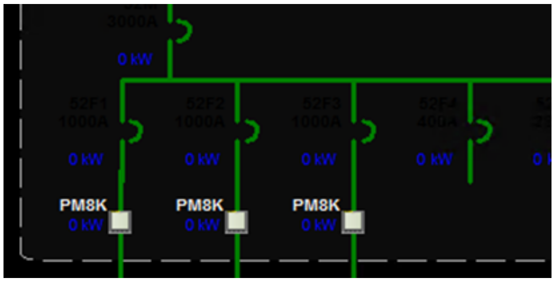
There are no changes required for Advanced one-line configuration for multi-clustering.

Multiple clusters must be selected for the Client during the Computer Setup Wizard Cluster Connection Setup portion.

The following image shows Advanced one-line with one cluster selected for the client. The devices in yellow are on the cluster that is not selected.



The following image shows Advanced one-line with multiple clusters selected for the client.



Servers

All the clusters in the site or on any global clients must be selected for the Alarm Servers, Report Servers, Trend Servers and the Client Server.

Notifications

Configure alarm notifications one cluster at a time. Select a cluster during initial startup.

You must also select a cluster when viewing alarms in the notifications results window.

Troubleshooting

During project compilation, if you get the “Tag usage is ambiguous” warnings, add `[CtEdit]SuppressCompilerWarning = W1039` to Citect.ini. This code suppresses the warnings. If your warning number is different, put that into Citect.ini.

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.

See the following topics for detailed information on configuring a redundant system:

Topic	Description
"Configure the Power SCADA Primary Server" on page 677	Lists the procedures to copy and export that will be subsequently imported on the Secondary Server.
"Configure the Power SCADA Secondary Server" on page 679	Lists the procedures to import the Primary Server files onto the Secondary Server.
"Updating on redundant systems" on page 681	Information on how to make changes on a redundant system without interruptions to the system.

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 677
- "Back up Application Configuration Utility settings" on page 678
- "Export One-Line Engine Encryption" on page 678
- "Export and import One-Time Password settings" on page 679

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\v2020 R2\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 680
- "Import the One-Time Password" on page 680
- "Import the Advanced One-Line Encryption (AES) File" on page 680
- In the Application Configuration Utility:
  - Re-enter SSO passwords in . For more information , see "Configure Single Sign-On (SSO)" on page 625.
  - Re-enter the Citect Data Platform password. For more information, see "Set up data acquisition parameters" on page 198.
- 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 346 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 Encryption (AES) 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.

Updating on redundant systems

When the system is in operation and the primary server becomes inoperative, or if you take it offline to perform maintenance or make changes, you can revert to the standby server with minimal or no interruption to the system.

When the primary server is brought back online, the system returns control of the I/O devices to the primary server.

First, make your desired changes, such as the following tasks:

- See topics within the Define one I/O device in a project section to learn how to add an I/O device.
- Remove an I/O device from a project
- Update TGML diagrams on redundant systems
- Update alarm thresholds on redundant systems

After implementing changes, do the following:

Compile and restart a redundant system to implement changes

Before you begin, observe the following:

> ***NOTICE***
>
> **LOSS OF DATA**
>
> Backup your primary and standby projects prior to implementing changes in a redundant system.
>
> **Failure to follow these instructions can result in a loss of data.**

- For details, see Back up the Power Operation Studio project. If you encounter issues, Restore the Power Operation Studio project.
- When updating projects on redundant systems, always begin with the standby server (the non-active server) unless it is not possible to do so.
- Verify that both the primary and standby servers are running.
- If using any event notifications, it is recommended that you put the system into maintenance mode while implementing changes during failover, as this may trigger changes for miscellaneous alarms and events.

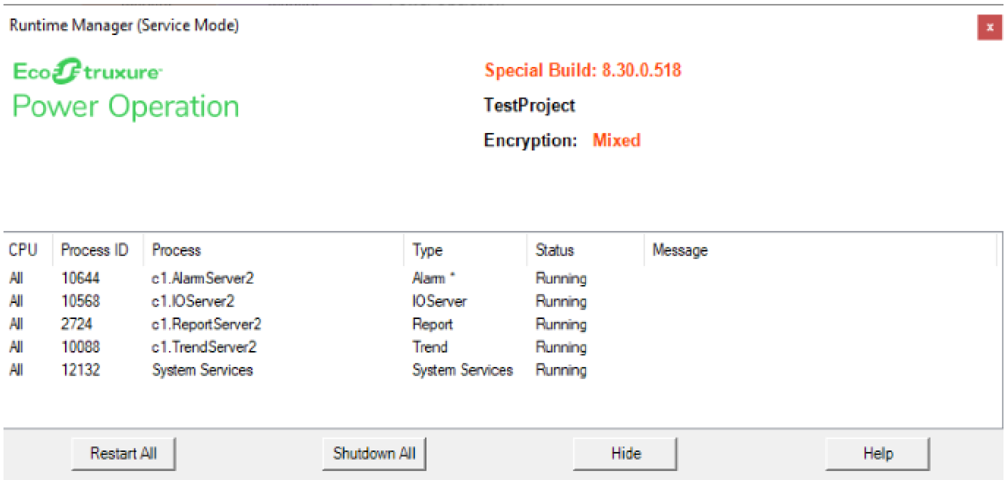
To compile and restart a redundant system to implement changes:

1. Implement your changes on the standby server.
2. In Power SCADA Studio, verify that the active project is selected > select **Compile**. Even if the project is compiled, it is recommended that you compile again.

3. (Optional) Select **Setup Wizard** > select **Next** through to the end of the Computer Setup Wizard. Do not change any settings.

4. Open the Power SCADA Runtime Manager.

5. In the Runtime Manager, confirm all Citect processes show their status as Running. It is recommended that you confirm all processes are running on the primary server as well, as the standby server will need to be restarted.



Runtime Manager (Service Mode)

EcoStruxure Power Operation

Special Build: 8.30.0.518

TestProject

Encryption: Mixed

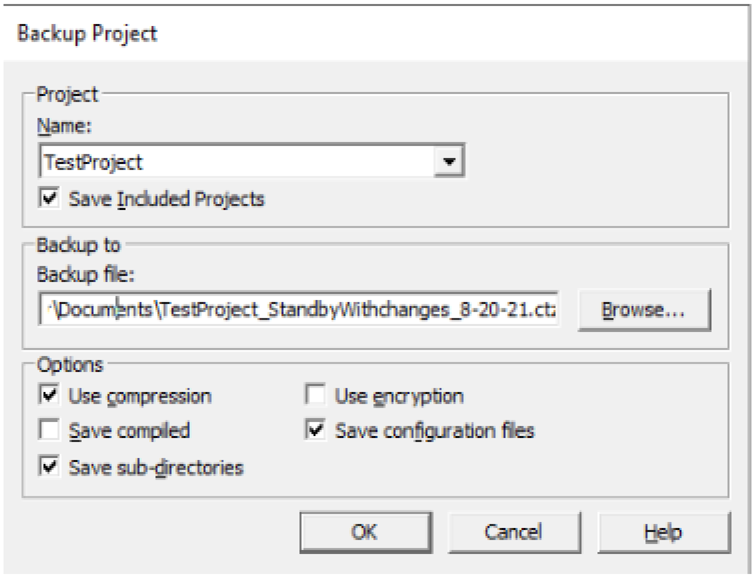
CPU	Process ID	Process	Type	Status	Message
All	10644	c1.AlarmServer2	Alarm *	Running	
All	10568	c1.IOServer2	IOServer	Running	
All	2724	c1.ReportServer2	Report	Running	
All	10088	c1.TrendServer2	Trend	Running	
All	12132	System Services	System Services	Running	

6. Select **Restart All**. Confirm that all processes restart and resume their Running status.

7. Confirm your changes are present on the standby server.

8. When the confirmed changes are complete, backup the standby server's project by navigating to Power SCADA Studio and select **Backup**.

9. In the Backup Project window, enable the **Save Included Projects** checkbox.



Backup Project

Project

Name: TestProject

☑ Save Included Projects

Backup to

Backup file: \Documents\TestProject_StandbyWithchanges_8-20-21.ct

Options

☑ Use compression ☐ Use encryption

☐ Save compiled ☑ Save configuration files

☑ Save sub-directories

10. Browse to the location where you want to store the backup file.

11. Maintain the default options enabled. Select **OK**.

12. When complete, copy the backup to the primary server.

13. On the primary server, open Power SCADA Studio.
14. In the Projects window, with the Active Project selected, from the Backup drop-down menu, choose **Restore**.
15. In the Restore Project dialog, browse to and select the project backup copied from the standby server.
16. In the Included Projects section, enable the **Select all included projects** checkbox. Select all projects, with the exception of PLS_Include. Do not overwrite PLS_Include with any backup.
17. In the To section, enable the **Current Project** radio button.
18. In the Options section, disable the **Configuration files** checkbox.
19. Select **OK**.
20. In the warning, select **Yes**.
21. When the restore is complete, compile the project.
22. When the compile is complete, select **Setup Wizard** > select **Next** through to the end of the Computer Setup Wizard. Do not change any settings.
23. When the Setup Wizard is complete, confirm the new project has the changes added on the Standby project.
24. When all changes are confirmed, open the Power SCADA Runtime Manager.
25. In the Runtime Manager, confirm all Citect processes show their status as "Running". It is recommended that you confirm all processes are running on the primary server as well, as the standby server will need to be restarted.
26. Select **Restart All**. Confirm that all processes restart and resume their Running status.
27. Confirm your changes are present on the standby server.
28. (Optional) Restart disabled or stopped notification services.

Updating TGML diagrams on redundant systems

Update existing TGML diagrams on redundant systems without interruption.

For detailed instructions on working with TGML diagrams, refer to the topics within the Graphics Editor section. The following is an example meant to demonstrate the process in the context of a redundant system.

To update TGML diagrams on redundant systems:

1. On the standby server, launch Graphics Editor.
2. Open the graphics page you want to update.
3. On the graphics page, using the Components library and Objects properties window, add the new components and binding for the device.
4. **File** > **Save**. If prompted, overwrite the existing graphics page.

5. On the standby server, open the WebHMI and navigate to the graphics page you edited. Refresh if the diagram does not appear updated. If a refresh is unsuccessful, clear your cache and try again.

6. Do one of the following:

   – If you have only made changes to your graphics, copy the TGML folder from the project directory and paste it to the primary server, overwriting the folder. In most cases, the project directory is located at C:\ProgramData\Schneider Electric\Power SCADA Operation\v2020\User\<*Your Project*>\TGML. This will apply the changes to the primary server.

   – If you have made other changes, such as adding or removing a device, a full backup and restore is required. Follow the Updating on redundant systems procedure.

Updating analog alarm thresholds on redundant systems

Update existing alarm thresholds on redundant systems without interruption. You can update analog alarm activation limits and push them to running projects. Some alarm types will have more limits that can be set than others.

Analog alarms are triggered when analog variables change beyond one or more specified limits. Each alarm can be configured in any of the following combinations:

- High and HighHigh alarms: The values reach an atypical high.
- Low and LowLow alarms: The values reach an atypical low.
- Deviation alarms: The values move away from a predefined set point.
- Rate of change alarms: Dramatic value changes occur within a specified period of time.

To update analog alarm thresholds on redundant systems:

1. On the standby server, launch Power SCADA Studio.
2. **System Model** > **Alarms** tab.
3. From the Alarm type drop-down menu, select **Analog Alarms** > select the alarm row you want to update.
4. Scroll to the right or use the Properties window on the far right, and then select the High setpoint.
5. Change the setpoint to the desired value for activation.
6. Select **Save**.

To complete the update, see Updating on redundant systems.

Administer

Use the information provided in this chapter to administer a deployed, running Power SCADA Operation system.

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

Topic	Description
"Updating a running system" on page 685	Lists the changes that you can make to a live, running system.
"Add and modify user accounts" on page 590	How to modify user access to a deployed Power SCADA Operation system.
"Editing a PSO to PME ETL job" on page 998	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 configuration changes that you can make on a running system. 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. 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 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.

Alarms, trends, reports

In the project, design an administration page containing a button that executes the `ServerReload` Cicode function.

> **NOTE**: Ensure that this page is only accessible by a logged-in user with the highest Administrator privileges.

While the project is running, ensure that the `[LAN]AllowRemoteReload` parameter is set to "1" in the Citect.INI file located on the target Power SCADA Operation server machine. 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 `ServerReload` 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 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\v2020 R2\bin`).

Web Applications Access and Privilege Levels

Power SCADA Operation 2020 Web Applications use the following access and privilege levels:

Access Levels	Privilege Levels					
	None = 0	Observer = 1	User = 2	Controller =3	Operator = 4	Administrator = 5
AlarmViewer.AcknowledgeAlarm				X	X	X
AlarmViewer.DeleteAny						X
AlarmViewer.EditAny						X
AlarmViewer.Owner				X	X	X
AlarmViewer.SetSystemDefaultItem						X
AlarmViewer.ViewIncidents			X	X	X	X
ApplicationAccess.AlarmViewer			X	X	X	X
ApplicationAccess.HmiApplication		X	X	X	X	X
ApplicationAccess.Event			X	X	X	X
ApplicationAccess.RealtimeData		X	X	X	X	X
ApplicationAccess.RealtimeTrend		X	X	X	X	X
ApplicationAccess.Tgml		X	X	X	X	X
ApplicationAccess.WebConfig		X	X	X	X	X
ConfigurationAccess.Alarms						X
ConfigurationAccess.CustomScripting						X
ConfigurationAccess.MyPreferences		X	X	X	X	X
ConfigurationAccess.Localization						X
ConfigurationAccess.Theme						X
ConfigurationAccess.Security						X
ConfigurationAccess.Tgml				X		X
Diagrams.Owner			X	X	X	X
Diagrams.EditAny						X
Diagrams.DeleteAny						X
Diagrams.SetSystemDefaultItem						X
Diagrams.ControlActions				X	X	X
RealtimeTrend.DeleteAny						X
RealtimeTrend.EditAny						X
RealtimeTrend.Owner				X	X	X
No Access	X					

These access and privilege levels are not the same as the roles and privileges in Citect, Windows, and Active Directory.

- For Citect users, privilege levels are mapped to the Web Applications access levels.
- For local Windows users, the local Windows groups are mapped to the Web Applications access levels.
- For Active Directory Windows users, the Active Directory groups are mapped to the Web Applications access levels.

The Schneider Electric Core Services configuration file is used to configure user access levels. The default installation path is: `C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020 R2\Applications\AppServices\bin\`

`configuration.xml` contains sections for Windows Groups and Citect privilege levels.

Citect Privilege Level Mapping to Web Applications Privilege Levels

Citect privilege levels map to the following Web Applications privilege (and access) levels:

Web Applications Privilege Levels	Citect Privilege Levels
None = 0	• `<ConfigurationItem Key="Priv0" Category="Security" Application="CitectPlatform"> <Value>0</Value> </ConfigurationItem>`
Observer = 1	• `<ConfigurationItem Key="Priv1" Category="Security" Application="CitectPlatform"> <Value>1</Value> </ConfigurationItem>`
User = 2	• `<ConfigurationItem Key="Priv2" Category="Security" Application="CitectPlatform"> <Value>2</Value> </ConfigurationItem>`
Controller = 3	• `<ConfigurationItem Key="Priv3" Category="Security" Application="CitectPlatform"> <Value>3</Value> </ConfigurationItem>` • `<ConfigurationItem Key="Priv4" Category="Security" Application="CitectPlatform"> <Value>3</Value> </ConfigurationItem>`

Web Applications Privilege Levels	Citect Privilege Levels
Operator = 4	• `</ConfigurationItem> <ConfigurationItem Key="Priv5" Category="Security" Application="CitectPlatform"> <Value>4</Value>` • `</ConfigurationItem> <ConfigurationItem Key="Priv6" Category="Security" Application="CitectPlatform"> <Value>4</Value> </ConfigurationItem>`
Administrator = 5	• `<ConfigurationItem Key="Priv7" Category="Security" Application="CitectPlatform"> <Value>5</Value> </ConfigurationItem>` • `<ConfigurationItem Key="Priv8" Category="Security" Application="CitectPlatform"> <Value>5</Value> </ConfigurationItem>`

For example, Citect privilege level 3 ("Priv3") maps to access level 3, which is a Controller. To make Citect privilege 3 be a ‘User’ access level, change the “Value” element for Priv3 to 2:

```
<ConfigurationItem Key="Priv3" Category="Security" Applic-
ation="CitectPlatform">
<Value>2</Value>
</ConfigurationItem>
```

Windows Groups Privilege Level Mapping to Web Applications Privilege Levels

Default local Windows Groups are created during the install. Adding local Windows Users to these groups will grant them the following mapped Web Applications privilege (and access) levels:

Web Applications Privilege Levels	Windows Group Privilege Levels
None = 0	• NA
Observer = 1	• `<ConfigurationItem Key="OsObservers" Category="Security" Application="CitectPlatform"> <Value>PSO_Observers</Value> </ConfigurationItem>`

Web Applications Privilege Levels	Windows Group Privilege Levels
User = 2	• `<ConfigurationItem Key="OsUsers" Category="Security" Application="CitectPlatform"> <Value>PSO_Users</Value> </ConfigurationItem>`
Controller = 3	• `<ConfigurationItem Key="OsControllers" Category="Security" Application="CitectPlatform"> <Value>PSO_Controllers</Value> </ConfigurationItem>`
Operator = 4	• `<ConfigurationItem Key="OsOperators" Category="Security" Application="CitectPlatform"> <Value>PSO_Operators</Value> </ConfigurationItem>`
Administrator = 5	• `<ConfigurationItem Key="OsAdministrators" Category="Security" Application="CitectPlatform"> <Value>PSO_Administrators</Value> </ConfigurationItem>`

For example, all local Windows Users added to the PSO_Controllers group will be granted the Web Application Controller = 3 access level. The values are a semicolon delimited list.

Active Directory Privilege Levels

Active Directory (AD) Windows Users added to local Windows Groups are not supported. AD Users will be authenticated against AD Windows Groups.

For example, if an AD User is in the AD Windows Group “Web_Controllers”, add that group to the **OsControllers** section:

```
<ConfigurationItem Key="OsControllers" Category="Security" Applic-
ation="CitectPlatform">
<Value>PSO_Controllers;Web_Controllers</Value>
</ConfigurationItem>
```

Cybersecurity

This section provides information on how to help secure your system during the Administering phase.

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

Topic	Content
"Windows Updates" on page 692	Windows Update recommendations.

Windows Updates

⚠ 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 arise from Windows updates, they are considered high priority by the Power SCADA Operation with Advanced Reporting and Dashboards development team. They will be evaluated and resolved to deliver patches to enable the continued use of Windows security updates.

Operate

Use the information provided in this chapter to use the Power SCADA Operation runtime.

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

Topic	Description
"Log in to the Power SCADA Runtime" on page 693	Log into the Power SCADA runtime to access system features and tools.
"View the interface" on page 694	How to navigate the Power SCADA Runtime
"View the Alarms/Events Page" on page 696	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 699
"Use Security Viewer" on page 703	How to use the Security Viewer to monitor your system, and how to use the "Security Viewer Filter" on page 705.
"Use the Analysis Page" on page 706	How to use the Analysis Page to view trend data.
"Use the Equipment Pop-Up Page" on page 708	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 714	Customizing advanced reports and design considerations for device communication in Power SCADA Operation with Advanced Reporting and Dashboards.
"Basic Reports" on page 715	A description of the Power SCADA Operation reports.
"Use basic reports" on page 719	How to use basic reports as well as how to create "Rapid access labels (QR codes)" on page 726.
"Web Applications" on page 728	How to use Alarms and Diagrams.

Log in 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 based on your user privilege 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.

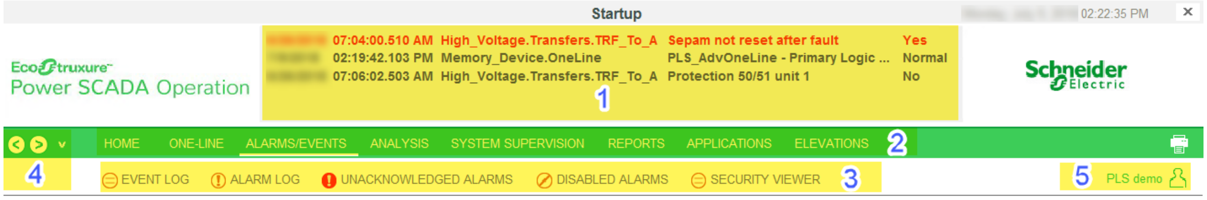
You can start using Power SCADA Runtime.

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 1231.

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:



Startup [redacted] 02:22:35 PM

[redacted] 07:04:00.510 AM High_Voltage.Transfers.TRF_To_A Sepam not reset after fault Yes
[redacted] 02:19:42.103 PM Memory_Device.OneLine PLS_AdvOneLine - Primary Logic ... Normal
[redacted] 07:06:02.503 AM High_Voltage.Transfers.TRF_To_A Protection 50/51 unit 1 No

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 313. 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.

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).

3 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 1246 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 1250 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**.

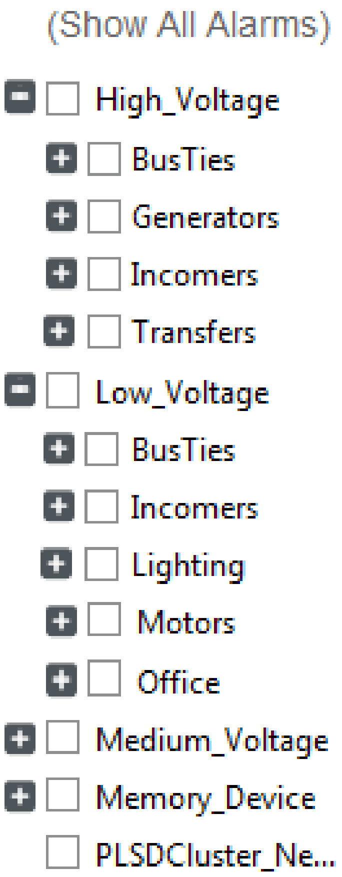
The Event Log displays all alarms and events that have occurred. The Alarms Log displays enabled alarms.

NOTE: If the alarms are not displaying correctly, see "When alarms do not display correctly" on page 815.

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 701 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 701.

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 256).

  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\v2020 R2\bin`).

  Waveforms must be correctly set up before they will display. See "Enabling waveforms for onboard alarms" on page 308 for more information. See "Use the Equipment Pop-Up Page" on page 708 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 2020 R2\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 708

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

Column Name	Description
Category	event; or high, medium, or low alarm
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 701 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

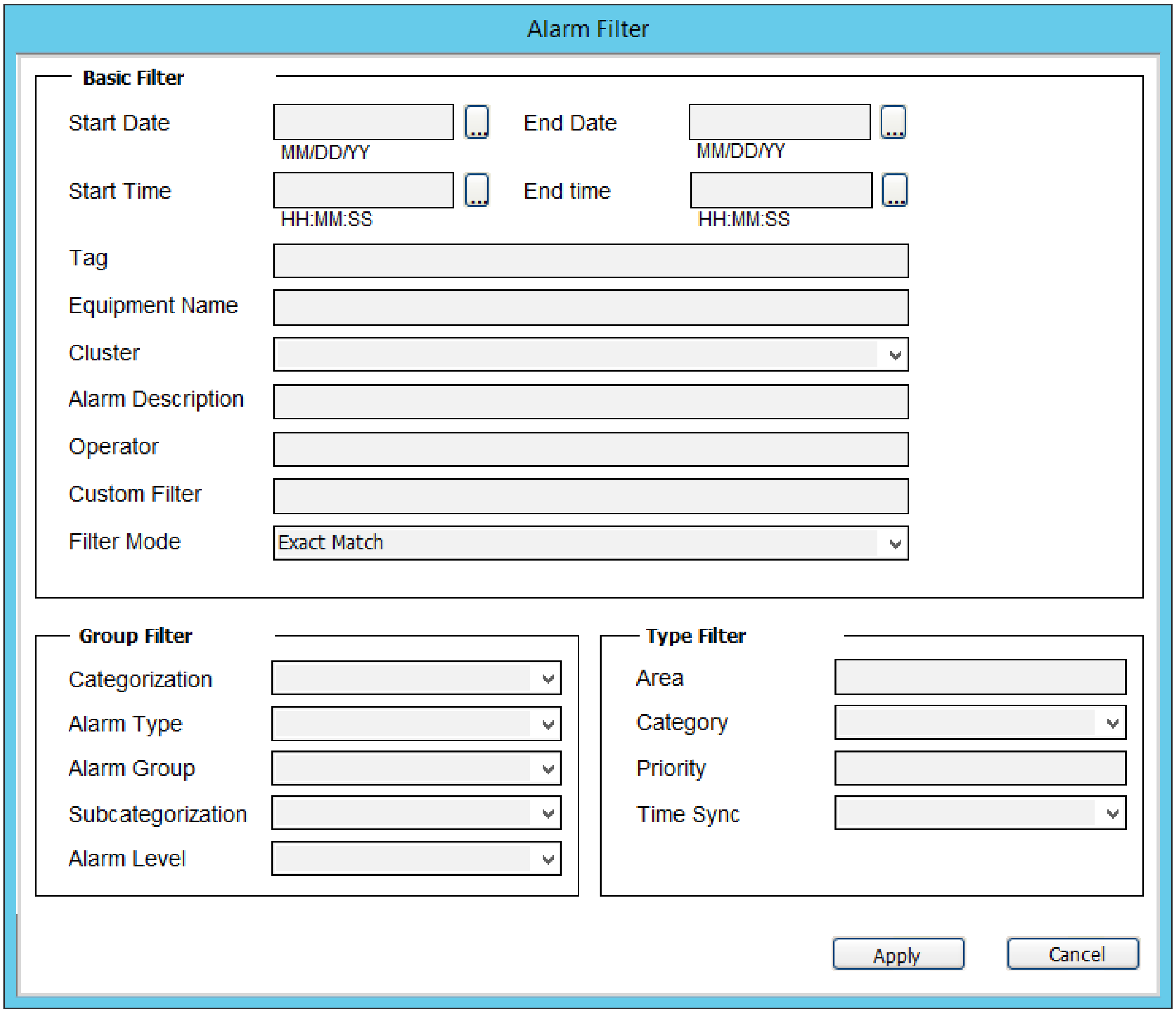
This topic describes the PLSCADA filter form. The information in the Citect filter form is the same, but is presented differently on the page. To change the filter form, use the UsePLSFilter parameter:

[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:



Filter Mode: Exact Match

The table below describes its settings.

Filter Option	Description: Display all alarms for:
Basic Filter box:	

Filter Option	Description: Display all alarms for:
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.
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

Filter Option	Description: Display all alarms for:
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).
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

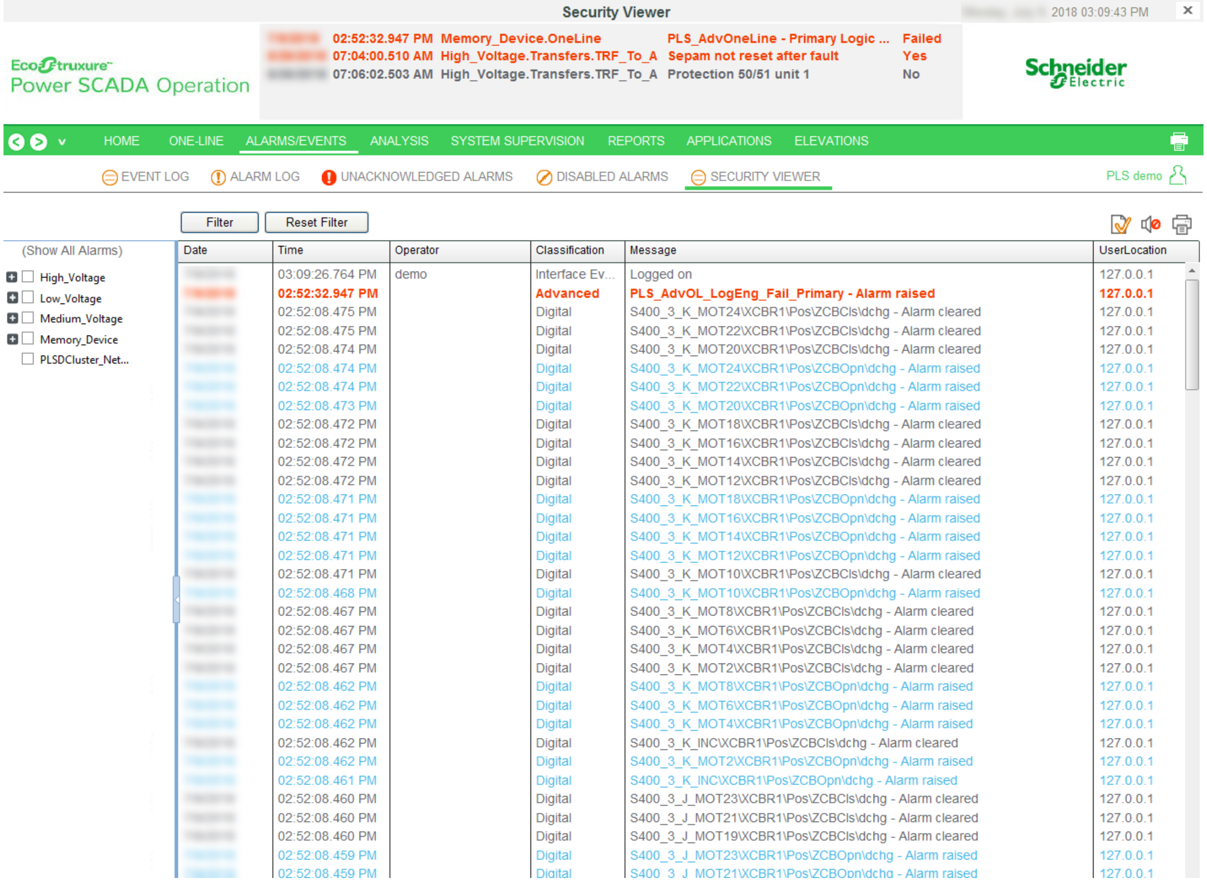
Category Label	Category Number	Priority Number
_PLSALM_HIGH	1001	1
_PLSALM_MEDIUM	1002	2
_PLSALM_LOW	1003	3
_PLSALM_EVENT	1004	0

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**.



[redacted]	02:52:32.947 PM	Memory_Device.OneLine	PLS_AdvOneLine - Primary Logic ...	Failed
[redacted]	07:04:00.510 AM	High_Voltage.Transfers.TRF_To_A	Sepam not reset after fault	Yes
[redacted]	07:06:02.503 AM	High_Voltage.Transfers.TRF_To_A	Protection 50/51 unit 1	No

Date	Time	Operator	Classification	Message	UserLocation
[redacted]	03:09:26.764 PM	demo	Interface Ev...	Logged on	127.0.0.1
[redacted]	02:52:32.947 PM		Advanced	PLS_AdvOL_LogEng_Fail_Primary - Alarm raised	127.0.0.1
[redacted]	02:52:08.475 PM		Digital	S400_3_K_MOT24\XCBR1\Pos\ZCBCls\dchg - Alarm cleared	127.0.0.1
[redacted]	02:52:08.475 PM		Digital	S400_3_K_MOT22\XCBR1\Pos\ZCBCls\dchg - Alarm cleared	127.0.0.1
[redacted]	02:52:08.474 PM		Digital	S400_3_K_MOT20\XCBR1\Pos\ZCBCls\dchg - Alarm cleared	127.0.0.1
[redacted]	02:52:08.474 PM		Digital	S400_3_K_MOT24\XCBR1\Pos\ZCBOpn\dchg - Alarm raised	127.0.0.1
[redacted]	02:52:08.474 PM		Digital	S400_3_K_MOT22\XCBR1\Pos\ZCBOpn\dchg - Alarm raised	127.0.0.1
[redacted]	02:52:08.473 PM		Digital	S400_3_K_MOT20\XCBR1\Pos\ZCBOpn\dchg - Alarm raised	127.0.0.1
[redacted]	02:52:08.472 PM		Digital	S400_3_K_MOT18\XCBR1\Pos\ZCBCls\dchg - Alarm cleared	127.0.0.1
[redacted]	02:52:08.472 PM		Digital	S400_3_K_MOT16\XCBR1\Pos\ZCBCls\dchg - Alarm cleared	127.0.0.1
[redacted]	02:52:08.472 PM		Digital	S400_3_K_MOT14\XCBR1\Pos\ZCBCls\dchg - Alarm cleared	127.0.0.1
[redacted]	02:52:08.472 PM		Digital	S400_3_K_MOT12\XCBR1\Pos\ZCBCls\dchg - Alarm cleared	127.0.0.1
[redacted]	02:52:08.471 PM		Digital	S400_3_K_MOT18\XCBR1\Pos\ZCBOpn\dchg - Alarm raised	127.0.0.1
[redacted]	02:52:08.471 PM		Digital	S400_3_K_MOT16\XCBR1\Pos\ZCBOpn\dchg - Alarm raised	127.0.0.1
[redacted]	02:52:08.471 PM		Digital	S400_3_K_MOT14\XCBR1\Pos\ZCBOpn\dchg - Alarm raised	127.0.0.1
[redacted]	02:52:08.471 PM		Digital	S400_3_K_MOT12\XCBR1\Pos\ZCBOpn\dchg - Alarm raised	127.0.0.1
[redacted]	02:52:08.471 PM		Digital	S400_3_K_MOT10\XCBR1\Pos\ZCBCls\dchg - Alarm cleared	127.0.0.1
[redacted]	02:52:08.468 PM		Digital	S400_3_K_MOT10\XCBR1\Pos\ZCBOpn\dchg - Alarm raised	127.0.0.1
[redacted]	02:52:08.467 PM		Digital	S400_3_K_MOT8\XCBR1\Pos\ZCBCls\dchg - Alarm cleared	127.0.0.1
[redacted]	02:52:08.467 PM		Digital	S400_3_K_MOT6\XCBR1\Pos\ZCBCls\dchg - Alarm cleared	127.0.0.1
[redacted]	02:52:08.467 PM		Digital	S400_3_K_MOT4\XCBR1\Pos\ZCBCls\dchg - Alarm cleared	127.0.0.1
[redacted]	02:52:08.467 PM		Digital	S400_3_K_MOT2\XCBR1\Pos\ZCBCls\dchg - Alarm cleared	127.0.0.1
[redacted]	02:52:08.462 PM		Digital	S400_3_K_MOT8\XCBR1\Pos\ZCBOpn\dchg - Alarm raised	127.0.0.1
[redacted]	02:52:08.462 PM		Digital	S400_3_K_MOT6\XCBR1\Pos\ZCBOpn\dchg - Alarm raised	127.0.0.1
[redacted]	02:52:08.462 PM		Digital	S400_3_K_MOT4\XCBR1\Pos\ZCBOpn\dchg - Alarm raised	127.0.0.1
[redacted]	02:52:08.462 PM		Digital	S400_3_K_INC\XCBR1\Pos\ZCBCls\dchg - Alarm cleared	127.0.0.1
[redacted]	02:52:08.462 PM		Digital	S400_3_K_MOT2\XCBR1\Pos\ZCBOpn\dchg - Alarm raised	127.0.0.1
[redacted]	02:52:08.461 PM		Digital	S400_3_K_INC\XCBR1\Pos\ZCBOpn\dchg - Alarm raised	127.0.0.1
[redacted]	02:52:08.460 PM		Digital	S400_3_J_MOT23\XCBR1\Pos\ZCBCls\dchg - Alarm cleared	127.0.0.1
[redacted]	02:52:08.460 PM		Digital	S400_3_J_MOT21\XCBR1\Pos\ZCBCls\dchg - Alarm cleared	127.0.0.1
[redacted]	02:52:08.460 PM		Digital	S400_3_J_MOT19\XCBR1\Pos\ZCBCls\dchg - Alarm cleared	127.0.0.1
[redacted]	02:52:08.459 PM		Digital	S400_3_J_MOT23\XCBR1\Pos\ZCBOpn\dchg - Alarm raised	127.0.0.1
[redacted]	02:52:08.459 PM		Digital	S400_3_J_MOT21\XCBR1\Pos\ZCBOpn\dchg - Alarm raised	127.0.0.1

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\v2020 R2\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.

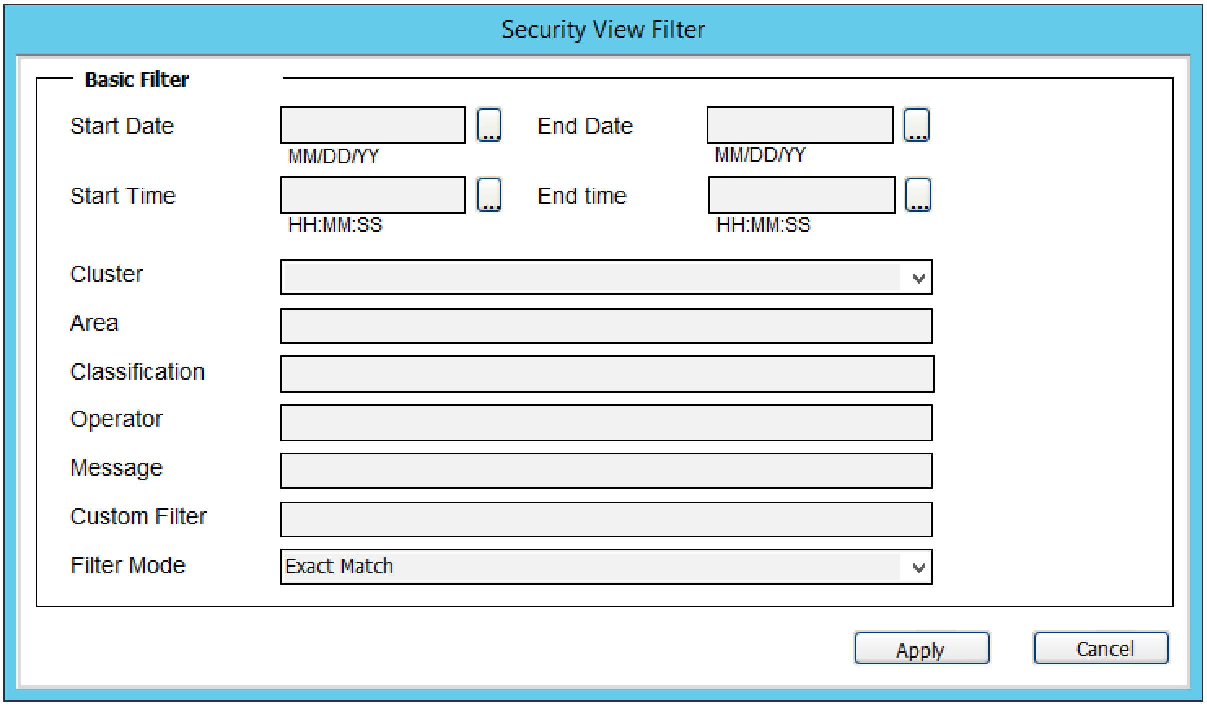
To remove a column:

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 705.

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.



Filter Mode: Exact Match

The following table describes the Security View Filter settings:

Filter option	Description: Display 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.

Filter option	Description: Display alarms for:
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\v2020 R2\bin\Help\Citect SCADA`).
Classification	The class of the event. See Alarm SOE fields in the Citect SCADA help file (`...\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020 R2\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\v2020 R2\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:

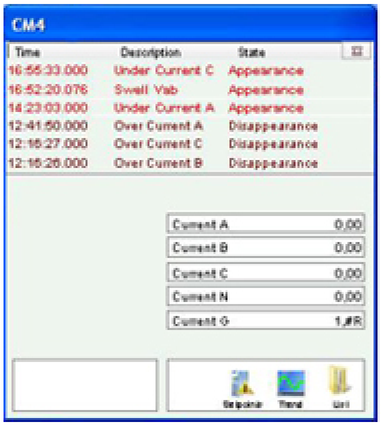
NOTE: If one of the pens returns a value of "1.#R,' the tag selected was not valid; no number could be reported for it. None of the values for any of the pens in the trend will be updated. To solve this issue, close the trend and open it again. This time, do not include the pen that gave the invalid 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 graphic 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 590 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.



CM4

Time	Description	State
16:55:33.000	Under Current C	Appearance
16:52:20.076	Swell Vab	Appearance
14:23:03.000	Under Current A	Appearance
12:41:50.000	Over Current A	Disappearance
12:16:27.000	Over Current C	Disappearance
12:16:26.000	Over Current B	Disappearance

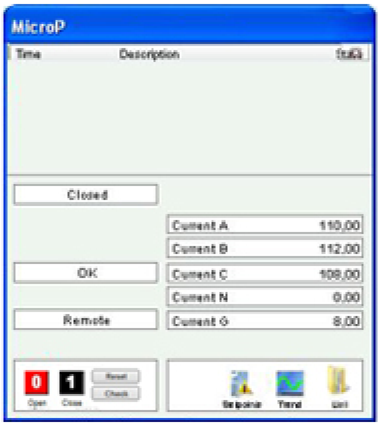
Current A 0,00

Current B 0,00

Current C 0,00

Current N 0,00

Current G 1,#R



MicroP

Time	Description	State

Closed

OK

Remote

Current A 110,00

Current B 112,00

Current C 108,00

Current N 0,00

Current G 8,00

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 256).

  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\v2020 R2\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 "Enabling waveforms for onboard alarms" on page 308 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 325 for information on setup. See "Perform IEC 61850 advanced control" on page 712 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 2020 R2 Bin folder (for example: `C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020 R2\bin`

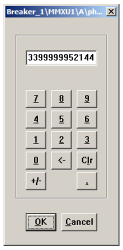
NOTE: Waveforms are not supported on View-only Clients.

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.



3399999952144

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 590. 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\v2020 R2\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 2020 > Power SCADA Operation web-based help).

See also: "View the Alarms/Events Page" on page 696

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.

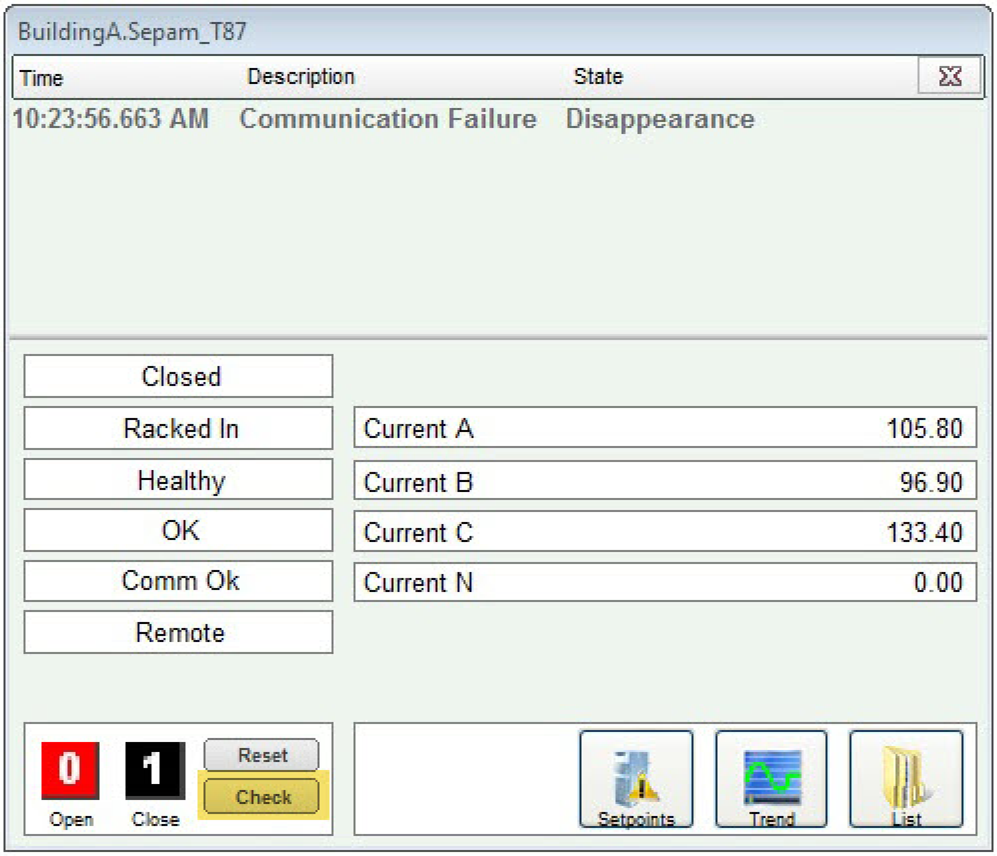
⚠ WARNING

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.

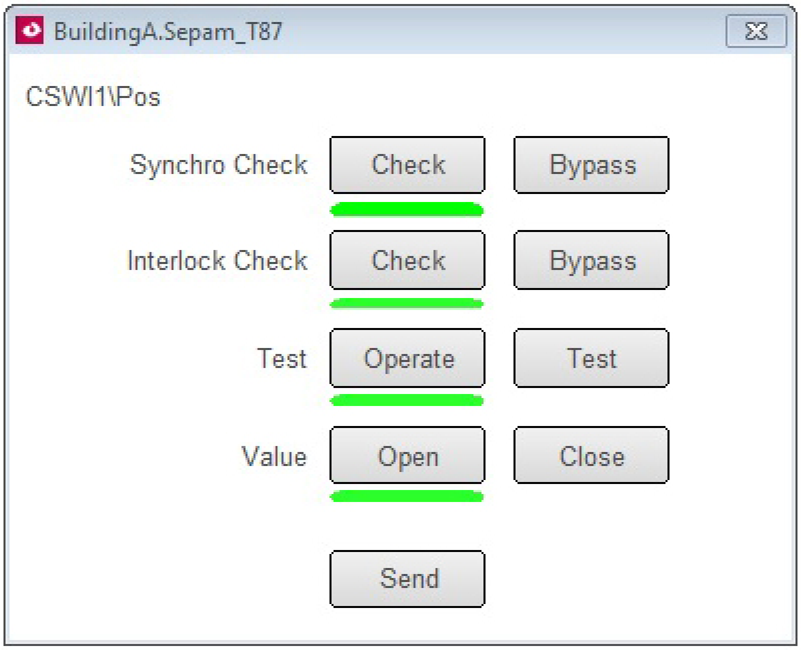
To access the advanced control window, open the equipment genie pop-up page on the one-line. Note that after you enable this feature, there is a **Check** button on the lower left:


BuildingA.Sepam_T87

Time	Description	State
10:23:56.663 AM	Communication Failure	Disappearance

Current A	105.80
Current B	96.90
Current C	133.40
Current N	0.00

Click **Check**. The advanced control window opens:


BuildingA.Sepam_T87

CSWI1\Pos

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.

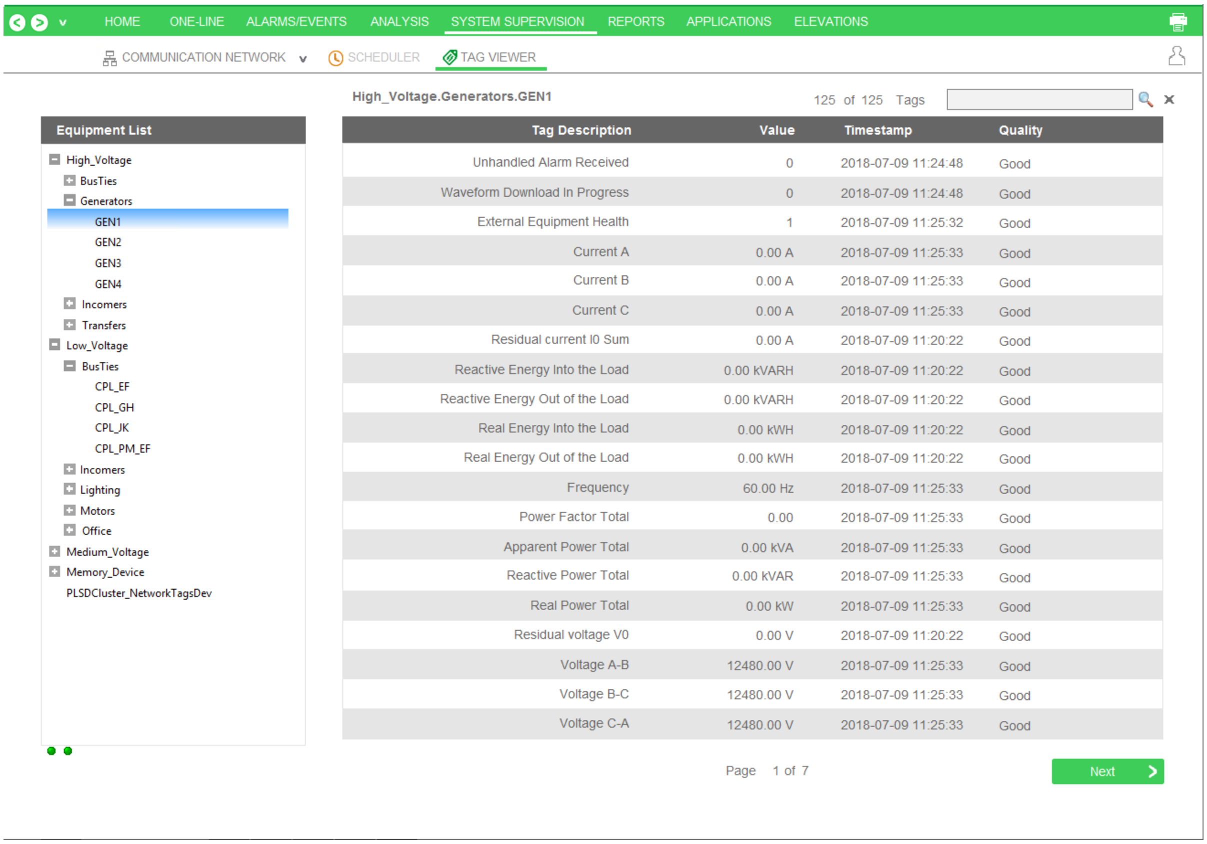
View the Tag Viewer

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:



High_Voltage.Generators.GEN1

125 of 125 Tags

Tag Description	Value	Timestamp	Quality
Unhandled Alarm Received	0	2018-07-09 11:24:48	Good
Waveform Download In Progress	0	2018-07-09 11:24:48	Good
External Equipment Health	1	2018-07-09 11:25:32	Good
Current A	0.00 A	2018-07-09 11:25:33	Good
Current B	0.00 A	2018-07-09 11:25:33	Good
Current C	0.00 A	2018-07-09 11:25:33	Good
Residual current I0 Sum	0.00 A	2018-07-09 11:20:22	Good
Reactive Energy Into the Load	0.00 kVARH	2018-07-09 11:20:22	Good
Reactive Energy Out of the Load	0.00 kVARH	2018-07-09 11:20:22	Good
Real Energy Into the Load	0.00 kWH	2018-07-09 11:20:22	Good
Real Energy Out of the Load	0.00 kWH	2018-07-09 11:20:22	Good
Frequency	60.00 Hz	2018-07-09 11:25:33	Good
Power Factor Total	0.00	2018-07-09 11:25:33	Good
Apparent Power Total	0.00 kVA	2018-07-09 11:25:33	Good
Reactive Power Total	0.00 kVAR	2018-07-09 11:25:33	Good
Real Power Total	0.00 kW	2018-07-09 11:25:33	Good
Residual voltage V0	0.00 V	2018-07-09 11:20:22	Good
Voltage A-B	12480.00 V	2018-07-09 11:25:33	Good
Voltage B-C	12480.00 V	2018-07-09 11:25:33	Good
Voltage C-A	12480.00 V	2018-07-09 11:25:33	Good

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 318.
- There must be data logged for the project. See "Use the Analysis Page" on page 706 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 609 for instructions.

After you have logged trend information, you can create and view basic reports. In the Power SCADA Operation, click the **Reports** tab and then choose the basic report type you want to create:

- "Single Device Usage reports" on page 716
- "Multi Device Usage reports" on page 716
- "Tabular reports" on page 717
- "Trend reports" on page 718
- "Rapid access labels (QR codes)" on page 726

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 "Read, Export, Print, and Edit Basic Reports" on page 722.

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 "Read, Export, Print, and Edit Basic Reports" on page 722.

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 "Read, Export, Print, and Edit Basic Reports" on page 722.

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 "Read, Export, Print, and Edit Basic Reports" on page 722.

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 719
- "Configure email settings to send basic reports" on page 320
- "Email basic reports" on page 723
- "Read, Export, Print, and Edit Basic Reports" on page 722

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 318.

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 726, 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 324.

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):

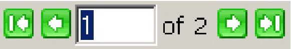
```
C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020
R2\Applications\Reporting\ReportConfigurations
```

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 following table describes the toolbar 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 printable 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 723

Exporting a Basic Report

Before you can print a basic report, you must export it into one of the following printable formats:

- PDF
- Web Archive
- Word Document
- XML File
- XLS Document

To export a report:

1. While viewing the report, select a printable 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 320 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 719 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

Text Field	Required Setting	Description
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\v2020 R2\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\v2020 R2\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.

Prerequisites

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 719 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.

Creating 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.

Web Applications

Web Applications is the main interface for accessing Power SCADA Operation power system information. Use Web Applications to view real-time data, alarms, historical trends, key performance indicators, reports, and other information about the power system you are monitoring. Web Applications also provides a number of configuration settings and tools to configure and customize your Power SCADA Operation system.

The following is a list of applications for accessing power system information through Web Applications:

Application	Function
Alarms	View and analyze Incidents, Alarms, and Events; Acknowledge alarms.
Diagrams	View low level, historical and real-time data in one-line and graphics diagrams.
Trends	View trends for real-time and historical data.

For a list of configuration tools and settings, see Web Applications settings.

When you open Web Applications, you are prompted to log in with your username and password. The access level assigned to your username determines which applications and which functions are available to you. See "Assign and control user privileges" on page 686 for details.

Opening Web Applications

Open Web Applications from Power SCADA Operation folder on your desktop, the Schneider Electric folder on the Start Screen, or by entering the Power SCADA Operation server URL into your browser Address bar. For example: `https://srv1.MyCompany.com/Web`

To reduce the risk of cybersecurity attacks, access Web Applications only from client computers and not from the Power SCADA Operation server.

Specifying which application to open first

When you connect to Web Applications through a client computer, the application whose link is on the left of the series of application links opens in the browser. To specify a different application to open first, add one of the following application query parameters into the Web address.

/#Diagrams	/#Alarms
/#Settings	/#Trends

For example: `http://srv1.MyCompany.com/Web/#Alarms` opens the Alarms application in the browser.

Opening Web Applications without a banner

You can open any of the Web Applications by itself without showing the Web Applications banner and navigation bar.

To open a Web Applications without a banner:

In the browser address bar, enter the PSO server URL with `/<application name>`.

For example: `http://srv1.MyCompany.com/Trends` opens the Trends application in the browser without the Web Applications UI elements.

Web Applications User Interface

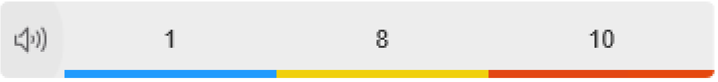
The top right of the banner contains:

- Your user name: The user name you used to log in.
- **Logout** link: Logs you out of Web Applications and returns you to the log in page.
- **Help** link: Opens the browser-based online help for the Web Applications component and the integrated applications.

Alarm Annunciator

The Alarm Annunciator shows information on the number of active and unacknowledged "Alarms" on page 744. It is displayed in the banner area of the Web Applications and is visible from any of the Power SCADA Operation Web Applications. The Annunciator alerts you to any new alarms that are occurring in the system. You can configure it to play a sound when certain alarm conditions are met.

The Annunciator looks like this:



In this example, the Annunciator shows:

- 1 low priority, active and unacknowledged Alarm (blue)
- 8 medium priority, active and unacknowledged Alarms (yellow)
- 10 high priority, active and unacknowledged Alarms (red)

The presence of the speaker icon indicates that it is configured to play a sound when new active and unacknowledged alarms occur. Click the speaker icon to mute or unmute the alarm sound.

You must have controller, operator, or supervisor-level access to see the Annunciator. If you have observer or user-level access, it is not displayed.

Library Pane

The library pane contains items and configuration options for the selected application. To show or hide the library pane, click the bar on the right or left side of the display area.

Display Pane

The display pane loads the data visualization selected in the configuration pane.

Time Display in Web Applications

Most of the information displayed in the Web Applications is time based, such as timestamped real-time data and historical data. In a Power SCADA Operation system the server converts timestamps to the local server time zone.

Power SCADA Operation supports multi-site configurations where the devices/sources, the server, and the client are located in different time zones. For example, a user in time zone A accesses the Power SCADA Operation server which is located in time zone B. The monitoring devices that are providing the data are located in time zone C. To configure devices in multiple time zones, see "Time zone settings" on page 646.

Alarms

The alarm viewer is the user interface (UI) for the Alarms application. Use the alarm viewer to see software generated and device based alarms in Power SCADA Operation.

The alarm viewer UI has two main areas, the view library and the alarms display. To see alarm information in the alarms display, select a view in the view library. The library has predefined system views and you can create additional custom views. For more information, see: Alarm Viewer UI.

TIP: You can open the alarm viewer from the **ALARMS** link in the Web Applications banner.

⚠ 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

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, or equipment damage.

View types

There are two types of views:

- Alarm Status – for all alarms.
- Alarm History – for recent alarms.

Status views

Use status views to see existing alarm definitions in the system, their present state, how often they occurred, their priority, and other relevant information. The following predefined status views are available in Power SCADA Operation:

View Name	Description
Active Alarms	This view shows alarms that are in the active state. It includes low, medium and high priority alarms from all sources and all categories. This view does not include General Event and Unassociated Dropout type alarms.
All Alarms	This view shows all low, medium and high priority alarms in the system regardless of state, category, and source.
Unacknowledged Alarms	This view shows unacknowledged alarms. It includes low, medium and high priority alarms from all sources and all categories that are in the active or inactive state.

History views

Use history views to see a record of Incidents, alarm instances, and events that happened in the past. The following predefined history views are available in Power SCADA Operation:

View Name	Description
Asset Monitoring Incidents	This view shows Incidents that are categorized as Asset Monitoring and are in the active or unacknowledged state. It includes low, medium and high priority Incidents from all sources.

View Name	Description
Clutter	This view shows Incidents that are categorized as General Clutter and are in the active or unacknowledged state. It includes low, medium and high priority Incidents from all sources.
Load Loss Incidents	This view shows incidents that are categorized as Power Quality (Over Voltage, Swell, Under Voltage, Interruption, Sag, Transient, or Unclassified Disturbance) and that recorded a sustained load loss after a voltage sag. It includes low, medium and high priority Incidents that are active or unacknowledged, from all sources.
Power Quality Incidents	This view shows Incidents that are categorized as Power Quality and are in the active or unacknowledged state. It includes low, medium and high priority Incidents from all sources.
Recent Alarms	This view shows alarm instances that are in the active or unacknowledged state. It includes low, medium and high priority alarms from all sources and all categories. This view does not include Unassociated Dropout and Clock/Time type alarms.
Recent Events	This view shows events of all priorities from all sources.
Recent Incidents	This view shows Incidents that are in the active or unacknowledged state. It includes low, medium and high priority Incidents from all sources and all categories. This view does not include General Alarms for type Clutter.
System Health	This view shows Diagnostics type alarm instances that are in the active or unacknowledged state. It includes low, medium and high priority alarms from all sources. This view does not include Diagnostics Alarms of type Clock/Time and Device Settings.

Incidents, Alarms, and Events

Incidents

Incidents provide a high-level view. They represent real world power events, such as disturbances or faults. An incident combines alarms, waveforms, and burst data from many sources in the system into a single representation of the power event. You can look at an incident and see how the different pieces of information are linked together, instead of having to analyze each data point individually. Use incidents as a starting point for your alarm analysis.

For more information, see:

- Incidents
- Viewing incidents
- Incident History UI

Alarms

Alarms provide information on the state and history of alarm conditions that are defined for specific sources and measurements in the system. Use alarms to monitor the state of your power system and to investigate specific details as part of an Incident analysis.

For more information, see:

- Alarms
- Viewing alarms
- Alarm status UI
- Alarm history UI

Events

Events are records of activities in the system. Activities are performed by users, the system software, or the connected devices. Events are logged and displayed as they happen in the system without any processing or aggregation. Power SCADA Operation uses event records to determine alarm types and states. Use events for low level investigations and detailed root cause analysis.

For more information, see:

- Events
- Viewing events
- Event history UI

Alarm Acknowledgment

You can acknowledge alarms in status views and history views. If you acknowledge alarms through an incident history view, all alarms that are part of this incident will be acknowledged. Whenever you acknowledge an alarm, from any of these locations, you are acknowledging the alarm definition itself, not a particular instance of it. That means acknowledging an alarm marks it as Acknowledged and resets its Unacknowledged occurrence counter. For more information, see Acknowledging alarms.

Analysis tools

The alarm viewer includes tools for analyzing the causes and impacts of alarm events. Some of these tools are for very specific alarm types, others can be used for a broad range of alarms.

For details on the different tools, see:

Timeline analysis

- Timeline analysis
- Viewing a timeline analysis
- Timeline Analysis UI

Waveforms

- Waveforms
- Viewing waveforms
- Waveforms UI

Time display

See for information on how time is displayed in a system where the monitoring devices, the Power SCADA Operation Web server, and the Web client (browser) are located in different time zones.

Terminology

See Alarms terminology for definitions of the terms used in the Alarms application.

For information on how to configure Alarms, see Alarms configuration.

Viewing incidents

View incidents to investigate system issues, to analyze what happened during a power disturbance or to identify root causes.

To view incidents:

1. In the alarm viewer, open an existing incident view from the view library or add a new View.
2. View the incident information displayed in the alarms display pane.

   (Optional) In the view library, right-click the view name or click **Options** ⋮, and then select **Edit** to open the view settings. You can also open the view settings by double-clicking the view name. Adjust the settings for View Type, Priority, State, Sources, and Categories to customize the view if necessary. **Save** the modified view settings or click **Cancel** to discard the changes.

For information on how to configure Alarms , see Alarms configuration.

Related topics:

- Viewing incidents
- Viewing alarms
- Viewing events
- Viewing a timeline analysis
- Viewing waveforms
- Acknowledging alarms
- "Displaying alarms in the runtime banner area" on page 751

Related references:

- Alarms
- Incidents

- Alarms
- Events
- Timeline analysis
- Waveforms
- Alarms terminology
- Alarms UI
- Timeline Analysis UI
- Waveforms UI

Viewing alarms

View Alarm Status to assess the state of the monitored power system and to respond to important events and issues. View Alarm History for root cause analysis and to understand the sequence of events.

To view Alarm Status or Alarm History:

1. In the alarm viewer, open an existing alarm status or alarm history view from the view library, or add a new View.
2. View the alarm information displayed in the alarms display pane.

   (Optional) In the view library, right-click the view name or click **Options** ⋮, and then select **Edit** to open the view settings. You can also open the view settings by double-clicking the view name. Adjust the settings for View Type, Priority, State, Sources, and Categories to customize the view if necessary. **Save** the modified view settings or click **Cancel** to discard the changes.

For information on how to configure Alarms , see Alarms configuration.

Related topics:

- Viewing incidents
- Viewing alarms
- Viewing events
- Viewing a timeline analysis
- Viewing waveforms
- Acknowledging alarms
- "Displaying alarms in the runtime banner area" on page 751

Related references:

- Alarms
- Incidents
- Alarms
- Events

- Timeline analysis
- Waveforms
- Alarms terminology
- Alarms UI
- Timeline Analysis UI
- Waveforms UI

Viewing events

View events to investigate system activities in Power SCADA Operation or to troubleshoot unexpected system behavior.

To view events:

1. In the alarm viewer, open an existing event view from the view library or add a new View.
2. View the event Information displayed in the alarms display pane.

   (Optional) In the view Library, right-click the view name or click **Options** ⋮ for this view, and select **Edit** to open the view settings. You can also open the view settings by double-clicking the view name. Adjust the settings for View Type, Priority and Sources to customize the view if necessary. **Save** the modified view settings or click **Cancel** to discard the changes.

TIP: Double-clicking an event in the events display table opens the associated alarm.

For information on how to configure Alarms , see Alarms configuration.

Related topics:

- Viewing incidents
- Viewing alarms
- Viewing events
- Viewing a timeline analysis
- Viewing waveforms
- Acknowledging alarms
- "Displaying alarms in the runtime banner area" on page 751

Related references:

- Alarms
- Incidents
- Alarms
- Events
- Timeline analysis
- Waveforms

- Alarms terminology
- Alarms UI
- Timeline Analysis UI
- Waveforms UI

Viewing a timeline analysis

View a timeline analysis to investigate the sequence of events that occurred during a single incident, multiple incidents, or alarms.

To view a timeline analysis for an incident:

1. In the alarm viewer, open an existing incident view from the view library or add a new View.
2. Find the incident for which you want to view the analysis, and click **Open Timeline Analysis** to open the timeline window.

   (Optional) Edit the view settings for the timeline analysis and save the view for future reference.

To view a timeline analysis for multiple incidents:

1. In the alarm viewer, open an existing Incident view from the view library or add a new View.
2. Find and select the incidents for which you want to view the analysis.

   > **TIP**: Use `Ctrl+Click` to select individual alarms, use `Shift+click` to select a block of alarms.

3. From the in the **Options** menu ≡ at the top of the alarms display pane, select **Open Timeline Analysis on selection**.

To view a timeline analysis for an alarm:

1. In the alarm viewer, open an existing alarm history view from the view library or add a new View.
2. Find the alarm for which you want to view the analysis and click **Open Details**.
3. In the alarm details window, click **Timeline Analysis**.

For information on how to configure Alarms , see Alarms configuration.

Related topics:

- Viewing incidents
- Viewing alarms
- Viewing events
- Viewing a timeline analysis
- Viewing waveforms
- Acknowledging alarms
- "Displaying alarms in the runtime banner area" on page 751

Related references:

- Alarms
- Incidents
- Alarms
- Events
- Timeline analysis
- Waveforms
- Alarms terminology
- Alarms UI
- Timeline Analysis UI
- Waveforms UI

Viewing waveforms

View waveforms to investigate power quality events and identify root causes of disturbances.

To view waveforms:

1. In the alarm viewer, open an existing Incident history view or alarm history view from the view library or add a new View.
2. Find the incident or alarm for which you want to view waveforms, and click **Details** .

   You can also open Details by double-clicking the incident or alarm instance.
3. In Details, click **Waveforms**.

   > **TIP**: Click **Open Representative Waveform** to see the representative waveform for this Incident or alarm instance.
4. View the waveforms associated with the incident or alarm instance.

   (Optional) Click **Inspect** a waveform to see more details and to analyze the waveform.

For information on how to configure Alarms , see Alarms configuration.

Related topics:

- Viewing incidents
- Viewing alarms
- Viewing events
- Viewing a timeline analysis
- Viewing waveforms
- Acknowledging alarms
- "Displaying alarms in the runtime banner area" on page 751

Related references:

- Alarms
- Incidents
- Alarms
- Events
- Timeline analysis
- Waveforms
- Alarms terminology
- Alarms UI
- Timeline Analysis UI
- Waveforms UI

Acknowledging alarms

Acknowledge alarms to show that these alarms are managed. Record relevant information related to the alarms, as part of the acknowledgment, for future reference. There are many ways to acknowledge alarms.

NOTE: You can acknowledge alarms in status views and history views. If you acknowledge alarms through an incident history view, all alarms that are part of this incident will be acknowledged. Whenever you acknowledge an alarm from any of these locations, you are acknowledging the alarm definition itself, not a particular instance of it. That means acknowledging an alarm marks it as Acknowledged and resets its Unacknowledged occurrence counter.

Acknowledging through an alarm status view

To acknowledge a single alarm:

1. In the alarm viewer, open an existing alarm status view from the view library or add a new View.
2. In the alarms display pane, find the alarm definition you want to acknowledge.

   (Optional) In the view library, right-click the view name or click **Options**⋮, and then select **Edit** to open the view settings. You can also open the view settings by double-clicking the view name. Adjust the settings for View Type, Priority, State, Sources, and Categories to customize the view if necessary. **Save** the modified view settings or click **Cancel** to discard the changes.
3. In the **Acknowledgment** column for this alarm definition, click **Acknowledge**. This opens the Acknowledge Alarms window. You can also open the details for this alarm definition and click **Acknowledge** in the details window to open Acknowledge Alarms.
4. In Acknowledge Alarms, click **Acknowledge**.
   (Optional) In the **Comment** box, enter notes related to the alarm definition.

> **TIP**: To later view the acknowledgment notes, open the alarm details and click **History** on the top right. The acknowledgment with the note is shown in the alarm instance history display.

To acknowledge multiple alarms:

1. In the alarm viewer, open an existing alarm status view from the view library or [add a new View](#).
2. In the alarms display pane, find and select the alarm definitions you want to acknowledge in the alarms table.

   > **TIP**: Use `Ctrl+Click` to select individual alarms, use `Shift+click` to select a block of alarms.

   (Optional) In the view library, right-click the view name or click **Options** ⋮, and then select **Edit** to open the view settings. You can also open the view settings by double-clicking the view name. Adjust the settings for View Type, Priority, State, Sources, and Categories to customize the view if necessary. **Save** the modified view settings or click **Cancel** to discard the changes.
3. Click **Options** ☰ in the top right corner of the alarms pane, and then click **Acknowledge Selected** in the options menu. This opens the Acknowledge Alarms window.
4. In Acknowledge Alarms, click **Acknowledge**.
   (Optional) In the **Comment** box, enter notes related to the alarm definitions.

   > **TIP**: To later view the acknowledgment notes, open the alarm details, for any of the alarms, and click **History** on the top right. The acknowledgment with the note is shown in the alarm instance history display.

To acknowledge all alarms in a view:

1. In the alarm viewer, open an existing alarm status view from the view library or [add a new View](#).
2. Click **Options** ☰ in the top right corner of the alarms pane, and then click **Acknowledge All** in the options menu. This opens the Acknowledge Alarms window.
3. In Acknowledge Alarms, click **Acknowledge**.
   (Optional) In the **Comment** box, enter notes related to the alarm definitions.

   > **TIP**: To later view the acknowledgment notes, open the alarm details, for any of the alarms, and click **History** on the top right. The acknowledgment with the note is shown in the alarm instance history display.

Acknowledging through an alarm history view

To acknowledge an alarm:

1. In the alarm viewer, open an existing alarm history view from the view library or add a new View.
2. In the alarms display pane, find the alarm you want to acknowledge.

   (Optional) In the view library, right-click the view name or click **Options** ⋮, and then select **Edit** to open the view settings. You can also open the view settings by double-clicking the view name. Adjust the settings for View Type, Priority, State, Sources, and Categories to customize the view if necessary. **Save** the modified view settings or click **Cancel** to discard the changes.
3. Open the details for this alarm by clicking on Open Details or double-clicking the alarm.
4. In Alarm Details, click **Acknowledge**. This opens the Acknowledge Alarms window.
5. In Acknowledge Alarms, click **Acknowledge**.
   (Optional) In the **Comment** box, enter notes related to the alarm.

   TIP: To later view the acknowledgment notes, open the alarm details and click **History** on the top right. The acknowledgment with the note is shown in the alarm instance history display.

Acknowledging through an incident history view

To acknowledge all alarms in an incident:

1. In the alarm viewer, open an existing incident history view from the view library or add a new View.
2. In the alarms display pane, find the incident you want to acknowledge.

   (Optional) In the view library, right-click the view name or click **Options** ⋮, and then select **Edit** to open the view settings. You can also open the view settings by double-clicking the view name. Adjust the settings for View Type, Priority, State, Sources, and Categories to customize the view if necessary. **Save** the modified view settings or click **Cancel** to discard the changes.
3. Open the details for this incident by clicking on Open Details or double-clicking the incident.
4. In Incident Details, click **Acknowledge**. This opens the Acknowledge Alarms window.
5. In Acknowledge Alarms, click **Acknowledge**.
   (Optional) In the **Comment** box, enter notes related to the alarms.

   TIP: To later view the acknowledgment notes, open the alarm details, for any of the alarms, and click **History** on the top right. The acknowledgment with the note is shown in the alarm instance history display.

For information on how to configure Alarms , see Alarms configuration.

Related topics:

- Viewing incidents
- Viewing alarms
- Viewing events
- Viewing a timeline analysis
- Viewing waveforms
- Acknowledging alarms
- "Displaying alarms in the runtime banner area" on page 751

Related references:

- Alarms
- Incidents
- Alarms
- Events
- Timeline analysis
- Waveforms
- Alarms terminology
- Alarms UI
- Timeline Analysis UI
- Waveforms UI

Incidents

Incidents in Power SCADA Operation represent real world power events, such as disturbances or faults. An incident combines alarms, waveforms, and burst data from many sources in the system into a single representation of the power event. Instead of having to analyze each data point individually, you can look at an incident and see how the different pieces of information are linked together.

Power SCADA Operation uses alarm types and alarm start times as criteria to determine which alarms to group into a specific incident. The start of an alarm marks the beginning of an incident. Any alarm of a similar type, that starts within a certain time interval is considered part this same incident. The grouping time interval is always based on the most recent alarm in the incident, which means that the counter is restarted every time a new alarm is added to the incident. If there is no more alarm that falls inside the interval, the incident is complete. The maximum duration for an incident is 24 hours and the maximum number of alarms in an incident is 500. A new incident is started the next time an alarm is recorded. See "Alarm to Incident Mapping" on page 1282 for more information.

The incident grouping time interval is different for different alarm types. For example, Over Voltage alarms have a time interval of 5 minutes. If a new Over Voltage alarm occurs within 5 minutes, for any source, it is grouped into the same incident. To make it easier to analyze incidents, Power SCADA Operation categorizes them into types. The incident types are based on the alarm types.

The following table shows the Incident types and the grouping time intervals for each type:

Category	Type	Grouping Time Interval
Power Quality	Flicker	5 minutes
	Frequency Variation	5 minutes
	Harmonics	5 minutes
	Interruption	5 minutes *
	Over Voltage	5 minutes *
	Sag	20 seconds *
	Swell	20 seconds *
	Transient	20 seconds *
	Unbalance	5 minutes
	Unclassified Disturbance	20 seconds *
	Under Voltage	5 minutes *
Asset Monitoring	Arc Flash	60 seconds
	Backup Power	80 minutes
	Current Monitor	5 minutes
	Protection	5 minutes
	Thermal Monitor	30 minutes
Energy Management	Air	5 minutes
	Demand	5 minutes
	Electricity	5 minutes
	Gas	5 minutes
	Power Factor	5 minutes
	Steam	5 minutes
	Water	5 minutes
General	Clutter	1 day
	General Setpoints	5 minutes
Diagnostics	Communication Status	10 minutes
	Device Status	5 minutes
	System Status	0 seconds (one incident per alarm)

* These grouping intervals time settings are default settings. The defaults are extended automatically to include power quality alarms that are outside the interval but close enough that they could be related to the incident.

For information on how to configure Alarms , see Alarms configuration.

Related topics:

- Viewing incidents
- Viewing alarms
- Viewing events
- Viewing a timeline analysis
- Viewing waveforms
- Acknowledging alarms

Related references:

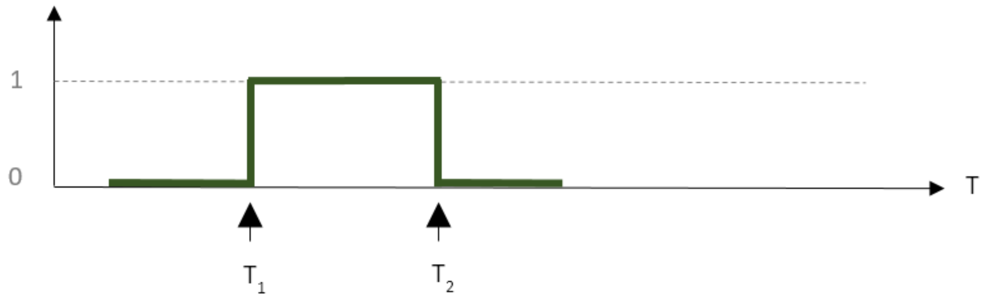
- Alarms
- Incidents
- Alarms
- Events
- Timeline analysis
- Waveforms
- Alarms terminology
- Alarms UI
- Timeline Analysis UI
- Waveforms UI

Alarms

An alarm is a defined condition for a particular source in Power SCADA Operation. The software or the device monitors this condition and records when the condition is met and when not. For example, you can define an Over Voltage alarm for a certain monitoring device in the system. When the voltage threshold is exceeded on this device, the alarm goes active. When the voltage drops below the threshold, the alarm goes inactive. The next time the voltage on this device goes above the threshold again, the same alarm goes active again. An alarm is always associated with a single source and a single measurement.

Some alarms are based on instantaneous events such as a voltage transient, others are based on a condition that lasts a certain period of time such as an over voltage condition. For lasting conditions, the alarm goes from an inactive state to an active state while the condition lasts and then back to an inactive state when the condition is over. Instantaneous alarms are always shown in an inactive state.

The following diagram shows an alarm that is based on a lasting condition. The alarm goes active at the time T_1 and inactive at T_2. The time interval between T_1 and T_2 can be short or long.

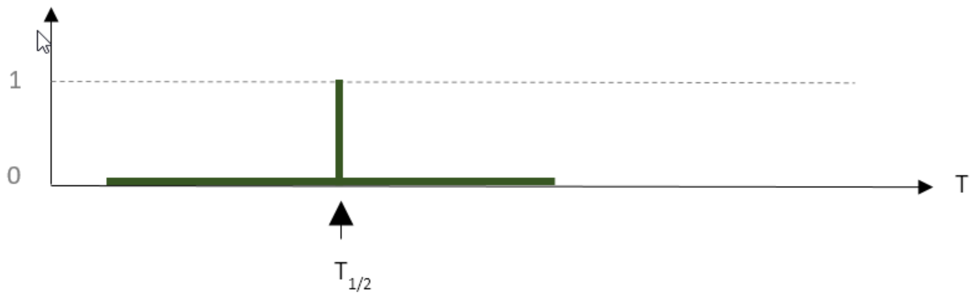


0 = inactive alarm state; 1 = active alarm state; T = time

T_1 = Alarm goes active

T_2 = Alarm goes inactive

The following diagram shows an instantaneous alarm. For this alarm, the start time T_1 and end time T_2 are identical.



0 = inactive alarm state, 1 = active alarm state; T = time

$T_{1/2}$ = Alarm goes active and immediately inactive again

For advanced alarms, if redundant alarm servers are used, alarm timestamps can be different by milliseconds. If this happens and timestamps are different by less than 500 milliseconds, Power SCADA Operation suppresses and logs any secondary alarms on redundant servers.

After an alarm has gone active, it can be acknowledged in the alarm viewer. When you acknowledge an alarm, the date and time of the acknowledgment is recorded together with an optional note that you can enter in the Acknowledge window.

An alarm stays unacknowledged until you acknowledge it. After you have acknowledged an alarm, it stays acknowledged until the next time it goes active. At that point it is reset to unacknowledged and is waiting for you to acknowledge it again.

Power SCADA Operation counts the number of times an alarm goes through an inactive to active state transition. The number of these transitions is displayed as Occurrences in the alarm viewer in the alarm status view. There are two counters for each alarm. One counter for the total number of occurrences, and one for occurrences since the alarm was last acknowledged.

The time period during which an alarm is active, starting when it goes active, ending when it goes inactive, is called an alarm instance.

Alarm conditions are defined either as software alarms in the Software Alarms tool, or as device based alarms in the monitoring devices, using the appropriate device configuration tool.

To make it easier to analyze alarms, Power SCADA Operation categorizes them into types and combines alarms of similar types into incidents, based on the alarm start times.

The following table shows the different alarm categories and types in Power SCADA Operation:

Category	Type
Power Quality	Flicker Frequency Variation Harmonics Harmonics (Current) Harmonics (Power) Harmonics (Voltage) Interruption Over Voltage Sag (Voltage) Swell (Voltage) Transient Unbalance Unbalance (Current) Unbalance (Voltage) Unclassified Disturbance Under Voltage
Asset Monitoring	Arc Flash Backup Power Over Current Protection Sag (Current) Swell (Current) Thermal Monitor Under Current
Energy Management	Air Demand Electricity Gas Power Factor Steam Water
General	General Event General Setpoint Unassociated Dropout

Category	Type
Diagnostics	Clock / Time Communication Status Device Settings Device Status System Status

For information on how to configure Alarms , see Alarms configuration.

Related topics:

- Viewing incidents
- Viewing alarms
- Viewing events
- Viewing a timeline analysis
- Viewing waveforms
- Acknowledging alarms

Related references:

- Alarms
- Incidents
- Events
- Timeline analysis
- Waveforms
- Alarms terminology
- Alarms UI
- Timeline Analysis UI
- Waveforms UI

Events

An event is a record of an activity or a condition that is logged in Power SCADA Operation. Events are generated by users, the system software, or the connected devices. Examples of events include resetting a measurement, logging into Power SCADA Operation, making a configuration change in a device, or a setpoint going active on a device. Some of these events are logged automatically, for others logging must be setup manually. Each event record that is logged has a timestamp and a number of fields that describe the activity. Each event record describes one single activity or condition, for example, a particular setpoint going active in a particular monitoring device.

Events are logged and displayed as they happen in the system without any processing or aggregation. For example, an Over Voltage setpoint going active and then inactive in a device will cause 3 events to be logged, one for the pickup, one for the dropout, and one for the extreme voltage value measured during the time the setpoint was active.

Here is an example of the event records for an over voltage setpoint:

Source	Timestamp	Event	Condition	Measurement	Value	Type
My.Device	8/10/2017 1:44:53.000 PM	Over Voltage	ON	Voltage Phase A	145.740	Pick up
My.Device	8/10/2017 1:44:53.000 PM	Over Voltage	Extreme	Voltage Phase A	145.740	Instantaneous
My.Device	8/10/2017 1:45:39.000 PM	Over Voltage	OFF	Voltage Phase A	125.230	Drop out

Power SCADA Operation uses event records to determine alarm types and states.

For information on how to configure Alarms , see Alarms configuration.

Related topics:

- Viewing incidents
- Viewing alarms
- Viewing events
- Viewing a timeline analysis
- Viewing waveforms
- Acknowledging alarms

Related references:

- Alarms
- Incidents
- Alarms
- Events
- Timeline analysis
- Waveforms
- Alarms terminology
- Alarms UI
- Timeline Analysis UI
- Waveforms UI

Timeline analysis

Timeline analysis is a sequence of event analysis for items that are associated with one or more incidents or alarms. The items are shown on a timeline, in chronological order. Items include alarms, waveforms and burst data recordings. The tools available in timeline analysis allow you to add or remove items from the timeline, add notes, zoom in or out, and include alarms previously not associated with this incident. You can save a timeline analysis as new view in the view library for future reference.

Use timeline analysis to investigate the sequence of events during an alarm or incident. See Timeline Analysis UI for more information.

Prerequisites

None. Any incident can be displayed using timeline analysis.

NOTE: Alarms and data measurements during an incident occur in very short time intervals. To show the correct sequence of events in the timeline analysis, the timestamps must be accurate. Consider using monitoring devices with Precision Time Protocol (PTP) or GPS time synchronization for accurate time stamping.

For information on how to configure Alarms , see Alarms configuration.

Related topics:

- Viewing incidents
- Viewing alarms
- Viewing events
- Viewing a timeline analysis
- Viewing waveforms
- Acknowledging alarms

Related references:

- Alarms
- Incidents
- Alarms
- Events
- Timeline analysis
- Waveforms
- Alarms terminology
- Alarms UI
- Timeline Analysis UI
- Waveforms UI

Waveforms

Waveforms are graphical representations of voltage and current that show their variations over time. The waveform displays in Power SCADA Operation are based on logged, historical measurements that were recorded by a monitoring device. The measurements recorded by a device for a waveform capture are called samples and the speed with which these samples are taken is called sampling rate. The higher the sampling rate, the more accurately the waveform capture represents the actual voltage or current waveform. Captures taken by different device types can have different sampling rates, depending on the capabilities and settings of the device.

Use Waveforms to analyze power quality events by viewing the individual wave shapes, the magnitudes, the phase angles between voltage and current, and the timing of wave shape variations. Waveform data is also used to show voltage and current phasors and the individual harmonic components.

Prerequisites

The monitoring device data associated with the alarm or incident must include waveform captures.

For information on how to configure Alarms , see Alarms configuration.

Related topics:

- Viewing incidents
- Viewing alarms
- Viewing events
- Viewing a timeline analysis
- Viewing waveforms
- Acknowledging alarms

Related references:

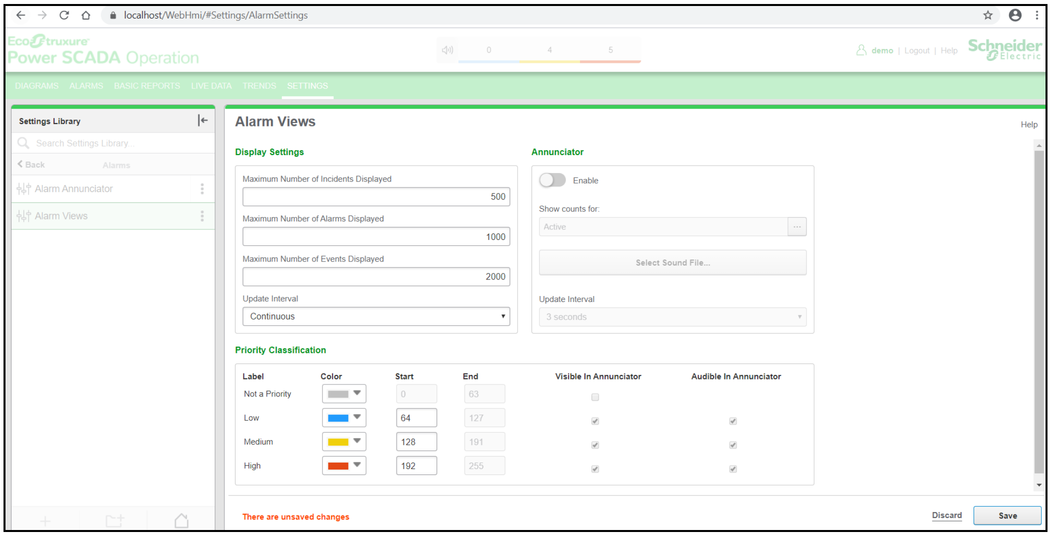
- Alarms
- Incidents
- Alarms
- Events
- Timeline analysis
- Waveforms
- Alarms terminology
- Alarms UI
- Timeline Analysis UI
- Waveforms UI

Displaying alarms in the runtime banner area

You can change how alarms are displayed in the Power SCADA Operation banner area. For example, instead of displaying the default Alarm Annunciator, you can display the three most recent alarms. This topic explains how to change how alarms display in the banner area.

To display the three most recent alarms in the banner:

1. Log in to the PSO Web Applications.
2. Click **SETTINGS**
3. In the **Setting Library** pane, click **Alarms**, and then click **Alarm Views**.
4. In the **Alarm Views** pane, under **Annunciator**, turn off the **Enable** toggle button.


localhost/WebHmi/#Settings/AlarmSettings
EcoStruxure Power SCADA Operation
0 4 5
demo | Logout | Help
Schneider Electric
DIAGRAMS ALARMS BASIC REPORTS LIVE DATA TRENDS SETTINGS
Settings Library
Search Settings Library...
Back Alarms
Alarm Annunciator
Alarm Views
Help
Discard
Save

Alarm Views

Display Settings

Maximum Number of Incidents Displayed: 500
Maximum Number of Alarms Displayed: 1000
Maximum Number of Events Displayed: 2000
Update Interval: Continuous

Annunciator

Enable
Show counts for: Active
Select Sound File...
Update Interval: 3 seconds

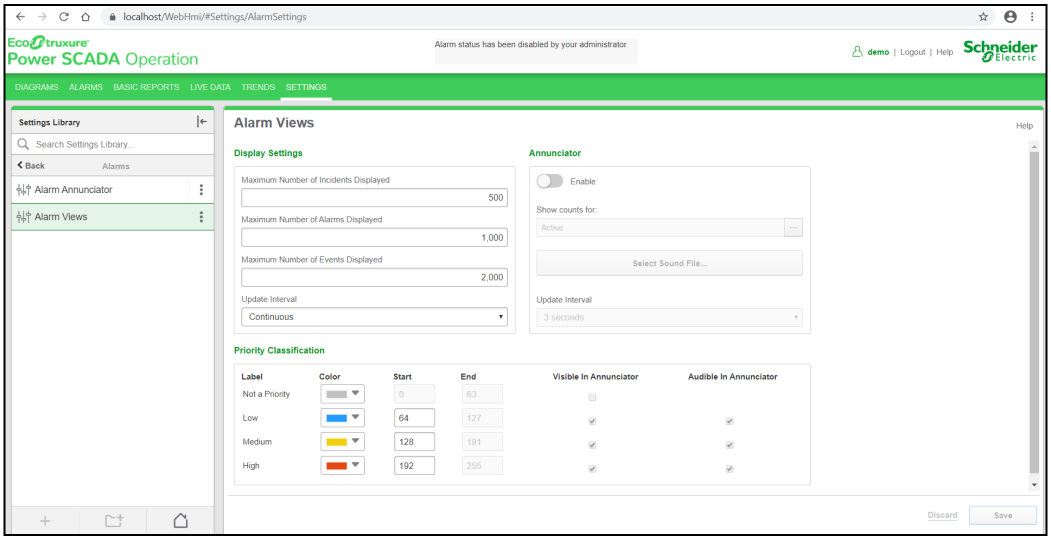
Priority Classification

Label	Color	Start	End	Visible In Annunciator	Audible In Annunciator
Not a Priority	▨	0	63	□	
Low	🟦	64	127	☑	☑
Medium	🟨	128	191	☑	☑
High	🟥	192	255	☑	☑

There are unsaved changes

5. Click **Save**, and then refresh the web page.

   Notice that the Alarm Annunciator is no longer displayed:


localhost/WebHmi/#Settings/AlarmSettings
EcoStruxure Power SCADA Operation
demo | Logout | Help
Schneider Electric
DIAGRAMS ALARMS BASIC REPORTS LIVE DATA TRENDS SETTINGS
Settings Library
Search Settings Library...
Back Alarms
Alarm Annunciator
Alarm Views
Help
Discard
Save

Alarm status has been disabled by your administrator.

Alarm Views

Display Settings

Maximum Number of Incidents Displayed: 500
Maximum Number of Alarms Displayed: 1,000
Maximum Number of Events Displayed: 2,000
Update Interval: Continuous

Annunciator

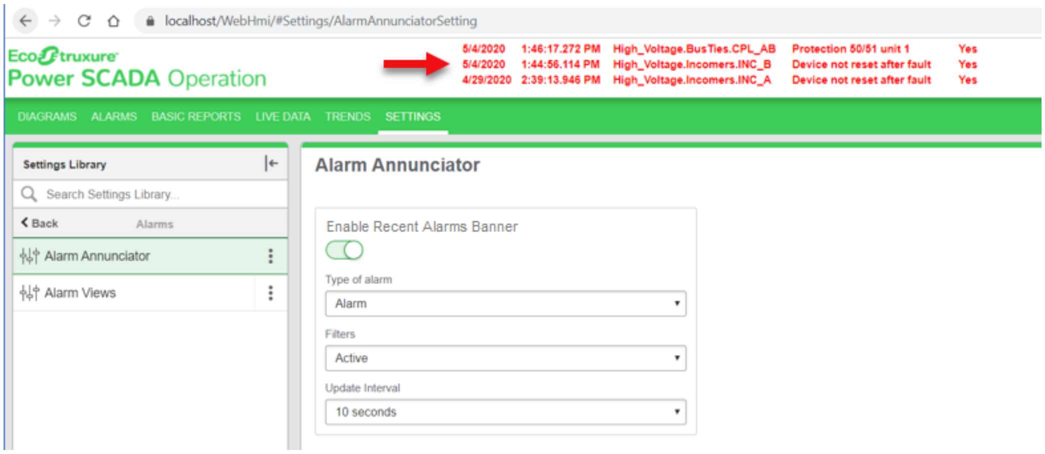
Enable
Show counts for: Active
Select Sound File...
Update Interval: 3 seconds

Priority Classification

Label	Color	Start	End	Visible In Annunciator	Audible In Annunciator
Not a Priority	▨	0	63	□	
Low	🟦	64	127	☑	☑
Medium	🟨	128	191	☑	☑
High	🟥	192	255	☑	☑

6. In the **Setting Library** pane, click **Alarm Annunciator**, and then turn on the **Enable Recent Alarms Banner** toggle button to display the three most recent alarms.
7. Click **Save**, and then refresh the web page.

The 3 most recent alarms are displayed. For example:



5/4/2020	1:46:17.272 PM	High_Voltage.BusTies.CPL_AB	Protection 50/51 unit 1	Yes
5/4/2020	1:44:56.114 PM	High_Voltage.Incomers.INC_B	Device not reset after fault	Yes
4/29/2020	2:39:13.946 PM	High_Voltage.Incomers.INC_A	Device not reset after fault	Yes

Alarm Annunciator

Enable Recent Alarms Banner

Type of alarm: Alarm

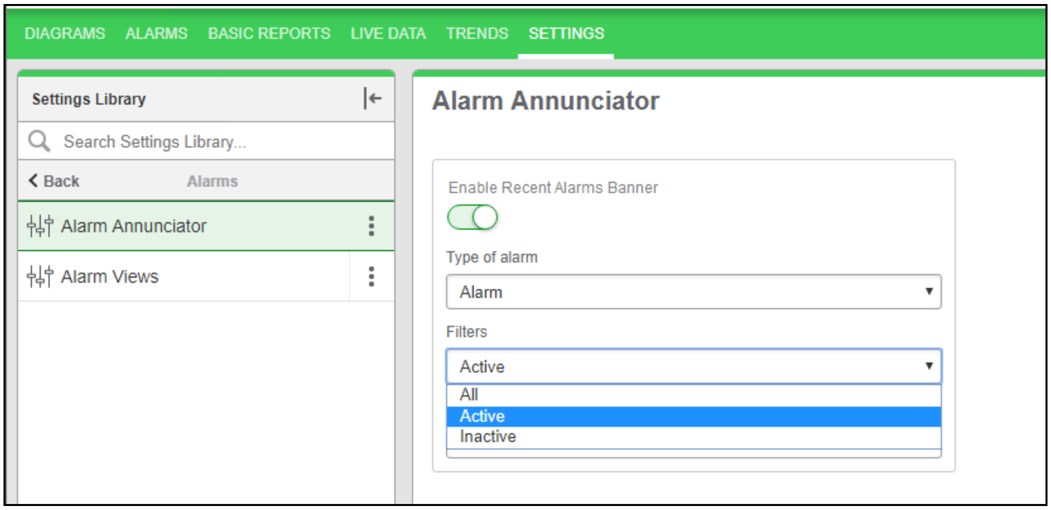
Filters: Active

Update Interval: 10 seconds

8. On the **Alarm Annunciator** page, select the relevant options as per your requirement to display the three most recent alarms in the banner.

   a. **Type of Alarm** – Select Alarms, Events, or Incidents from the drop-down.

   b. **Filters** – Select All, Active, or Inactive from the drop-down.



Alarm Annunciator

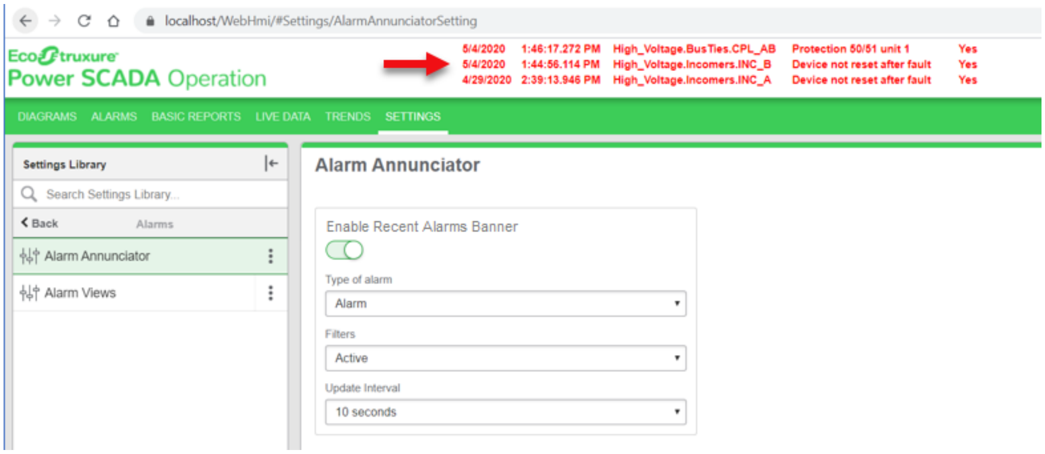
Enable Recent Alarms Banner

Type of alarm: Alarm

Filters: Active

All
Active
Inactive

   c. **Update Interval** – You can select the required refresh rate by clicking the **Update Interval** drop-down.



```
5/4/2020   1:46:17.272 PM  High_Voltage.BusTies.CPL_AB  Protection 50/51 unit 1      Yes
5/4/2020   1:44:56.114 PM  High_Voltage.Incomers.INC_B  Device not reset after fault Yes
4/29/2020  2:39:13.946 PM  High_Voltage.Incomers.INC_A  Device not reset after fault Yes
```

Alarm Annunciator

Enable Recent Alarms Banner

Type of alarm: Alarm

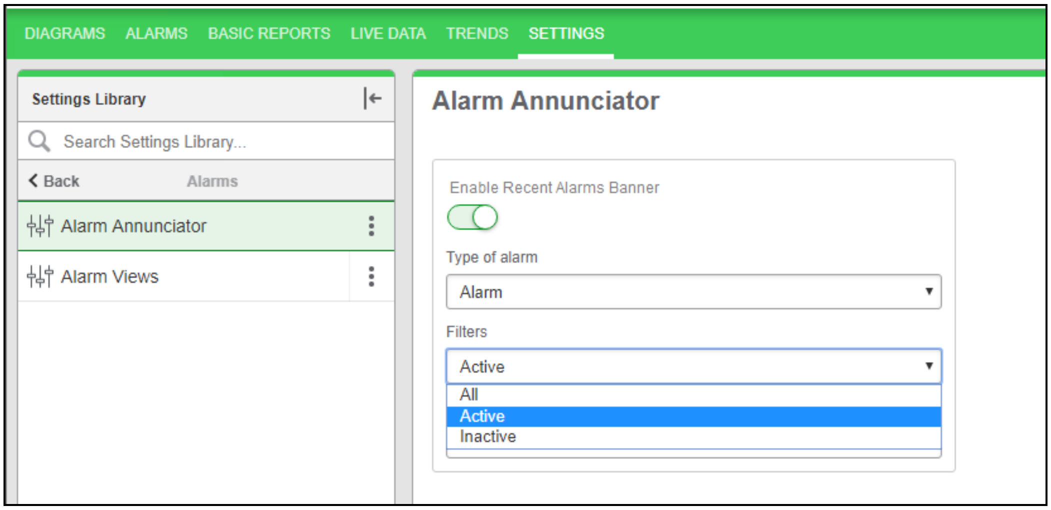
Filters: Active

Update Interval: 10 seconds

Filter selections in the banner

You can filter the display in the banner using the following alarm filters:

- All
- Active
- Inactive



Alarm Annunciator

Enable Recent Alarms Banner

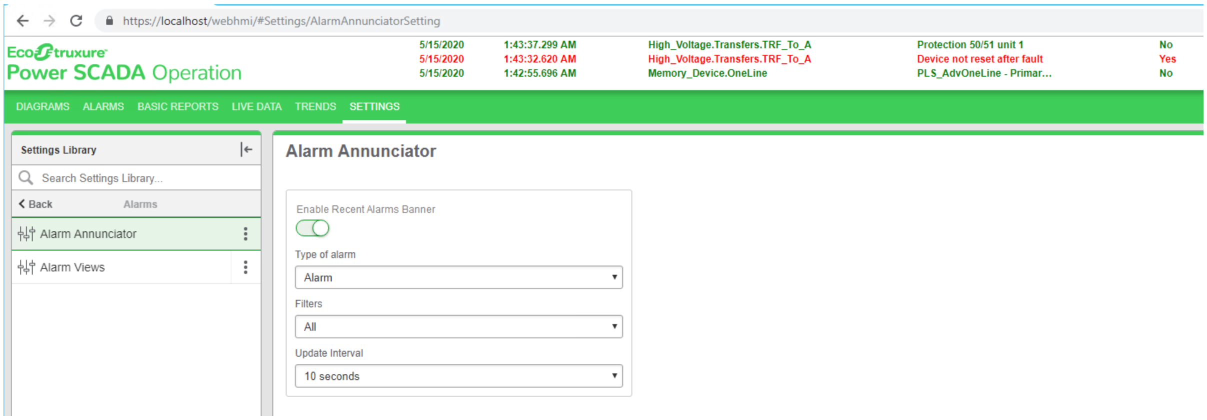
Type of alarm: Alarm

Filters: Active

NOTE: The alarms are displayed based on the selection of the filter.

Type	Active Records	Inactive Records	Acknowledged	Unacknowledged
Alarm	Color: red	Color: green	Font: normal	Font: bold

All Alarms – Displays all active unacknowledged, acknowledged and inactive unacknowledged, acknowledged with status **Yes** or **No**. For example:



5/15/2020 1:43:37.299 AM High_Voltage.Transfers.TRF_To_A Protection 50/51 unit 1 No
5/15/2020 1:43:32.620 AM High_Voltage.Transfers.TRF_To_A Device not reset after fault Yes
5/15/2020 1:42:55.696 AM Memory_Device.OneLine PLS_AdvOneLine - Primar... No

Alarm Annunciator

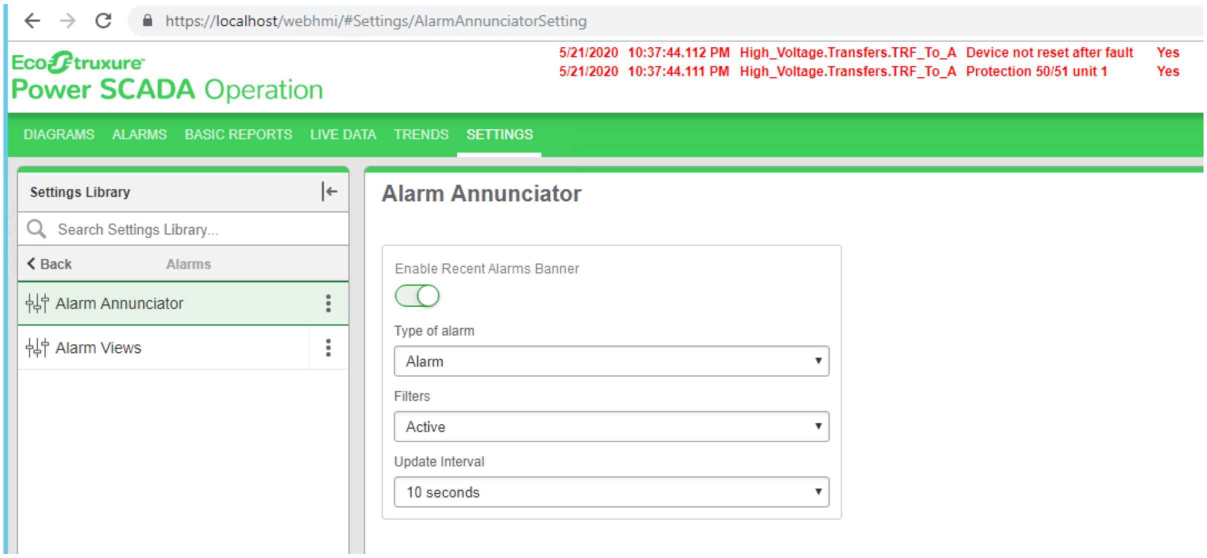
Enable Recent Alarms Banner

Type of alarm: Alarm

Filters: All

Update Interval: 10 seconds

Active Alarms – Displays all active unacknowledged, acknowledged alarms with status **Yes**. For example:



5/21/2020 10:37:44.112 PM High_Voltage.Transfers.TRF_To_A Device not reset after fault Yes
5/21/2020 10:37:44.111 PM High_Voltage.Transfers.TRF_To_A Protection 50/51 unit 1 Yes

Alarm Annunciator

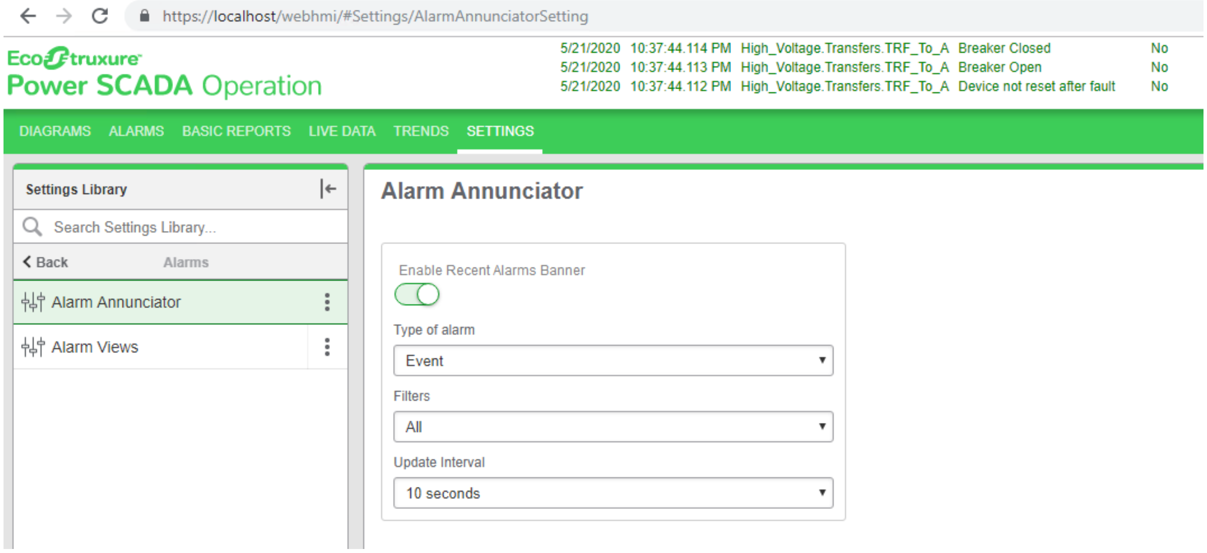
Enable Recent Alarms Banner

Type of alarm: Alarm

Filters: Active

Update Interval: 10 seconds

Inactive Alarms – Displays inactive unacknowledged, acknowledged alarms which are not active with status **No**. For example:



5/21/2020 10:37:44.114 PM High_Voltage.Transfers.TRF_To_A Breaker Closed No
5/21/2020 10:37:44.113 PM High_Voltage.Transfers.TRF_To_A Breaker Open No
5/21/2020 10:37:44.112 PM High_Voltage.Transfers.TRF_To_A Device not reset after fault No

Alarm Annunciator

Enable Recent Alarms Banner

Type of alarm: Event

Filters: All

Update Interval: 10 seconds

Related topics:

- Viewing incidents
- Viewing alarms
- Viewing events
- Viewing a timeline analysis
- Viewing waveforms
- Acknowledging alarms
- Displaying alarms in the runtime banner area

For reference information, see:

- Alarms
- Incidents
- Alarms
- Events
- Timeline analysis
- Waveforms
- Alarms terminology
- Alarms UI
- Timeline Analysis UI
- Waveforms UI

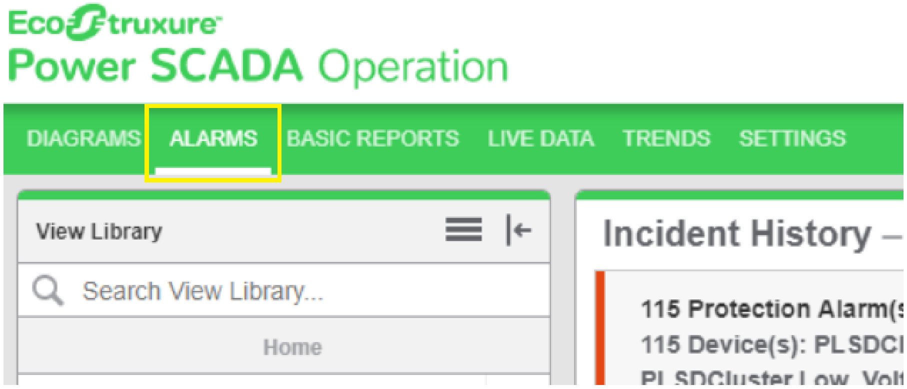
For information on how to configure Alarms , see Alarms configuration.

Creating alarm menus

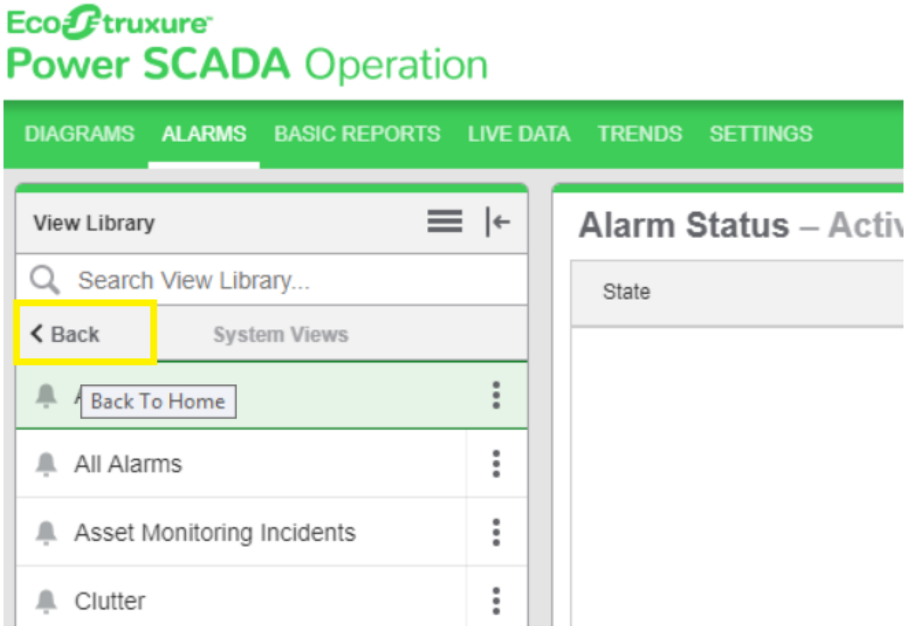
To create alarm menus:

1. Log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)

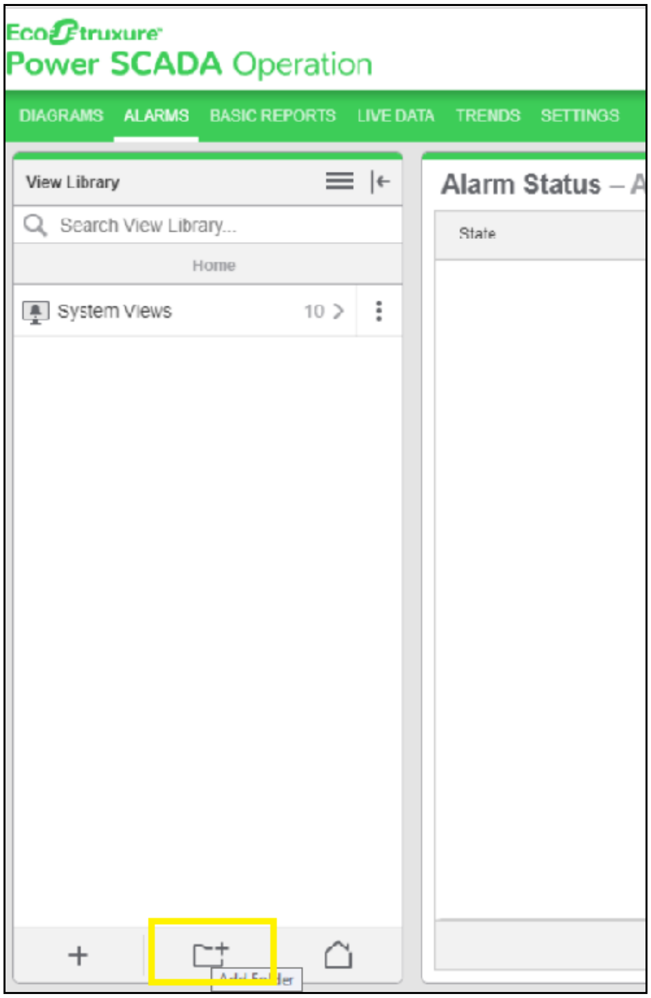
2. Click **ALARMS**.



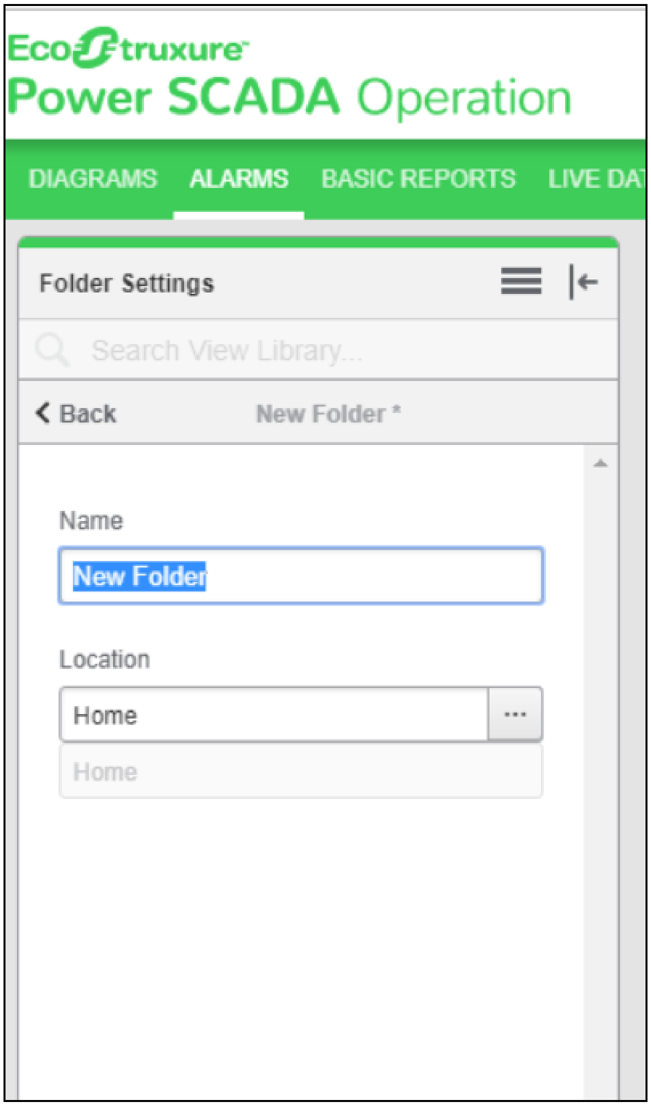
3. Click on **Back** to go back to the Home page.



4. At the bottom of the **View Library**, click **Add Folder**:


EcoStruxure
Power SCADA Operation
DIAGRAMS
ALARMS
BASIC REPORTS
LIVE DATA
TRENDS
SETTINGS
View Library
Search View Library...
Home
System Views
10
Alarm Status – A
State
Add Folder

5. Enter the folder **Name**:



Name

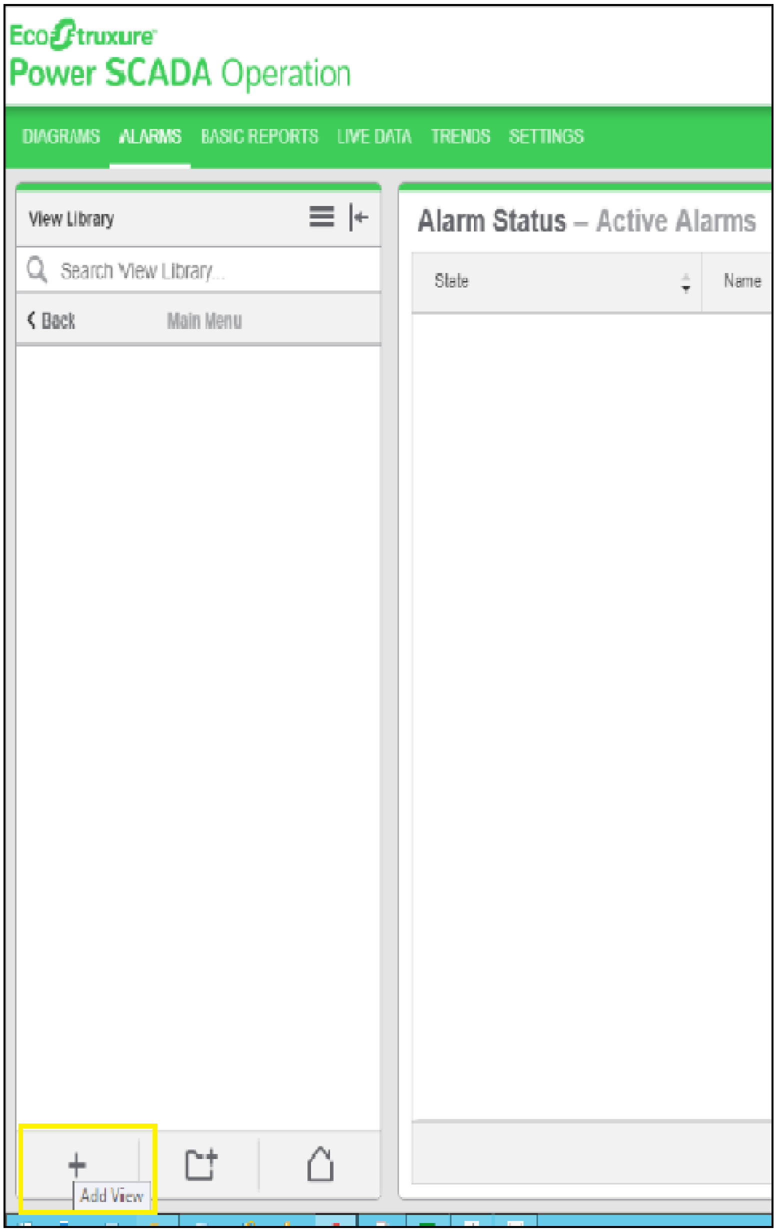
New Folder

Location

Home

Home

6. At the bottom of the **View Library**, click **Add View**:



7. Configure the **View** by setting the following values based on your requirements:

   a. **View Name**: Type the view name.

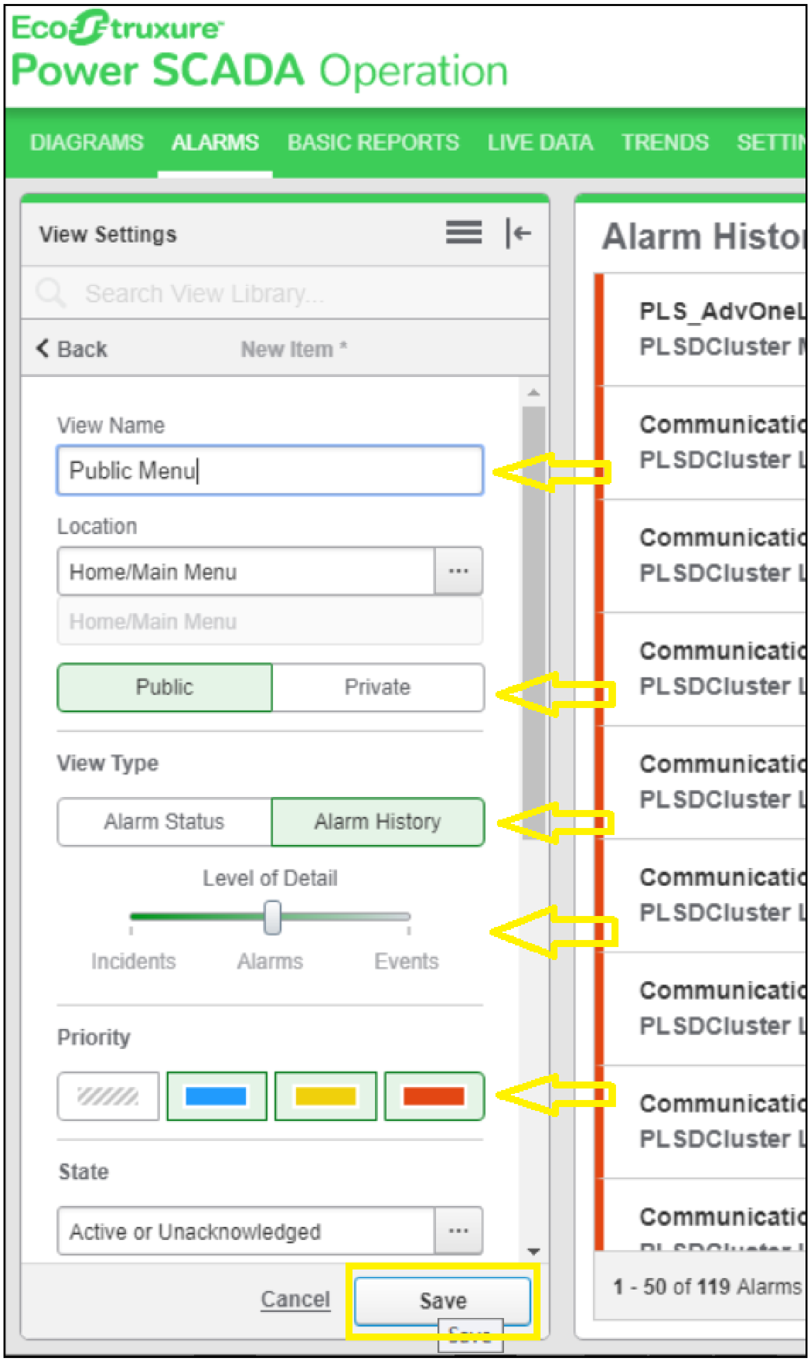
   b. **Location**: Select the location to display.

   c. Select **Public** or **Private**.

   d. **View Type**

   e. **Priority**

f. **State**



View Name: Public Menu

Location: Home/Main Menu

Home/Main Menu

[✓ Public] Private

View Type: Alarm Status [✓ Alarm History]

Level of Detail: Incidents [✓ Alarms] Events

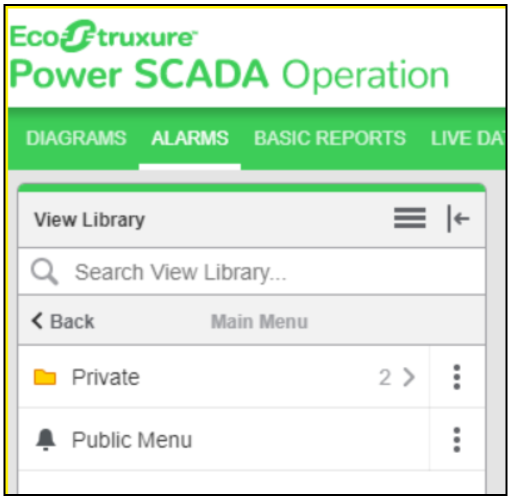
Priority: ▨ 🟦 🟨 🟥

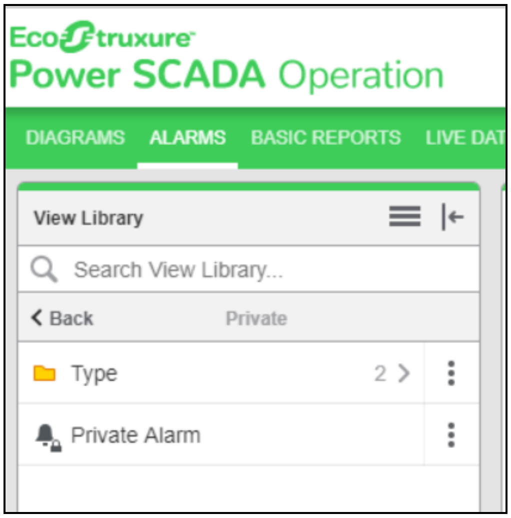
State: Active or Unacknowledged

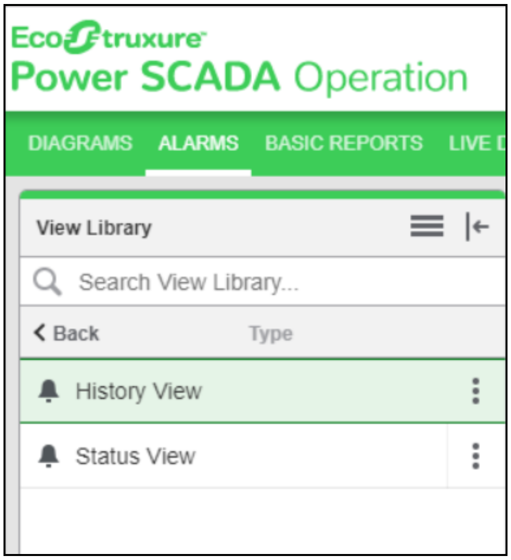
g. Click **Save**.

8. (Optional) Repeat steps from 4 through 6 to add more sub-folders or views inside the folder.

For reference, see the following images to add sub-folders or views:







Related topics:

- "Adding a new Alarms view" on page 370
- "Copying an Alarms view" on page 372
- "Editing an Alarms view" on page 372
- "Moving an Alarms view" on page 373
- "Deleting an Alarms view" on page 374
- "Setting a default Alarms view" on page 375
- Creating alarm menus
- "Exporting alarm menus" on page 382
- "Importing alarm menus" on page 384
- "Alarm Views" on page 575

For reference information see:

- "Alarms UI" on page 1263

For information on how to use Alarms, see "Alarms" on page 730.

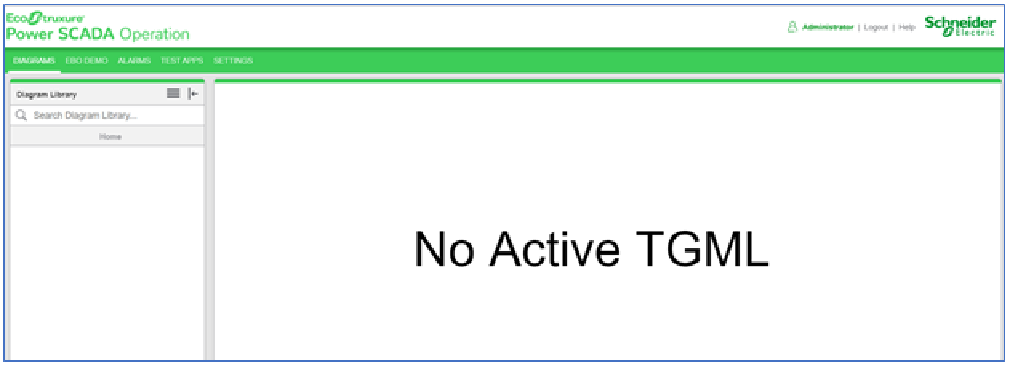
Diagrams

Diagrams is the web application that display TGML graphic pages.

TGML graphic pages are TGML files available in the Diagrams Library. When you select a TGML graphic in the Diagrams Library, it displays on the screen.

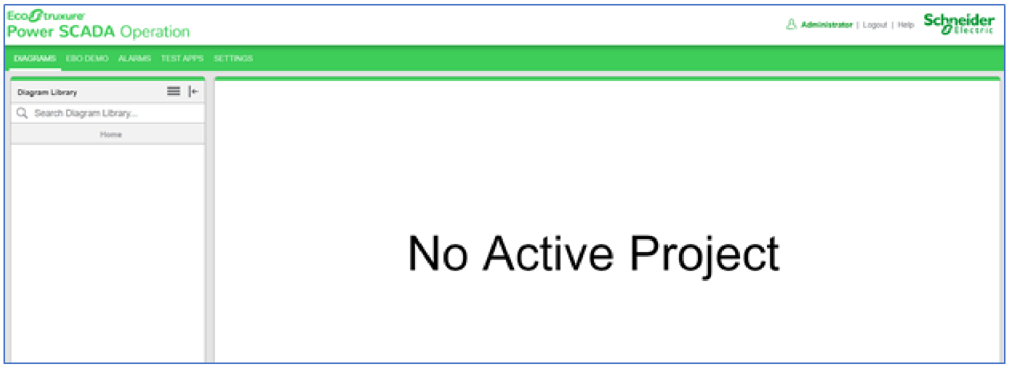
To set a specific TGML graphic page as the default diagram, in the Diagrams Library right-click the TGML graphic page, and then click **Set as default**.

If there are no TGML graphic pages available in the running project, or TGML graphic pages were not created for the running project, the following screen is displayed:



No Active TGML

If there are no projects running in Power SCADA Operation, then the following screen is displayed:



No Active Project

You can add or modify TGML graphic pages and save those files in the TGML folder. For more information see " TGML graphics and TGML graphic templates" on page 768.

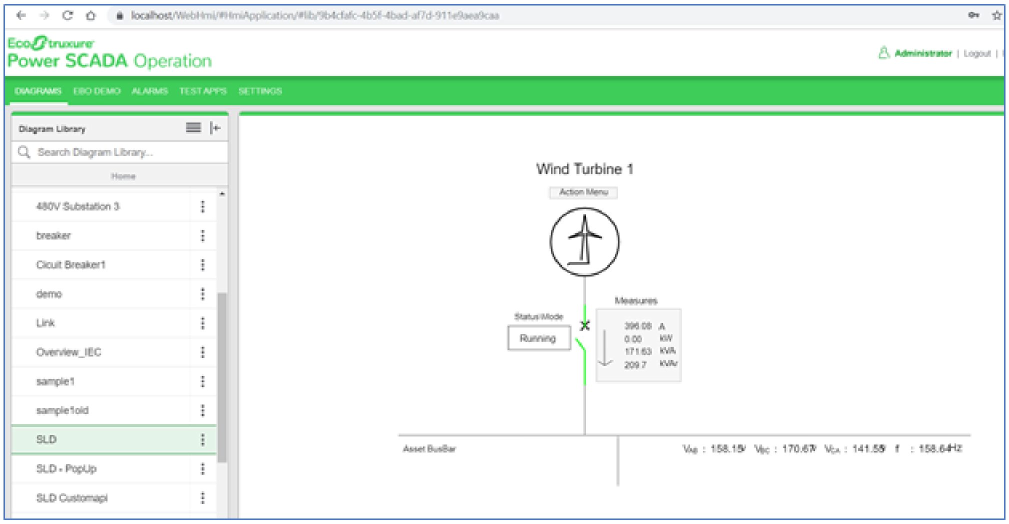
NOTE: All the TGML files must be saved in the TGML folder of the running project so that the files are displayed in Diagrams. For more information, see Defining the Diagrams Menu Structure.

Display View

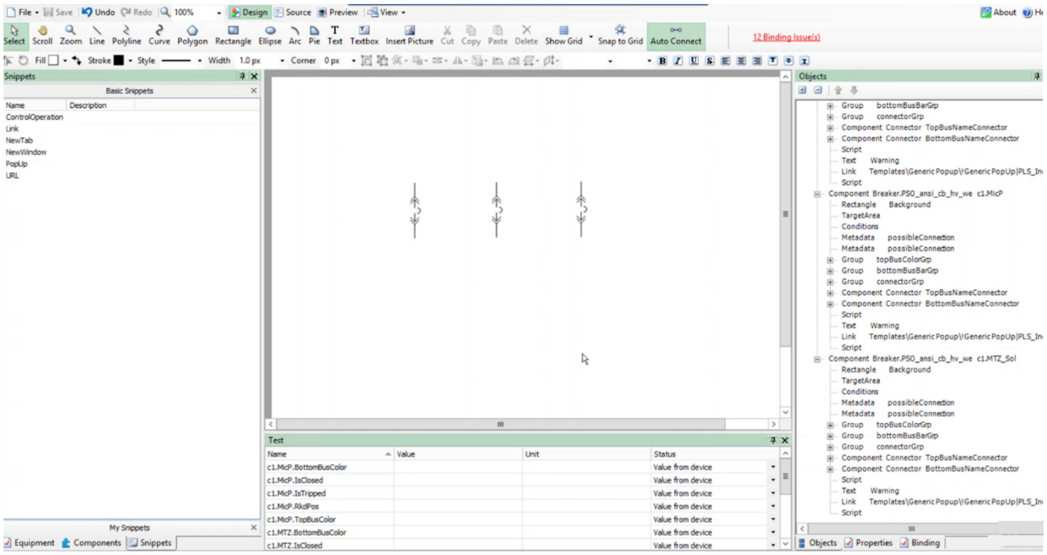
To view a one-line diagram:

In the **Diagram Library** select the TGML graphics page. Click a component to see detailed information displayed in one of the following formats:

- **Link:** Navigate from one TGML graphics page to another.



- **Pop-Up:** A pop-up window appears displaying relevant information about the equipment.

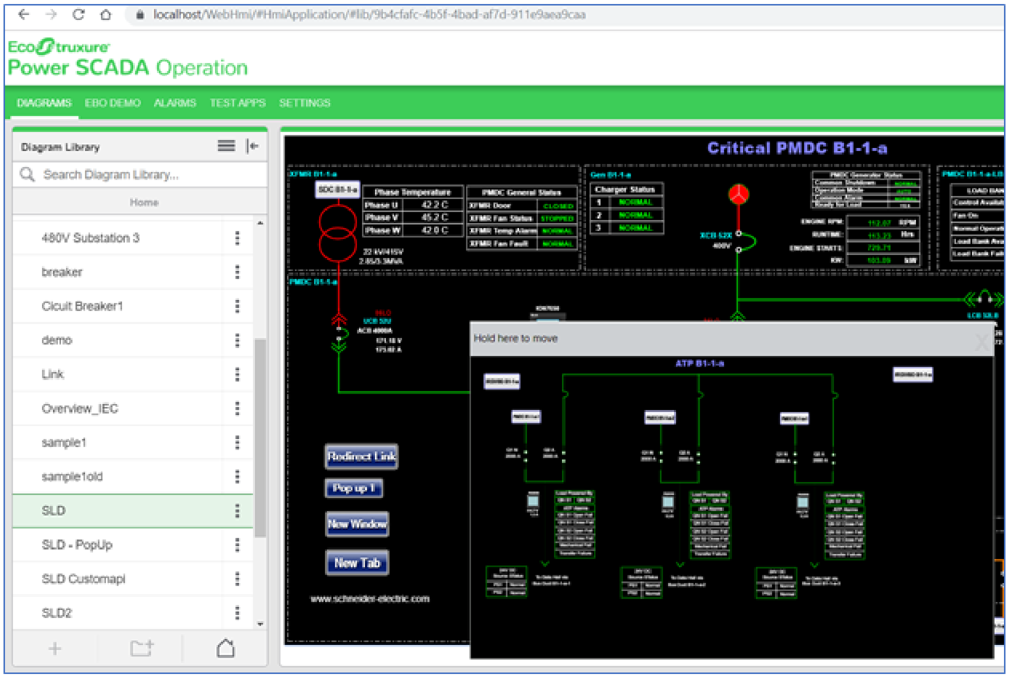


Test

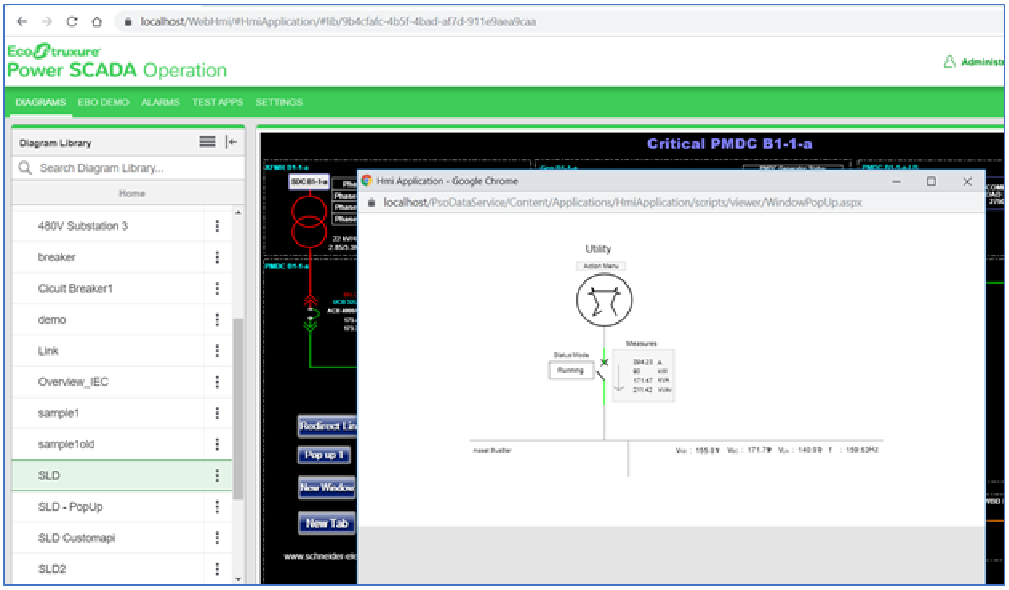
Name	Value	Unit	Status
c1.MicP.BottomBusColor			Value from device
c1.MicP.IsClosed			Value from device
c1.MicP.IsTripped			Value from device
c1.MicP.RkdPos			Value from device
c1.MicP.TopBusColor			Value from device
c1.MTZ.BottomBusColor			Value from device
c1.MTZ.IsClosed			Value from device

Objects

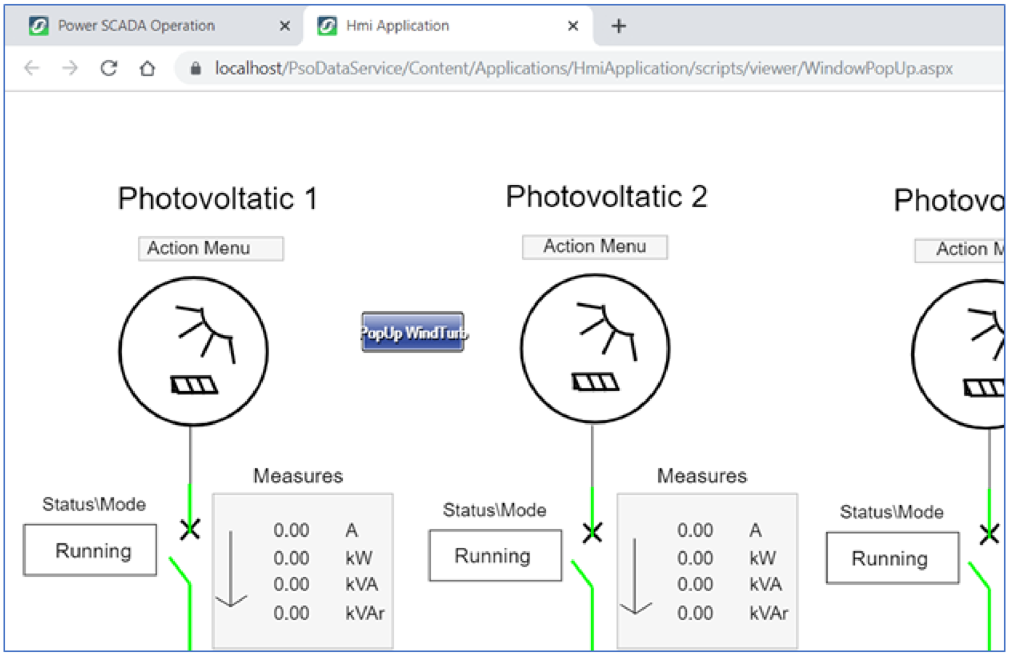
```
    Group      bottomBusBarGrp
    Group      connectorGrp
    Component  Connector  TopBusNameConnector
    Component  Connector  BottomBusNameConnector
    Script
    Text     Warning
    Link     Templates\GenericPopup\!GenericPopUp|PLS_In
    Script
Component Breaker.PSO_ansi_cb_hv_we  c1.MicP
    Rectangle    Background
    TargetArea
    Conditions
    Metadata    possibleConnection
    Metadata    possibleConnection
    Group      topBusColorGrp
    Group      bottomBusBarGrp
    Group      connectorGrp
    Component  Connector  TopBusNameConnector
    Component  Connector  BottomBusNameConnector
    Script
    Text     Warning
    Link     Templates\GenericPopup\!GenericPopUp|PLS_In
    Script
Component Breaker.PSO_ansi_cb_hv_we  c1.MTZ_Sol
    Rectangle    Background
    TargetArea
    Conditions
    Metadata    possibleConnection
    Metadata    possibleConnection
    Group      topBusColorGrp
    Group      bottomBusBarGrp
    Group      connectorGrp
    Component  Connector  TopBusNameConnector
    Component  Connector  BottomBusNameConnector
    Script
    Text     Warning
    Link     Templates\GenericPopup\!GenericPopUp|PLS_In
    Script
```



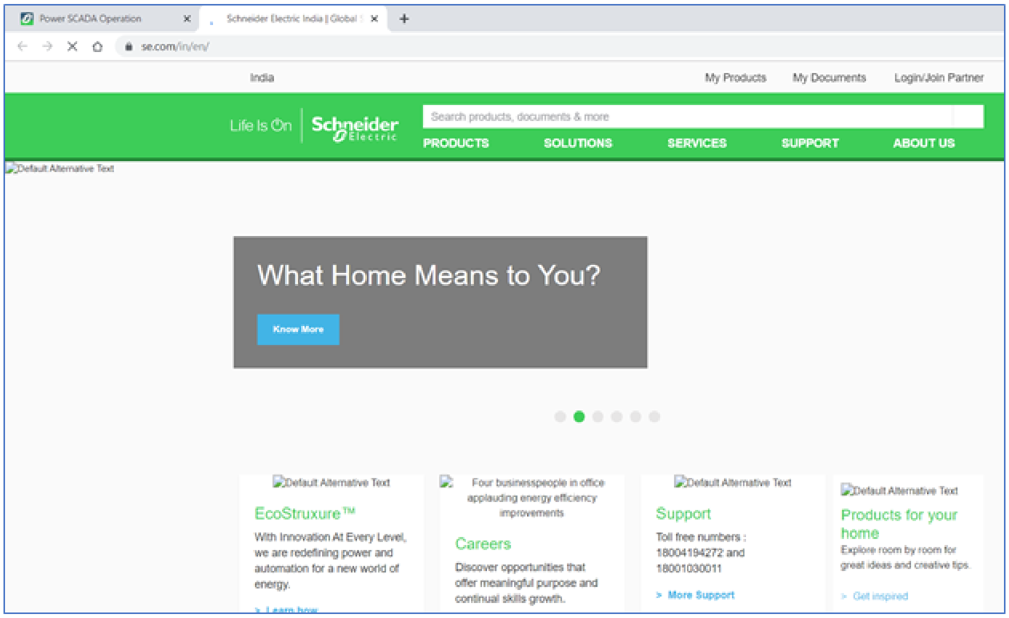
- **New Window:** Opens the TGML page in a new window.



- **New Tab:** Opens the TGML page in a new tab.



- **Url:** Navigates to a different site or another Web Applications page.



What Home Means to You?

With Innovation At Every Level, we are redefining power and automation for a new world of energy.

Discover opportunities that offer meaningful purpose and continual skills growth.

Toll free numbers : 18004194272 and 18001030011

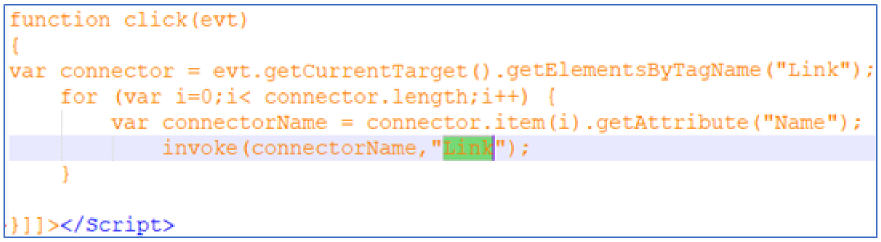
Explore room by room for great ideas and creative tips.

NOTE: You can open multiple Web Applications pop-up windows at the same time. However, it is recommended you only open one pop-up window at a time, as it may slow the performance of the web page.

Display configuration

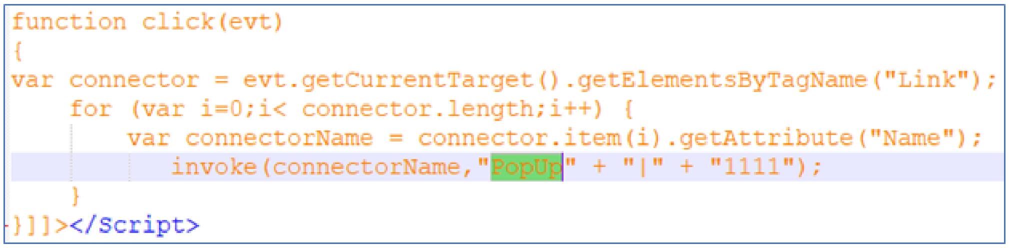
You can define the type of information to display for each component when creating the TGML graphic pages. The following code snippets help to add the different navigation types described above.

- Link



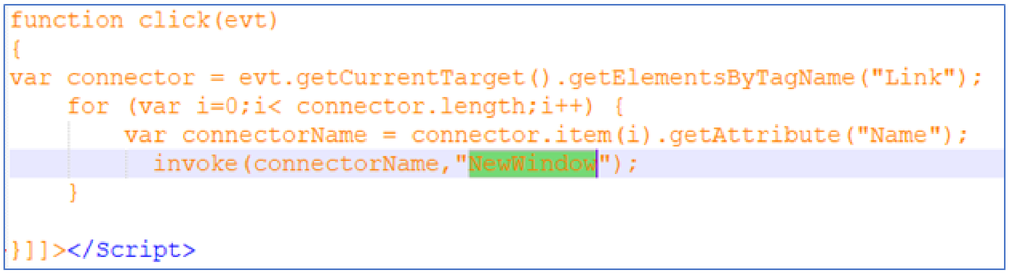
```
function click(evt)
{
var connector = evt.getCurrentTarget().getElementsByTagName("Link");
    for (var i=0;i< connector.length;i++) {
        var connectorName = connector.item(i).getAttribute("Name");
            invoke(connectorName,"Link");
    }

}]]></Script>
```

- Pop-Up



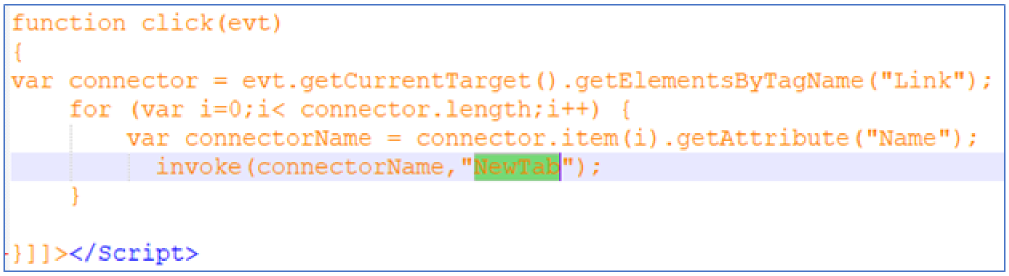
```
function click(evt)
{
var connector = evt.getCurrentTarget().getElementsByTagName("Link");
    for (var i=0;i< connector.length;i++) {
        var connectorName = connector.item(i).getAttribute("Name");
            invoke(connectorName,"PopUp" + "|" + "1111");
    }
}]]></Script>
```

- New Window



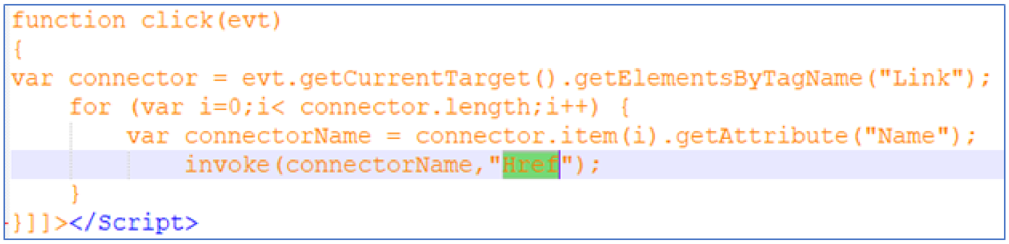
```
function click(evt)
{
var connector = evt.getCurrentTarget().getElementsByTagName("Link");
    for (var i=0;i< connector.length;i++) {
        var connectorName = connector.item(i).getAttribute("Name");
          invoke(connectorName,"NewWindow");
    }

}]]></Script>
```

- New Tab



```
function click(evt)
{
var connector = evt.getCurrentTarget().getElementsByTagName("Link");
    for (var i=0;i< connector.length;i++) {
        var connectorName = connector.item(i).getAttribute("Name");
          invoke(connectorName,"NewTab");
    }

}]]></Script>
```

- URL



```
function click(evt)
{
var connector = evt.getCurrentTarget().getElementsByTagName("Link");
    for (var i=0;i< connector.length;i++) {
        var connectorName = connector.item(i).getAttribute("Name");
            invoke(connectorName,"Href");
    }
}]]></Script>
```

TGML graphics and TGML graphic templates

TGML graphics

A TGML graphic is a graphic component that is configured to provide detailed information about it. For information on how to create a TGML graphic, see "Designing TGML graphics" on page 768.

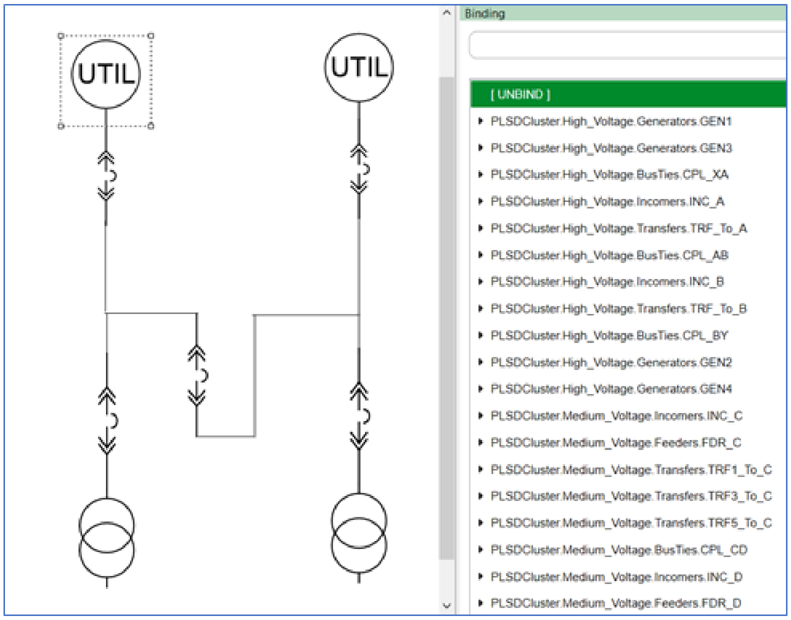
TGML graphic templates

TGML graphics templates are non-instantiated pages that are linked to a TGML graphic. When a TGML graphic file name is saved starting with ! (exclamation mark), it acts as a template rather than a regular TGML graphic. For information on how to create a TGML graphic template, see "Designing TGML graphic templates" on page 769.

Designing TGML graphics

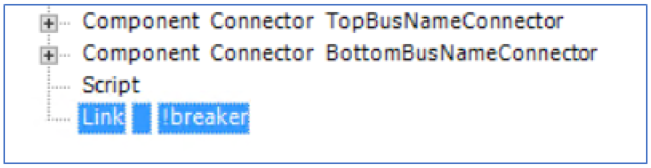
To design a TGML graphic:

1. In the Graphics Editor, drag a component to the workspace, and then assign the bind.


UTIL
UTIL
Binding
[UNBIND]
PLSDCluster.High_Voltage.Generators.GEN1
PLSDCluster.High_Voltage.Generators.GEN3
PLSDCluster.High_Voltage.BusTies.CPL_XA
PLSDCluster.High_Voltage.Incomers.INC_A
PLSDCluster.High_Voltage.Transfers.TRF_To_A
PLSDCluster.High_Voltage.BusTies.CPL_AB
PLSDCluster.High_Voltage.Incomers.INC_B
PLSDCluster.High_Voltage.Transfers.TRF_To_B
PLSDCluster.High_Voltage.BusTies.CPL_BY
PLSDCluster.High_Voltage.Generators.GEN2
PLSDCluster.High_Voltage.Generators.GEN4
PLSDCluster.Medium_Voltage.Incomers.INC_C
PLSDCluster.Medium_Voltage.Feeders.FDR_C
PLSDCluster.Medium_Voltage.Transfers.TRF1_To_C
PLSDCluster.Medium_Voltage.Transfers.TRF3_To_C
PLSDCluster.Medium_Voltage.Transfers.TRF5_To_C
PLSDCluster.Medium_Voltage.BusTies.CPL_CD
PLSDCluster.Medium_Voltage.Incomers.INC_D
PLSDCluster.Medium_Voltage.Feeders.FDR_D

2. Create a Link property inside the component. It helps in navigating to the details page in pop-up window. Use a common template name (TGML graphic template) as Link value in component.

   In the following image, `!breaker.tgml` is the TGML graphic template that shows detailed information:



```
⊞ Component Connector TopBusNameConnector
⊞ Component Connector BottomBusNameConnector
  Script
  Link  !breaker
```

3. Add the following script on the click event inside the component:

```
function click(evt)
{
    var connector = evt.getCurrentTarget().getParentNode
().getElementsByTagName("Link");
    var instanceId = evt.getCurrentTarget().getParentNode().getAttribute
("InstanceId");
    var componentName = "PLSDCluster.High_Voltage.BusTies.CPL_AB";
    var title = "PLSDCluster.High_Voltage.BusTies.CPL_AB";
    var width = "60%";
    var Height = "40%";
    var showTitleBar = "True";

    for (var i=0;1< connector.length;i++) {
        var connectorName = connector.item(i).getAttribute("Name");
        invoke("tgmlPath="+ connector + "Type = PopUp | ComponentName=" +
componentName + " | InstanceId=" + instanceId + " | Title=" + title + " |
Width=" + width + " | Height=" + height + " | ShowTitleBar =" +
ShowTitleBar);
    }
}
```

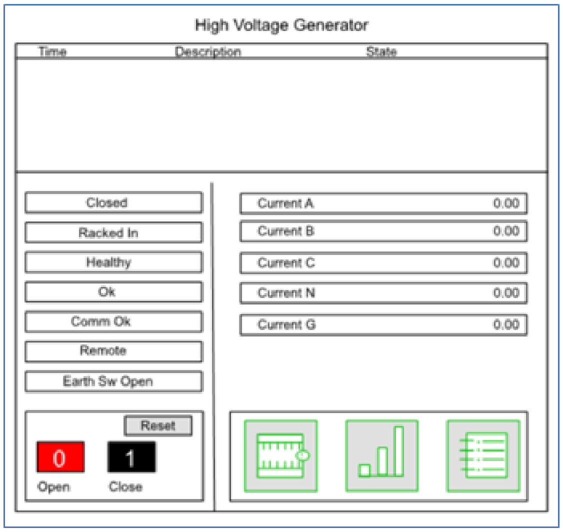
NOTE: This can be also done using snippets.

4. Save the TGML graphic.

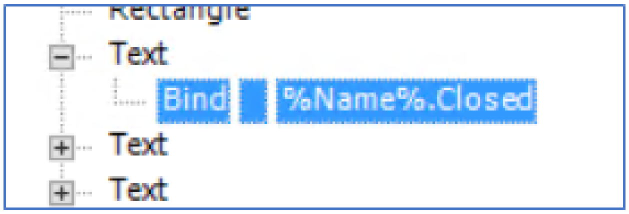
Designing TGML graphic templates

To design a TGML graphic template:

1. In the Graphics Editor, create a TGML graphic with the detailed information that should display about the device or equipment. For example:



2. Create the binds based on the following naming convention: `%Name%.bindName`. For example:

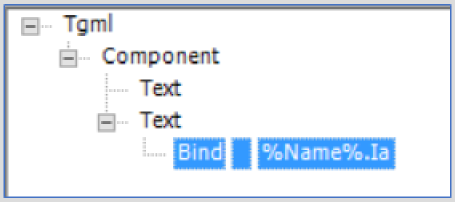


Rectangle
Text
Bind %Name%.Closed
Text
Text

NOTE: `%Name%` replacement is supported only for bind or component names. Therefore, we can have binds that are not part of any component with `%Name%`. For example, if `%name%.Ia` is part of a component and the component name uses `%name%`, then any other replacement or combination of both are not supported.

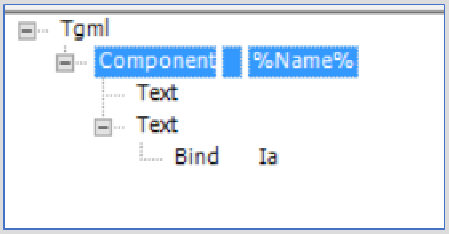
The following images show the different valid scenarios:

At Bind name:



Tgml
Component
Text
Text
Bind %Name%.Ia

At Component name:

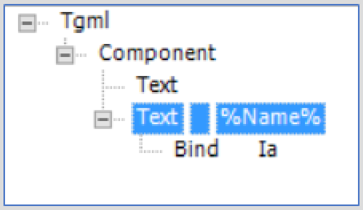


Tgml
Component %Name%
Text
Text
Bind Ia

Except for the above two cases, the use of %Name% (variables) at other places is not valid.

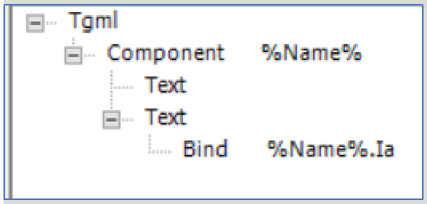
The following images show invalid scenarios:

At Text name:



Tgml
Component
Text
Text %Name%
Bind Ia

At Component name and inherited bind name:



```
Tgml
  Component   %Name%
    Text
    Text
      Bind   %Name%.Ia
```

This combination is also invalid in component name and within the bind.

3. Save the TGML graphic template with ! (exclamation mark). For example: `!breaker.tgml`.

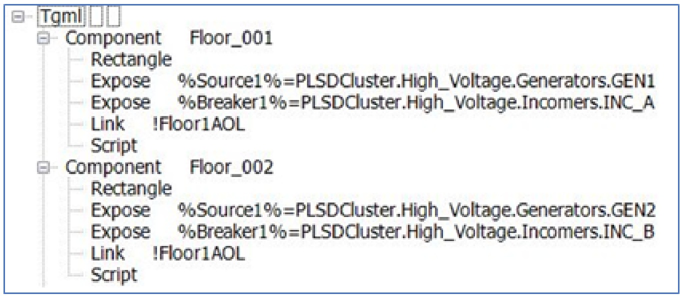
NOTE: The Web Applications consider `%Name%` to be a variable and will replace it with the respective equipment or TGML graphic at runtime.

TGML graphics templates for multiple equipment

You can extend TGML graphics templates to create TGML graphics that render data from multiple equipment, providing system or area-wide statistics.

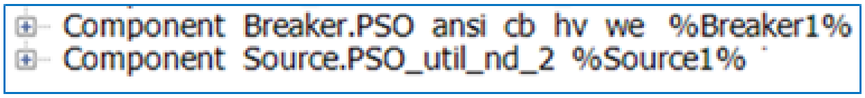
You can also create TGML graphics templates based on a common design and then apply it to minimize rework. For example: If the graphics for all the floors are the same, but only the equipment differs, you can use a single TGML graphics template for multiple floors to create a map that could provide information about the respective equipment.

Refer the following parent TGML structure:



```
Tgml
  Component   Floor_001
    Rectangle
    Expose   %Source1%=PLSDCluster.High_Voltage.Generators.GEN1
    Expose   %Breaker1%=PLSDCluster.High_Voltage.Incomers.INC_A
    Link   !Floor1AOL
    Script
  Component   Floor_002
    Rectangle
    Expose   %Source1%=PLSDCluster.High_Voltage.Generators.GEN2
    Expose   %Breaker1%=PLSDCluster.High_Voltage.Incomers.INC_B
    Link   !Floor1AOL
    Script
```

Refer the following child TGML structure:



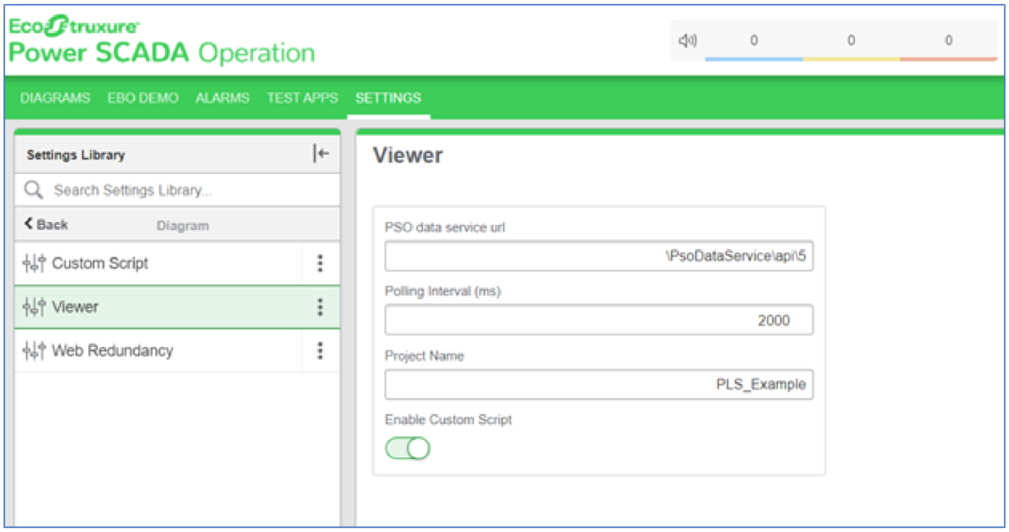
```
Component  Breaker.PSO  ansi  cb  hv  we  %Breaker1%
Component  Source.PSO_util_nd_2  %Source1%
```

Creating custom scripts for TGML graphics

In PSO Web Applications, you can use custom scripts to add new functions for use in TGML graphics, including data values that can be represented or animated. You can write custom JavaScript and deploy it to the server for use in your TGML graphics. This topic includes information on how to create, modify, and delete custom scripts.

Prerequisites

Enable Custom Script must be on. You can enable or disable the custom script menu in SETTINGS:



Viewer

PSO data service url: \PsoDataService\api\5

Polling Interval (ms): 2000

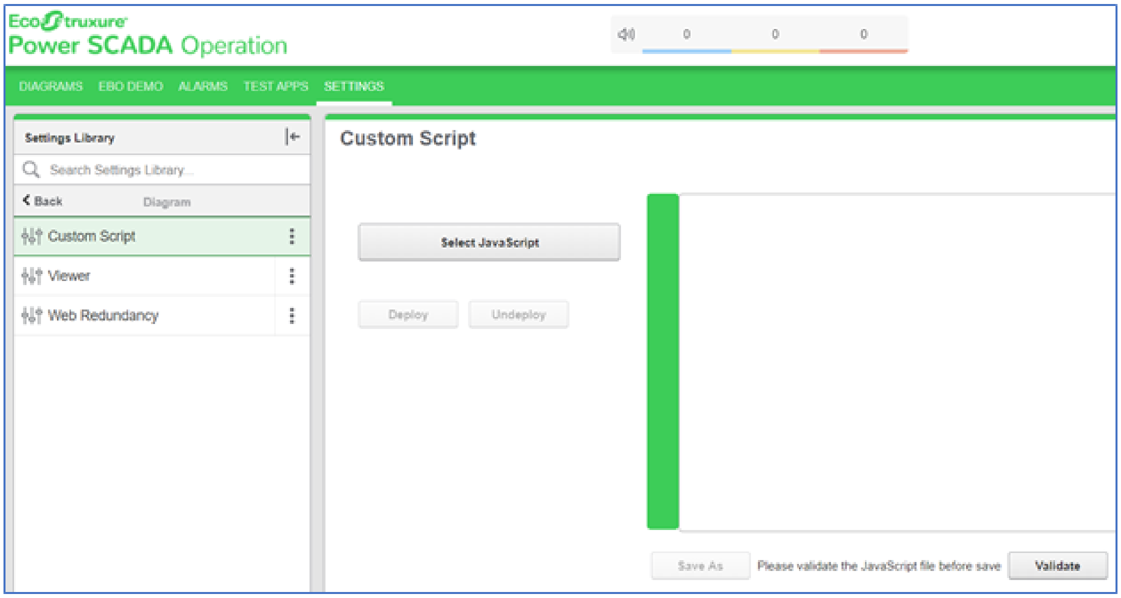
Project Name: PLS_Example

Enable Custom Script: on

Creating custom TGML graphics scripts

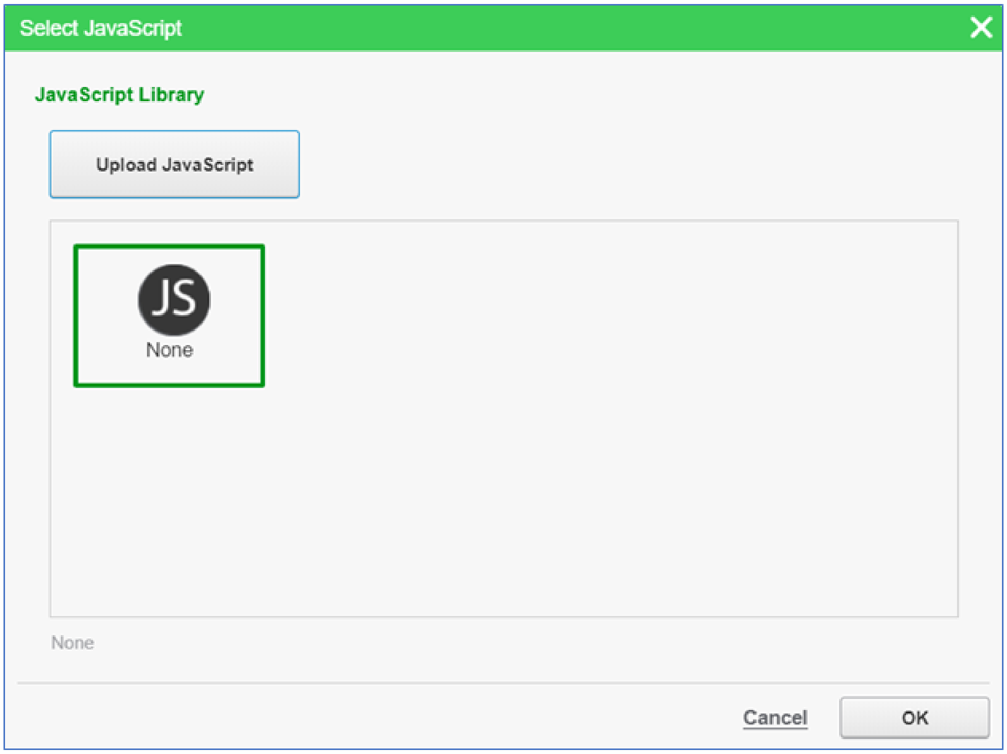
To create a custom TGML graphics script:

1. In PSO Web Applications, navigate to **Settings > Diagrams > Custom Script**.
2. Click **Select JavaScript**.



Please validate the JavaScript file before save

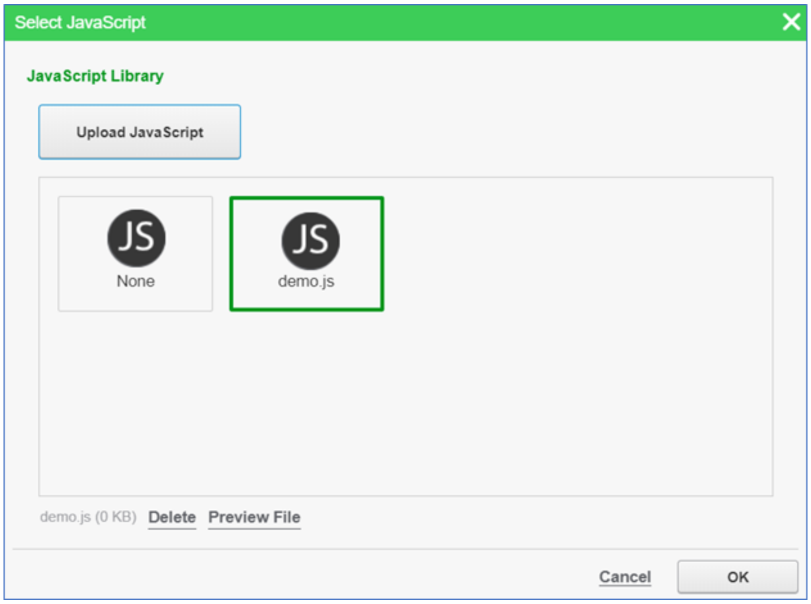
3. Click **Upload JavaScript**.



4. Click **Choose Files...**. You can choose multiple files with the `*.js` extension.

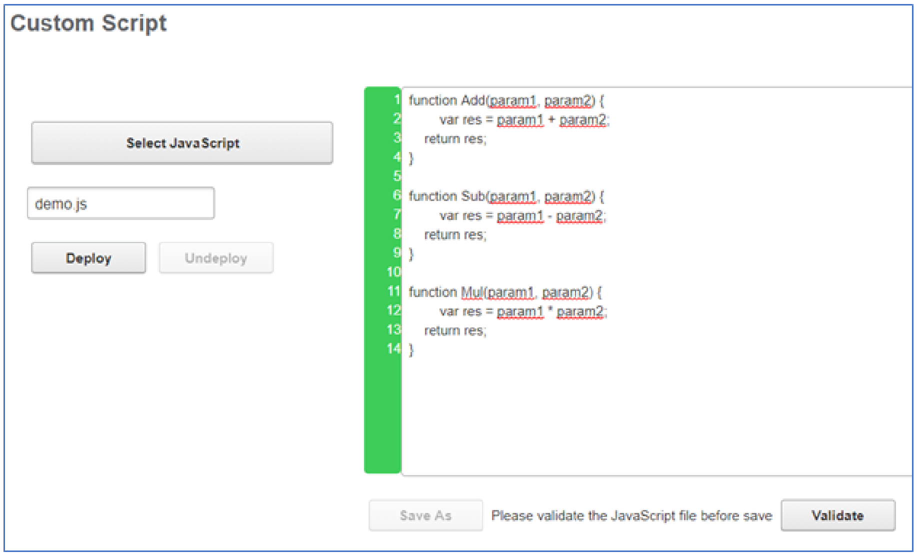
NOTE: Make sure that the file size does not exceed 2 MB.

5. In the Upload new files window, select the file, and then click **Finish**.
6. In the Select JavaScript window, select the file, and then click **OK**.



NOTE: You can delete or preview the selected JavaScript file by clicking **Delete** or **Preview File** at the bottom of the screen.

The file content is populated in the text area with line numbers displayed based on the number of lines in the file.

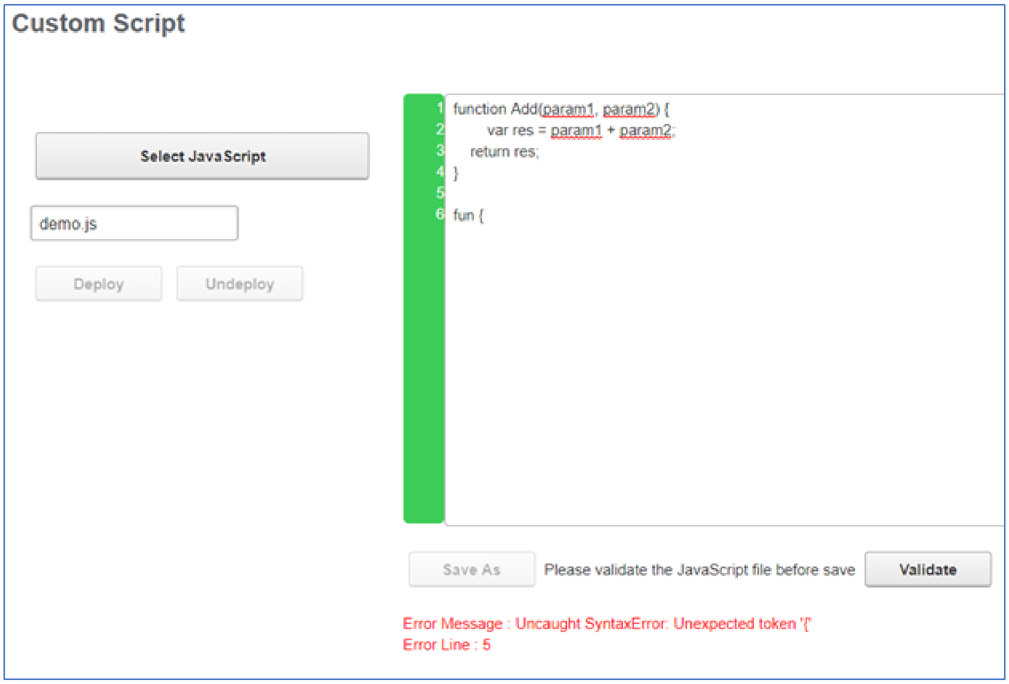


```
function Add(param1, param2) {
      var res = param1 + param2;
   return res;
}

function Sub(param1, param2) {
      var res = param1 - param2;
   return res;
}

function Mul(param1, param2) {
      var res = param1 * param2;
   return res;
}
```

Please validate the JavaScript file before save

7. Click **Validate** to validate the file.

> **NOTE**: If there is an error in the JavaScript file, the line number with the error is displayed and you can fix the error by editing the script in the text box window.



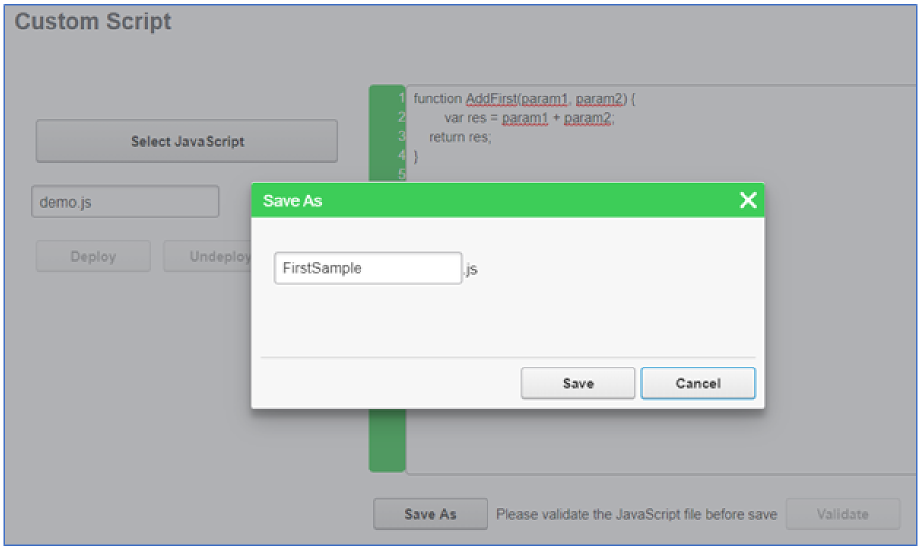
```
function Add(param1, param2) {
      var res = param1 + param2;
   return res;
}

fun {
```

Please validate the JavaScript file before save

Error Message : Uncaught SyntaxError: Unexpected token '{'
Error Line : 5

8. Once the validation process completes successfully, click **Save As**.

9. Enter the file name in the Save As window and click **Save**.

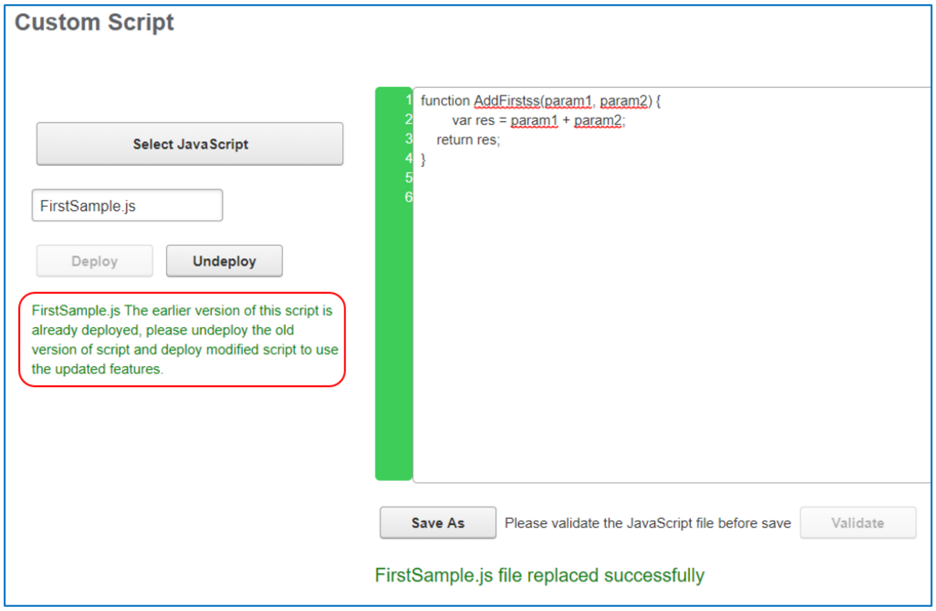


```
function AddFirst(param1, param2) {
        var res = param1 + param2;
    return res;
}
```

Please validate the JavaScript file before save

After the file is saved, the new file is displayed in the Select JavaScript window.

NOTE: The file is saved to `web\SystemDataService\App_Data\CustomScript\UploadJS` folder and is not yet deployed.



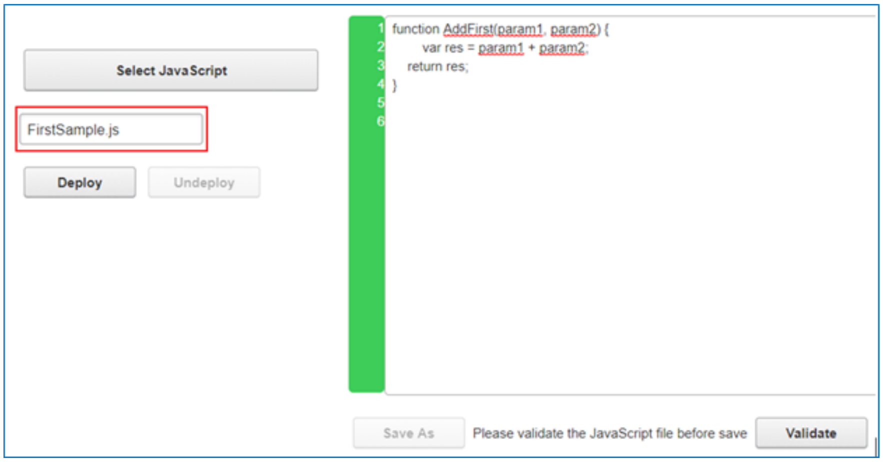
```
function AddFirstss(param1, param2) {
        var res = param1 + param2;
    return res;
}
```

FirstSample.js The earlier version of this script is already deployed, please undeploy the old version of script and deploy modified script to use the updated features.

Please validate the JavaScript file before save

FirstSample.js file replaced successfully

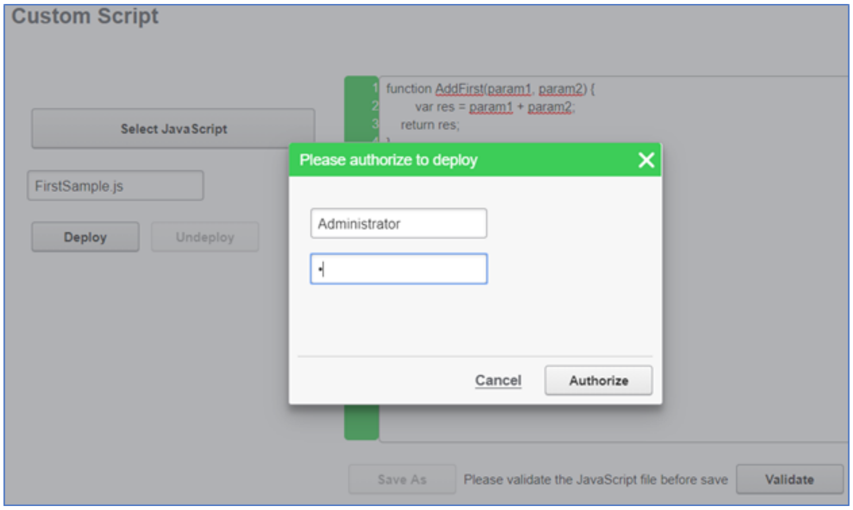
10. In the Select JavaScript window, select the required JavaScript file and click **OK**. The selected file appears on the left of the text box area.



```
function AddFirst(param1, param2) {
        var res = param1 + param2;
    return res;
}
```

NOTE: The options available for the selected JavaScript file are displayed based on the state of deployment.

To deploy the JavaScript, click **Deploy**, and then enter your log in credentials.



```
function AddFirst(param1, param2) {
        var res = param1 + param2;
    return res;
```

After the log in details are verified, the files are copied to: `Web\SystemDataService\App_Data\CustomScript\DeployJS`.

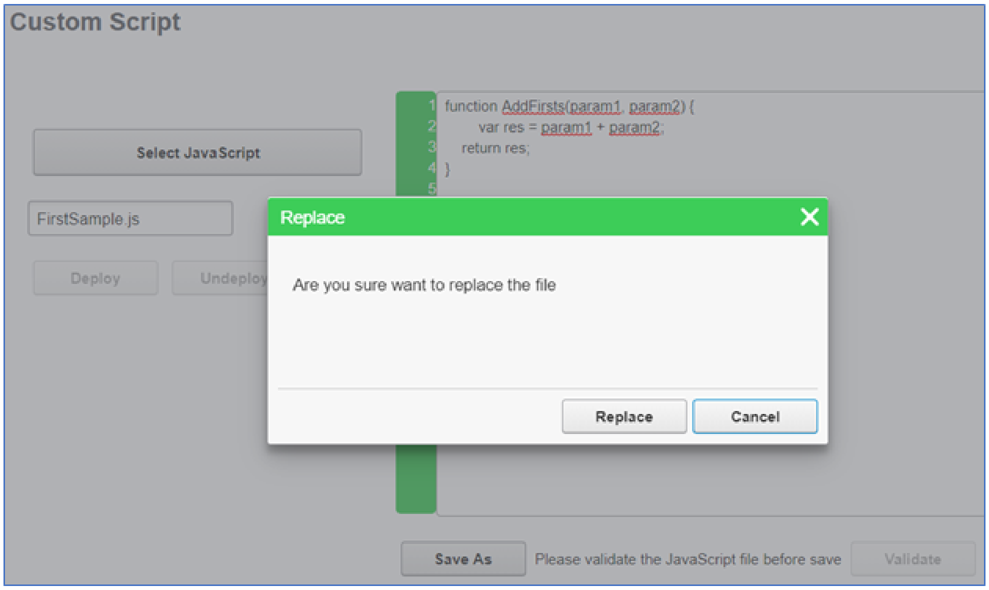
To undeploy the JavaScript, click **Undeploy** and enter your login credentials.

After the login details are verified, the file is removed from the DeployJS folder but is still available in the UploadJS folder.

Modifying deployed custom TGML graphics scripts

To modify a deployed file:

1. After you make your changes in the deployed file, click **Validate**, then click **Save As**.
2. In the confirmation window, click **Replace**.

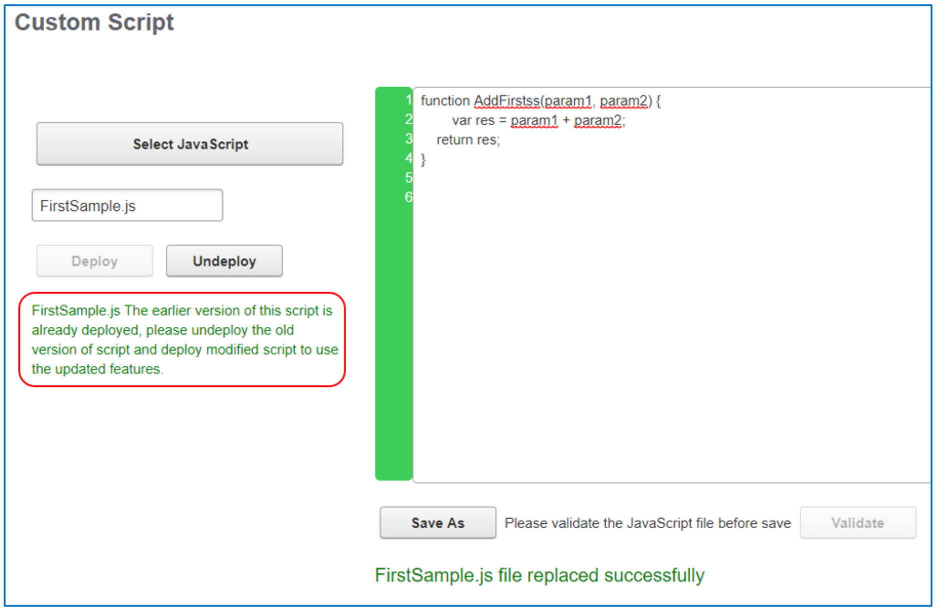


```
function AddFirsts(param1, param2) {
     var res = param1 + param2;
   return res;
}
```

Are you sure want to replace the file

Please validate the JavaScript file before save

3. If the following message appears, you have two versions of the same file. You must undeploy the previous version of the file, and then deploy the new version.



```
function AddFirstss(param1, param2) {
     var res = param1 + param2;
   return res;
}
```

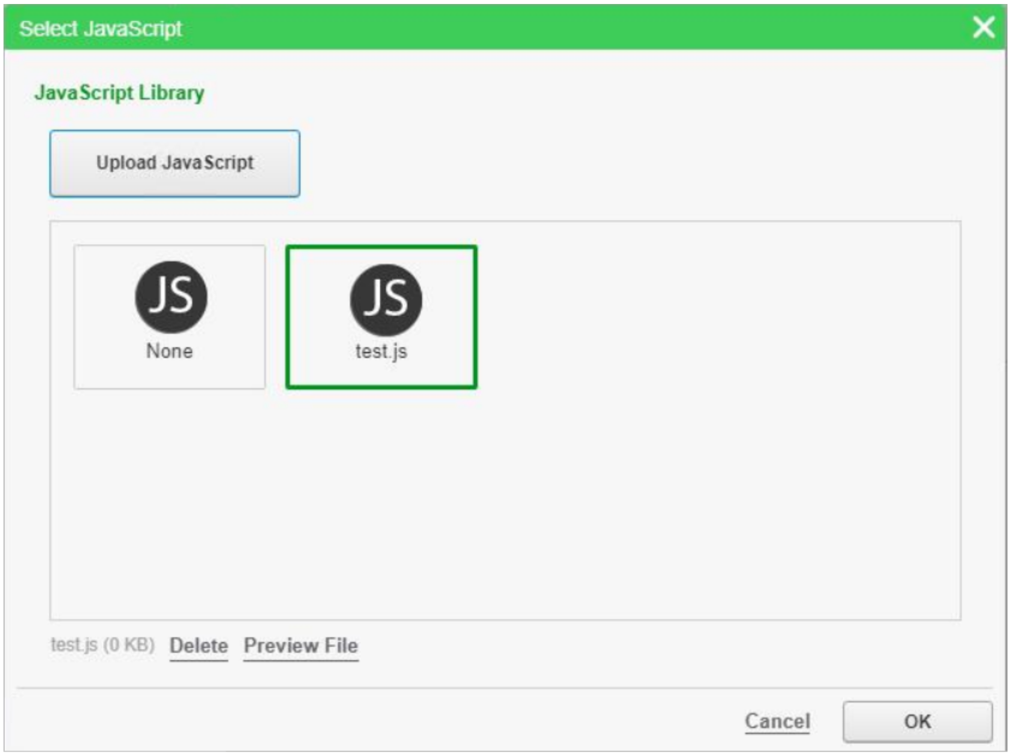
FirstSample.js The earlier version of this script is already deployed, please undeploy the old version of script and deploy modified script to use the updated features.

Please validate the JavaScript file before save

FirstSample.js file replaced successfully

Deleting custom TGML graphics scripts

To delete a file, you must undeploy it. Deployed files cannot be deleted. If you delete an undeployed file, you will lose the functions available in the script. Be sure to save a copy.



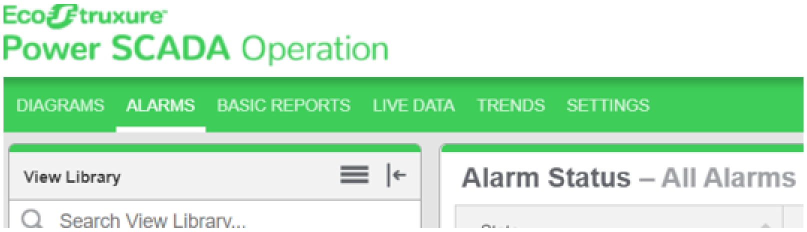
Navigate to associated graphics page

This feature enables you to navigate to an associated device diagram from an alarms details page.

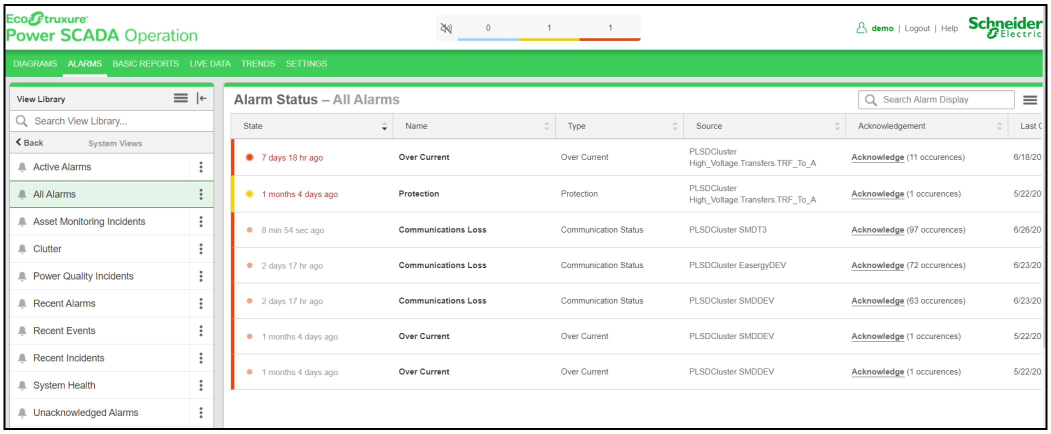
This topic lists the steps to view a device diagram associated to an alarm, incident, and event.

Alarms Workflow

1. Log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**).
2. Click **Alarms**:



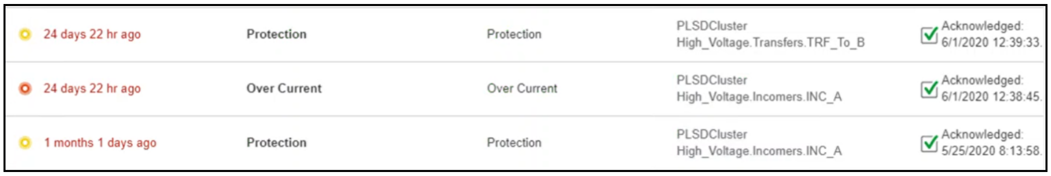
3. Click **All Alarms**.



Alarm Status – All Alarms

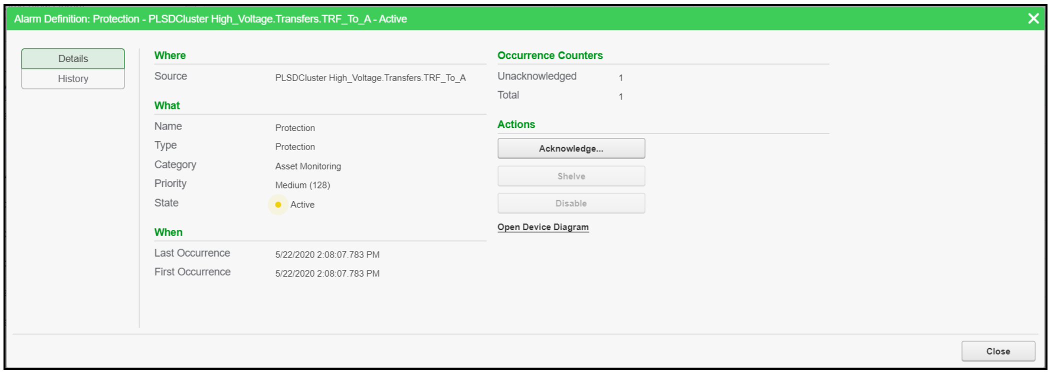
State	Name	Type	Source	Acknowledgement	Last O
● 7 days 18 hr ago	Over Current	Over Current	PLSDCluster High_Voltage.Transfers.TRF_To_A	Acknowledge (11 occurences)	6/18/20
● 1 months 4 days ago	Protection	Protection	PLSDCluster High_Voltage.Transfers.TRF_To_A	Acknowledge (1 occurences)	5/22/20
● 8 min 54 sec ago	Communications Loss	Communication Status	PLSDCluster SMDT3	Acknowledge (97 occurences)	6/26/20
● 2 days 17 hr ago	Communications Loss	Communication Status	PLSDCluster EasergyDEV	Acknowledge (72 occurences)	6/23/20
● 2 days 17 hr ago	Communications Loss	Communication Status	PLSDCluster SMDDEV	Acknowledge (63 occurences)	6/23/20
● 1 months 4 days ago	Over Current	Over Current	PLSDCluster SMDDEV	Acknowledge (1 occurences)	5/22/20
● 1 months 4 days ago	Over Current	Over Current	PLSDCluster SMDDEV	Acknowledge (1 occurences)	5/22/20

4. Double-click on any of the alarms listed to open the **Details** page.



● 24 days 22 hr ago	Protection	Protection	PLSDCluster High_Voltage.Transfers.TRF_To_B	☑ Acknowledged: 6/1/2020 12:39:33
● 24 days 22 hr ago	Over Current	Over Current	PLSDCluster High_Voltage.Incomers.INC_A	☑ Acknowledged: 6/1/2020 12:38:45
● 1 months 1 days ago	Protection	Protection	PLSDCluster High_Voltage.Incomers.INC_A	☑ Acknowledged: 5/25/2020 8:13:58

5. Click **Open Device Diagram** link to list associated TGMLs and render TGML in viewer.



Alarm Definition: Protection - PLSDCluster High_Voltage.Transfers.TRF_To_A - Active

Where

Source PLSDCluster High_Voltage.Transfers.TRF_To_A

What

Name Protection
Type Protection
Category Asset Monitoring
Priority Medium (128)
State ● Active

When

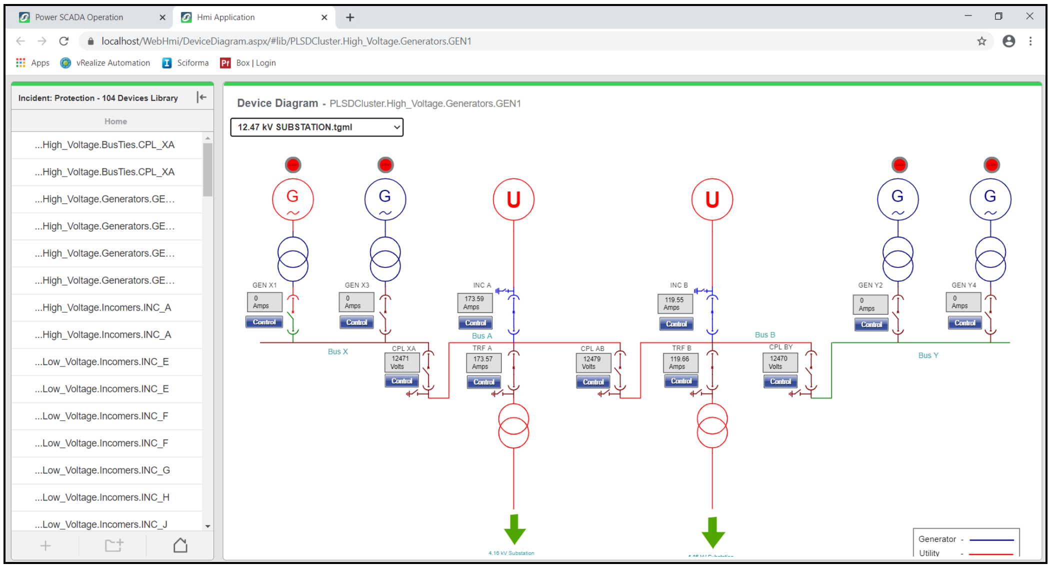
Last Occurrence 5/22/2020 2:08:07.783 PM
First Occurrence 5/22/2020 2:08:07.783 PM

Occurrence Counters

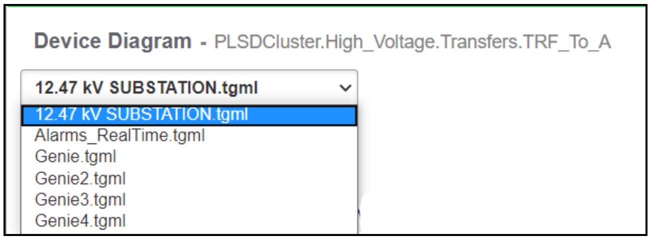
Unacknowledged 1
Total 1

Actions

The following screen is displayed:


Power SCADA Operation
Hmi Application
localhost/WebHmi/DeviceDiagram.aspx/#lib/PLSDCluster.High_Voltage.Generators.GEN1
Apps
vRealize Automation
Sciforma
Box | Login
Incident: Protection - 104 Devices Library
Home
...High_Voltage.BusTies.CPL_XA
...High_Voltage.BusTies.CPL_XA
...High_Voltage.Generators.GE...
...High_Voltage.Generators.GE...
...High_Voltage.Generators.GE...
...High_Voltage.Generators.GE...
...High_Voltage.Incomers.INC_A
...High_Voltage.Incomers.INC_A
...Low_Voltage.Incomers.INC_E
...Low_Voltage.Incomers.INC_E
...Low_Voltage.Incomers.INC_F
...Low_Voltage.Incomers.INC_F
...Low_Voltage.Incomers.INC_G
...Low_Voltage.Incomers.INC_H
...Low_Voltage.Incomers.INC_J
Device Diagram - PLSDCluster.High_Voltage.Generators.GEN1
12.47 kV SUBSTATION.tgml
G
U
GEN X1
GEN X3
INC A
INC B
GEN Y2
GEN Y4
Control
Bus X
Bus A
Bus B
Bus Y
CPL XA
TRF A
CPL AB
TRF B
CPL BY
Generator
Utility

6. Click the drop-down below Device Diagram, and then select the required TGML to display.

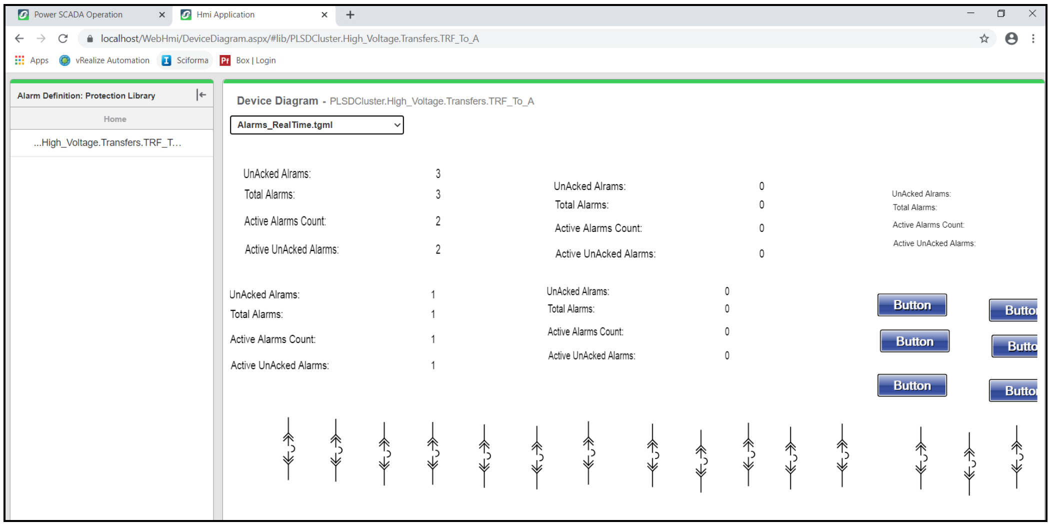


Device Diagram - PLSDCluster.High_Voltage.Transfers.TRF_To_A

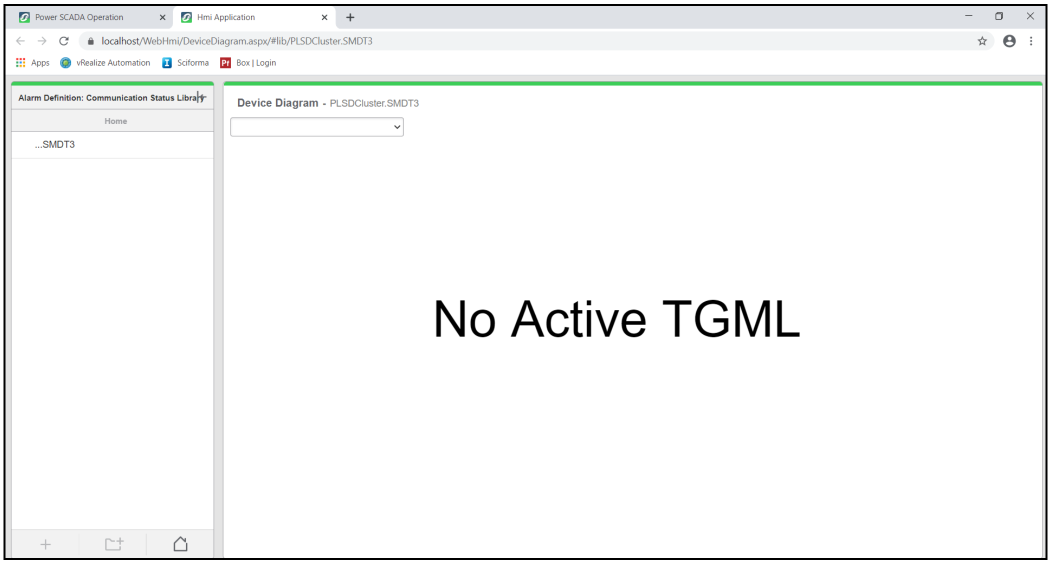
12.47 kV SUBSTATION.tgml

12.47 kV SUBSTATION.tgml
Alarms_RealTime.tgml
Genie.tgml
Genie2.tgml
Genie3.tgml
Genie4.tgml

7. Based on the user selection, the appropriate TGML is rendered:

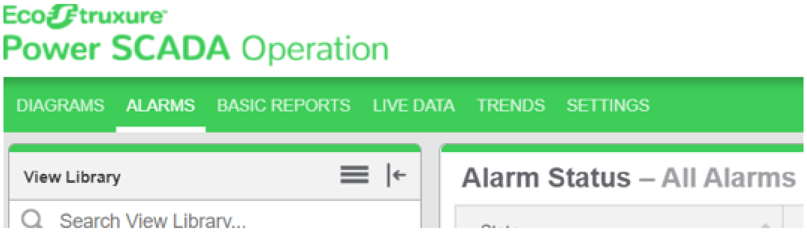


If there are no TGML graphics found for specific device name, then **No Active TGML** is displayed in the viewer:

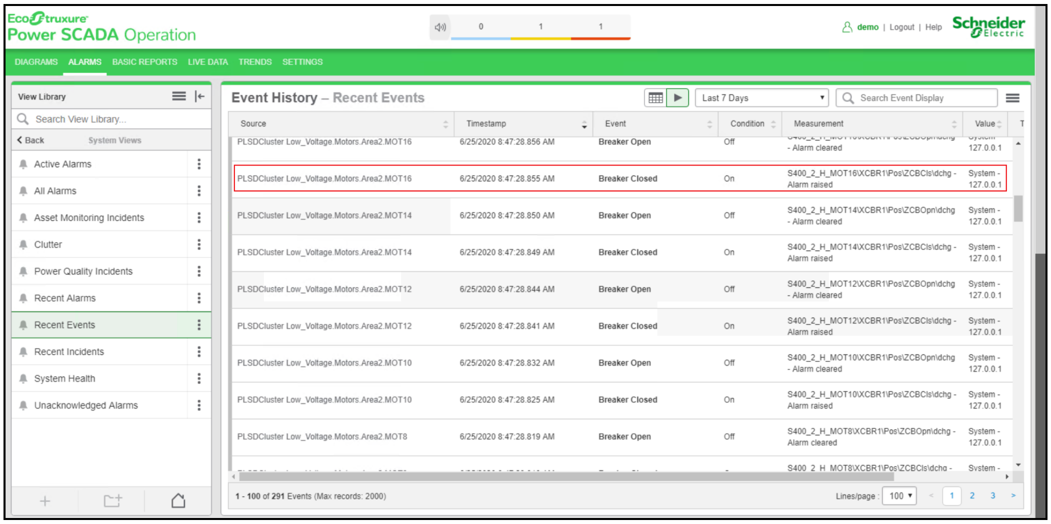

Power SCADA Operation
Hmi Application
localhost/WebHmi/DeviceDiagram.aspx/#lib/PLSDCluster.SMDT3
Apps
vRealize Automation
Sciforma
Box | Login
Alarm Definition: Communication Status Library
Home
...SMDT3
Device Diagram - PLSDCluster.SMDT3
No Active TGML

Events Workflow

1. Log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**).
2. Click **Alarms**:



3. Click **Recent Events**, and then double-click **Recent Events** to open details page.



- Active Alarms
- All Alarms
- Asset Monitoring Incidents
- Clutter
- Power Quality Incidents
- Recent Alarms
- Recent Events
- Recent Incidents
- System Health
- Unacknowledged Alarms

Event History – Recent Events

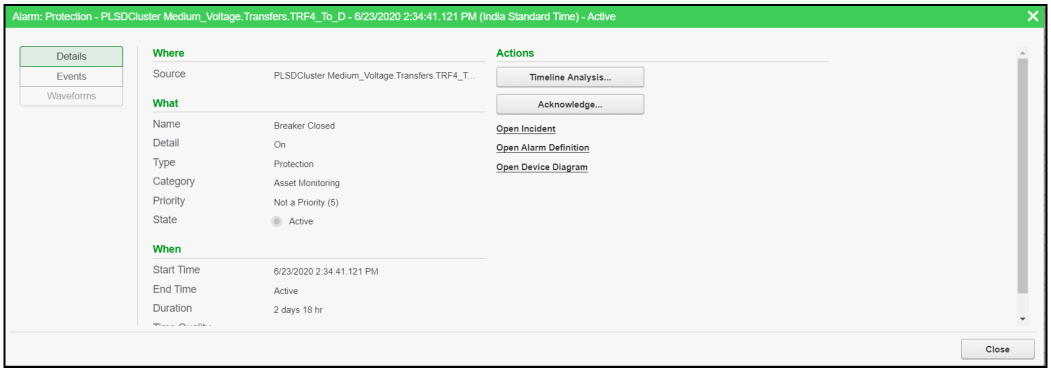
Last 7 Days | Search Event Display

Source	Timestamp	Event	Condition	Measurement	Value
PLSDCluster Low_Voltage.Motors.Area2.MOT16	6/25/2020 8:47:28.856 AM	Breaker Open	Off	- Alarm cleared	127.0.0.1
PLSDCluster Low_Voltage.Motors.Area2.MOT16	6/25/2020 8:47:28.855 AM	Breaker Closed	On	S400_2_H_MOT16\XCBR1\Pos\ZCBCls\dchg - Alarm raised	System - 127.0.0.1
PLSDCluster Low_Voltage.Motors.Area2.MOT14	6/25/2020 8:47:28.850 AM	Breaker Open	Off	S400_2_H_MOT14\XCBR1\Pos\ZCBOpn\dchg - Alarm cleared	System - 127.0.0.1
PLSDCluster Low_Voltage.Motors.Area2.MOT14	6/25/2020 8:47:28.849 AM	Breaker Closed	On	S400_2_H_MOT14\XCBR1\Pos\ZCBCls\dchg - Alarm raised	System - 127.0.0.1
PLSDCluster Low_Voltage.Motors.Area2.MOT12	6/25/2020 8:47:28.844 AM	Breaker Open	Off	S400_2_H_MOT12\XCBR1\Pos\ZCBOpn\dchg - Alarm cleared	System - 127.0.0.1
PLSDCluster Low_Voltage.Motors.Area2.MOT12	6/25/2020 8:47:28.841 AM	Breaker Closed	On	S400_2_H_MOT12\XCBR1\Pos\ZCBCls\dchg - Alarm raised	System - 127.0.0.1
PLSDCluster Low_Voltage.Motors.Area2.MOT10	6/25/2020 8:47:28.832 AM	Breaker Open	Off	S400_2_H_MOT10\XCBR1\Pos\ZCBOpn\dchg - Alarm cleared	System - 127.0.0.1
PLSDCluster Low_Voltage.Motors.Area2.MOT10	6/25/2020 8:47:28.825 AM	Breaker Closed	On	S400_2_H_MOT10\XCBR1\Pos\ZCBCls\dchg - Alarm raised	System - 127.0.0.1
PLSDCluster Low_Voltage.Motors.Area2.MOT8	6/25/2020 8:47:28.819 AM	Breaker Open	Off	S400_2_H_MOT8\XCBR1\Pos\ZCBOpn\dchg - Alarm cleared	System - 127.0.0.1
				S400_2_H_MOT8\XCBR1\Pos\ZCBCls\dchg -	System -

1 - 100 of 291 Events (Max records: 2000)

Lines/page : 100 ▾ 1 2 3 >

4. Click on **Open Device Diagram** link to list associated TGMLs and render TGML in viewer.


Alarm: Protection - PLSDCluster Medium_Voltage.Transfers.TRF4_To_D - 6/23/2020 2:34:41.121 PM (India Standard Time) - Active

Where

Source	PLSDCluster Medium_Voltage.Transfers.TRF4_T...

What

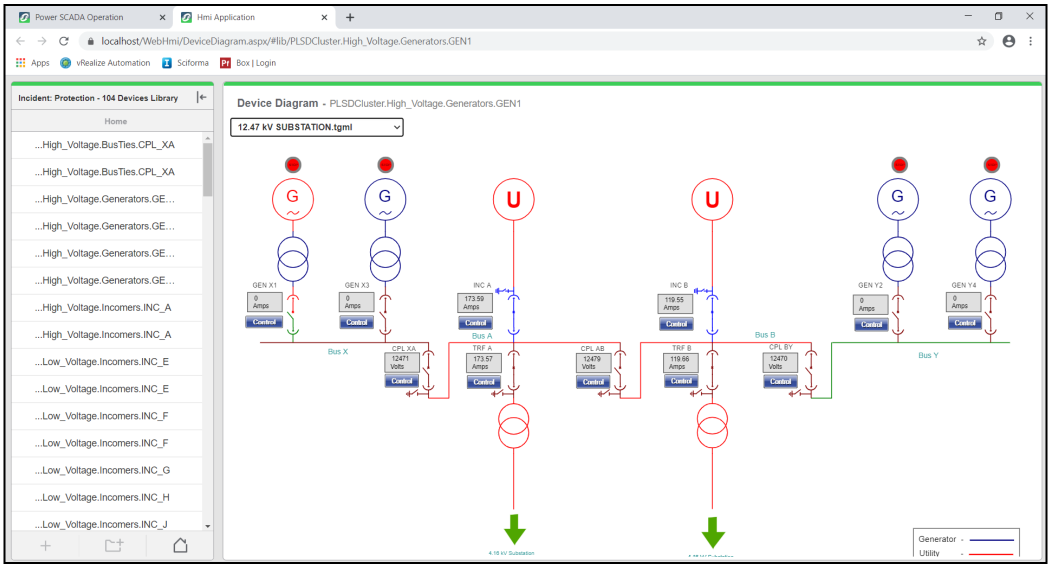
Name	Breaker Closed
Detail	On
Type	Protection
Category	Asset Monitoring
Priority	Not a Priority (5)
State	● Active

When

Start Time	6/23/2020 2:34:41.121 PM
End Time	Active
Duration	2 days 18 hr

Actions

The following screen is displayed.

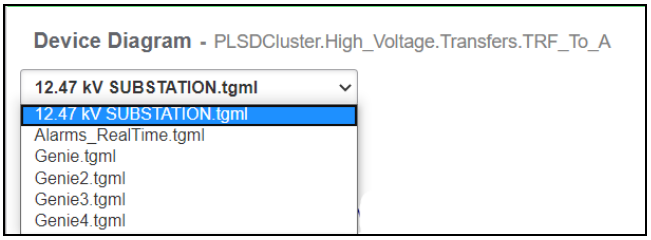

Power SCADA Operation
Hmi Application
localhost/WebHmi/DeviceDiagram.aspx/#lib/PLSDCluster.High_Voltage.Generators.GEN1
Apps
vRealize Automation
Sciforma
Box | Login
Incident: Protection - 104 Devices Library
Home

- ...High_Voltage.BusTies.CPL_XA
- ...High_Voltage.BusTies.CPL_XA
- ...High_Voltage.Generators.GE...
- ...High_Voltage.Generators.GE...
- ...High_Voltage.Generators.GE...
- ...High_Voltage.Generators.GE...
- ...High_Voltage.Incomers.INC_A
- ...High_Voltage.Incomers.INC_A
- ...Low_Voltage.Incomers.INC_E
- ...Low_Voltage.Incomers.INC_E
- ...Low_Voltage.Incomers.INC_F
- ...Low_Voltage.Incomers.INC_F
- ...Low_Voltage.Incomers.INC_G
- ...Low_Voltage.Incomers.INC_H
- ...Low_Voltage.Incomers.INC_J

Device Diagram - PLSDCluster.High_Voltage.Generators.GEN1

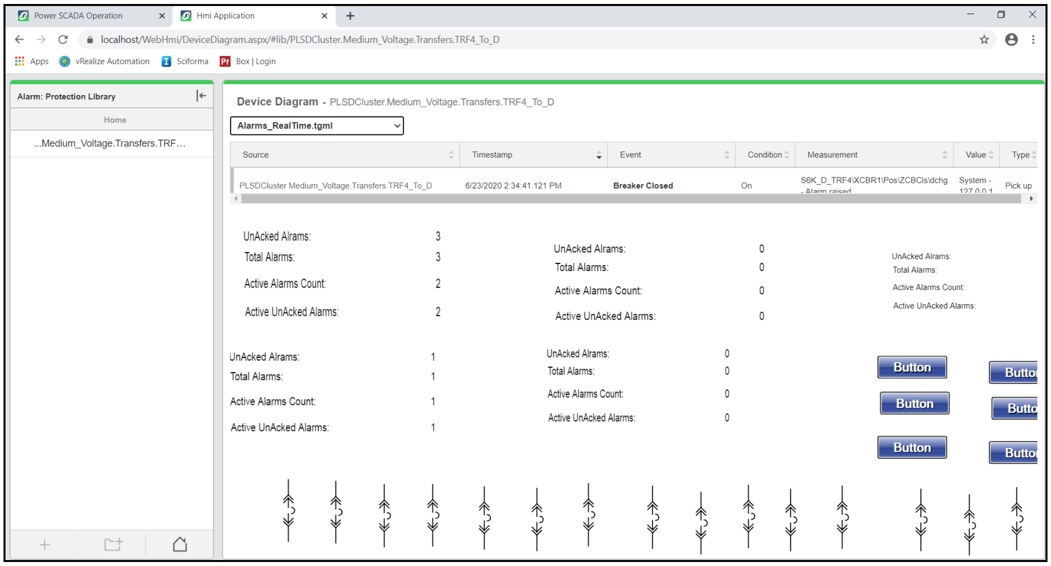
12.47 kV SUBSTATION.tgml

5. Click drop down below Device Diagram and select required TGML to display.



Device Diagram - PLSDCluster.High_Voltage.Transfers.TRF_To_A

6. Based on the user selection, the appropriate TGML is rendered:



Power SCADA Operation
Hmi Application
localhost/WebHmi/DeviceDiagram.aspx/#lib/PLSDCluster.Medium_Voltage.Transfers.TRF4_To_D
Apps
vRealize Automation
Sciforma
Box | Login
Alarm: Protection Library
Home
...Medium_Voltage.Transfers.TRF...
Alarms_RealTime.tgml
Button
Button
Button
Button
Button
Button

Device Diagram - PLSDCluster.Medium_Voltage.Transfers.TRF4_To_D

Source	Timestamp	Event	Condition	Measurement	Value	Type
PLSDCluster Medium_Voltage.Transfers.TRF4_To_D	6/23/2020 2:34:41.121 PM	**Breaker Closed**	On	S6K_D_TRF4\XCBR1\Pos\ZCBCls\dchg - Alarm raised	System - 127.0.0.1	Pick up

UnAcked Alrams: 3
Total Alarms: 3
Active Alarms Count: 2
Active UnAcked Alarms: 2

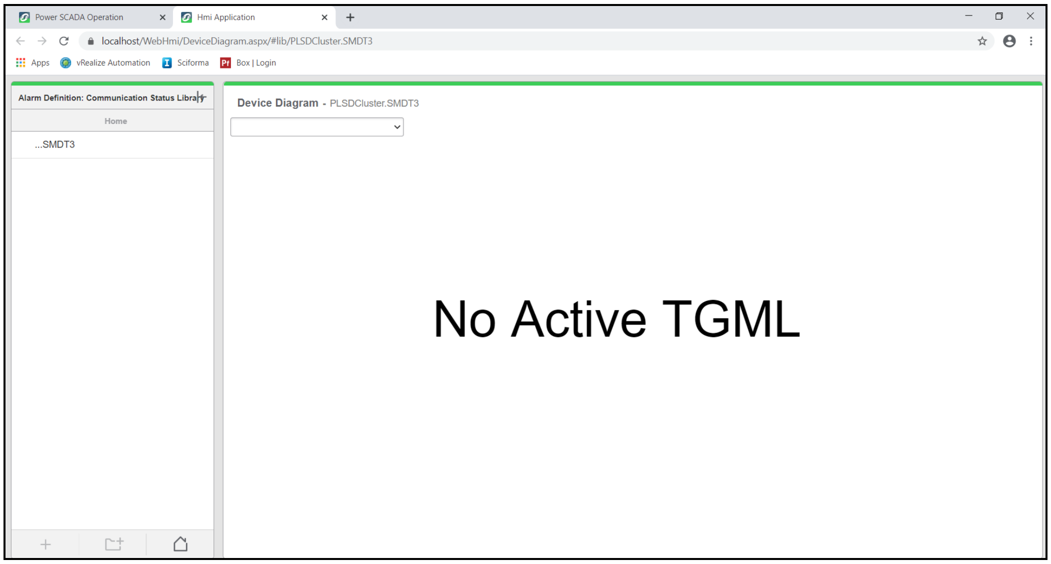
UnAcked Alrams: 0
Total Alarms: 0
Active Alarms Count: 0
Active UnAcked Alarms: 0

UnAcked Alrams:
Total Alarms:
Active Alarms Count:
Active UnAcked Alarms:

UnAcked Alrams: 1
Total Alarms: 1
Active Alarms Count: 1
Active UnAcked Alarms: 1

UnAcked Alrams: 0
Total Alarms: 0
Active Alarms Count: 0
Active UnAcked Alarms: 0

If there are no TGML graphics found for specific device name, then **No Active TGML** is displayed in the viewer.



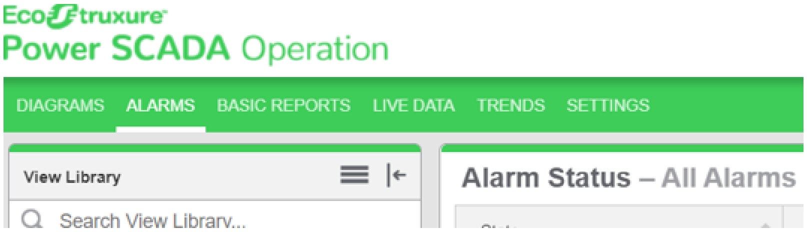
Power SCADA Operation
Hmi Application
localhost/WebHmi/DeviceDiagram.aspx/#lib/PLSDCluster.SMDT3
Apps
vRealize Automation
Sciforma
Box | Login
Alarm Definition: Communication Status Library
Home
...SMDT3

Device Diagram - PLSDCluster.SMDT3

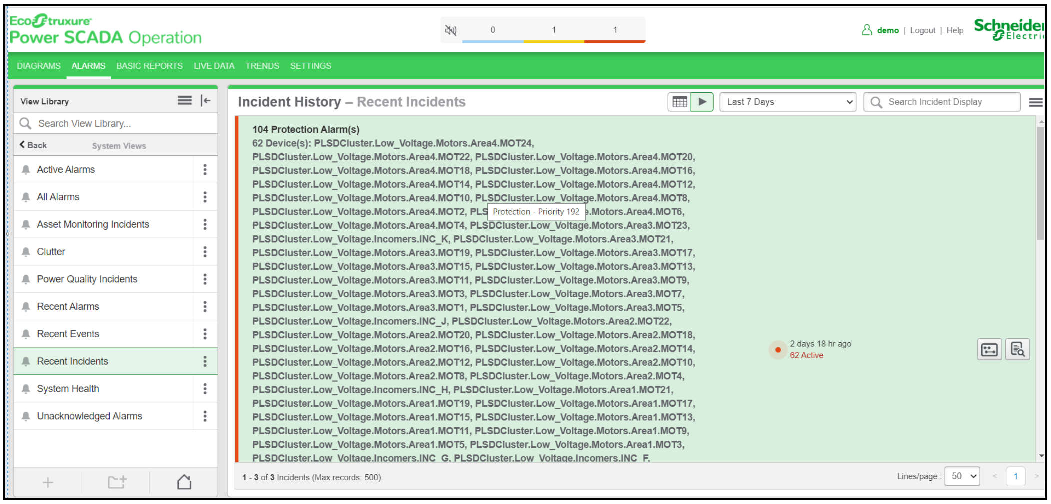
No Active TGML

Incidents Workflow

1. Log in to PSO Web Applications (**https://localhost/webhmi or https://ipaddress/webhmi**)
2. Click **Alarms**:



3. Click **Recent Incidents**, and then double-click **Recent Incidents** to open details page.



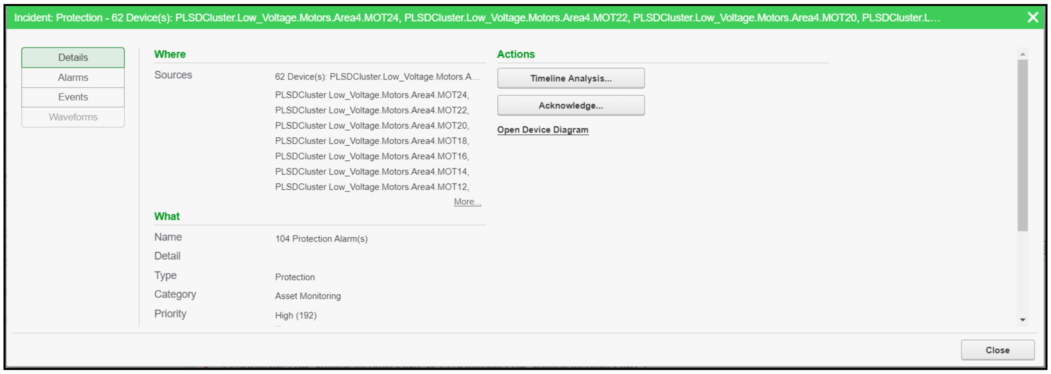
Incident History – Recent Incidents

104 Protection Alarm(s)
62 Device(s): PLSDCluster.Low_Voltage.Motors.Area4.MOT24,
PLSDCluster.Low_Voltage.Motors.Area4.MOT22, PLSDCluster.Low_Voltage.Motors.Area4.MOT20,
PLSDCluster.Low_Voltage.Motors.Area4.MOT18, PLSDCluster.Low_Voltage.Motors.Area4.MOT16,
PLSDCluster.Low_Voltage.Motors.Area4.MOT14, PLSDCluster.Low_Voltage.Motors.Area4.MOT12,
PLSDCluster.Low_Voltage.Motors.Area4.MOT10, PLSDCluster.Low_Voltage.Motors.Area4.MOT8,
PLSDCluster.Low_Voltage.Motors.Area4.MOT2, PLS Protection - Priority 192 e.Motors.Area4.MOT6,
PLSDCluster.Low_Voltage.Motors.Area4.MOT4, PLSDCluster.Low_Voltage.Motors.Area3.MOT23,
PLSDCluster.Low_Voltage.Incomers.INC_K, PLSDCluster.Low_Voltage.Motors.Area3.MOT21,
PLSDCluster.Low_Voltage.Motors.Area3.MOT19, PLSDCluster.Low_Voltage.Motors.Area3.MOT17,
PLSDCluster.Low_Voltage.Motors.Area3.MOT15, PLSDCluster.Low_Voltage.Motors.Area3.MOT13,
PLSDCluster.Low_Voltage.Motors.Area3.MOT11, PLSDCluster.Low_Voltage.Motors.Area3.MOT9,
PLSDCluster.Low_Voltage.Motors.Area3.MOT3, PLSDCluster.Low_Voltage.Motors.Area3.MOT7,
PLSDCluster.Low_Voltage.Motors.Area3.MOT1, PLSDCluster.Low_Voltage.Motors.Area3.MOT5,
PLSDCluster.Low_Voltage.Incomers.INC_J, PLSDCluster.Low_Voltage.Motors.Area2.MOT22,
PLSDCluster.Low_Voltage.Motors.Area2.MOT20, PLSDCluster.Low_Voltage.Motors.Area2.MOT18,
PLSDCluster.Low_Voltage.Motors.Area2.MOT16, PLSDCluster.Low_Voltage.Motors.Area2.MOT14,
PLSDCluster.Low_Voltage.Motors.Area2.MOT12, PLSDCluster.Low_Voltage.Motors.Area2.MOT10,
PLSDCluster.Low_Voltage.Motors.Area2.MOT8, PLSDCluster.Low_Voltage.Motors.Area2.MOT4,
PLSDCluster.Low_Voltage.Incomers.INC_H, PLSDCluster.Low_Voltage.Motors.Area1.MOT21,
PLSDCluster.Low_Voltage.Motors.Area1.MOT19, PLSDCluster.Low_Voltage.Motors.Area1.MOT17,
PLSDCluster.Low_Voltage.Motors.Area1.MOT15, PLSDCluster.Low_Voltage.Motors.Area1.MOT13,
PLSDCluster.Low_Voltage.Motors.Area1.MOT11, PLSDCluster.Low_Voltage.Motors.Area1.MOT9,
PLSDCluster.Low_Voltage.Motors.Area1.MOT5, PLSDCluster.Low_Voltage.Motors.Area1.MOT3,
PLSDCluster.Low_Voltage.Incomers.INC_G, PLSDCluster.Low_Voltage.Incomers.INC_F,

2 days 18 hr ago
62 Active

1 - 3 of 3 Incidents (Max records: 500)

4. Click **Open Device Diagram** link to list associated TGMLs and render TGML in viewer.



Incident: Protection - 62 Device(s): PLSDCluster.Low_Voltage.Motors.Area4.MOT24, PLSDCluster.Low_Voltage.Motors.Area4.MOT22, PLSDCluster.Low_Voltage.Motors.Area4.MOT20, PLSDCluster.L...

Where

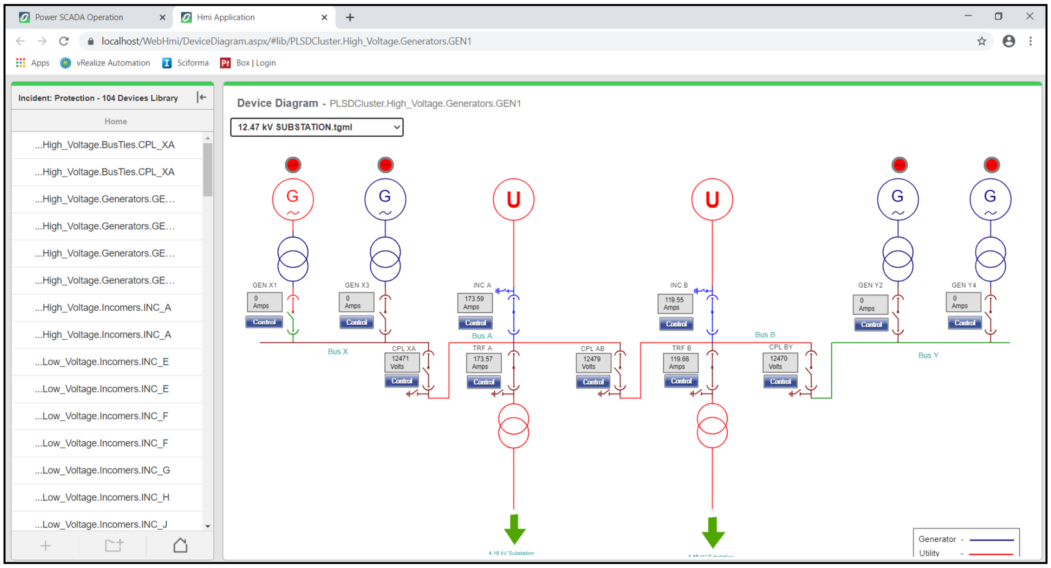
Sources 62 Device(s): PLSDCluster.Low_Voltage.Motors.A...
PLSDCluster Low_Voltage.Motors.Area4.MOT24,
PLSDCluster Low_Voltage.Motors.Area4.MOT22,
PLSDCluster Low_Voltage.Motors.Area4.MOT20,
PLSDCluster Low_Voltage.Motors.Area4.MOT18,
PLSDCluster Low_Voltage.Motors.Area4.MOT16,
PLSDCluster Low_Voltage.Motors.Area4.MOT14,
PLSDCluster Low_Voltage.Motors.Area4.MOT12,
More...

What

Name	104 Protection Alarm(s)
Detail	
Type	Protection
Category	Asset Monitoring
Priority	High (192)

Actions

The following screen is displayed.



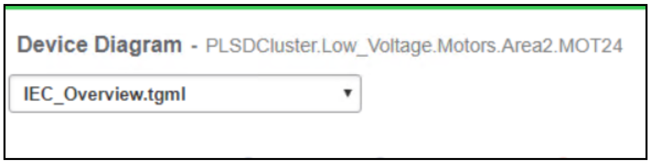
localhost/WebHmi/DeviceDiagram.aspx/#lib/PLSDCluster.High_Voltage.Generators.GEN1

Device Diagram - PLSDCluster.High_Voltage.Generators.GEN1

12.47 kV SUBSTATION.tgml

Power SCADA Operation
Hmi Application
Apps
vRealize Automation
Sciforma
Box | Login
Incident: Protection - 104 Devices Library
Home
...High_Voltage.BusTies.CPL_XA
...High_Voltage.BusTies.CPL_XA
...High_Voltage.Generators.GE...
...High_Voltage.Generators.GE...
...High_Voltage.Generators.GE...
...High_Voltage.Generators.GE...
...High_Voltage.Incomers.INC_A
...High_Voltage.Incomers.INC_A
...Low_Voltage.Incomers.INC_E
...Low_Voltage.Incomers.INC_E
...Low_Voltage.Incomers.INC_F
...Low_Voltage.Incomers.INC_F
...Low_Voltage.Incomers.INC_G
...Low_Voltage.Incomers.INC_H
...Low_Voltage.Incomers.INC_J
G
G
U
U
G
G
GEN X1
0 Amps
Control
GEN X3
0 Amps
Control
INC A
173.59 Amps
Control
INC B
119.55 Amps
Control
GEN Y2
0 Amps
Control
GEN Y4
0 Amps
Control
Bus X
Bus A
Bus B
Bus Y
CPL XA
12471 Volts
Control
TRF A
173.57 Amps
Control
CPL AB
12479 Volts
Control
TRF B
119.66 Amps
Control
CPL BY
12470 Volts
Control
4.16 kV Substation
Generator
Utility

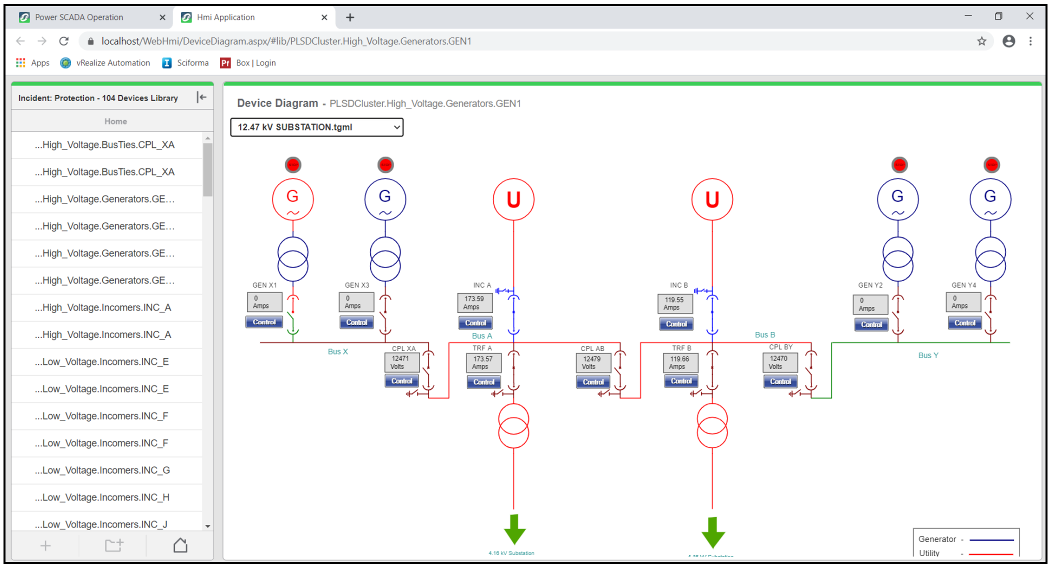
5. Click drop down below Device Diagram and select required TGML to display.



Device Diagram - PLSDCluster.Low_Voltage.Motors.Area2.MOT24

IEC_Overview.tgml

6. Based on the user selection, the appropriate TGML is rendered:



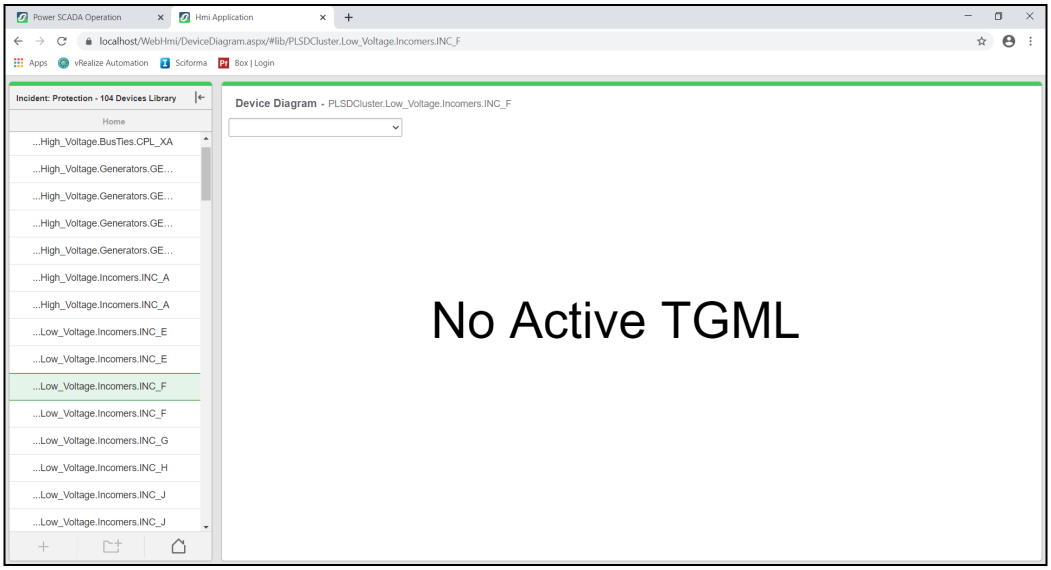
localhost/WebHmi/DeviceDiagram.aspx/#lib/PLSDCluster.High_Voltage.Generators.GEN1

Device Diagram - PLSDCluster.High_Voltage.Generators.GEN1

12.47 kV SUBSTATION.tgml

Power SCADA Operation
Hmi Application
Apps
vRealize Automation
Sciforma
Box | Login
Incident: Protection - 104 Devices Library
Home
...High_Voltage.BusTies.CPL_XA
...High_Voltage.BusTies.CPL_XA
...High_Voltage.Generators.GE...
...High_Voltage.Generators.GE...
...High_Voltage.Generators.GE...
...High_Voltage.Generators.GE...
...High_Voltage.Incomers.INC_A
...High_Voltage.Incomers.INC_A
...Low_Voltage.Incomers.INC_E
...Low_Voltage.Incomers.INC_E
...Low_Voltage.Incomers.INC_F
...Low_Voltage.Incomers.INC_F
...Low_Voltage.Incomers.INC_G
...Low_Voltage.Incomers.INC_H
...Low_Voltage.Incomers.INC_J
G
G
U
U
G
G
GEN X1
0 Amps
Control
GEN X3
0 Amps
Control
INC A
173.59 Amps
Control
INC B
119.55 Amps
Control
GEN Y2
0 Amps
Control
GEN Y4
0 Amps
Control
Bus X
Bus A
Bus B
Bus Y
CPL XA
12471 Volts
Control
TRF A
173.57 Amps
Control
CPL AB
12479 Volts
Control
TRF B
119.66 Amps
Control
CPL BY
12470 Volts
Control
4.16 kV Substation
Generator
Utility

If there are no TGMLs found for specific device name, then **No Active TGML** is displayed in the viewer:


Power SCADA Operation
Hmi Application
localhost/WebHmi/DeviceDiagram.aspx/#lib/PLSDCluster.Low_Voltage.Incomers.INC_F
Apps
vRealize Automation
Sciforma
Box | Login
Incident: Protection - 104 Devices Library
Home
...High_Voltage.BusTies.CPL_XA
...High_Voltage.Generators.GE...
...High_Voltage.Generators.GE...
...High_Voltage.Generators.GE...
...High_Voltage.Generators.GE...
...High_Voltage.Incomers.INC_A
...High_Voltage.Incomers.INC_A
...Low_Voltage.Incomers.INC_E
...Low_Voltage.Incomers.INC_E
...Low_Voltage.Incomers.INC_F
...Low_Voltage.Incomers.INC_F
...Low_Voltage.Incomers.INC_G
...Low_Voltage.Incomers.INC_H
...Low_Voltage.Incomers.INC_J
...Low_Voltage.Incomers.INC_J
Device Diagram - PLSDCluster.Low_Voltage.Incomers.INC_F

No Active TGML

Trends

⚠ 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.

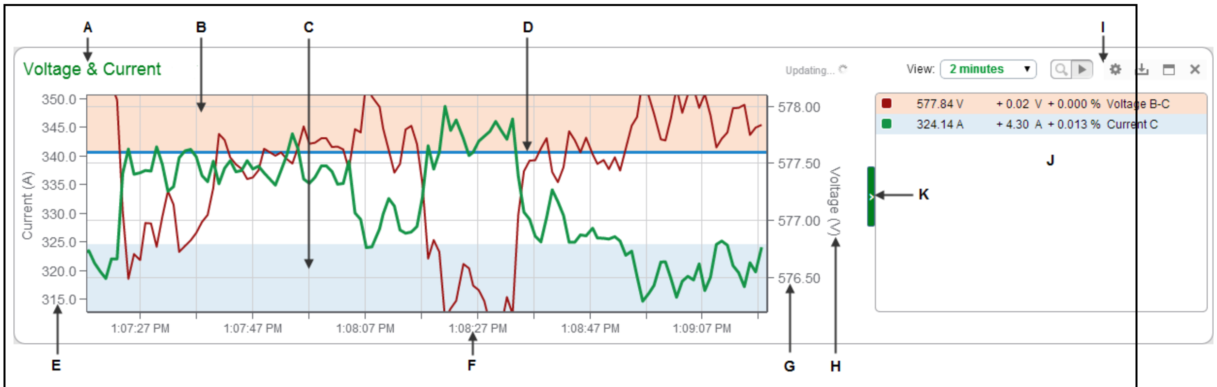
⚠ WARNING

UNINTENDED EQUIPMENT OPERATION

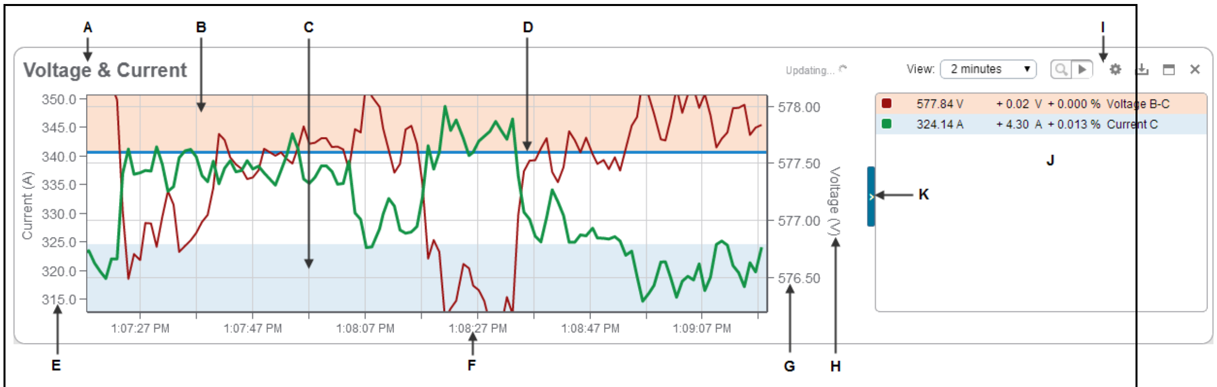
- Do not use the software or devices for critical control or protection applications where human or equipment safety relies on the operation of the control action.
- Do not use the software to control time-critical functions.
- Do not use the software to control remote equipment without proper access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

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

Use the Trends application to monitor current system conditions by displaying real-time data in a graphical format. In addition, you can save the trend data as a CSV file.



A	Title	**B**	Upper threshold	**C**	Lower threshold
D	Target line	**E**	Left axis	**F**	Scale (from View setting)
G	Right axis	**H**	Axis title	**I**	Trend options
J	Legend	**K**	Close/open toggle		



A	Title	**B**	Upper threshold	**C**	Lower threshold
D	Target line	**E**	Left axis	**F**	Scale (from View setting)
G	Right axis	**H**	Axis title	**I**	Trend options
J	Legend	**K**	Close/open toggle		

TIP: You can open the Trends application from the **TRENDS** link in the Web Applications banner.

Time display

See Time Display in Web Applications for information on how time is displayed in a system where the monitoring devices, the Power SCADA Operation/Web server, and the Web client (browser) are located in different time zones.

For information on how use the Trends application, see Trends UI.

For information on how to configure Trends, see Trends configuration.

Trends configuration

Configuring General settings

To configure general settings:

1. In the Trend Setup dialog, on the **General** tab, enter a title for the trend.
2. To add a new data series, click **Add** under **Data Series**. This opens the Add Data Series dialog.
3. To edit an existing series, select it, and then click **Edit**. This opens the Edit Data Series dialog.
4. For the selected source, expand a measurement type, for example **Voltage**, and click the specific measurement you want to include in your trend, for example **Voltage A-B**.

   The measurements are listed in alphabetical order by measurement category. You can use the **Search Measurements** field to find a specific measurement category or measurement.

   (Optional) Click **Show Advanced** to open options for filtering the measurements.
5. (Optional) Select **Display Name** if you want to enter a series name of your choice for trend data purposes. By default, a series name is a combination of source and measurement information formatted as `group.source measurement`, for example `BldgA.meterA Voltage A-B`.
6. (Optional) Select **Display Units** and enter a unit description of your choice.
7. You can modify the following settings for each source measurement:
   - **Style**: select the color and line thickness from the available choices in the dropdown menus.
   - **Decimals**: select the number of decimal places for the data displayed in the legend.
   - **Plot on**: select **Right** or **Left Axis** for the location of the measurement values for the selected measurement.
   - **Overlay**: select the values that you want to overlay on the trend. By default, no items are selected. The selections are **Min**, **Max**, and **Mean**.
   - **Data Source**: select where to access the data for the trend. The options are to gather series data from the source in real-time, gather series data from the database as it is being logged, or gather real-time series data from the source and historical data from the database to fill the trend, if possible.
8. Click **OK** to save your changes and close the Add (or Edit) Data Series dialog and to return to the Trend Setup dialog.
9. Click **Add** to specify additional sources and measurements for the trend.
10. Select **Private Trend** to keep this trend private, or clear the check box to make it public.

    NOTE: A public item is visible to all users in your user group. A private item is visible to you and any user in your user group with Edit permissions on this item type. See "Assign and control user privileges" on page 686 for details.

Configuring Axes settings

To configure axes settings:

1. In the Trend Setup dialog, on the **Axes** tab, enter a label for the axes in the **Title** field under **Right Axis (Primary)** or **Left Axis (Secondary)**.

   Axis titles only appear if you have configured at least one measurement series and it appears on the trend.

2. For **Right Axis (Primary)**, **Max Value** and **Min Value** are set to **Auto** by default.

   a. (Optional) Select **Fixed** and enter the maximum or minimum values in the respective input fields.

      i. When you select **Upper Threshold**:

      - Select a color from the color selector for area shading on the trend between the maximum value and the upper threshold value.
      - Enter a value for the upper threshold in the input field.

        Each time the latest data point of a measurement series occurs in an upper or lower threshold, the color defined for the threshold also colors the background of the measurement series in the legend.

      ii. When you select **Lower Threshold**:

      - Select a color from the color selector for area shading on the trend between the minimum value and the lower threshold value.
      - Enter a value for the lower threshold in the input field.

        If the latest data point of a measurement series occurs in an upper or lower threshold, the color defined for the threshold also colors the background of the measurement series in the legend.

   b. (Optional) Select **Target Line**, then select a color from the color selector and enter a value for the target line in the input field.

      You can select the **Target Line** independently from the **Upper Threshold** or **Lower Threshold** settings.

3. For **Left Axis (Secondary)**, **Max Value** and **Min Value** are set to **Auto** by default.

   For **Fixed** maximum or minimum, enter the values in the respective input fields.

Configuring Chart settings

To configure chart settings:

1. In the Trend Setup dialog, on the **Chart** tab, select the text size from the list.

   The text size property is applied to trend axis labels, the size of the legend, the legend text size, and trend data point tooltips.

The default setting is **Medium**, and the choices are **Small**, **Medium**, or **Large**.

2. Select the position of the legend included in the trend display area from the list.

   The default setting is **Right**, which places the legend on the right side of the trend. The available choices are **Off**, **Left**, or **Right**.

3. Select the content that you want to include in the legend from the available settings.

   The default selections are **Name** and **Value**. The additional selections are **Difference** and **Difference (%)**.

   Name is either the default measurement name in the form of `group.device measurement`, or the custom name that you specified on the **Add** or **Edit Data Series** dialogs.

   Value is latest data value and the unit of measurement. For example, for voltage measurements, the default value is *`numeric_value`* `V` such as `415.2 V`.

   Difference is the change in the measurement from one update to the next. For example, if the voltage is `415.8` and it changes to `416.1` at the next trend update, the difference appears as `+0.3` in the legend.

   Difference (%) is the percentage change in the measurement from one update to the next. For example, if the voltage changes from `415.8` to `416.1` at the next trend update, the difference expressed as a percentage appears as `+0.072%` in the legend.

Configuring Data display settings

To configure data display settings:

1. In the Trend Setup dialog, on the **Data** tab, specify the **Data Update Intervals** in the **From device** and **From database** dropdown lists.

   The default setting is `5 seconds` for data updates for trends using the data directly from a device, and `5 minutes` for data updates for trends with data from a database.

2. Specify the **Data Points** for the x-axis of the trend in the **Max per series** input field.

   The default setting is `40000`.

   The value must be between 100 and 500,000. Increasing the value adds more data points per series but this can result in a degradation of trend performance.

   Examples:

   - A data interval of 1 second equates to 3600 data points per hour (60 points per minute X 60 minutes per hour). At a setting of 40000 points, approximately 11.1 hours of data is retained for viewing (40, 000 points / 3600 points per hour = approximately 11.1 hours).
   - A data interval of 5 seconds equates to 720 data points per hour (12 points per minute X 60 minutes per hour). At a setting of 40000 points, approximately 55.5 hours of data is retained for viewing (40, 000 points / 720 points per hour = approximately 55.5 hours).

- A data interval of 10 seconds equates to 360 data points per hour (6 points per minute X 60 minutes per hour). At a setting of 40000 points, approximately 111.1 hours of data is retained for viewing (40, 000 points / 360 points per hour = approximately 111.1 hours).

Applications

You can extend the capabilities of Power SCADA Operation by configuring applications. This chapter contains information on the applications you can add to Power SCADA Operation.

Thermal Monitoring of Medium Voltage Substations Application

This section describes the components, procedures, and best practices for setting up the Thermal Monitoring of Medium Voltage (MV) Substations application.

Hardware setup is beyond the scope of this section. For information on configuring and troubleshooting hardware, refer to the User Guide for the selected device on www.se.com or the Exchange Extranet.

Overview

This application provides remote, continuous monitoring of the thermal conditions of MV substations equipped with Easergy TH110 temperature sensors and Easergy CL110 environmental sensors. Thermal monitoring can help you find exceptional conditions in the substation equipment, such as overloads or faulty power connections of cables, busbars, circuit breakers, or transformers. In addition, this application can monitor the environmental conditions, ambient temperature and humidity, inside the substation and switchgear cubicles. You can see temperature data in real-time, analyze historical trends, and receive alarms and notifications. Any applicable Power SCADA Operation power monitoring features can be used with the substation monitoring data.

For more information on the value of Continuous Thermal Monitoring see the EcoStruxure Power Digital Applications for Large Buildings & Critical Facilities Design Guide on www.se.com or the Exchange Extranet.

⚠ 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.

Components

The Thermal Monitoring of MV Substations application is based on the following system components and features:

- Power SCADA Operation 2020 software.
- Substation with Substation Monitoring Devices (SMDs), Harmony ZBRN32 access points, Easergy TH110 thermal sensors and Easergy CL110 environmental sensors.

Prerequisites

The following is required to set up the Thermal Monitoring of MV Substations application in Power SCADA Operation 2020:

- Power SCADA Operation 2020 must be installed and commissioned.
- The TH110 and CL110 sensors, the ZBRN32 access points, and the SMDs in the substation must be configured, connected, and communicating.

  NOTE: This application supports SMD v4.0 and v3.0. SMD v2.0 is not supported.

- The SMDs must be accessible from the Power SCADA Operation server by Ethernet.
- You must know the IP address of the SMDs.

  NOTE: If you are using Power SCADA Operation 2020 with Advanced Reporting and Dashboards and want to set up dashboards to display substation data, refer to the Thermal Monitoring of Medium Voltage (MV) Substations section in the Power Monitoring Expert System Guide.

Limitations

There are no specific, software-based limitations for this application in addition to the general Power SCADA Operation performance and scalability limits.

The SMDs and ZBRN32 access points have the following limitations:

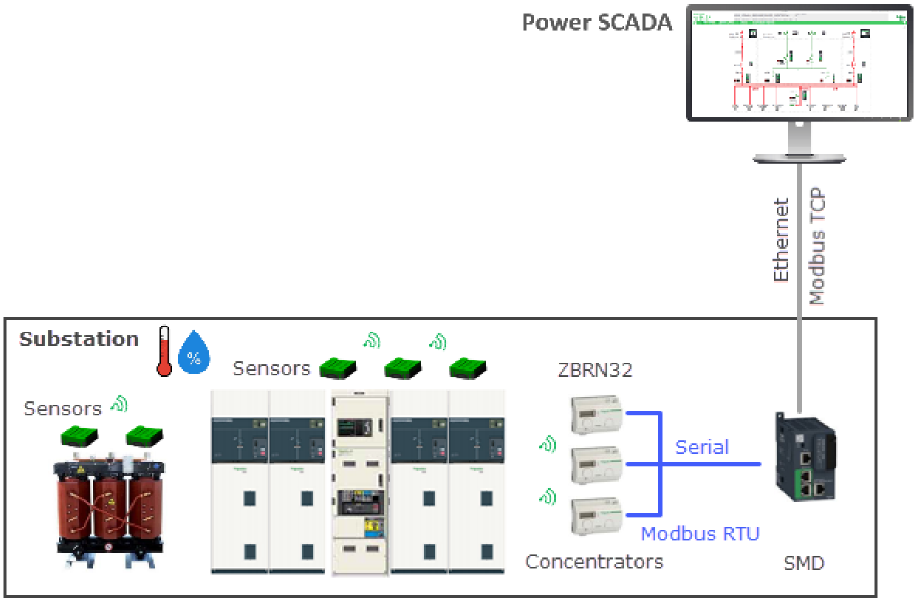
- Each SMD can support up to three ZBRN32 access points.
- Each ZBRN32 access point can support up to 60 sensors.
- Each SMD can support up to 16 cubicles and 16 transformers
  - For each cubicle, the SMD can support sensors for 1 busbar, 2 circuit breakers, 2 cables, and a set of environmental measurements.
  - For each transformer, the SMD can support sensors for MV taps, LV taps, windings, and tapping links.

NOTE: Only those cubicles, transformers, and sensor locations that are configured in the SMD are shown in Power SCADA Operation. For more information on configuring your SMD, refer to the correct version SMD Installation Manual on www.se.com or the Exchange Extranet.

Design

Easergy CL110 and TH110 sensors are installed in the substation. The sensors wirelessly send measurement data to the ZBRN32 access points. The access points are connected to SMDs through Modbus serial communications. The SMDs are connected to Power SCADA Operation through an Ethernet connection.

Power SCADA Operation has a pre-configured device profile for the SMD. See "Configuring a thermal monitoring device profile" on page 795 for more information.



NOTE: If you are using Power SCADA Operation 2020 with Advanced Reporting and Dashboards and want to set up reports and dashboards to display substation data, refer to the Thermal Monitoring of Medium Voltage (MV) Substations section in the PME System Guide on www.se.comor the Exchange Extranet.

Configuration

Before configuring the Thermal Monitoring of MV Substations application, confirm that the Prerequisites are in place for your system.

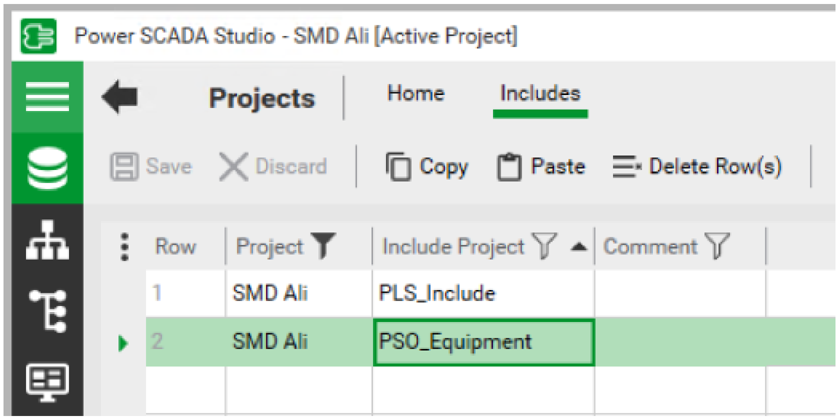
Configuring this application requires the following steps:

- Adding a Thermal Monitoring Include to a Power SCADA Project
- Configuring a thermal monitoring device profile
- Adding a thermal monitoring device to a Power SCADA Project
- Configuring a thermal monitoring popup in a graphic

Adding a thermal monitoring Include to a Power SCADA Project

To add a thermal monitoring Include to a Power SCADA Operation project:

1. Open the Power SCADA Project Studio.
2. Select **Projects** > **Includes**.
3. In a new row, select the Project name, and select **PSO_Equipment** as the Include Project.



Row	Project	Include Project	Comment
1	SMD Ali	PLS_Include	
2	SMD Ali	PSO_Equipment	

4. Click **Save**.

Configuring a thermal monitoring device profile

To create a thermal monitoring device profile:

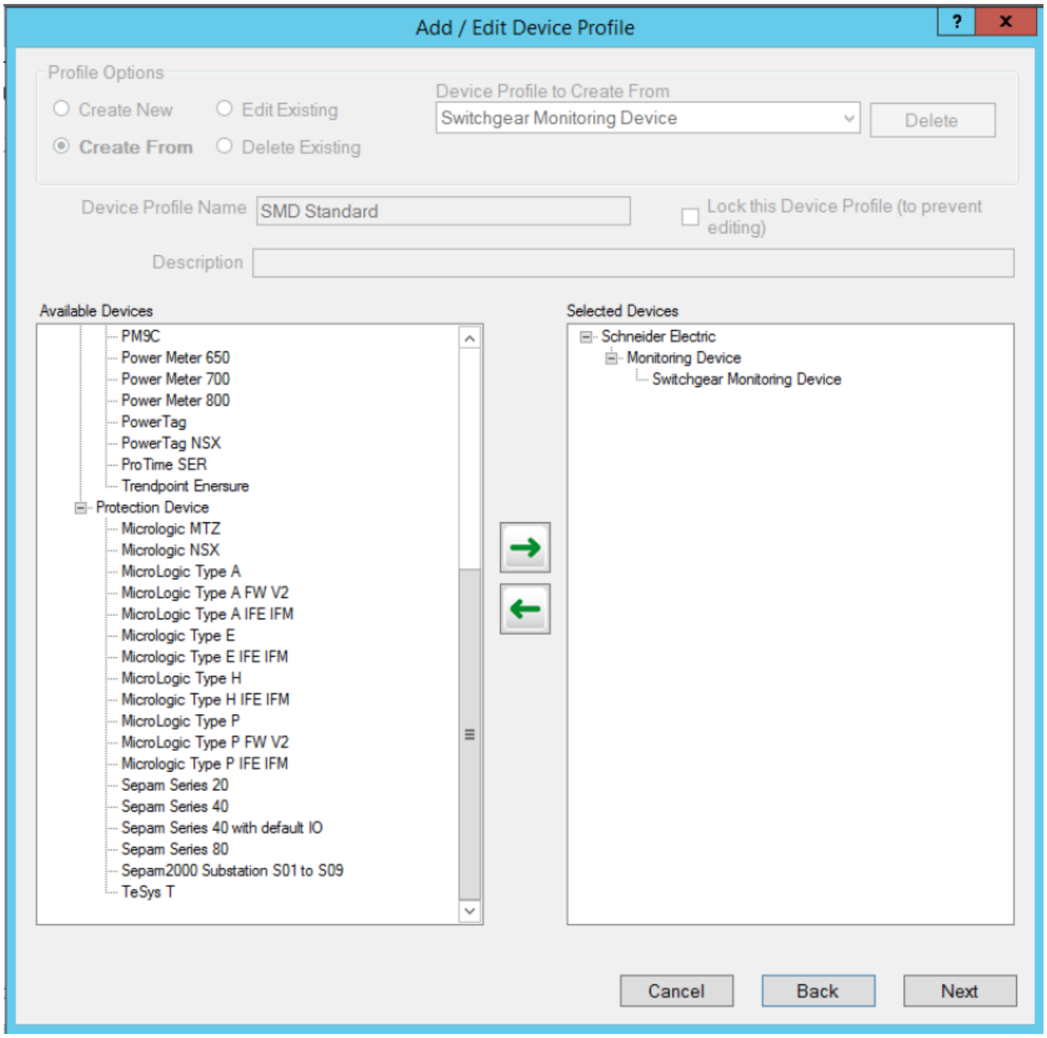
> **NOTE**: By default, a Switchgear Monitoring Device has Cubicle 1, Transformer 1, and Location 1 tags in the profile. If you only need those tags, skip to step 10.

1. Open the Profile Editor and select the **Create Device Profiles** tab.
2. In the Device Profile drop-down, select **Switchgear Monitoring Device** and click **OK**.
3. Click **Add/Edit**.
4. In the Add/Edit Device Profile window, select **Create From** and click **Next**.

   > **NOTE**: Make sure **Switchgear Monitoring Device** is still selected in the **Device Profile to Create From** drop-down.

5. Enter a new name for the device profile, for example **SMD Standard**, and click **Next**.

6. Confirm **Switchgear Monitoring Device** appears in the Selected Devices section, and then click **Next**.



Add / Edit Device Profile

Profile Options

○ Create New ○ Edit Existing

● Create From ○ Delete Existing

Device Profile to Create From: Switchgear Monitoring Device

Delete

Device Profile Name: SMD Standard

□ Lock this Device Profile (to prevent editing)

Description:

Available Devices

- PM9C
- Power Meter 650
- Power Meter 700
- Power Meter 800
- PowerTag
- PowerTag NSX
- ProTime SER
- Trendpoint Enersure
- Protection Device
  - Micrologic MTZ
  - Micrologic NSX
  - MicroLogic Type A
  - MicroLogic Type A FW V2
  - MicroLogic Type A IFE IFM
  - Micrologic Type E
  - Micrologic Type E IFE IFM
  - MicroLogic Type H
  - Micrologic Type H IFE IFM
  - MicroLogic Type P
  - MicroLogic Type P FW V2
  - Micrologic Type P IFE IFM
  - Sepam Series 20
  - Sepam Series 40
  - Sepam Series 40 with default IO
  - Sepam Series 80
  - Sepam2000 Substation S01 to S09
  - TeSys T

Selected Devices

- Schneider Electric
  - Monitoring Device
    - Switchgear Monitoring Device

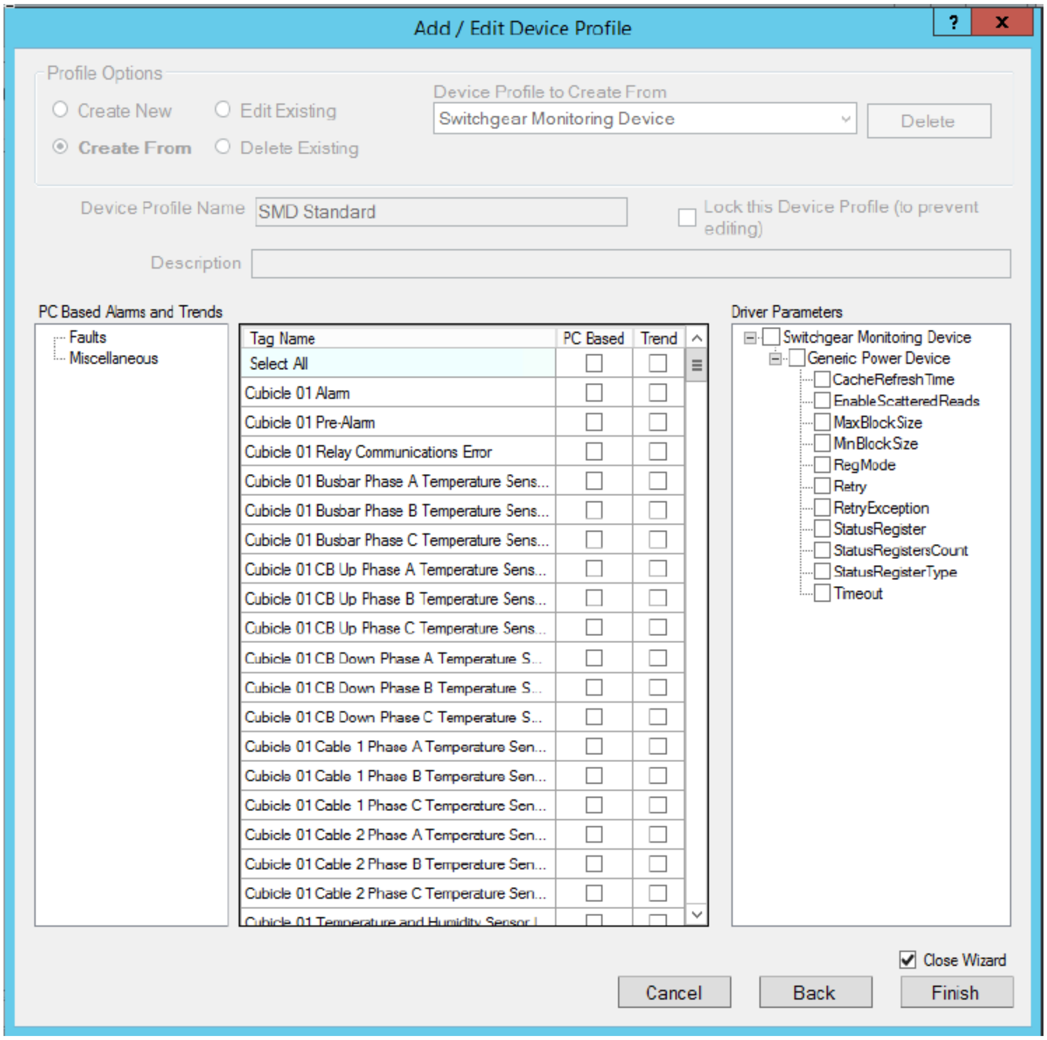
7. In the Device Types Tags section, select the Cubicle, Transformer, and Location tags you need for the SMD, and click the → to add them to the device profile.

NOTE: By default, a Switchgear Monitoring Device has Cubicle 1, Transformer 1, and Location 1 tags in the profile. If you only need those tags, click Cancel and skip to step 10.

8. Select all the tags you need and click **Next**.

9. Select **Close Wizard** and click **Finish**.

NOTE: Select any SMD trending tags you need in this window. By default, no tags are selected for PC Based or Trend.



Add / Edit Device Profile

Profile Options

○ Create New ○ Edit Existing

● Create From ○ Delete Existing

Device Profile to Create From: Switchgear Monitoring Device

Delete

Device Profile Name: SMD Standard

□ Lock this Device Profile (to prevent editing)

Description:

PC Based Alarms and Trends

- Faults
- Miscellaneous

Tag Name	PC Based	Trend
Select All	□	□
Cubicle 01 Alarm	□	□
Cubicle 01 Pre-Alarm	□	□
Cubicle 01 Relay Communications Error	□	□
Cubicle 01 Busbar Phase A Temperature Sens...	□	□
Cubicle 01 Busbar Phase B Temperature Sens...	□	□
Cubicle 01 Busbar Phase C Temperature Sens...	□	□
Cubicle 01 CB Up Phase A Temperature Sens...	□	□
Cubicle 01 CB Up Phase B Temperature Sens...	□	□
Cubicle 01 CB Up Phase C Temperature Sens...	□	□
Cubicle 01 CB Down Phase A Temperature S...	□	□
Cubicle 01 CB Down Phase B Temperature S...	□	□
Cubicle 01 CB Down Phase C Temperature S...	□	□
Cubicle 01 Cable 1 Phase A Temperature Sen...	□	□
Cubicle 01 Cable 1 Phase B Temperature Sen...	□	□
Cubicle 01 Cable 1 Phase C Temperature Sen...	□	□
Cubicle 01 Cable 2 Phase A Temperature Sen...	□	□
Cubicle 01 Cable 2 Phase B Temperature Sen...	□	□
Cubicle 01 Cable 2 Phase C Temperature Sen...	□	□
Cubicle 01 Temperature and Humidity Sensor I...	□	□

Driver Parameters

- □ Switchgear Monitoring Device
  - □ Generic Power Device
    - □ CacheRefreshTime
    - □ EnableScatteredReads
    - □ MaxBlockSize
    - □ MinBlockSize
    - □ RegMode
    - □ Retry
    - □ RetryException
    - □ StatusRegister
    - □ StatusRegistersCount
    - □ StatusRegisterType
    - □ Timeout

☑ Close Wizard

10. Select the Set Up Projects tab and click **Add/Edit**.

11. Select **Create New**, then click 🔍, select your project, and click **OK**.

12. In the Device Profiles section, select the device profile you created and click ➔ to move it to the Selected Device Profiles section.

> **NOTE**: If you skipped directly to step 10, select Switchgear Monitoring Device.

13. Click **Save & Exit**.

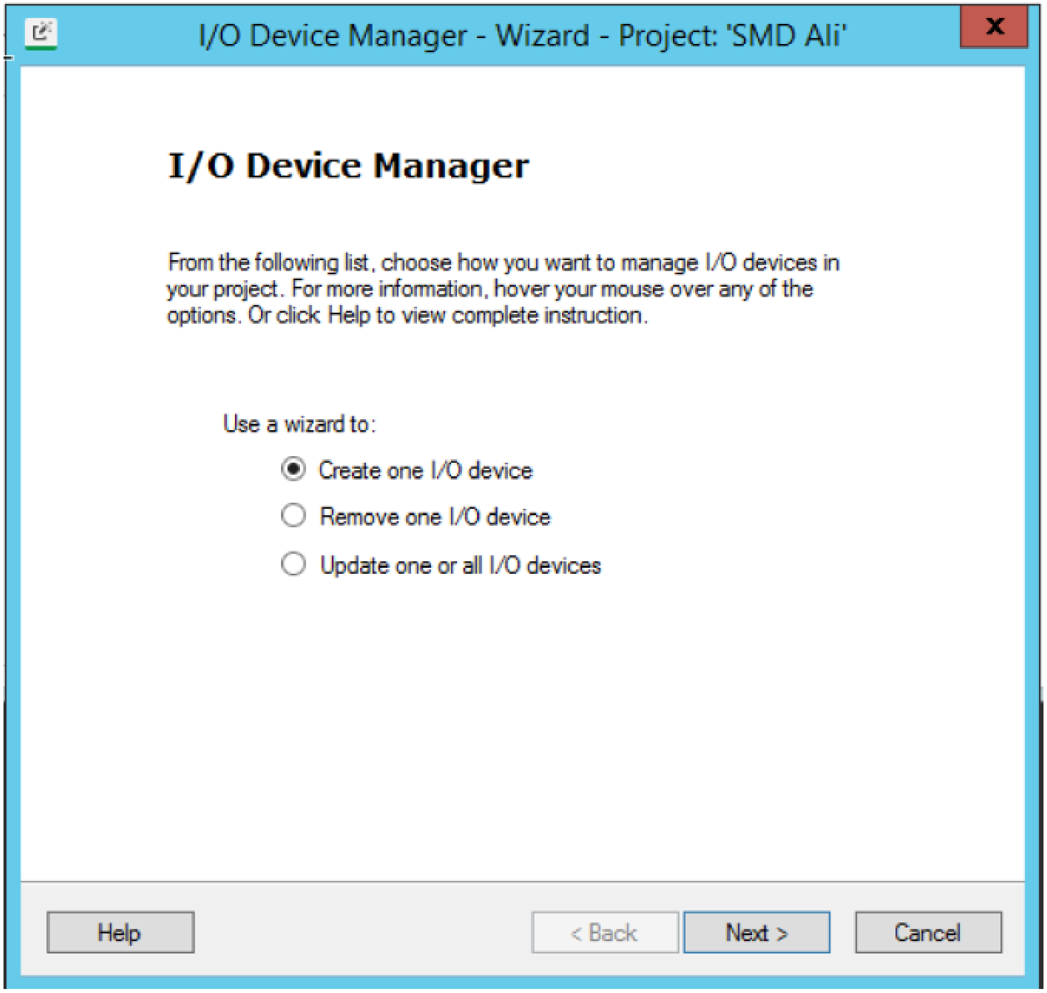
14. Click **Export Project** to export the new profile to the Power SCADA project.

Adding a thermal monitoring device to a Power SCADA project

To add a thermal monitoring Device to a Power SCADA project using the profile you created in Configuring a thermal monitoring device profile:

1. Open the I/O Device Manager.
2. In the I/O Device Manager, confirm the correct project is selected in the Project Name drop down.
3. Click **Manage a Single Device** to open the I/O Device Manager Wizard.

4. Select **Create one I/O Device** and click **Next**.



I/O Device Manager - Wizard - Project: 'SMD Ali'

I/O Device Manager

From the following list, choose how you want to manage I/O devices in your project. For more information, hover your mouse over any of the options. Or click Help to view complete instruction.

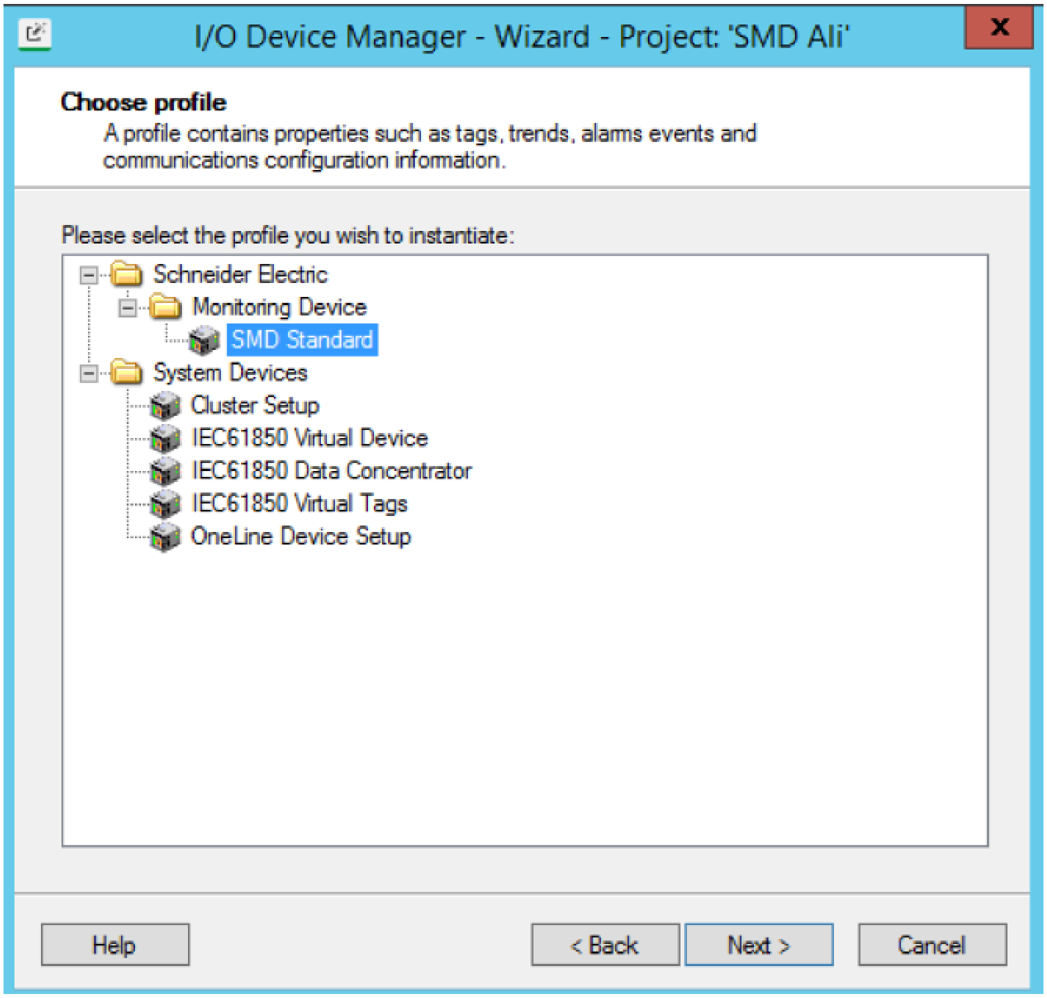
Use a wizard to:

● Create one I/O device

○ Remove one I/O device

○ Update one or all I/O devices

5. In Choose Profile, select the thermal monitoring device profile you created and click **Next**.



I/O Device Manager - Wizard - Project: 'SMD Ali'

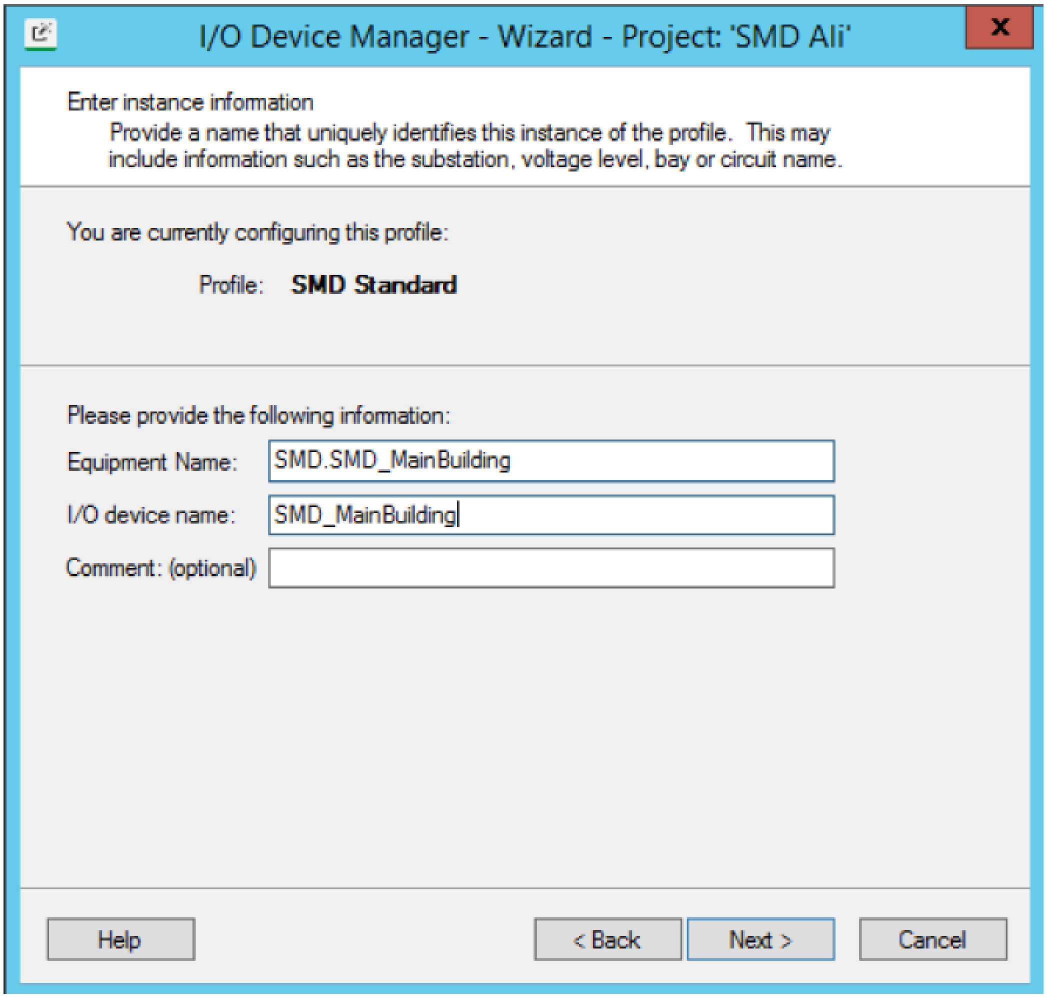
Choose profile

A profile contains properties such as tags, trends, alarms events and communications configuration information.

Please select the profile you wish to instantiate:

- Schneider Electric
  - Monitoring Device
    - SMD Standard
- System Devices
  - Cluster Setup
  - IEC61850 Virtual Device
  - IEC61850 Data Concentrator
  - IEC61850 Virtual Tags
  - OneLine Device Setup

6. Enter an Equipment name and I/O device name.



I/O Device Manager - Wizard - Project: 'SMD Ali'

Enter instance information

Provide a name that uniquely identifies this instance of the profile. This may include information such as the substation, voltage level, bay or circuit name.

You are currently configuring this profile:

Profile: **SMD Standard**

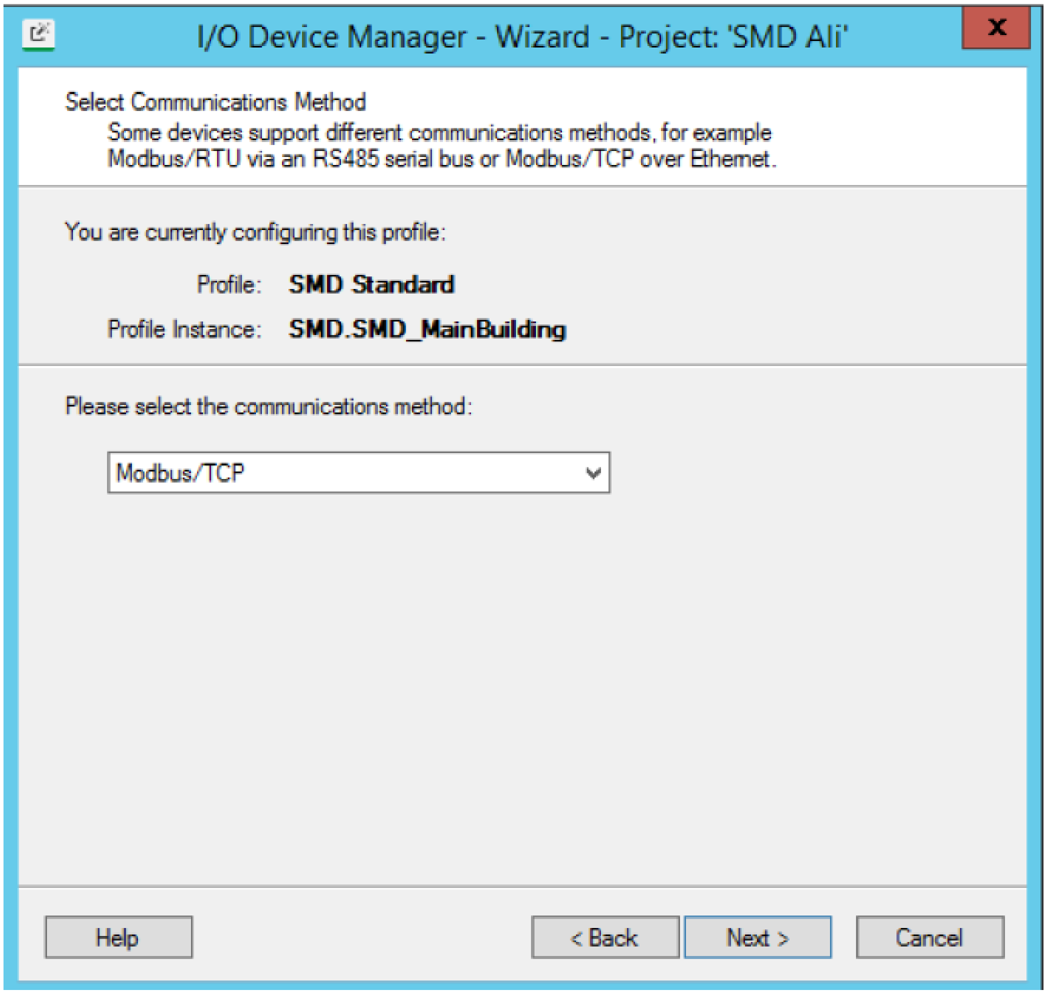
Please provide the following information:

Equipment Name: SMD.SMD_MainBuilding

I/O device name: SMD_MainBuilding

Comment: (optional)

7. Select your communications method, Modbus/TCP or Modbus/RTU Via Gateway.



Select Communications Method

Some devices support different communications methods, for example Modbus/RTU via an RS485 serial bus or Modbus/TCP over Ethernet.

You are currently configuring this profile:

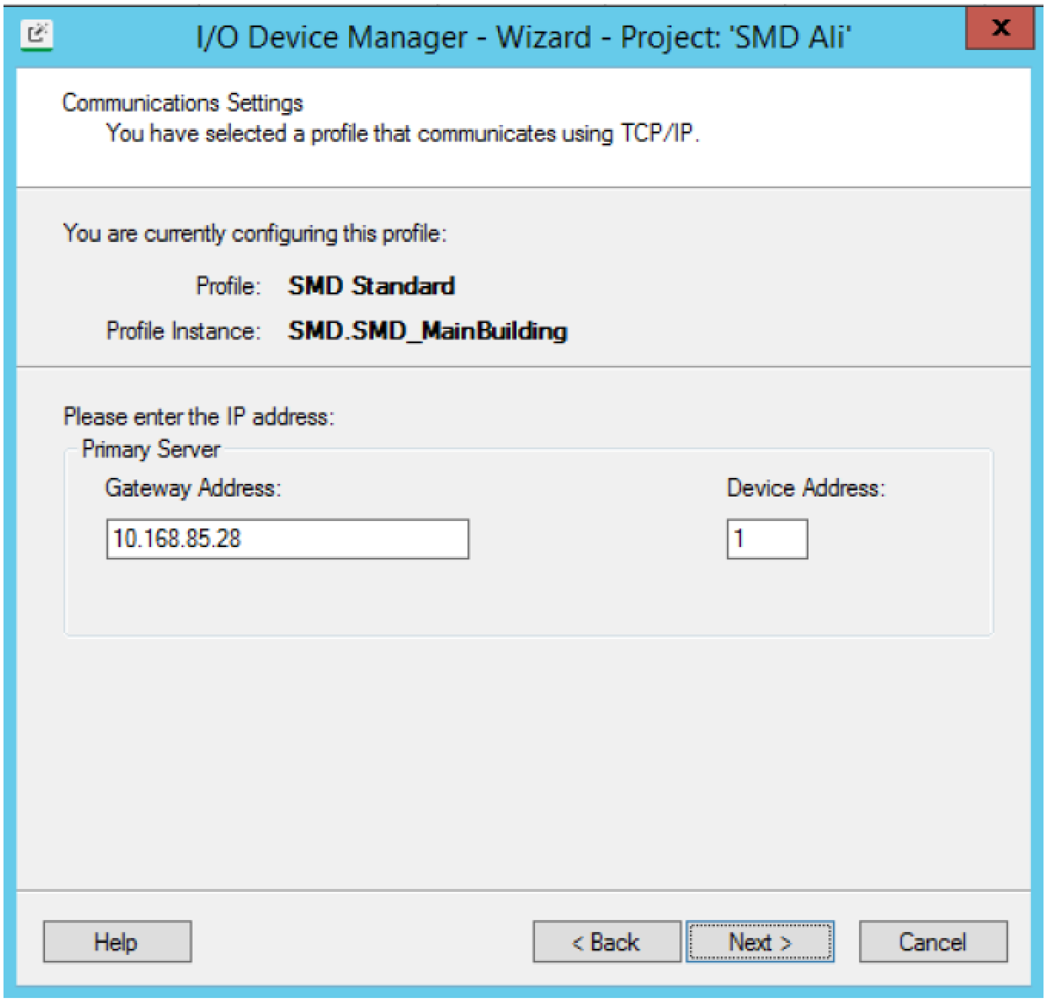
Profile: **SMD Standard**

Profile Instance: **SMD.SMD_MainBuilding**

Please select the communications method:

Modbus/TCP

8. Enter the SMD communication settings.



I/O Device Manager - Wizard - Project: 'SMD Ali'

Communications Settings

You have selected a profile that communicates using TCP/IP.

You are currently configuring this profile:

Profile: **SMD Standard**

Profile Instance: **SMD.SMD_MainBuilding**

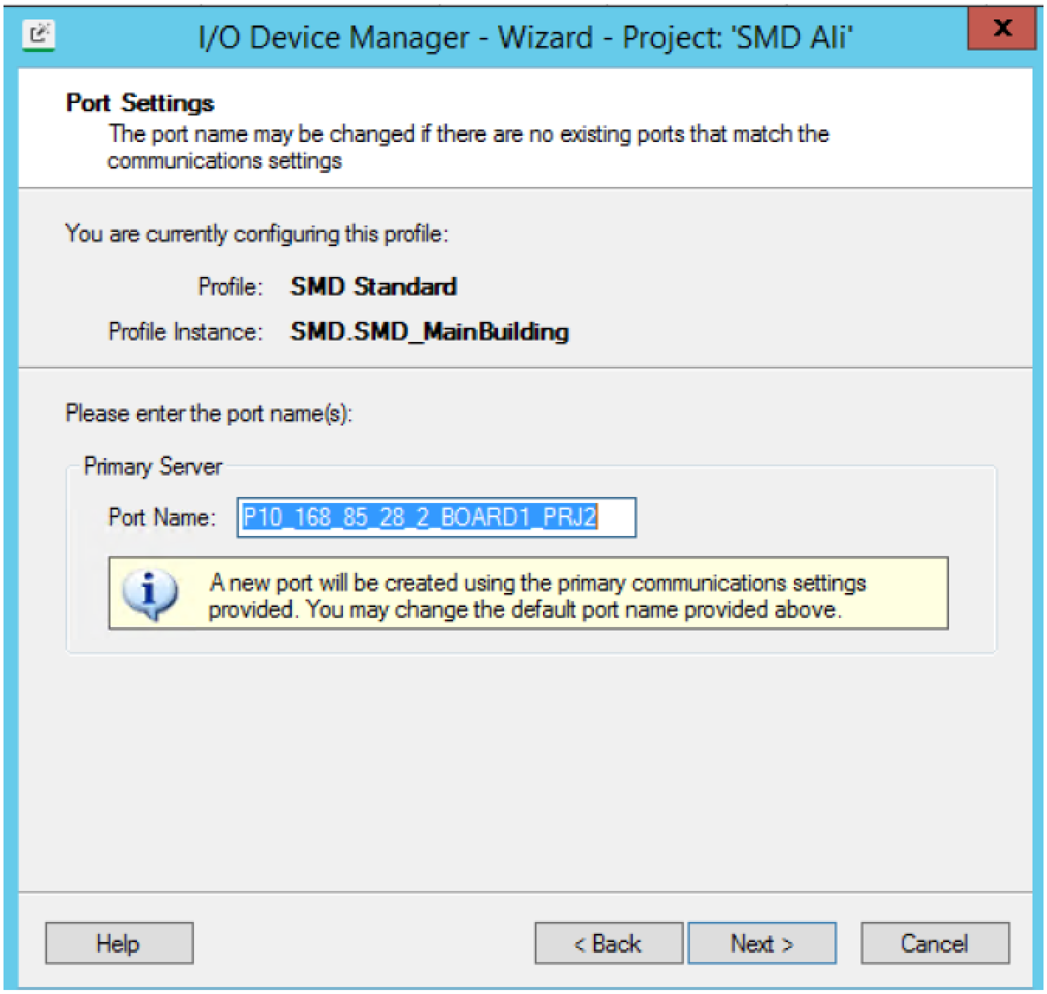
Please enter the IP address:

Primary Server

Gateway Address: 10.168.85.28

Device Address: 1

9. Enter a custom port name or keep the generated default, and click **Next**.



Port Settings

The port name may be changed if there are no existing ports that match the communications settings

You are currently configuring this profile:

Profile: **SMD Standard**

Profile Instance: **SMD.SMD_MainBuilding**

Please enter the port name(s):

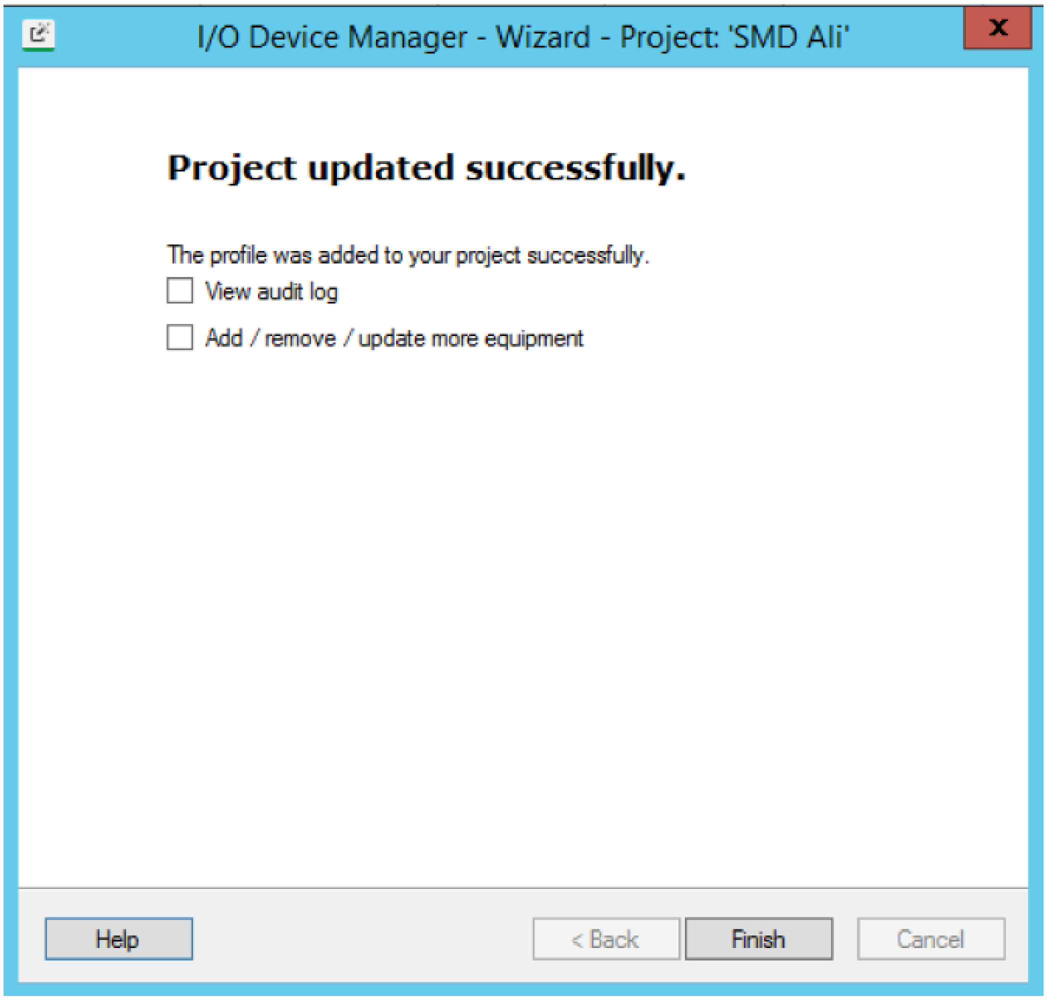
Primary Server

Port Name: P10_168_85_28_2_BOARD1_PRJ2

A new port will be created using the primary communications settings provided. You may change the default port name provided above.

10. Click **Finish**.

NOTE: If you are adding multiple SMD devices, select Add / Remove / Update more equipment and repeat steps 1-10 for each.



I/O Device Manager - Wizard - Project: 'SMD Ali'

Project updated successfully.

The profile was added to your project successfully.

□ View audit log

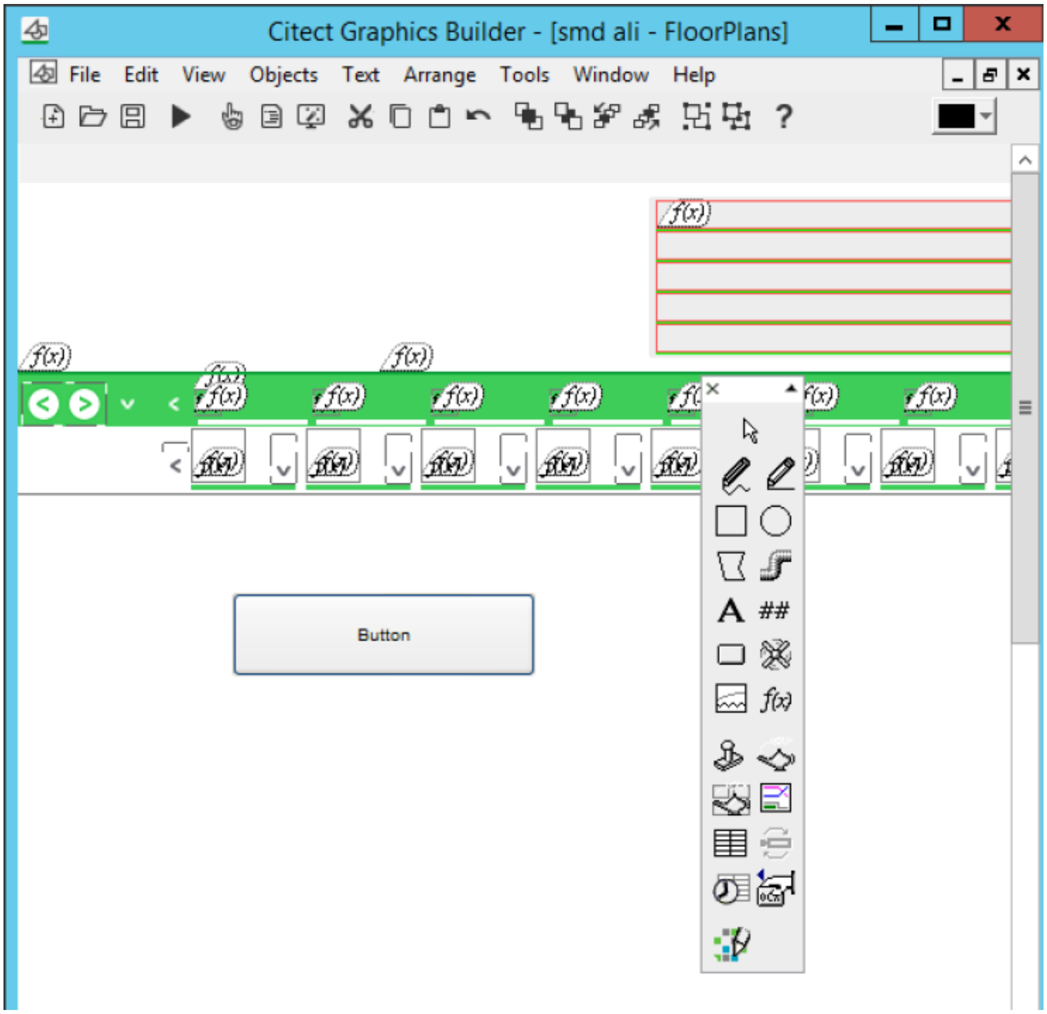
□ Add / remove / update more equipment

Configuring a thermal monitoring device popup in a graphic

To create a button to open a thermal monitoring device popup and view the cubicle, transformer, and location information:

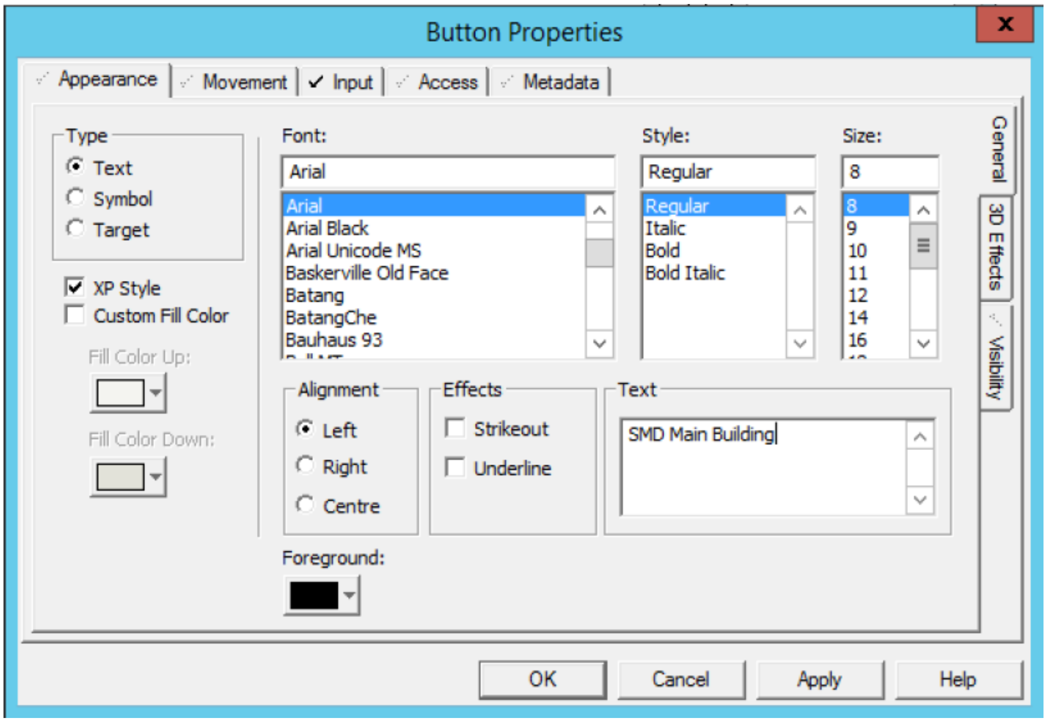
1. Open the Citect Graphics Builder.
2. Open the graphic page on which you want to create the button.

3. Using the toolbox, create a button on the page.


Citect Graphics Builder - [smd ali - FloorPlans]
File Edit View Objects Text Arrange Tools Window Help
Button

4. Select the button, right-click, and then click **Properties**.

5. In the Button Properties Appearance tab, enter unique text for the button.



Button Properties

Type: ● Text ○ Symbol ○ Target

☑ XP Style

□ Custom Fill Color

Font: Arial

Style: Regular

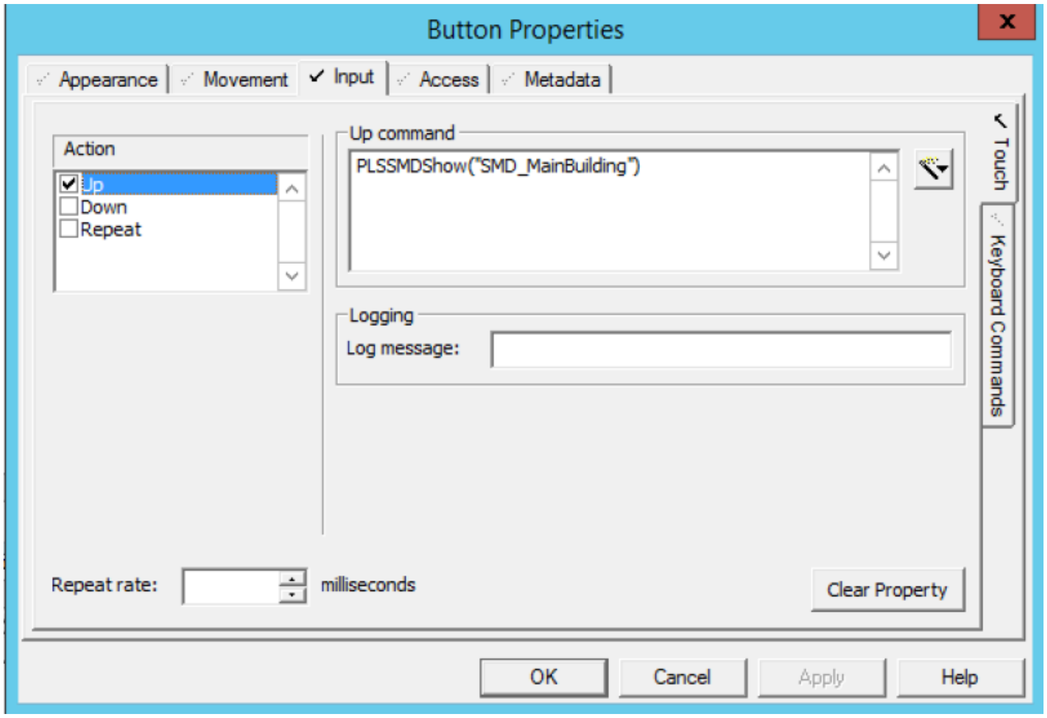
Size: 8

Alignment: ● Left ○ Right ○ Centre

Effects: □ Strikeout □ Underline

Text: SMD Main Building

6. In the Input Tab Up Command section, enter PLSMDShow(“Your I/O device Name”).



Button Properties

Action: ☑ Up □ Down □ Repeat

Up command: PLSSMDShow("SMD_MainBuilding")

Logging

Log message:

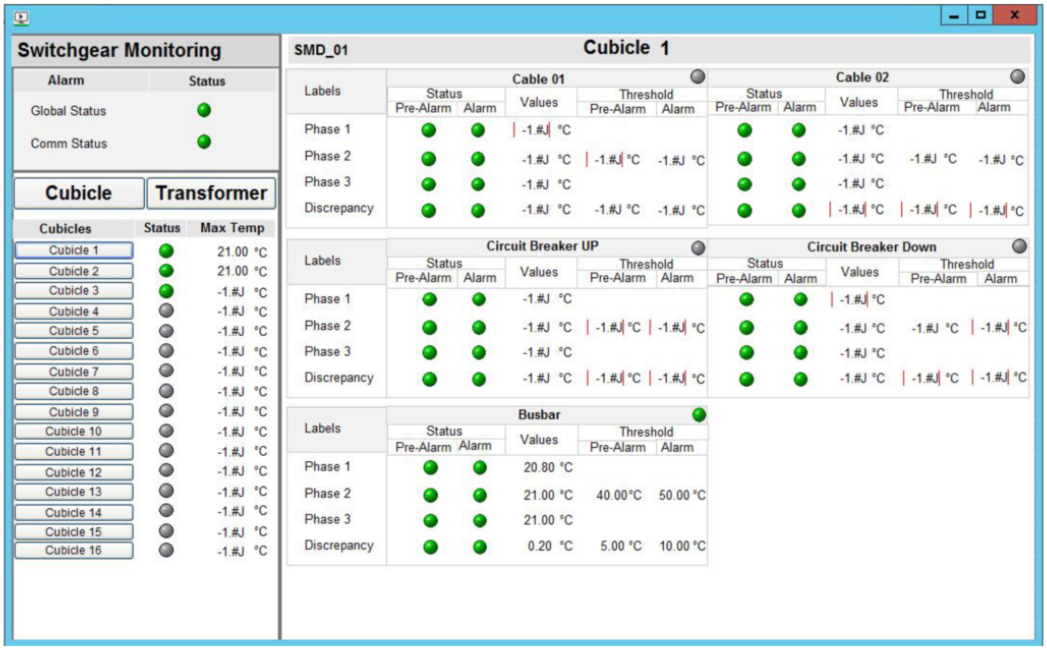
Repeat rate: milliseconds

NOTE: NOTE: The quotation marks are necessary. The device name must match what was entered as the **I/O device name** in Adding a thermal monitoring device to a Power SCADA project.

7. Click **OK**.

8. Compile and run the project.

9. Confirm that the button you created appears in the correct graphic page, and click the button to see the popup.



Switchgear Monitoring

Alarm	Status
Global Status	🟢
Comm Status	🟢

Cubicle | Transformer

Cubicles	Status	Max Temp
Cubicle 1	🟢	21.00 °C
Cubicle 2	🟢	21.00 °C
Cubicle 3	🟢	-1.#J °C
Cubicle 4	⚪	-1.#J °C
Cubicle 5	⚪	-1.#J °C
Cubicle 6	⚪	-1.#J °C
Cubicle 7	⚪	-1.#J °C
Cubicle 8	⚪	-1.#J °C
Cubicle 9	⚪	-1.#J °C
Cubicle 10	⚪	-1.#J °C
Cubicle 11	⚪	-1.#J °C
Cubicle 12	⚪	-1.#J °C
Cubicle 13	⚪	-1.#J °C
Cubicle 14	⚪	-1.#J °C
Cubicle 15	⚪	-1.#J °C
Cubicle 16	⚪	-1.#J °C

SMD_01

Cubicle 1

Labels	Cable 01 ⚪					Cable 02 ⚪				
	Status		Values	Threshold		Status		Values	Threshold	
	Pre-Alarm	Alarm		Pre-Alarm	Alarm	Pre-Alarm	Alarm		Pre-Alarm	Alarm
Phase 1	🟢	🟢	-1.#J °C			🟢	🟢	-1.#J °C		
Phase 2	🟢	🟢	-1.#J °C	-1.#J °C	-1.#J °C	🟢	🟢	-1.#J °C	-1.#J °C	-1.#J °C
Phase 3	🟢	🟢	-1.#J °C			🟢	🟢	-1.#J °C		
Discrepancy	🟢	🟢	-1.#J °C	-1.#J °C	-1.#J °C	🟢	🟢	-1.#J °C	-1.#J °C	-1.#J °C

Labels	Circuit Breaker UP ⚪					Circuit Breaker Down ⚪				
	Status		Values	Threshold		Status		Values	Threshold	
	Pre-Alarm	Alarm		Pre-Alarm	Alarm	Pre-Alarm	Alarm		Pre-Alarm	Alarm
Phase 1	🟢	🟢	-1.#J °C			🟢	🟢	-1.#J °C		
Phase 2	🟢	🟢	-1.#J °C	-1.#J °C	-1.#J °C	🟢	🟢	-1.#J °C	-1.#J °C	-1.#J °C
Phase 3	🟢	🟢	-1.#J °C			🟢	🟢	-1.#J °C		
Discrepancy	🟢	🟢	-1.#J °C	-1.#J °C	-1.#J °C	🟢	🟢	-1.#J °C	-1.#J °C	-1.#J °C

Labels	Busbar 🟢				
	Status		Values	Threshold	
	Pre-Alarm	Alarm		Pre-Alarm	Alarm
Phase 1	🟢	🟢	20.80 °C		
Phase 2	🟢	🟢	21.00 °C	40.00 °C	50.00 °C
Phase 3	🟢	🟢	21.00 °C		
Discrepancy	🟢	🟢	0.20 °C	5.00 °C	10.00 °C

Troubleshoot and FAQs

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 808.
- Use the One-Line Configuration Utility to repair problems with equipment on graphics pages: see "One-line errors and warnings" on page 814.
- General troubleshooting questions: see "Frequently Asked Questions (FAQs)" on page 818.

To learn the status of all of your project tags, see "View the Tag Viewer" on page 714.

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.

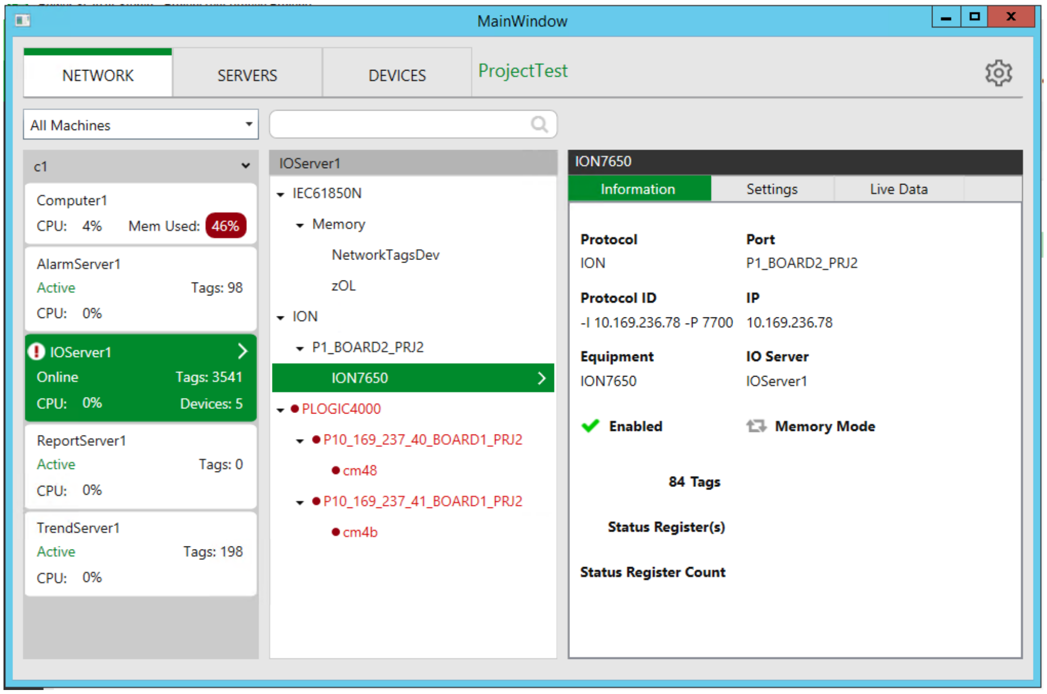
For more information, see "Offline and Online Mode" on page 809.

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 809 only) in the far-right pane.
- Click ⚙ to set the refresh rate.



Computer1
CPU: 4% Mem Used: 46%

AlarmServer1
Active Tags: 98
CPU: 0%

IOServer1
Online Tags: 3541
CPU: 0% Devices: 5

ReportServer1
Active Tags: 0
CPU: 0%

TrendServer1
Active Tags: 198
CPU: 0%

- IEC61850N
  - Memory
    - NetworkTagsDev
    - zOL
- ION
  - P1_BOARD2_PRJ2
    - ION7650
- ● PLOGIC4000
  - ● P10_169_237_40_BOARD1_PRJ2
    - ● cm48
  - ● P10_169_237_41_BOARD1_PRJ2
    - ● cm4b

Protocol ION

Port P1_BOARD2_PRJ2

Protocol ID -I 10.169.236.78 -P 7700

IP 10.169.236.78

Equipment ION7650

IO Server IOServer1

✓ **Enabled**

Memory Mode

84 Tags

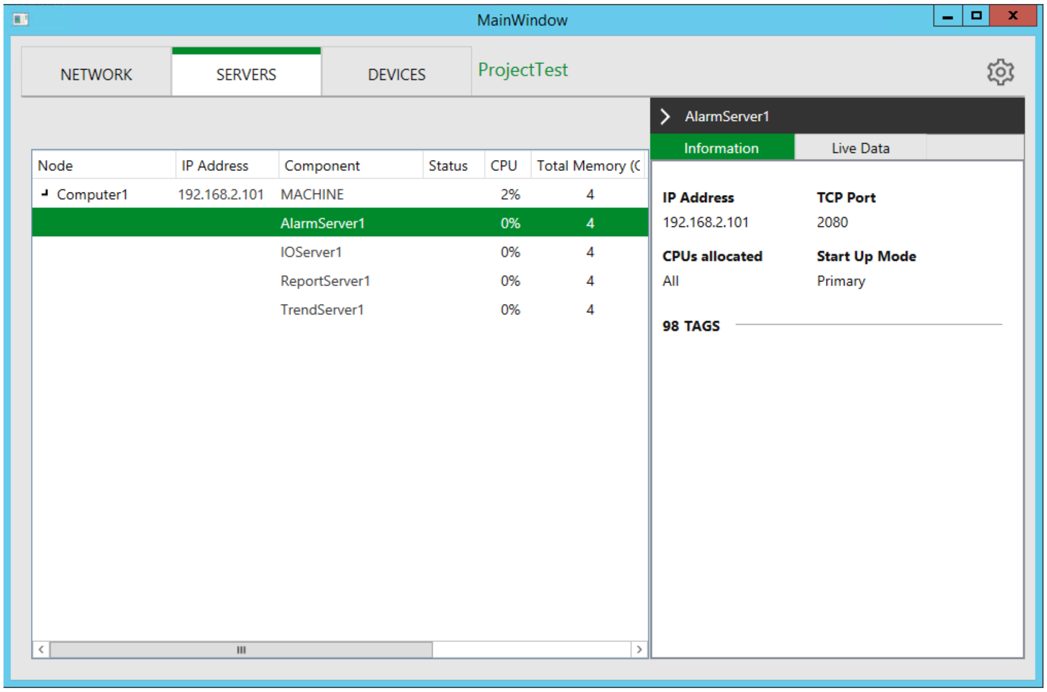
Status Register(s)

Status Register Count

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 809 only) in the right pane.
- Click < to open and > to close the right pane.
- Click ⚙ to set the refresh rate.



Node	IP Address	Component	Status	CPU	Total Memory (C
Computer1	192.168.2.101	MACHINE		2%	4
		AlarmServer1		0%	4
		IOServer1		0%	4
		ReportServer1		0%	4
		TrendServer1		0%	4

IP Address	TCP Port
192.168.2.101	2080

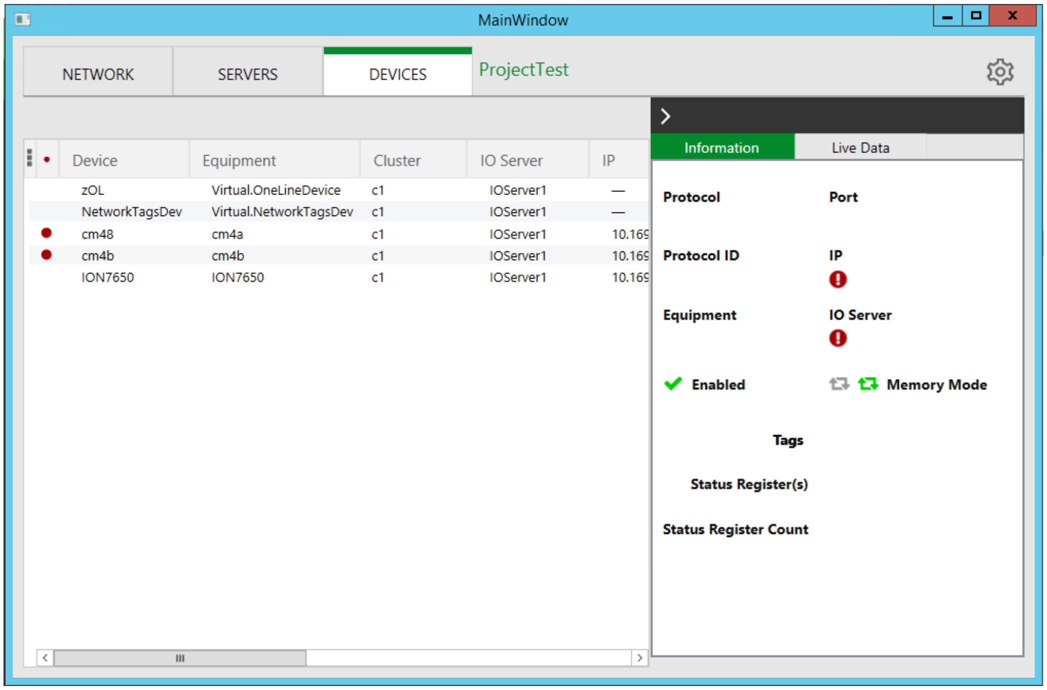
CPUs allocated	Start Up Mode
All	Primary

98 TAGS

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 809 only) in the right pane.
- Click < to open and > to close the right pane.
- Click ⚙ to set the refresh rate.



●	Device	Equipment	Cluster	IO Server	IP
	zOL	Virtual.OneLineDevice	c1	IOServer1	—
	NetworkTagsDev	Virtual.NetworkTagsDev	c1	IOServer1	—
●	cm48	cm4a	c1	IOServer1	10.169
●	cm4b	cm4b	c1	IOServer1	10.169
	ION7650	ION7650	c1	IOServer1	10.169

Protocol Port

Protocol ID IP ❗

Equipment IO Server ❗

✓ Enabled Memory Mode

Tags

Status Register(s)

Status Register Count

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.

Troubleshooting web diagrams

This section provides hints and instructions for correcting issues with web diagrams.

Fix WebHMI when showing incorrect or missing data

Address web diagrams with bad data quality, such as a 0 returned for timestamp and a 0 for quality, or an exclamation icon indicating bad data quality in the product default components:

1. Check your bind names using browser developer tools to inspect network activity and read-data-point request responses. The response data contains the name of each datapoint requested from the server. Confirm that these datapoints exist in your server SCADA project as valid tag definitions and/or equipment item definitions. Browser debugging tools:
   - Chrome
   - Safari
   - Microsoft Edge

2. Ensure your variable binds for template TGML/graphics adhere to the following format in order to allow them to be replaced with parent components during run time:

   %Name%.BindName

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. By default, only exceptions are logged.

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.

Error message	Description
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: 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

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 alarms 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.
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.	There is no resolution.

Issue	Cause	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).

Web Applications

WEB_REQUEST_FAILED message

There are multiple reasons that you may see the message "Single Use Token failed with error code. #WEB_REQUEST_FAILED" when you access Advanced Reporting and Dashboards in PSO.

To troubleshoot:

1. Install the Advanced Reports security certificate on the PSO server.
2. Set the Hostname to the Advanced Reports server name.
3. Disable IPV6.

Install the Advanced Reports security certificate on the PSO server

To Install the Advanced Reports security certificate on the PSO server, see **Installing and binding security certificates** in the PME System Guide.

Set the hostname

To set the hostname:

1. Open the Computer Setup Editor: In Power SCADA Studio, click **Projects > Setup Wizard** drop down, and then click Setup Editor.

2. In **Applications > Hostname**, enter the Advanced Reports server name.

Disable IPv6

To disable IPv6 on the network adapter:

1. Open **Windows Settings > Network & Internet > Change adapter options**.
2. Select the network adapter. For example, Ethernet.
3. Right-click the network adapter and click **Properties**.
4. Remove the check mark next to **Internet Protocol Version 6 (TCP/IPv6)**.

Permission Denied

If you see a Permission Denied error in a web application, verify the following:

- `%2f` is the escape character for forward slash ('/'). Any other value in the `Target` field will result in Permission Denied.
- `ApplicationMenuConfig.json`
  - `Id`: The Id cannot contain spaces.
  - `Target`: The Reports application must use '`reporter`' as its name. Any other value will result in Permission Denied.

    For example, the following will result in a Permission Denied error: `/psodataservice/pme/auth?returnUrl=%2freports`
- `HmiConfiguration.json`:
  - `Value`: Cannot include a trailing slash or anything after the server name.

    For example, the following will result in a Permission Denied error: https://pme.se.com/web

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 subcategories 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 282 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 produce 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 587. For additional information, see **Using Security** in the `citectSCADA.chm` help file (`...\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020 R2\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:

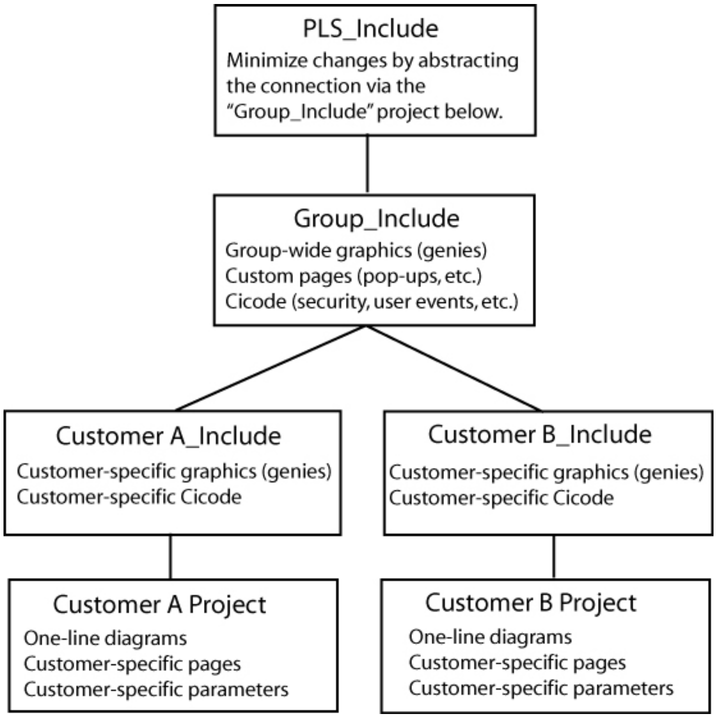
- Keep original ‘Master’ copies of the PLS_Example and the PLS_Include projects for reference.
- The Services Group may develop a group-wide “include” project that will act as a conduit between the PLS_Include project and all customer projects (for example: "Group_Include"). This will make the upgrading 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.


PLS_Include
Minimize changes by abstracting the connection via the “Group_Include” project below.
Group_Include
Group-wide graphics (genies)
Custom pages (pop-ups, etc.)
Cicode (security, user events, etc.)
Customer A_Include
Customer-specific graphics (genies)
Customer-specific Cicode
Customer B_Include
Customer-specific graphics (genies)
Customer-specific Cicode
Customer A Project
One-line diagrams
Customer-specific pages
Customer-specific parameters
Customer B Project
One-line diagrams
Customer-specific pages
Customer-specific parameters

- 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 an 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 an 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 2020 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 2020 Help Manual (Viewing Device Profiles: "Alarm On Text" and "Alarm Off Text").

What can I modify during runtime?

See the Power SCADA Operation 2020 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 1229 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 2020 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 2020 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 2020 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 325
- "Perform IEC 61850 advanced control" on page 712

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

Decommission

Decommissioning removes Power SCADA files to prevent potential disclosure of sensitive, confidential and proprietary data and software from your system. You risk disclosing your power system data, system configuration, user information, and other sensitive information if you don’t decommission. We strongly recommend you decommission your system at the end of its life.

⚠ **WARNING**

UNINTENDED EQUIPMENT OPERATION

Before decommissioning, verify that the system is not performing critical control actions that may affect human or equipment safety.

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

⚠ **WARNING**

INACCURATE DATA RESULTS

Before decommissioning, verify that the system data results are not used for critical decision making that may affect human or equipment safety.

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

To decommission Power SCADA you have two choices, **Destroy** or **Overwrite:**

Destroy: Choose this if you don’t need to use your hard drives for any other software. If you need to use your hard drives for other software, choose **Overwrite**.

Overwrite: Choose this if you still need to use your hard drives for other software. This method uses a commercial tool to put random data in place of Power SCADA files on your hard drives.

See "Decommission references" on page 1298 for detailed instructions.

Reference

The Reference chapter contains detailed reference information related to planning, installing and upgrading, configuring, administering, decommissioning, and using Power SCADA Operation. This information is referenced in the other chapters of this guide.

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

Topic	Description
"Upgrade references" on page 833	Detailed information on the Cicode functions, Citect INI settings, and upgrade information specific to previous versions of Power SCADA Operation.
"Configure references" on page 884	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.
"Operate references" on page 1262	Detailed information on operating Power SCADA OperationWeb Applications.
"Decommission references" on page 1298	Detailed information on removing potentially sensitive, confidential and proprietary data and software your Power SCADA Operation system at the end its' life.
"Glossary" on page 1305	A list of commonly used terms and acronyms, and their definitions.

Upgrade references

The topics in this section contain detailed reference information that pertains to upgrading to Power SCADA Operation 2020

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 834
- "Upgrade Information for versions 8.1 and 8.0 SR1" on page 835
- "Upgrade Information for versions 7.40 and 8.0" on page 839
- "Upgrade Information for Version 7.30" on page 840
- "Upgrade Information for Version 7.20" on page 843
- "Citect SCADA Migration Information" on page 881

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:

- "Cicode Functions in version 8.2" on page 845
- "Cicode Functions in versions 8.1 and 8.0 SR1" on page 846
- "Cicode Functions in 7.40 and 8.0" on page 847
- "Cicode Functions in 7.30" on page 849
- "Cicode Functions in 7.20" on page 857

Citect.ini Parameters

Refer to the following topics for detailed information on the Citect.ini parameters that were added for each release:

- "Citect.ini parameters in 8.2" on page 864
- "Citect.ini parameters in 8.1 and 8.0 SR1" on page 864
- "Citect.ini parameters in 7.40 SP1" on page 867
- "Citect.ini parameters in 7.40" on page 868
- "Citect.ini parameters in 7.30" on page 869
- "Citect.ini parameters in 7.20" on page 874

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 lot 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 several 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 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 (like version 7.20), the cluster replication parameter has been added: [General]ClusterReplication.

Computer Setup Wizard

You can now assign a CPU to a component using the CPU Configuration page.

Default Trend Storage

- Storage Method

  Trend records must 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 behavior 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 SetLanguage 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 2020 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 2020. 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 1261

  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.

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 Power 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.

Citect SCADA Migration Information

This section contains the required and optional steps to migrate from pre-7.x versions of Citect SCADA to Power SCADA Operation. For an overview, see "Migrating from Citect SCADA" on page 172.

NOTE: It is possible to restore and run the Citect project and includes on Power SCADA Operation and modify them to fully convert them to Power SCADA Operation projects. However, we recommend using the following migration steps to take advantage of the built-in power management features of Power SCADA Operation.

The required migration steps are:

1. "Create a New Project" on page 881
2. "Import Citect Customizations" on page 882
3. "Create Device Type Tags and Devices" on page 883
4. "Export Alarm History" on page 883
5. "Enable Waveforms" on page 883

The optional migrations steps are:

1. "Re-create One-line Animation" on page 884
2. "Add Notifications" on page 884
3. "Set Up Diagnostics" on page 884
4. "Add Basic Reports and LiveView" on page 884
5. "Set Up Two-Factor Authentication" on page 884

Required Steps

The following steps are required to migrate from pre-7.x versions of Citect SCADA to Power SCADA Operation.

Create a New Project

To create and configure a new Power SCADA Operation project, see "Power SCADA Projects" on page 206.

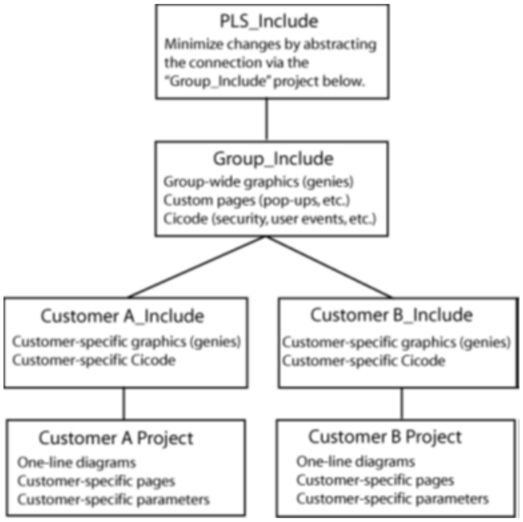
The new project will have a default cluster, computer, network, and servers (Alarm, Trend, Report, and I/O). The project will also include two defined I/O devices, zOL (used for Advanced One-line) and NetworkTagsDev (used for communication purposes). Furthermore, the PLS_Include project is included with the main project. The PLS_Include contains the standard Power SCADA Operation cicode, labels, fonts, alarm categories, parameters, graphics, and tools.

Import Citect Customizations

The new Power SCADA Operation project has all defaults for users, fonts, parameters, etc. All the custom users, fonts, parameters from the original Citect projects and includes must be added to the new project.

To import Citect customizations:

1. Create a group-wide “include” project that will act as a link between the PLS_Include project and all customer projects (for example: "Group_Include"). This will make upgrading the 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.
2. Make any changes to the PLS_Include project in the Group_Include project: remove portions of the code from the PLS_Include project, modify the code, and save it in the Group_Include project. Removing (or commenting out) the original code and placing the new code in the Group_Include project simplifies the upgrade process and preserves a layer of abstraction. The only changes to PLS_Include should be code removal. The following image shows the include structure:


PLS_Include
Minimize changes by abstracting the connection via the "Group_Include" project below.
Group_Include
Group-wide graphics (genies)
Custom pages (pop-ups, etc.)
Cicode (security, user events, etc.)
Customer A_Include
Customer-specific graphics (genies)
Customer-specific Cicode
Customer B_Include
Customer-specific graphics (genies)
Customer-specific Cicode
Customer A Project
One-line diagrams
Customer-specific pages
Customer-specific parameters
Customer B Project
One-line diagrams
Customer-specific pages
Customer-specific parameters

3. Review all custom Cicode functions to check that they match the new I/O device name and naming convention. All customizations must be saved in the Global_Include, main projects, or any other includes.
4. Re-save any custom pages, genies, etc. in the Global_Include project or the main project and re-link the graphic objects and functions (device names, tag names, etc. are now different).
5. Add any other project customizations (menus, new users, fonts, parameters, etc.) to the Group_Include project or the main project. The PLS_Include can be used as reference for adding the new custom customization. See structure below.

For more detailed information, see "How do I manage projects in the Power SCADA Studio of Power SCADA Operation?" on page 821

Create Device Type Tags and Devices

Use the Profile Editor to create device type tags (real-time tags, PC-based alarm tags, onboard alarm tags, trend tags, and reset tags), device types, and device profiles. The Profile Editor is a productivity tool for commissioning Modbus, ION, IEC-61850 devices types faster and more efficiently. Power SCADA Operation uses this information to create graphics pages and device types. For more information, see "The Profile Editor" on page 222.

NOTE: By default, the Profile Editor includes the standard Schneider Electric monitoring and protection device types. For third-party devices, a custom device type must be created using the Profile Editor.

NOTE: The Power SCADA Operation tag naming convention follows the IEC 61850 standard. Convert all the tags in the Citect project to the IEC 61850 naming convention. Before converting the tags, consider the structure of the tag. If the default genies, popups, etc. that come with Power SCADA Operation will be used, the tag naming must follow the same tag structure. This will prevent the need to rework any default genies, popups, etc. that already exist to fit your custom tag structure. For detailed information on the tag naming convention, see "Tag naming convention" on page 247.

Use the I/O Device Manager to create, remove, or update devices. The I/O Device Manager uses the device types created with the Profile Editor. For step by step information on using the I/O Device Manager, see "Manage I/O devices in a project" on page 289.

NOTE: When you add a Modbus device type to the project using the I/O Device Manager, it will automatically use the Power Modbus (PwrModbus) driver unless the device is added via the Express Wizard. All devices added to the system require an Equipment Name and a Device Name. For more information on PwrModbus see "Power Modbus (PwrModbus) Driver for Modbus Devices" on page 967.

Export Alarm History

The alarm and trend history from the Citect projects will be deleted when the project is migrated to Power SCADA Operation. The devices alarm tags and trend tags will be redefined in Power SCADA Operation and will no longer have any linking to the Citect history.

To keep a back up Citect alarm history, export all alarm history from the Citect project before fully shutting the project down. It is not possible to export the trend history.

Enable Waveforms

Power SCADA Operation supports the ability to view electrical waveforms for ION PQ meters and any device that supports COMTRADE waveforms.

To enable waveforms for PQ devices, see "Enable Waveforms" on page 256.

Optional Steps

The following steps are optional to migrate from pre-7.x versions of Citect SCADA to Power SCADA Operation.

Re-create One-line Animation

If the Citect graphics included a flat one-line configuration, we recommend that you re-create the one-line animation using the Advanced One-line Configuration Utility in Power SCADA Operation. See "Legacy animated one-line diagrams" on page 1232 for more information.

Add Notifications

Add Notifications to alert specific people in your facility about critical power incidents no matter where they are. For more information, see "Notifications" on page 338.

Set Up Diagnostics

Set up Diagnostics to provide visibility into your system, help you understand how the system is organized, monitor performance, and troubleshoot issues. For more information on setup and accessing diagnostics, see "Diagnostics Overview" on page 808.

Add Basic Reports and LiveView

For installation and information on adding Basic Reports, see "Basic Reports" on page 317.

For information on LiveView, see "Create Real-Time Data Views" on page 325.

Set Up Two-Factor Authentication

For information on setting up two-factor authentication, see "Two-Factor Authentication" on page 593.

Configure references

The topics in this section contain detailed reference information that pertains to configuring Power SCADA Operation.

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

Topic	Content
Citect INI Parameters	A detailed listing of the Citect INI parameters you can use in your projects.
Logic code definitions	Design considerations and sample architectures for the Power SCADA Operation components.
Default Genie Library	A detailed listing of the Power SCADA Operation PLS_* genie library and its naming conventions.
Deadbands and ignored devices and topics	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	Detailed information on how to set up and add engineering units and conversions.
LiveView Tables	Detailed information on the information contained in each LiveView table.

Topic	Content
Notifications references	Detailed information on the notifications user interfaces (UIs).
Web Applications references	Detailed information on configuration the Web Applications.

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 886

"Performance Tuning Parameters" on page 893

"Waveform parameters" on page 900

"Sepam event reading parameters" on page 906

"MicroLogic modules configuration parameters" on page 904

"Data replication parameters" on page 902

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:

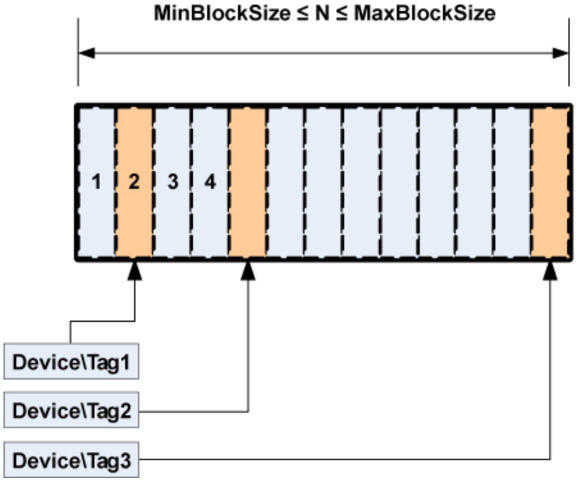
```
[PWRMODBUS.MYCLUSTER.PORT_1.DEVICE_A]
enableScatteredReads 1
```

```
[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.

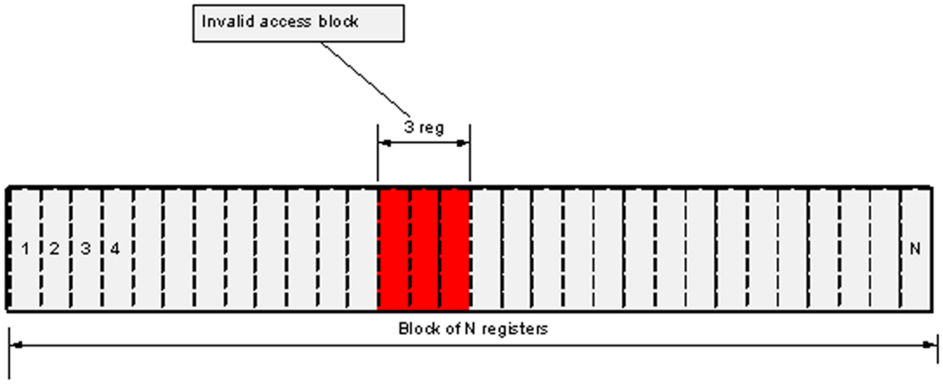
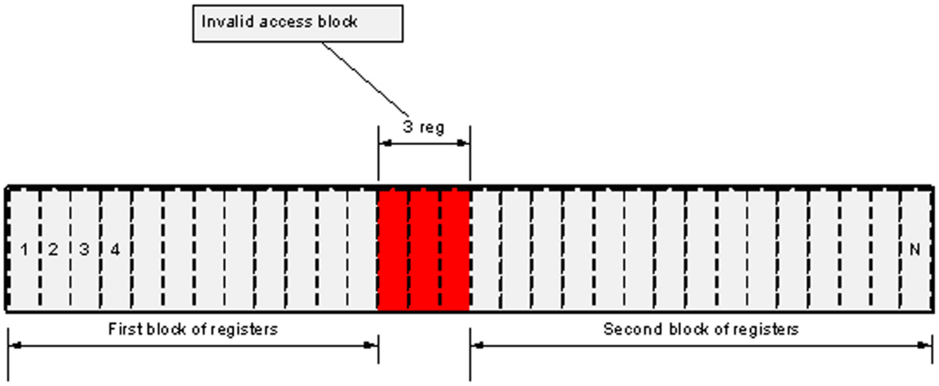


Figure 1 shows how the "invalid memory access block" affects MODBUS register blocking. In this situation, if the driver does not know that the block of 3 inaccessible registers exists, it will try to block all registers from 1 to N (depending on data that was requested by the real-time data collector). 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

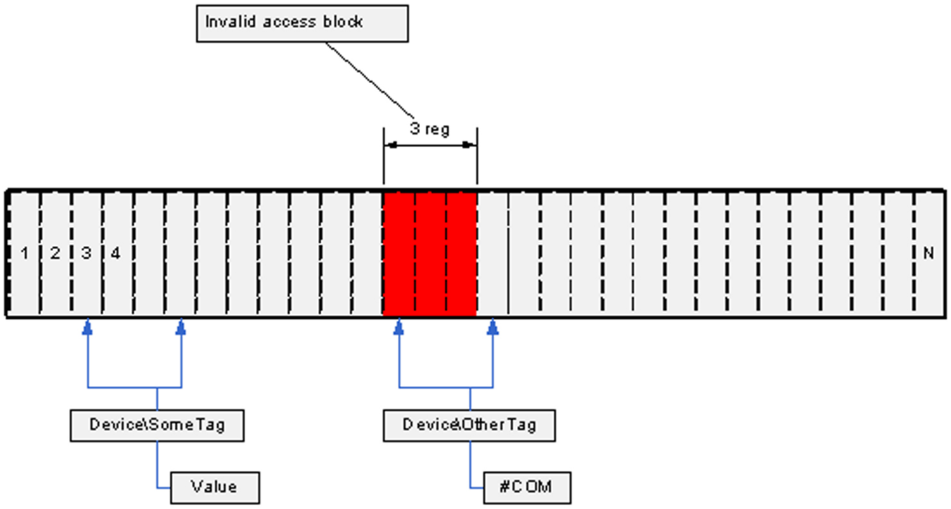
Example:

```
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
```

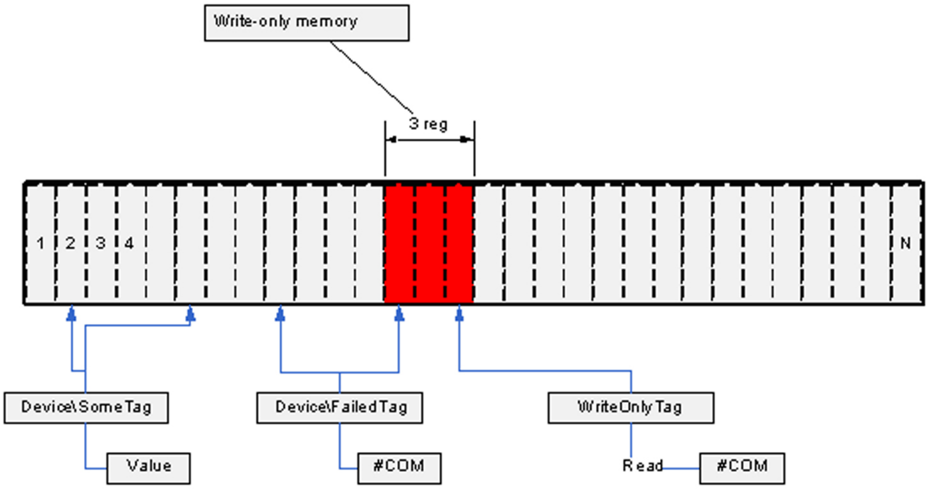
```
[ERROR] [MISC] [RealTimeData::BlockBuilder::AddDataPoint()] Cannot add
datapoint, one or more invalid memory addresses fall into non-splittable
block
```

```
[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.

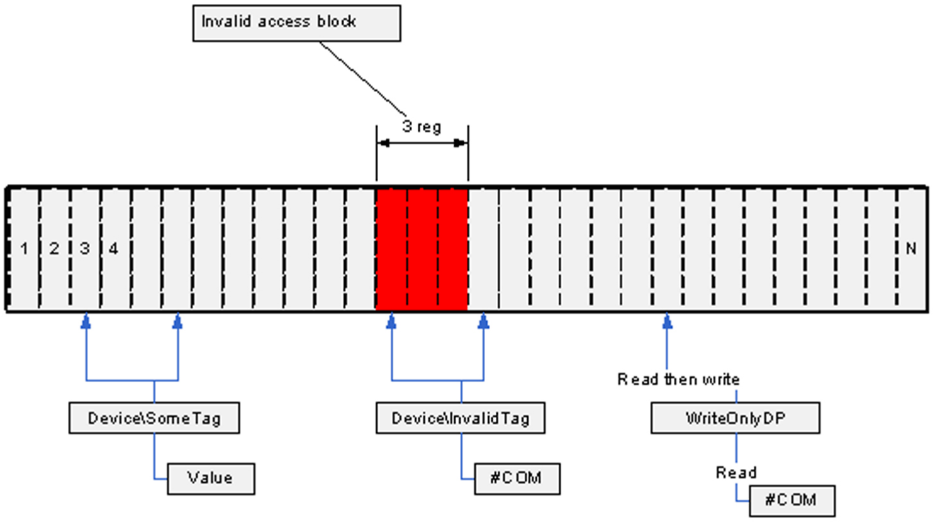
Write-only Memory

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.

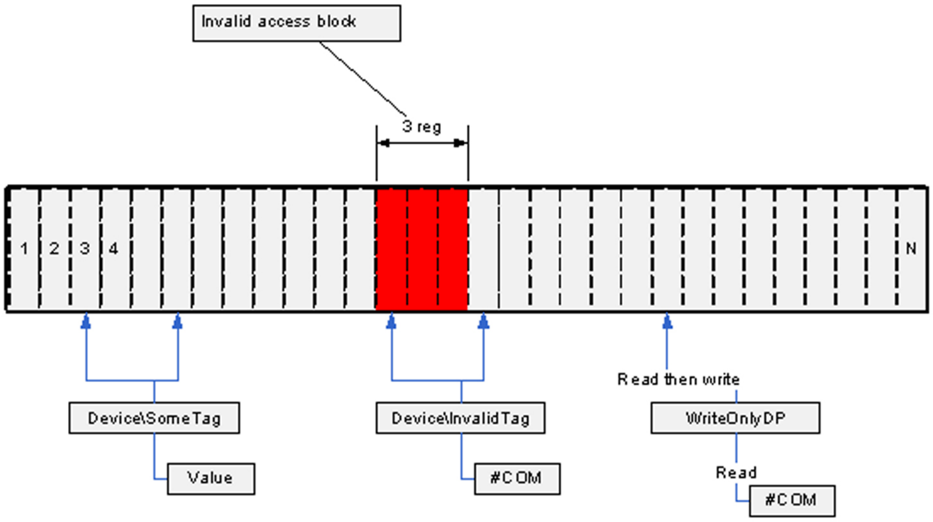
Tags found invalid, due to intersection with invalid memory areas, are not blocked with “good” real-time tags and will not therefore interfere with them.

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\2020\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 607):

```
[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 "Editing tag addresses" on page 236 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						

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
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
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)

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
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)
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)

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
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
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)

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
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
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.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
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.
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.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
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.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
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.
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.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
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.
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.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
CM4 Power Factor IEEE (L:P:50)	MV/CM	REAL	1 register	No	CM4	Returns the IEEE power factor.
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.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
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.
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.

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
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.
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

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
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.
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

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
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).

Logic Code	IEC Type	Power SCADA Operation Data Type	Register Definition	Scaled Required? (register or fixed)	Device Specific?	Notes
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	Micrologic X	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 IEEE32 (L:P:171)	CR	REAL	2 to 4 registers	Fixed	Micrologic X	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	Micrologic X	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 Unsigned (L:P:173)	CR	REAL	2 to 4 registers	Fixed	Micrologic X	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
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

Genie Abbreviation	Description
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
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
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

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
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
SL_In	incoming bus

Genie Abbreviation	Description
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
gen_nd_1	generator 1: no current designation

Genie Abbreviation	Description
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
sl_out_cb_rack	feeder, circuit breaker when racked out, plug
SL_Out_Rack	feeder, circuit breaker when racked out, socket
sl_resistor	resistor

Genie Abbreviation	Description
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
hv_cb_nes_dr_nc	high-voltage drawout circuit breaker, no remote control, no earth

Genie Abbreviation	Description
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 metersl

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\v2020 R2\Applications\AppServices\bin\Configuration.xml).

Deadbands

```
<ConfigurationItem Key="Deadbands" Category="Platform Mapping" Applic-
ation="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).

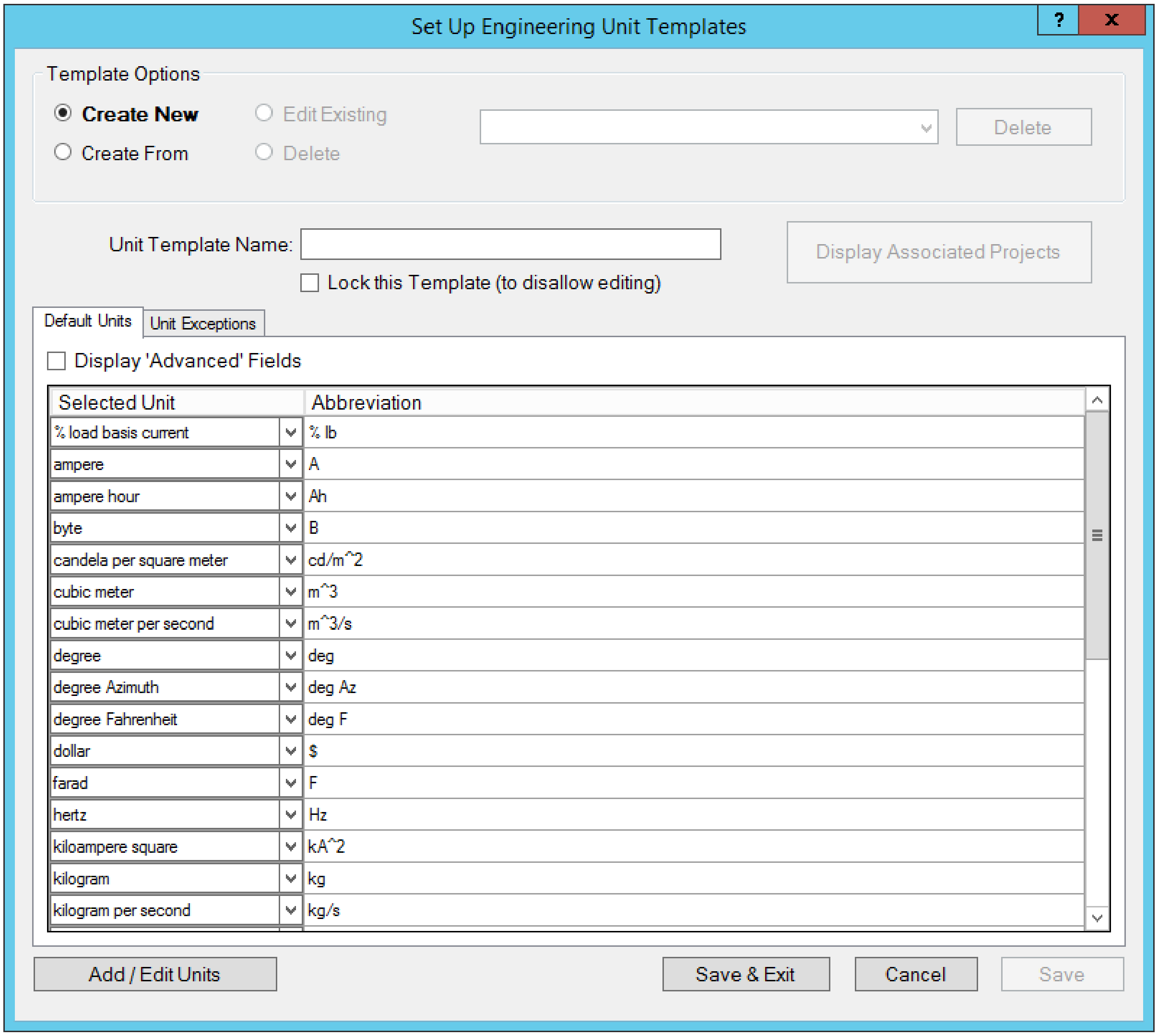
To configure engineering units or conversions, see:

- "Set up engineering templates and select conversions" on page 951
- "Add or edit a base engineering unit or conversion" on page 955

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**.



Template Options: ● Create New ○ Edit Existing ○ Create From ○ Delete

Unit Template Name:

□ Lock this Template (to disallow editing)

□ Display 'Advanced' Fields

Selected Unit	Abbreviation
% load basis current	% lb
ampere	A
ampere hour	Ah
byte	B
candela per square meter	cd/m^2
cubic meter	m^3
cubic meter per second	m^3/s
degree	deg
degree Azimuth	deg Az
degree Fahrenheit	deg F
dollar	$
farad	F
hertz	Hz
kiloampere square	kA^2
kilogram	kg
kilogram per second	kg/s

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 955.
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 955.

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 955. 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 955.
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 955.

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 955).
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.

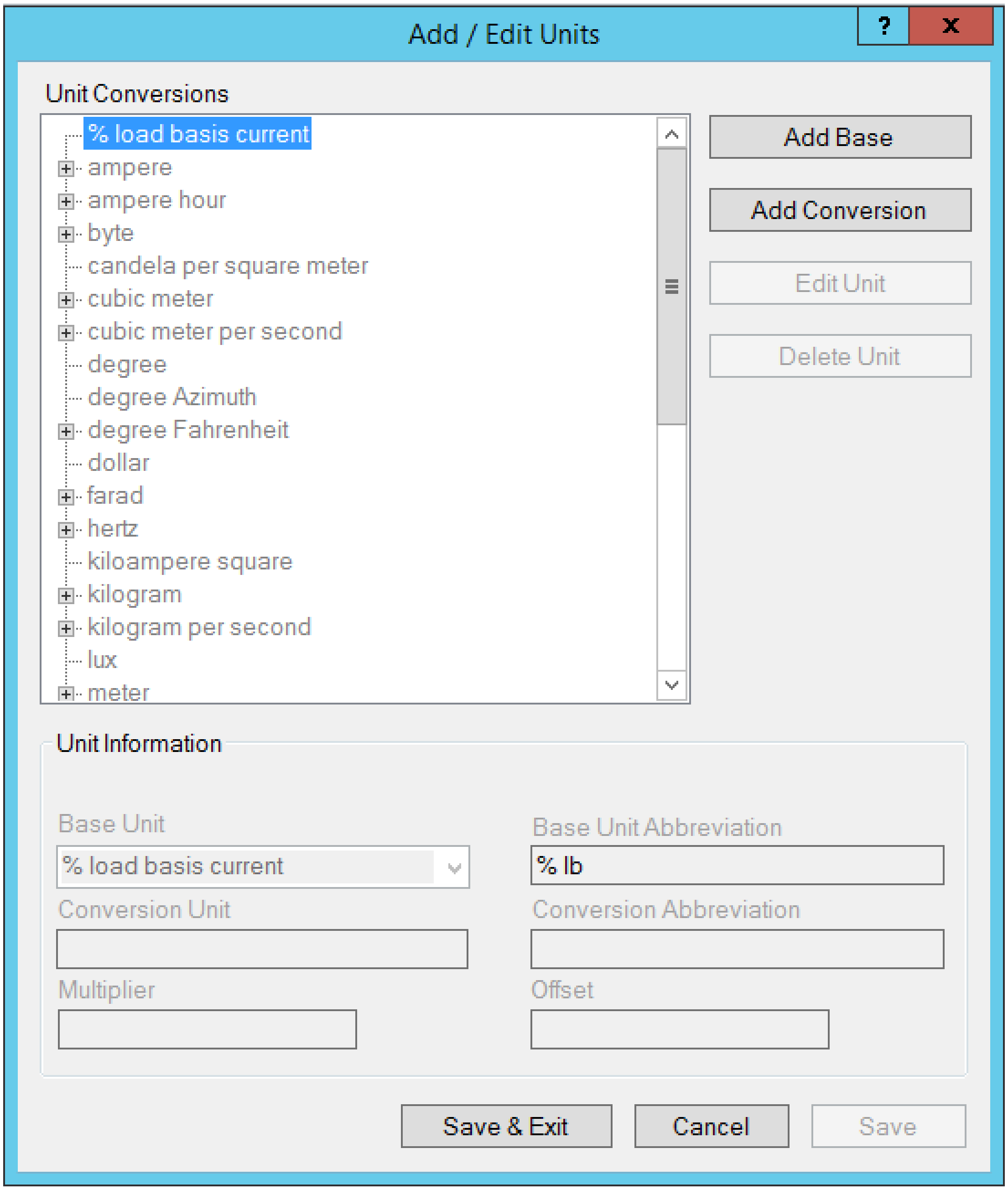
To delete a template:

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.



Unit Conversions

- % load basis current
- ampere
- ampere hour
- byte
- candela per square meter
- cubic meter
- cubic meter per second
- degree
- degree Azimuth
- degree Fahrenheit
- dollar
- farad
- hertz
- kiloampere square
- kilogram
- kilogram per second
- lux
- meter

Unit Information

Base Unit: % load basis current

Base Unit Abbreviation: % lb

Conversion Unit:

Conversion Abbreviation:

Multiplier:

Offset:

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" on page 958
- "LiveView Power Flow Summary" on page 959
- "LiveView Energy Summary" on page 959
- "LiveView Energy Readings" on page 959
- "LiveView Fundamental Phasor Readings" on page 960
- "LiveView THD Current Summary" on page 960
- "LiveView THD Voltage Summary" on page 960
- "LiveView Uptime Summary" on page 961
- "LiveView Incremental Reactive Energy Summary" on page 961
- "LiveView Incremental Real Energy Summary" on page 961
- "LiveView Harmonic Apparent Power Flows" on page 962
- "LiveView Harmonic Reactive Power Flows" on page 962
- "LiveView Harmonic Real Power Flows" on page 963
- "LiveView Demand Current Summary" on page 964
- "Live View Demand Voltage Summary" on page 964

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 references

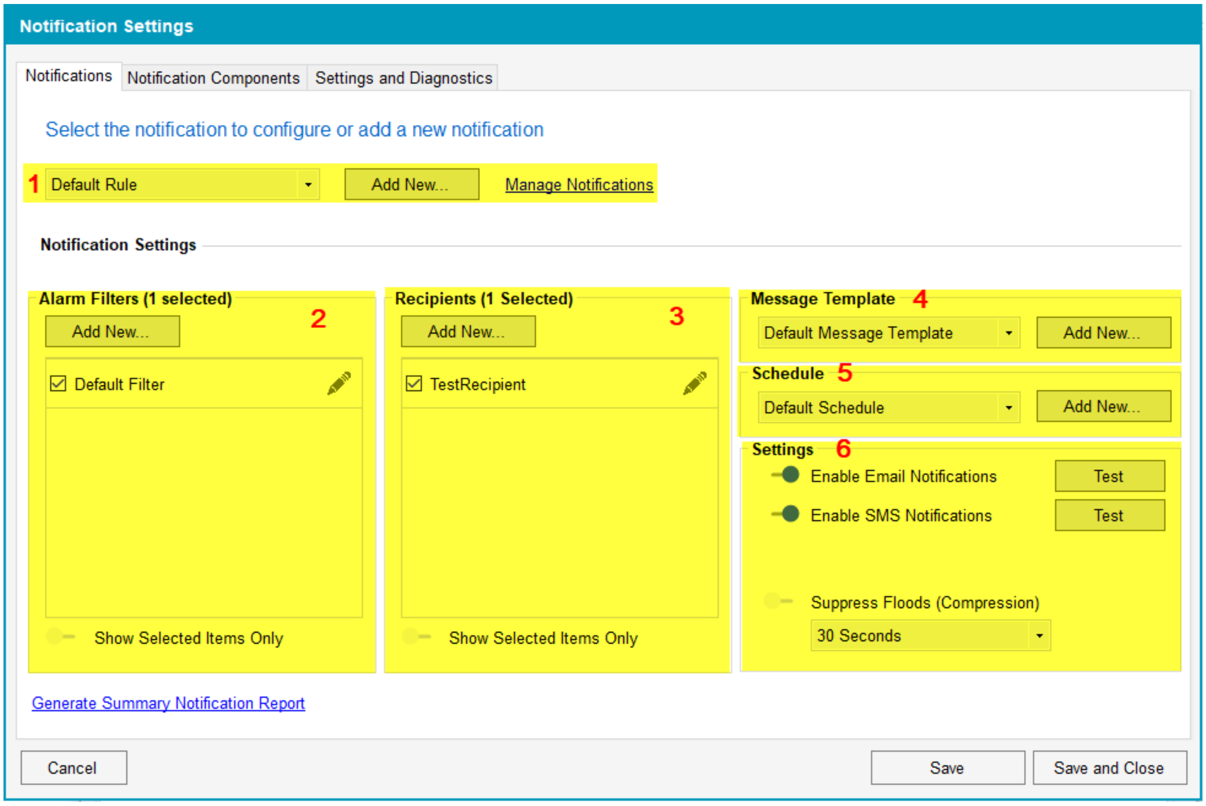
This section contains information on the Notifications Settings user interface (UI) as well as more detailed information on configuration options.

For detailed information on the notifications UIs, see the following topics:

- "Notifications UI" on page 965
- "Notifications Components UI" on page 966
- "Settings and Diagnostics UI" on page 966
- "Alarm Filter System Views" on page 967

Notifications UI

The **Notifications** pane lists all the system notifications and displays all the component information of a selected notification.


Notification Settings

Notifications | Notification Components | Settings and Diagnostics

Select the notification to configure or add a new notification

1 Default Rule ▾ | Add New... | Manage Notifications

Notification Settings

Alarm Filters (1 selected) 2
Add New...
☑ Default Filter
Show Selected Items Only

Recipients (1 Selected) 3
Add New...
☑ TestRecipient
Show Selected Items Only

Message Template 4
Default Message Template ▾ | Add New...

Schedule 5
Default Schedule ▾ | Add New...

Settings 6
● Enable Email Notifications | Test
● Enable SMS Notifications | Test
Suppress Floods (Compression)
30 Seconds ▾

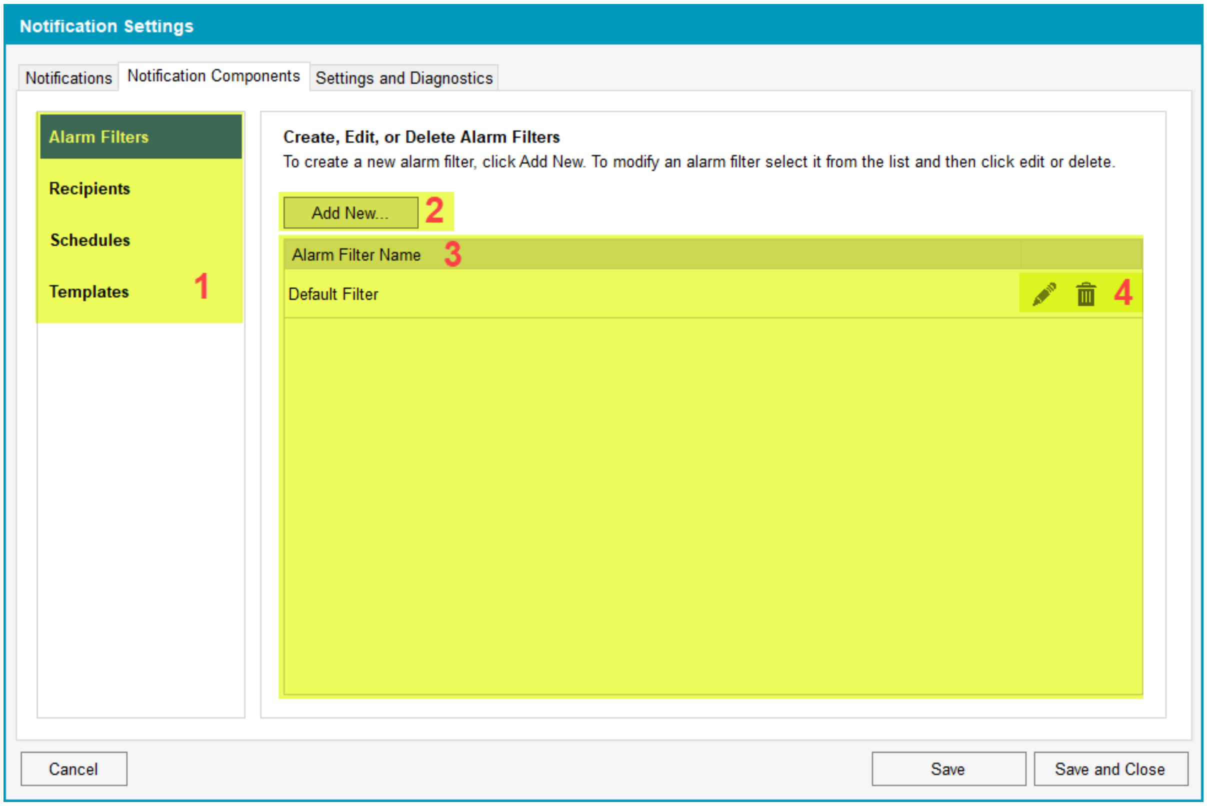
Generate Summary Notification Report

Cancel | Save | Save and Close

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

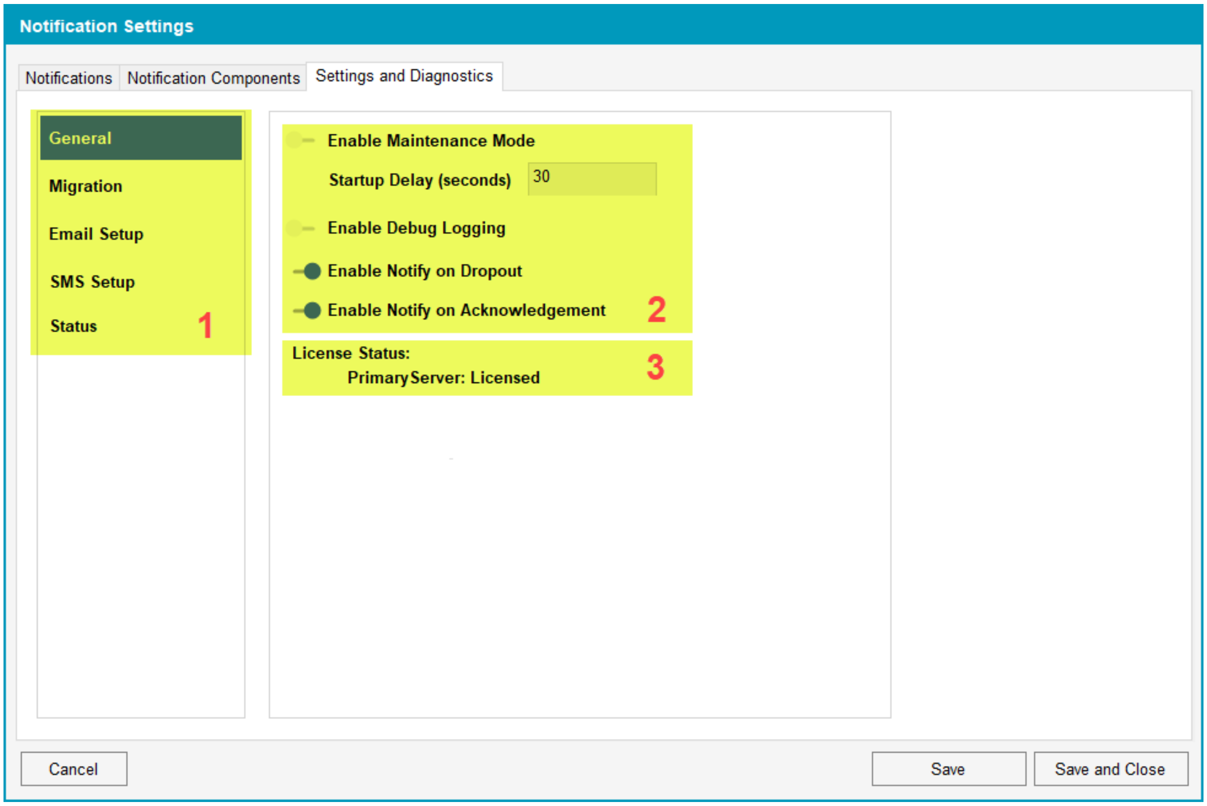
Notification Components consist of alarm filters, recipients, schedules, and templates. Use the **Notification Components** pane to manage notification components.



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.



○ Enable Maintenance Mode

Startup Delay (seconds) 30

○ Enable Debug Logging

● Enable Notify on Dropout

● Enable Notify on Acknowledgement

License Status:
PrimaryServer: Licensed

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

Power Modbus (PwrModbus) Driver for Modbus Devices

Power SCADA Operation uses the Power Modbus (PwrModbus) driver to communicate to Modbus devices. Although Modnet is available to use, PwrModbus gives the user more flexibility and ease of use than Modnet.

NOTE: Modnet should still be used for some PLC Devices, such as the TSX Quantum and TSX Premium PLC ranges. There are two special Modnet Protocols that exist for these PLCs that will select the correct addressing modes, select certain INI values, and other settings to provide compatibility with these devices without the need to set any other special parameters.

Benefits of PwrModbus

These are the benefits of using PwrModbus for Modbus devices:

- Writing Real-Time tags – PwrModbus creates variable tags with different attributes that are specific to the device registers. Attributes such as register type, scaling, and priority, etc. can be added. Modnet cannot add attributes to the tag and manipulation has to be done after the value is read in.
- Logic Codes – Logic codes tell Power SCADA Operation how to mathematically operate on the values in device registers to give users the desired values. Logic codes include Date / Time, Scaled Registers Signed/Unsigned, Coil Register, and IEEE variations. Logic codes allow you to view the device register values without any manipulation to get the desired value. Logic codes allow you to use fewer tags to get the data. For example, to read an Energy (Consumption) tag with logic codes, you would need a single tag with the specific Mod10K logic code. In comparison, to read an Energy (Consumption) tag with the Modnet driver, you would need four tags (at minimum) to read each register, then you would need to perform the Mod10K algorithm on the registers (in Cicode), and finally read the result.
- On-board alarms – If a device has onboard alarms, the Power Modbus allows the creation of alarm tags that can retrieve the onboard alarms. This feature allows users to retrieve historical alarms if the device stops communicating to the Power SCADA Operation or for any other issue.
- Control Tags – Creation of tags that can do predefined controls on some standard devices for example Operate (ENERGIZE).
- Reset Tags – Standard device types include some pre-defined resets. These pre-defined commands cause proprietary functions within the device.
- Driver Parameters in Citect.ini – Numerous driver parameters available to fine-tune performance of the devices using the driver.
- Waveform – Supports download of waveforms on certain device types with waveform capabilities (Comtrade files).
- Advanced Logging Parameters – a debug logging system is implemented for the driver. Logging is integrated with the I/O server system log and it produces messages in order of appearance that is vital for troubleshooting issues with devices.

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 969 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 "Synchronizing devices" on page 996 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\v2020 R2\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.

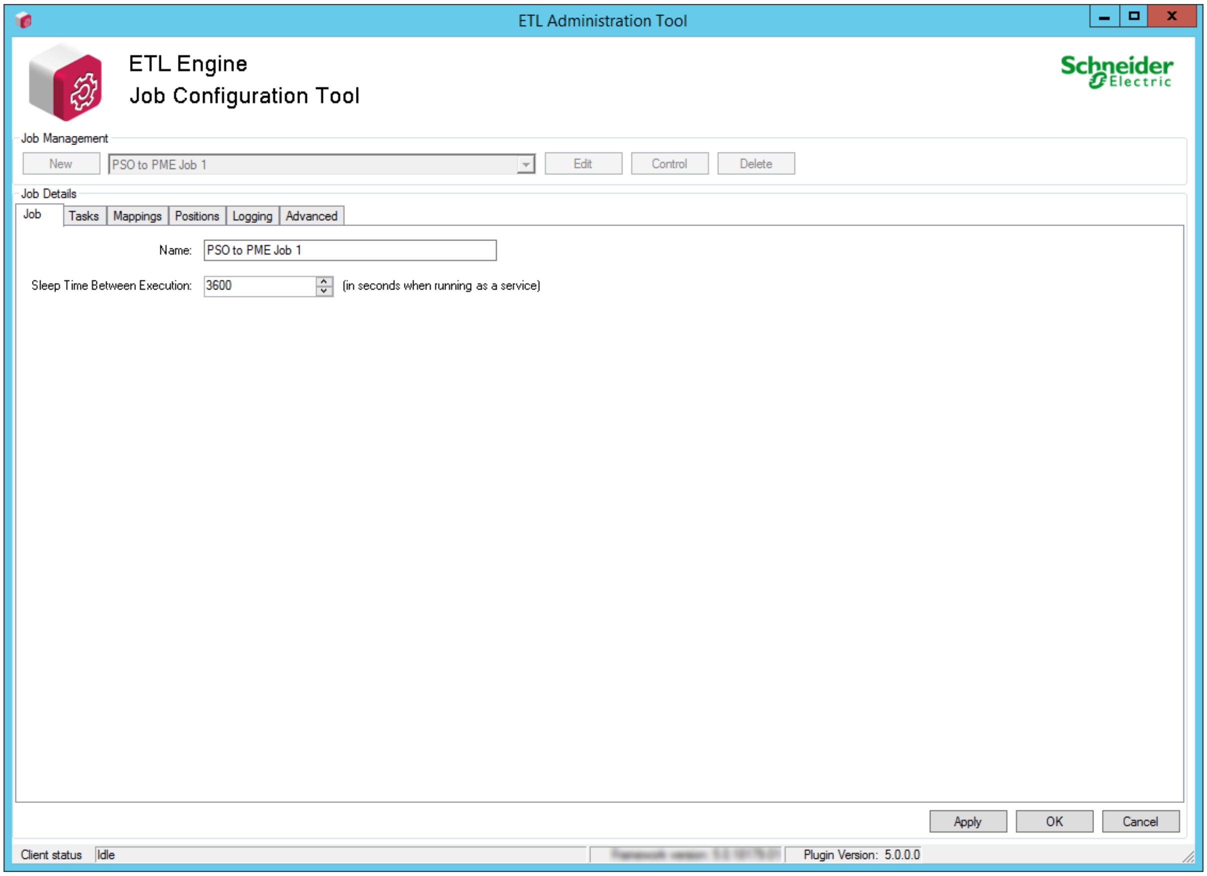
Opening the ETL Administration Tool

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:



Job Management: PSO to PME Job 1

Name: PSO to PME Job 1

Sleep Time Between Execution: 3600 (in seconds when running as a service)

Client status Idle

Plugin Version: 5.0.0.0

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 2020 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).

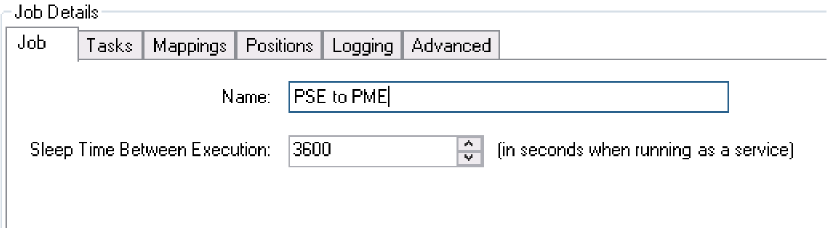
   When the mapping upgrade process is complete, detailed results are displayed. This information is also save to a new XML file in the MappingResults folder (under the ETL root). This XML file is useful for tracking and troubleshooting device and topic mappings before and after the upgrade operation ran.

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.



Name: PSE to PME

Sleep Time Between Execution: 3600 (in seconds when running as a service)

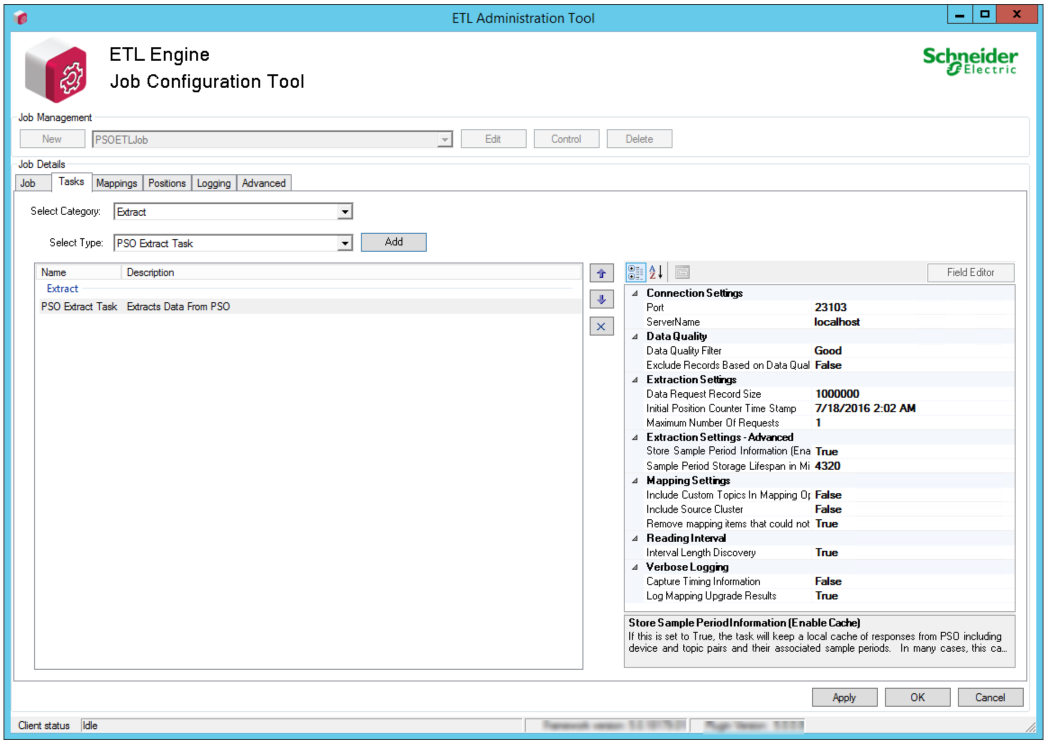
NOTE: The ETL job name has to be unique. Make sure your ETL job name does not conflict with any other ETL jobs on your system. This is particularly important to consider when registering ETL jobs to run as Windows services.

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:



Job Management: PSOETLJob

Select Category: Extract

Select Type: PSO Extract Task

Name	Description
Extract	
PSO Extract Task	Extracts Data From PSO

Connection Settings	
Port	23103
ServerName	localhost
Data Quality	
Data Quality Filter	Good
Exclude Records Based on Data Qual	False
Extraction Settings	
Data Request Record Size	1000000
Initial Position Counter Time Stamp	7/18/2016 2:02 AM
Maximum Number Of Requests	1
Extraction Settings - Advanced	
Store Sample Period Information (Ena	True
Sample Period Storage Lifespan in Mi	4320
Mapping Settings	
Include Custom Topics In Mapping O	False
Include Source Cluster	False
Remove mapping items that could not	True
Reading Interval	
Interval Length Discovery	True
Verbose Logging	
Capture Timing Information	False
Log Mapping Upgrade Results	True

Store Sample Period Information (Enable Cache)
If this is set to True, the task will keep a local cache of responses from PSO including device and topic pairs and their associated sample periods. In many cases, this ca...

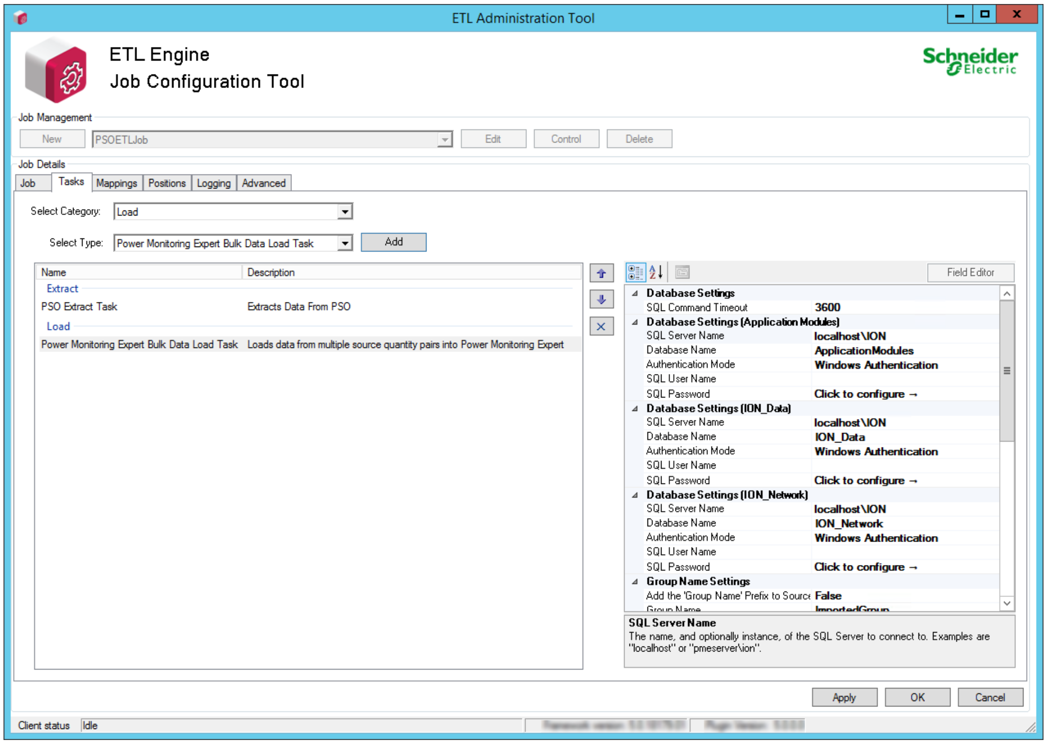
Client status Idle

8. In the Field Editor pane, configure the extract task. See "Configuring the PSO to PME extract task" on page 973 for details.
9. (Optional) Add and configure a transform task. See "Configuring the PSO to PME transform task" on page 979 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:



Job: PSOETLJob

Select Category: Load

Select Type: Power Monitoring Expert Bulk Data Load Task

Name	Description
Extract	
PSO Extract Task	Extracts Data From PSO
Load	
Power Monitoring Expert Bulk Data Load Task	Loads data from multiple source quantity pairs into Power Monitoring Expert

Database Settings	
SQL Command Timeout	3600
Database Settings (Application Modules)	
SQL Server Name	localhost\ION
Database Name	ApplicationModules
Authentication Mode	Windows Authentication
SQL User Name	
SQL Password	Click to configure →
Database Settings (ION_Data)	
SQL Server Name	localhost\ION
Database Name	ION_Data
Authentication Mode	Windows Authentication
SQL User Name	
SQL Password	Click to configure →
Database Settings (ION_Network)	
SQL Server Name	localhost\ION
Database Name	ION_Network
Authentication Mode	Windows Authentication
SQL User Name	
SQL Password	Click to configure →
Group Name Settings	
Add the 'Group Name' Prefix to Sourc	False
Group Name	ImportedGroup

SQL Server Name
The name, and optionally instance, of the SQL Server to connect to. Examples are "localhost" or "pmeserver\ion".

Client status Idle

13. In the Field Editor pane, configure the load task settings. See "Configuring the PSO to PME load task" on page 980 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 980 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 984 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	The port used for communicating with PSO.	Default: 23103; leave as is.
Server Name	The name or IP address of the PSO server.	Default: localhost
Data Quality		

Setting Name	Description	Setting Parameters / Recommended Values
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 Additional Notes below)		
Data Request Record Size	The maximum number of records in each data request, sent to SCADA. See "Grouping" on page 976 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 976 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 970	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**

Additional Notes:

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.

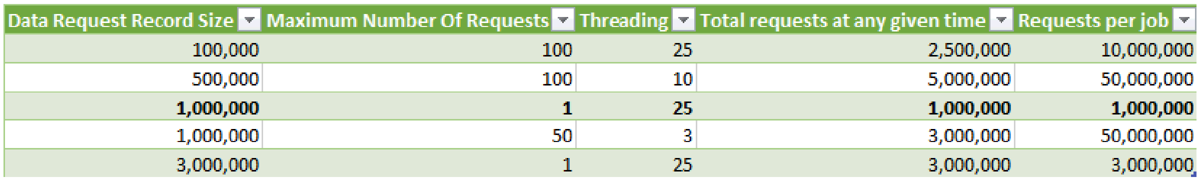
The total records requested at any given time can be calculated as Data Request Record Size x Number of Threads.

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 977 .

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.

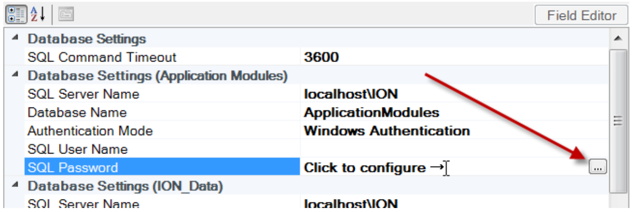
Setting Name	Description	Setting Parameters / Recommended Values
Transform		
Intervalization Method	The method for converting the values from an irregular interval to a regular interval.	LastKnownValue
Intervalize to present time	When set to True, the data is intervalized up to the current system time. 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.  Database Settings SQL Command Timeout 3600 Database Settings (Application Modules) SQL Server Name localhost\ION Database Name ApplicationModules Authentication Mode Windows Authentication SQL User Name SQL Password Click to configure Database Settings (ION_Data) SQL Server Name localhost\ION
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		

Setting Name	Description	Setting Parameters / Recommended Values
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
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		

Setting Name	Description	Setting Parameters / Recommended Values
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		
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 984 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**.

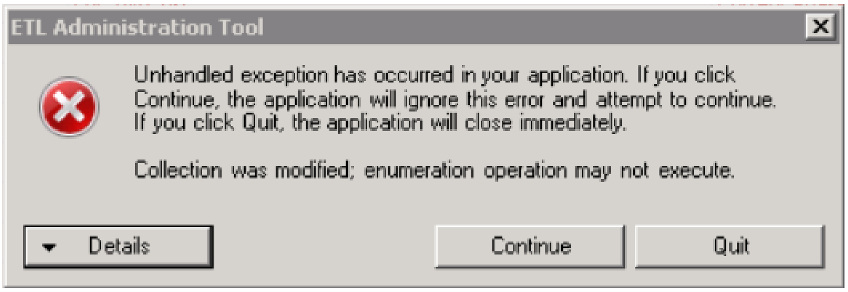
   NOTE: You might need to restart the Schneider Electric CoreServiceHost service and reset Internet Information Services (IIS) on the Power SCADA Server if the device that you add to Power SCADA Operation does not appear in the **Mappings** pane after you click **Load Sources**. Performing a restart could 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:



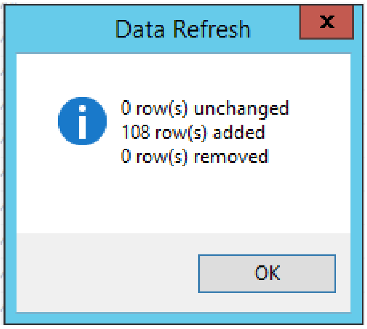
ETL Administration Tool

Unhandled exception has occurred in your application. If you click Continue, the application will ignore this error and attempt to continue. If you click Quit, the application will close immediately.

Collection was modified; enumeration operation may not execute.

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:

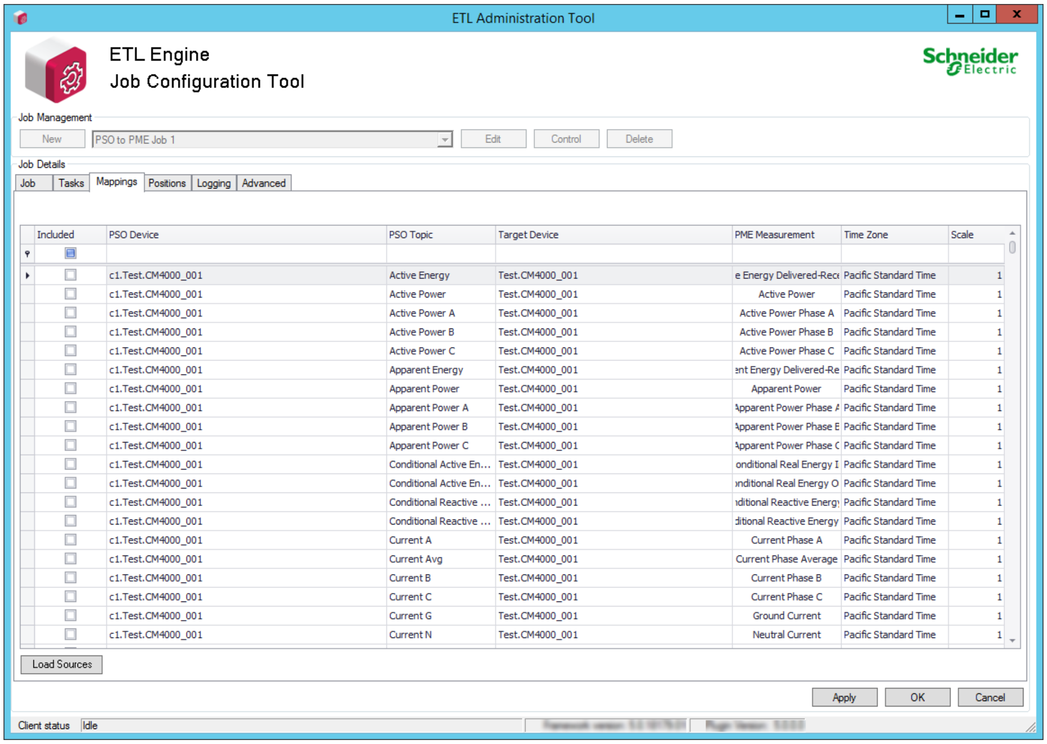


0 row(s) unchanged
108 row(s) added
0 row(s) removed

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:



PSO to PME Job 1

Included	PSO Device	PSO Topic	Target Device	PME Measurement	Time Zone	Scale
☐	c1.Test.CM4000_001	Active Energy	Test.CM4000_001	e Energy Delivered-Rec	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Active Power	Test.CM4000_001	Active Power	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Active Power A	Test.CM4000_001	Active Power Phase A	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Active Power B	Test.CM4000_001	Active Power Phase B	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Active Power C	Test.CM4000_001	Active Power Phase C	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Apparent Energy	Test.CM4000_001	ent Energy Delivered-Re	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Apparent Power	Test.CM4000_001	Apparent Power	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Apparent Power A	Test.CM4000_001	Apparent Power Phase A	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Apparent Power B	Test.CM4000_001	Apparent Power Phase B	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Apparent Power C	Test.CM4000_001	Apparent Power Phase C	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Conditional Active En…	Test.CM4000_001	onditional Real Energy I	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Conditional Active En…	Test.CM4000_001	onditional Real Energy O	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Conditional Reactive …	Test.CM4000_001	ditional Reactive Energy	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Conditional Reactive …	Test.CM4000_001	ditional Reactive Energy	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Current A	Test.CM4000_001	Current Phase A	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Current Avg	Test.CM4000_001	Current Phase Average	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Current B	Test.CM4000_001	Current Phase B	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Current C	Test.CM4000_001	Current Phase C	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Current G	Test.CM4000_001	Ground Current	Pacific Standard Time	1
☐	c1.Test.CM4000_001	Current N	Test.CM4000_001	Neutral Current	Pacific Standard Time	1

Client status: Idle [redacted] [redacted]

4. Review the PSO to PME mappings.

5. (Optional) Edit the default mappings if they do not meet your needs. See"Editing PSO to PME mappings" on page 986 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"Tips for working with mappings" on page 988 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 "Resetting and resending data" on page 999. Or you can run the ETL job; see "Running an ETL job" on page 992

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**.

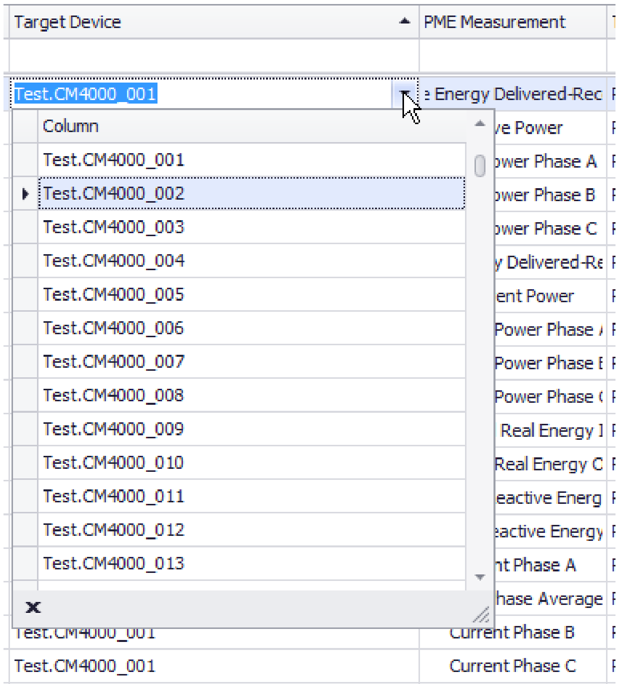
      > **NOTE**: The **PME Source** name has to match the Power Monitoring Expert device naming convention of *Group*.*DeviceName* with no special characters, such as: \ | + = : ; < > ? or , .

      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.



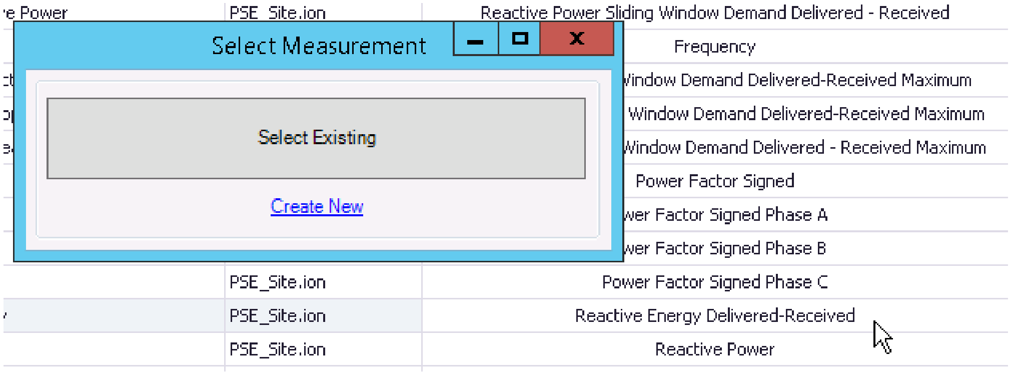
Editing the PME quantity

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:

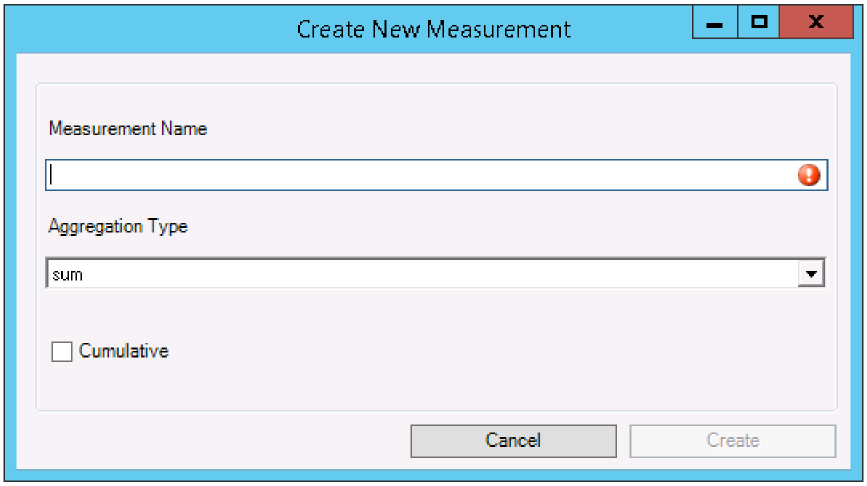


2. Assign a different or new PME quantity to the PSO device-topic pair :

   To assign a new PME quantity:

   a. Click **Create New**.

      The Create New Measurement dialog appears:

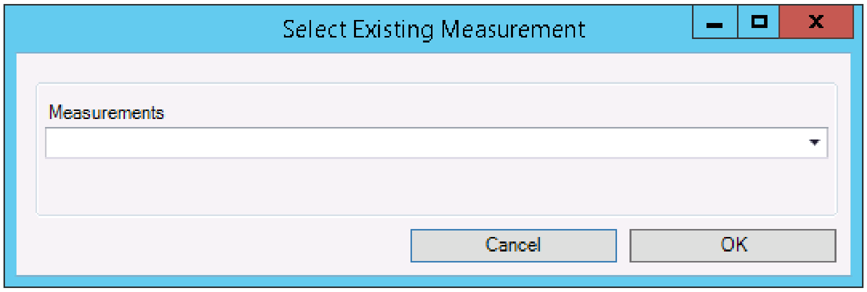


   b. Enter the new measurement name, set its values, and then click **Create**.

   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**.

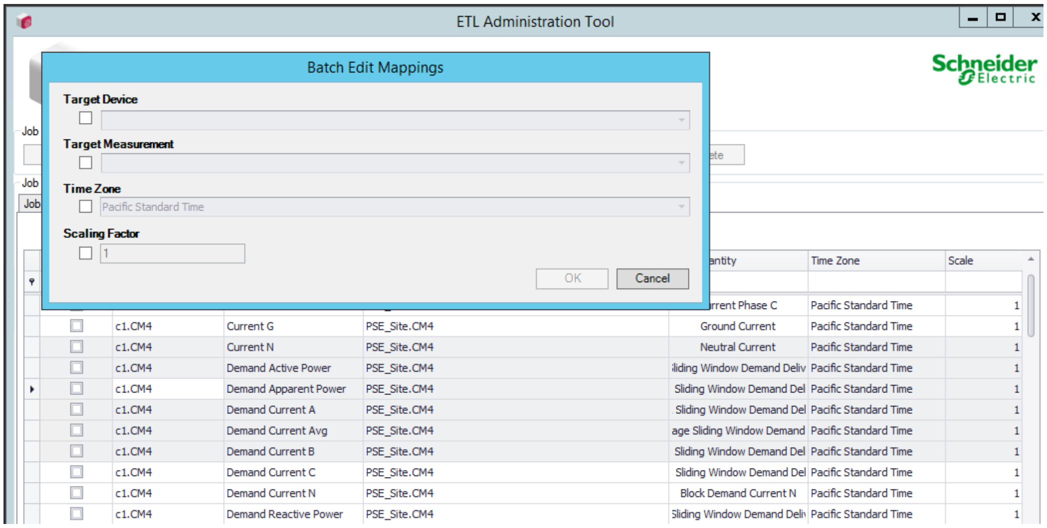
Batch Edits

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.



Batch Edit Mappings

Target Device
□

Target Measurement
□

Time Zone
□ Pacific Standard Time

Scaling Factor
□ 1

OK Cancel

				antity	Time Zone	Scale
				rrent Phase C	Pacific Standard Time	1
□	c1.CM4	Current G	PSE_Site.CM4	Ground Current	Pacific Standard Time	1
□	c1.CM4	Current N	PSE_Site.CM4	Neutral Current	Pacific Standard Time	1
□	c1.CM4	Demand Active Power	PSE_Site.CM4	liding Window Demand Deliv	Pacific Standard Time	1
□	c1.CM4	Demand Apparent Power	PSE_Site.CM4	Sliding Window Demand Del	Pacific Standard Time	1
□	c1.CM4	Demand Current A	PSE_Site.CM4	Sliding Window Demand Del	Pacific Standard Time	1
□	c1.CM4	Demand Current Avg	PSE_Site.CM4	age Sliding Window Demand	Pacific Standard Time	1
□	c1.CM4	Demand Current B	PSE_Site.CM4	Sliding Window Demand Del	Pacific Standard Time	1
□	c1.CM4	Demand Current C	PSE_Site.CM4	Sliding Window Demand Del	Pacific Standard Time	1
□	c1.CM4	Demand Current N	PSE_Site.CM4	Block Demand Current N	Pacific Standard Time	1
□	c1.CM4	Demand Reactive Power	PSE_Site.CM4	Sliding Window Demand Deli	Pacific Standard Time	1

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

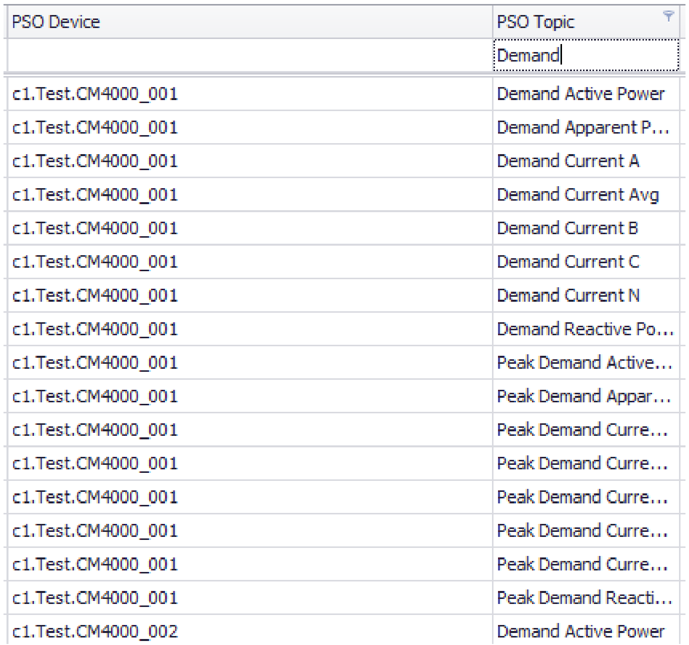
To sort contents by column:

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.

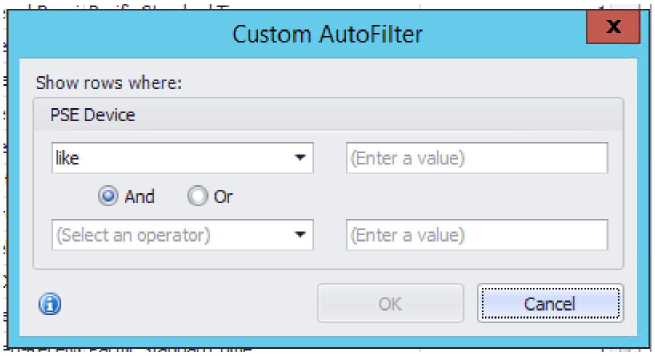


PSO Device	PSO Topic
	Demand
c1.Test.CM4000_001	Demand Active Power
c1.Test.CM4000_001	Demand Apparent P...
c1.Test.CM4000_001	Demand Current A
c1.Test.CM4000_001	Demand Current Avg
c1.Test.CM4000_001	Demand Current B
c1.Test.CM4000_001	Demand Current C
c1.Test.CM4000_001	Demand Current N
c1.Test.CM4000_001	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**.



Custom AutoFilter

Show rows where:

PSE Device

like (Enter a value)

● And ○ Or

(Select an operator) (Enter a value)

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**: Create an HTML load task as part of 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.

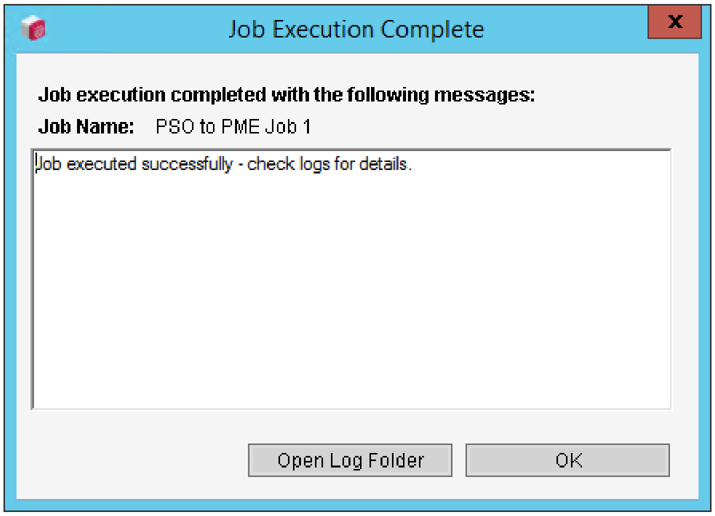
To enable the ETL logs:

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:



Job execution completed with the following messages:

Job Name: PSO to PME Job 1

Job executed successfully - check logs for details.

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"Editing a PSO to PME ETL job" on page 998 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"Editing a PSO to PME ETL job" on page 998 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"Editing a PSO to PME ETL job" on page 998 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"Editing a PSO to PME ETL job" on page 998.

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 "Tips for working with mappings" on page 988 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 "Verifying PSO data transfer to PME" on page 1000 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 997 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.

Web Applications references

This section contains reference information related to Web Applications.

For more information, see the following sections:

- Alarms and Incidents customization
- System and personal localization settings
- "Graphics references" on page 1003

Alarms and Incidents customization

You can customize alarms and incidents in Power SCADA. To override the default classifications or to add new classifications, create a Classifications.json file in your project directory. The service will merge the default and the project classifications files. Anything in the project classifications file will take priority.

There is an example classifications.json file located in:
C:\ProgramData\Schneider Electric\Power SCADA Operation\v2020 R2\Examples.

This file contains a few key json objects: Incident Categories, Incident Types, Alarm Categories, and Alarm Types.

Incident and Alarm categories

Incident and Alarm categories contain an Id, Display Name, Rank, and (Incident or Alarm) Types.

Id: Id of the incident or alarm category that is used in translation. If the Id key is not found in translation, the Id will be displayed in the UI.

DisplayName: Not currently used. Intended to be the default display name if the id key is not found in translation.

Rank: Orders the incident and alarm types in the UI. Lowest rank displays first.

Types: The list of incident or alarm types that will fall into the category.

Incident Types

An incident type contains information about how to group alarms into this incident type.

Id: Id of the incident type that is used in translation. If the Id key is not found in translation, the Id will be displayed in the UI.

DisplayName: Not currently used. Intended to be the default display name if the id key is not found in translation.

Priority: Incident priority (0-255) with 255 being the highest priority.

TimeWindowSeconds: Determines the time window for incident grouping with a few exceptions:

- -1 = a 'full day' group.
- -2 = State based. If any alarm in the incident is active, new alarms are added to that incident. If all alarms are not active, a new incident is created.
- Anything less than 0 (not specified above) will default to -1
- Anything above 0 is a sliding windows for the alarm start times in seconds.

AlarmTypeIds: List of alarm types that are grouped into this incident type.

Alarm Types

An alarm type contains information about how to group events into this alarm type.

Id: Id of the alarm type that is used in translation. If the Id key is not found in translation, the Id will be displayed in the UI.

DisplayName: Not currently used. Intended to be the default display name if the id key is not found in translation.

Searches: List of items to group an event into this alarm type.

Confidence: The matching confidence. The event with the highest matched confidence will be the alarm type for the event.

Custom3-6: List of string matches for the Custom3-6 fields on the alarm from Citect (if any).

TagDescriptionRegEx: List or regular expressions to match to the alarm description.

TagNameRegEx: List or regular expressions to match to the alarm tag name.

EquipmentRegEx: List or regular expressions to match to the equipment name.

During the matching, each of the search items will be checking. All matching fields are evaluated in this manner. For an event to be grouped into the type, all of the criteria must be met. For example, if an event has **Custom3 = Loss**, but **Custom5 = Currents**, the event is not matched.

System and personal localization settings

NOTE: The language settings in System Language and Personal Preferences determine the language the web applications are displayed in.

By default, the localization settings in Personal Preferences are the same as the ones in System localization. Changes to the settings in System Language are automatically copied to the Personal Preferences settings as long as the Personal Preferences settings have never been customized. After you customized the Personal Preferences localization settings once, they will no longer change when the System Language settings are changed.

NOTE: Your personal localization settings overrule the system localization settings for your user account.

Example 1: Language settings in Personal Preferences follow System Language if they have never been customized.

Condition	Language Settings	Comments
Default	System Language: English Personal Preferences: English	This is assuming the software was installed as an English system.
Change System Language to French	System Language: French Personal Preferences: French	The Personal Preferences language settings follow the System Language settings.

Example 2: Personal Preferences remain at customized setting after having been customized at some point.

Condition	Language Settings	Comments
Default	System Language: English Personal Preferences: English	This is assuming the software was installed as an English system.
Change Personal Preferences to French	System Language: English Personal Preferences: French	The Personal Preferences have been customized.
Change Personal Preferences back to English	System Language: English Personal Preferences: English	The settings are back to their defaults, but the Personal Preferences have been customized.
Change System Language to French	System Language: French Personal Preferences: English	The Personal Preferences language settings no longer follow the System Language settings.

Graphics references

Graphics can be customized to provide the user interface required to operate a site. EcoStruxure Power SCADA Operation software uses TGML (TAC Graphics Markup Language) graphics that are created and edited using Graphics Editor.

For more information, see the following sections:

- TGML References Overview
- Graphics Editor References Overview
- Workflows

For information on the legacy Graphics Builder, see Legacy Graphics Builder References.

TGML references Overview

TAC Graphics Markup Language (TGML) is a declarative XML-based language for dynamic 2D graphics.

TGML is inspired by the XML based Scalable Vector Graphics (SVG) which is an open standard for 2D graphics.

TGML specifies a hierarchy of runtime objects with a set of properties and logic. Each markup element (XML element) represents a TGML object which can be edited, or configured, in the Graphics Editor. However, not all of the objects are graphical (visible). Several objects are used to add a specific behavior to a graphical object, such as enabling dynamic update of attributes, transformations and gradients.

The TGML object model is based on the W3C Document Object Model (DOM). The TGML graphics elements are accessible for applications through the exposed TGML DOM interfaces.

TGML version

The TGML version is specified in an XML processing instruction:

```
<?xml version="1.0"?>
<?TGML Version="1.2"?>

<TGML Width="800" Height="600" Stretch="Uniform" Background="#FFFFFF">
    ...
</TGML>
```

Namespaces

The TGML graphics elements specified in this specification belong to the default XML namespace, TGML.

TGML allows inclusion of elements from foreign namespaces anywhere with the TGML content. In general, the TGML loader will include the unknown (foreign) elements in the DOM, but will otherwise ignore the unknown elements.

For more information, see the following sections:

- TGML Overview
- Basic TGML Elements
- Interactive TGML Elements
- TGML Appendices

TGML Overview

TGML overview information:

- TGML Properties and Attributes
- TGML Coordinate System
- TGML Rendering Model
- TGML Types and Enumerations
- TGML File Format
- TGML Code Snippets
- TGML Common Attributes
- TGML Components
- TGML Document Structure
- TGML Scripting

For more TGML information, see the following chapters:

- Basic TGML Elements
- Interactive TGML Elements
- TGML Appendices

TGML Properties and Attributes

In the underlying class library, public data members are exposed as properties. Each TGML element attribute has a corresponding property in the TGML class that implements the TGML element.

Default Values

Most of the object properties have a default value. Element attributes that are omitted in the TGML document are considered to be undefined. An undefined attribute will result in assigning a default value to the corresponding property, unless the value is inherited from a parent element.

Attribute Inheritance

TGML supports attribute inheritance similar to the SVG and XAML attribute inheritance.

The attribute inheritance means that a child element inherits (gets) the attribute value from an ancestor element if the attribute value is omitted and if the attribute has been specified for an ancestor (any of the parents).

In the example below, Line will inherit the Stroke value from the Group. StrokeWidth is not defined either, but since it is not specified at the parent level, StrokeWidth will be assigned the default StrokeWidth value.

```
<Group Stroke="FF0000">
    <Line X1="10.00" Y1="10.00" X2="100.00" Y2="100.00"/>
</Group>
```

Custom Attributes

The implementation of attribute inheritance also enables the user to specify custom attributes, since an element will accept attributes that are actually unknown for the element. For example, in the attribute inheritance example, Stroke was specified, and accepted at the group level, despite the Group does not have a Stroke attribute.

Custom attributes can, for example, be used to create "local" variables. The custom attribute can be bound to a signal, or animated, as any other attribute of an element and it can be accessed from scripts in the graphics.

Error Notifications

TGML does not specify any error or warning notifications. However, the TGML implementation (e.g. Graphics Services) and the viewer and editor applications should notify the user about any error conditions.

Implicit Syntax

TGML uses an implicit syntax. The object model implementation is not exposed in the serialized TGML.

The following example defines a group containing a line. The TGML code does not reveal how the containment is implemented in Group. The Group implementation includes for example a child list that is the actual container, but such information is not serialized.

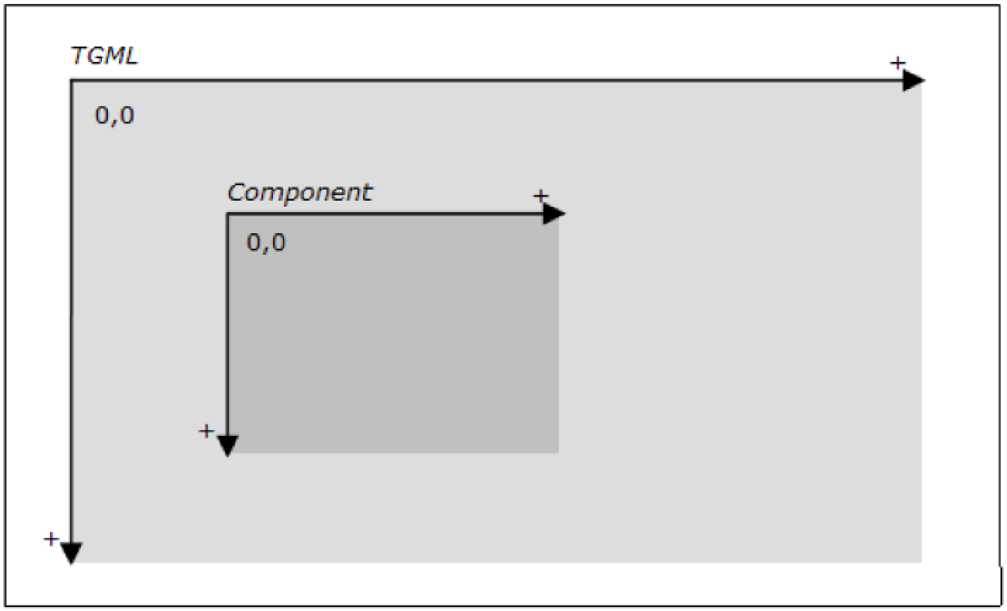
```
<Group>
    <Line
X1="50.00" Y1="100.00" X2="150.00" Y2="200.00" Fill="#FF0000" Stroke="#000000"/>
</Group>
```

Undefined (non-specified) attributes values are not serialized, to avoid breaking the attribute inheritance.

TGML Coordinate System

The origin of the TGML default coordinate system is in the upper-left corner. Values of x increase as you move right, and values of y increase as you move down.

Container elements such as TGML and Component establish new coordinate systems. Group is also a container element, but does not establish a new coordinate system.



Coordinates

The unit of measurement for coordinates is the device independent pixel, which is 1/96 of an inch (96dpi). The data type for coordinates and lengths (that is, Width and Height) is Double.

Initial Scale

A TGML viewer uses the Width, Height and Stretch attributes of the outermost TGML element to determine the initial scale. Stretch specifies if the document is scaled to fit within the work area (preserving the aspect ratio or not) or if the original size is preserved (scale 1:1).

TGML Rendering Model

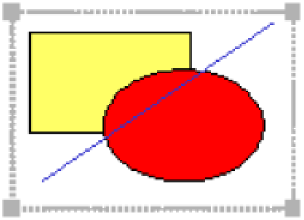
The following description is a summary of the TGML rendering model, which is very similar to the SVG rendering model.

Elements in a TGML document have an implicit drawing order, with the first elements in the TGML document getting "painted" first. Subsequent elements are painted on top of previously painted elements.

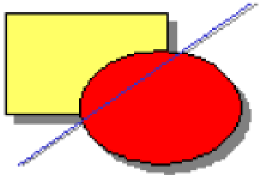
Grouping elements have the effect of producing a temporary separate canvas onto which child elements are painted. Upon the completion of the group, any filter effects specified for the group are applied to create a modified temporary canvas. The modified temporary canvas is composited into the background, taking into account any group-level settings, such as opacity, on the group.

Individual graphics elements are rendered as if each graphics element represented its own group (applicable when for example opacity is defined for the element).

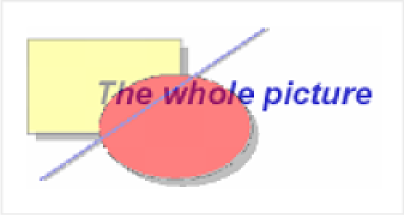
Grouping elements have the effect of producing a temporary separate canvas onto which child elements are painted.



Upon the completion of the group, any filter effects specified for the group are aööoled to create a modified temporary canvas.



The modified temporary canvas is composited into the background, taking into account any group-level settings, such as opacity, on the group.



TGML Types and Enumerations

All TGML attributes are of a certain type or enumeration.

Type / Enumeration	Description
Animation	An enumeration that controls an animation: **"Start"**, **"Stop"**
Bool	A boolean value: **"True"**, **"False"**
Brush	Describes how an area (Fill) or a stroke (Stroke) is painted. Accepted values (WP1): **"None"**, "<Color>"
CalcMode	Specifies how values are interpolated when animated: **"Discrete"**, **"Linear"** **NOTE**: **"Linear"** is only applicable when you animate attributes of the types Double, Color, Point and array of Point.

Type / Enumeration	Description
Color	An RGB or an ARGB color. RGB is described as the hexadecimal representation of the red, green and blue components. ARGB is described as the hexadecimal representation of the alpha (00-FF, where 00 is fully transparent), red, green, and blue components (in that order). **Example, opaque red:** Fill=**"#FF0000"** **Example, 50% transparent red:** Fill=**"#7FFF0000"**
Double	Double-precision floating-point numbers. For example, heights and widths are Double. **Example:"50, 25"**
FontStyle	Describes the style of the font: **"Normal"**, **"Italic"**
FontWeight	Describes the weight of the font: **"Normal"**, **"Bold"**
Format	This enumeration describes the formatting of the subscribed data. **"None"**: No formatting. The data type of the received data matches the data type of the server variable (integer, float, boolean, string etc). **"Presentation"**: Formatted as text. The server is expected to deliver the text representation of the value, if any (e.g. **On/Off** instead of **0/1**).
HorizontalAlign	Describes the horizontal alignment of a text string. **"Left"**, **"Center"**, **"Right"**
Point	Represents an XY coordinate. Syntax: "<Double>, <Double>". **Example: "25.00 , 50.00"**
Repeat	Describes the way the animation will be repeated. "<Iterations>" **"Forever"** <Iterations> specifies the number of times the animation is repeated.
SpreadMethod	Specifies how a gradient should be drawn outside of the specified gradient vector or space: **"Pad"**: The color values at the ends of the gradient vector are used to fill the remaining space. **"Reflect"**: The gradient is repeated in the reverse direction until the space is filled. **"Repeat"**: The gradient is repeated in the original direction until the space is filled.

Type / Enumeration	Description
Stretch	An enumeration that specifies how the content will be stretched. **"None"**: Preserve the original size. **"Uniform"**: Resize the content, preserving the native aspect ratio. **"Fill"**: The content is resized but aspect ratio is not preserved.
String	A string value, i.e. plain text. Reserved XML characters (&<>) are escaped using the standard XML escaping.
TextDecoration	Describes decorations that are added to the texts. **"None", "Underline", "Strikethrough"**
VerticalAlign	Describes the vertical alignment of a text string. **"Top", "Middle", "Bottom"**
Visibility	An enumeration that specifies the visibility of an element: **"Visible", "Hidden"**

Arrays

Some attributes accepts arrays of values, such as arrays of Double and Point. Arrays are written as a sequence of values, delimited by space.

```
StrokeDashArray="5.0 3.0 2.0 3.0"
Points="50.0,150.0 100.0,50.0 150.0,150.0"
```

TGML Code Snippets

A snippet is a stored piece of TGML code. It can be used for reusing constructs such as preconfigured animations and gradients.

A snippet file contains only one TGML snippet. The root element includes at least two Metadata elements describing the snippet; one for Name and one for Description. It is recommended that the file name matches the Name metadata.

```
<Metadata Name="Name" Value="Blink"/>
<Metadata Name="Description" Value="Blink twice per second"/>
```

The example below is an animation snippet that can be inserted as a child to any graphical (renderable) TGML element:

```
<Animate Attribute="Visibility" Duration="1.0" From="Visible" To="Hidden">
    <Metadata Name="Name" Value="Blink"/>
    <Metadata Name="Description" Value="Blink twice per second"/>
</Animate>
```

TGML Common Attributes

The following attributes are applicable to all TGML elements:

Attribute	Type	Description
ID	String	The identity of the element. Reserved for scripts and other entities that need to use unique element identifiers to access specific elements. **Inheritable:** No **Animatable:** No
Name	String	The name of the element. The primary use is to identify exposed elements such as Bind. **Inheritable:** No **Animatable:** No

TGML Components

Components are standardized, predefined graphics for defined use.

Component Library

Components are stored the same way as TGML code snippets. Component is the root element and the associated Metadata elements describe the component.

ComponentContent Element

The document type ComponentContent is the root of the document when the content of a component is edited in the TGML graphics editor.

ComponentContent is replaced with Component when the component is stored in the library.

ComponentContent has the following attributes:

- Height
- Opacity
- Visibility
- Width

Width and Height are replaced with (copied to) ContentWidth and ContentHeight of the Component element when the component is stored.

Initial Viewport

Width and Height of the stored Component are the initial viewport, that is, the initial size of the component when it is pasted into a TGML document.

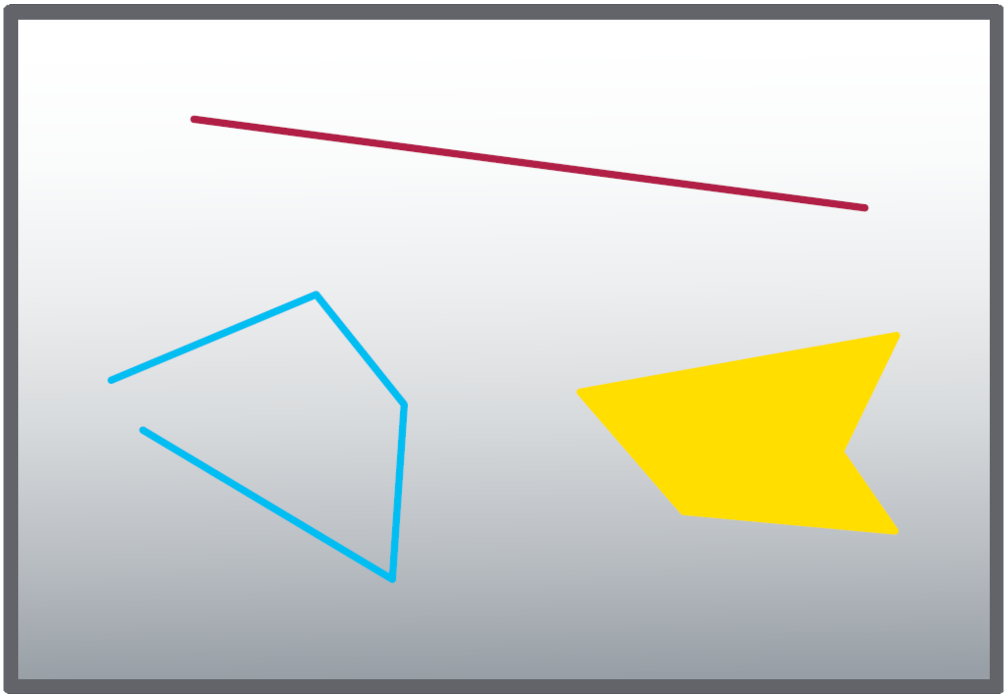
TGML Document Structure

A TGML Graphics document always consists of at least one element, the root ("outermost") TGML element. This TGML element is the ancestor of all other elements in the document.

Layer and Group are examples of grouping (container) elements that are used to structure the graphics. Elements that describe shapes and other graphical (visible) objects can also have children, but these children are not visible objects, but elements that add a specific behavior, such as gradients, animations and dynamic bindings. The example below shows TGML code with the TGML root element and a number of child elements:

```
<TGML Width="400" Height="250" Stretch="Uniform" >
    <Layer Name="Background">
        <Rectangle Left="0.0" Top="0.0" Width="400.0" Height="250.0" Fill="None"
Stroke-"None">
            <LinearGradient Attribute="Fill
" EndPoint="0.0,1.0" SpreadMethod="Pad" StartPoint="0.0,0.0">
                <GradientStop Color="#FFFFFF" Offset="0.0"/>
                <GradientStop Color="#803080" Offset="l.0"/>
            </LinearGradient>
        </Rectangle>
    </Layer>
    <Layer Name="Foreground">
        <Group>
            <Line
Xl="75.0" Yl="50.0" X2="325.0" Y2="75.0" Stroke="#FF0000" StrokeWidth="2.0"/>
            <Polyline Points="50.0,100.0 150.0,75.0 175.0,125.0 150.0,200.0
75.0,125.0" Stroke="#0000FF"/>
            <Polygon Points="200.0,125.0 350.0,100.0 325.0,150.0 350.0,200.0
250.0,175.0" Stroke="#00FF00" Fill="#FFFF00" />
        </Group>
    </Layer>
</TGML>
```

The graphic as seen in Graphics Editor:



TGML Scripting

TGML scripting supports a number of event methods:

- TGML About Data Types
- TGML Common Event Methods
- TGML Mouse Event Methods
- TGML SignalChange Event Methods
- TGML DOM Methods
- TGML Standard DOM Methods (Commonly Used)
- TGML JavaScript Functions

TGML About Data Types

JavaScript variables are untyped. For example, if you use global variables or custom attributes to store numeric values:

Convert the variables to a Number:

```
<TGML>
<Rectangle
Step
="1"

Fill
="#FFFFFFF" Height="50.0" Left="50.0" Stroke="#000000" Top="50.0" Width="50.0">
<Bind Name="Value"/>
<Script OnMouseClick="click" OnSignalChange="signal"><![CDATA [

// Global variable, initialized with Not a Number.
value = NaN;

function signal(evt)
{
    value = new Number{evt.getValue());
    //Note: getValue may return a string,
}

function click(evt)
{
    // If value is a numeric value (not a NaN).
    if(!isNaN(value))
    {
        var target = evt.getTarget();
        //Convert the custom attribute step to a Number
        //stored as string)
        var increase = new Number(target.setAttribute("Step"|);
        // Increase the value
        var newValue = value + increase;
        // Get the name of the binding
        bind = target.get Child("Value");
        var fullName = bind.getFullBindName();
        //set the new value
        setValue(fullName, newValue);
    }
    else
    {
        alert("Not a numeric value.")
    }
    //Do not display any "change value" dialogs when clicking
    evt.preventDefault();
}

}]]></Script>
</Rectangle>
</TGML>
```

TGML Common Event Methods

An overview of the methods that are common for all event objects.

Method	Description
getCurrentTarget()	Returns the element which the Script belongs to (i.e. the EventTarget whose EventListeners are currently being processed.)
getTarget()	Returns the element to which the event was originally dispatched (for example, the element you clicked on).
Redundancy Client	If the event is cancelable, preventDefault cancels the default action normally taken by the implementation (e.g. the viewer, see Remarks below).
stopPropagation()	Prevents further propagation of an event.

Remarks

When an element contains Bind or Link, the viewer is supposed to respond (e.g. show a "change value" dialog or open the linked presentation object) when the user clicks on the element. This is the "default action" for the viewer which is canceled by the preventDefault function.

preventDefault in an OnMouseClick function cancels the change value dialog (when the element contains a Bind) or the link function (when the element contains a Link).

preventDefault in an OnSignalChange function cancels the error indication (the red cross).

Example of a common event method:

```
<TGML>
    <Component
Left
="99.5"

Top
="99.5" Width="101.0" Height="101.0" ContentHeight="101.0" ContentWidth="101.0">
```

```
    <Script OnMouseDown="down"><![CDATA [
        function down(evt)
        {
            // The Rectangle will be the target since the Component
            // has no painted (clickable) surface
            var rectangle = evt.getTarget();

            // The Component is the current target because
            // it is the immediate parent of
            // the Script (i.e. the executed event listener)
            var component = evt.getCurrentTarget();
        }
    ]]></Script>

    <Rectangle
Left
="0.5" Top="0.5" Width="100.0" Height="100.0" Fill="#FFFFFF" Stroke="#000000"/>

    </Component>
</TGML>
```

TGML Mouse Event Methods

The following table contains an overview of the methods that are specific for the mouse event object:

Method	Description
getButton()	Returns an integer describing which button was pressed or released. Applicable for MouseDownEvent and MouseUpEvent. 0 = Left button 1 = Middle button 2 = Right button
getClientX()	Returns the X coordinate of the cursor, relative the origin of the target coordinate system. The coordinate is calculated using the transformations of the target element.
getClientY()	Returns the Y coordinate of the cursor, relative the origin of the target coordinate system. The coordinate is calculated using the transformations of the target element.
getCurrentTargetX()	Returns the X coordinate of the cursor, relative the origin of the current target coordinate system. The coordinate is calculated using the transformations of the current target element.

Method	Description
getCurrentTargetY()	Returns the Y coordinate of the cursor, relative the origin of the current target coordinate system. The coordinate is calculated using the transformations of the current target element.
getCurrentTargetParentX()	Returns the X coordinate of the cursor, relative the origin of the current target's parent coordinate system. The coordinate is calculated using the transformations of the current target's parent.
getCurrentTargetParentY()	Returns the Y coordinate of the cursor, relative the origin of the current target's parent coordinate system. The coordinate is calculated using the transformations of the current target's parent.
getScreenX()	Returns the X coordinate of the cursor, relative to the origin of the document coordinate system.
getScreenY()	Returns the Y coordinate of the cursor, relative to the origin of the document coordinate system.

TGML SignalChange Event Methods

The following table contains an overview of the methods that are specific for the SignalChange event object:

Method	Description
getStatus()	Returns the status of the signal: 0: Error (Bad quality) 1: Stored value (Uncertain quality) 2: Real value (Good quality) 3: Forced value (Good quality)
getPresentationValue()	Returns the value of the bound signal as a "presentation value". For more information, see section 33.12 “TGML Signal Binding: <Bind>” on page XXX.
getUnit()	Returns the unit of the bound signal as a string.
getValue()	Returns the value of the bound signal.

TGML DOM Methods

An overview of the methods that are specific for the TGML DOM methods that do not exist in standard DOM implementations, that is, methods unique for TGML.

For more information, see the W3C Document Object Model (DOM) Level 3 Core Specification.

Attribute	Type	Description
Any element	**getChildByName ("‹name›")** Obsolete. This function may be removed in a future release.	Returns the child element that has the Name attribute with the given value. If no such element exists, this returns null. If more than one element has a Name attribute with that value, what is returned is undefined.
Any element	**getChild ("‹name›")**	Returns the immediate child element that has the Name attribute with the given value. If no such child exists, this returns null. If more than one immediate child element has a Name attribute with that value, what is returned is undefined.
Any element	**getChildRecursive ("‹name›")**	Returns the child element at any level that has the Name attribute with the given value. If no such element exists, this returns null. If more than one element has a Name attribute with that value, what is returned is undefined.
Bind or Link	**getFullBindName()**	Returns the exposed name of the Bind or Link element, including names of parent components.

Example:

```
<TGML>
    <Component Name="MyComponent" Left="50.0" Top="50.0 Width="100.0"
Height="100.0" ContentHeight="100.0" ContentWidth=100.0>

        <Script OnMouseClick="click"><![CDATA [
            function click(evt)
            {
                var rectangle = evt.getTarget();
                var bind = rectangle.getChild("Value");
                var name = bind.getFullBindName(); //"MyComponent.Value"
                ....
            }
        ]]></Script>

        <Rectangle Left="0.0" Top="0.0 Width="100.0" Height="100.0"
Fill="#FFFFFF" Stroke=#000000">
            <Bind Name="Value" Attribute="Fill" />
        </Rectangle>

    </Component>
</TGML>
```

TGML Standard DOM Methods (Commonly Used)

An overview of standard DOM methods that can be used to access graphics elements and attributes.

See W3C Document Object Model (DOM) Level 3 Core Specification for more information.

Object	Method	Description
Any element	**getAttribute("‹attribute›")**	Returns the value of ‹attribute›.
Any element	**getChildNodes()**	Returns a NodeList that contains all children of this node.
Any element	**getOwnerDocument()**	Returns the document.
Any element	**getParentNode()**	Returns the parent element.
Any element	**getTagName()**	Returns the element tag name, e.g. Rectangle or Bind.
Any element	**setAttribute ("‹attribute›", "‹value›")**	Sets the value of ‹attribute› to ‹value›. If the element does not have the attribute, it is created (as a TGML custom attribute).
Document	**getDocumentElement()**	Returns the TGML (root) element.
Document	**getElementById("‹Id›")**	Returns the child element that has the Id attribute with the given value.
Document	**getElementsByTagName ("‹tagName›")**	Returns a NodeList of all the elements in document order with a given tag name.
NodeList	**getLength()**	Returns the number of elements in the NodeList.
NodeList	**item("‹index›")**	Returns the element at ‹index› in the NodeList.

Example:

```
<TGML>
    <Rectangle Left="50" Top="50 Width="100" Height="100" Fill="#C0C0C0"
Stroke="#000000">
        <Script OnMouseOver="over" OnMouseOut="out"><![CDATA [
            function over(evt)
            {
                // Change fill color hile hovering
                var rectangle = evt.getTarget();

                // Get the original fill color and store it
                // as a custom attribute of the rectangle
                var color = rectangle.getAttribute("Fill");
                rectangle.setAttribute("originalColor", color);

                // Change the color to blue
                rectangle.setAttribute("Fill", "#0000FF");
            }

            function out(evt)
            {
                // Restore the fill color
                var rectangle = evt.getTarget();
                var color = rectangle.getAttribute("originalColor");
                rectanle.setAttribute("Fill", color);
            }
        ]]></Script>

    </Rectangle>
</TGML>
```

TGML JavaScript Functions

An overview of the TGML JavaScript functions unique for TGML.

Method	Description
alert("‹message›")	Displays a message box.
clearInterval(intervalID) *Obsolete. This function is replaced by JavaScript standard implementation.	Cancels the interval previously started using the setInterval function. intervalID is an identifier returned by a previous call to the setInterval function.
clearTimeout(timeoutID) *Obsolete. This function is replaced by JavaScript standard implementation.	Cancels a time-out that was set with the setTimeout function. timeoutID is an identifier returned by a previous call to the setTimeout function.
confirm("‹message›")	Displays a confirm box with "Yes" and "No" buttons. Returns true if the user clicks "Yes" or false if the user clicks "No".

execute("‹command›")
execute("‹command›", "‹options›")
	Requests an execute operation to be performed by the TGML viewer (start a Windows program).
	command is the name of the program (full path) and options is the command line options.
	Returns true if succeeded or false if failed.
	NOTE: The implementation of this function is system dependent. May not be implemented in some systems.

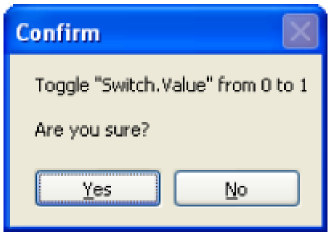
Method	Description
invoke("‹bindingName›", "‹operation›")	Requests an operation to be performed on a bound object by the TGML viewer. The bindingName is the full name (as it is exposed to the binding tools) of a Bind or Link element. Returns true if succeeded or false if failed. **NOTE**: The implementation of this function is system dependent. May not be implemented in some systems.
openFile("‹path›", "‹operation›")	Requests the TGML viewer to open a file. The operation is typical Windows object verbs. Returns true if succeeded or false if failed. **NOTE**: The implementation of this function is system dependent. May not be implemented in some systems.
prompt("‹message›", "‹defaultValue›")	Prompts the user to enter a value. Returns the entered value or null if canceled.
setForce ("‹bindingName›", "true|false")	Sets the force state of a bound signal object. The bindingName is the full name (as it is exposed to the binding tools) of a Bind element. Returns true if succeeded or false if failed. **NOTE**: The implementation of this function is system dependent. May not be implemented in some systems.
setInterval ("‹expression›", "‹milliseconds›") *Obsolete. This function is replaced by JavaScript standard implementation.	Evaluates (executes) the expression each time the specified number of milliseconds has elapsed. Returns an identifier that cancels the timer with the clearInterval method.
setTimeout ("‹expression›", "‹milliseconds›") *Obsolete. This function is replaced by JavaScript standard implementation.	Evaluates (executes) the expression after the specified number of milliseconds has elapsed. Returns an identifier that cancels the timer with the clearTimeout method.
setValue ("‹bindingName›", "‹value›")	Sets the value of a bound signal object. The bindingName is the full name (as it is exposed to the binding tools) of a Bind element. Returns true if succeeded or false if failed. **NOTE**: The implementation of this function is system dependent. May not be implemented in some systems.

*This is valid for the TGML specific implementations and may be removed in future releases. Made obsolete in favor of the JavaScript standard implementation.

TGML JavaScript Functions - Example 1

This is an example of how to create an interactive rectangle that toggles a value.

```
<TGML>
   <Component
Name="Switch" Left="50" Top="50" Width="102" Height="102" ContentHeight="102" ContentWidth="102">

   <Rectangle Left="l" Top="l" Width="100.0" Height="100.0" Fill="#C0C0C0" Stroke="#000000">

      <Script OnMouseClick="click"
      OnSignalChange="signal"><![CDATA [

      currentValue = NaN;

      function click(evt)
      {
         var rect = evt.getTarget();

         var oldVal = new Number(currentValue);
         // Note: currentValue is set in function signal.

         if (!isNaN(oldVal)) // Skip if no value has arrived
         {
         // Toggle
         var newVal = (oldVal==0) ? 1 : 0;

         // Get the Bind element (named Value)
         var bind = rect.getChild("Value");

         // Get the full binding name
         var fullName = bind.getFulIBindName();

         var toggle = confirm("Toggle \"" + fullName + "\" from " + oldVal + " to " + newVal +
"\n\nAre you sure?");

         if(toggle == true)
         setValue(fullName, newVal);
      }

      // The rect contains a Bind element. Cancel the
      // default viewer action (do not display the
      // default "edit value" dialog).
      evt.preventDefault();
      )

      function signal(evt)
      {
         var currentValue = evt.getValue ();
      }
      ] ]></Script>

      <Bind Attribute="Fill" Name="Value">
         <ConvertValue AttributeValue="#FF0000" SignalEqualTo="0"/>
         <ConvertValue AttributeValue="#00FF00" SignalEqualTo="l"/>
      </Bind>

   </Rectangle>
   </Component>
</TGML>
```



Toggle "Switch.Value" from 0 to 1

Are you sure?

TGML JavaScript Functions - Example 2

The example below assumes that the Link named "Report" is bound to a TACOS report object. The function invoke will result in a "print report" operation in Diagrams.

```
<TGML>
    <Text Left="100" Top="100">
    Print TACOS report

        <Script OnMouseClick="click"><![CDATA [
            function click(evt)
            (
                var rect = evt.getTarget();
                var link = rect.getChild("Report");
                var fullName = link.getFullBindName();

                // Print the linked report
                invoke(fullName, "PrintReport");

                // The text contains a Link. Cancel the default
                // Link operation
                evt .preventDefault();
            }
        ]] ></Script>

        <Link Name="Report"/>

    </Text>
</TGML>
```

TGML JavaScript Functions - Example 3

This example starts an interval timer and animates (toggles) the fill color when the cursor is over the rectangle. The timer is stopped and the color is restored when the cursor leaves the rectangle.

```
<TGML>
    <Rectangle
Fill="#FFFFFF" Height="100.0" Left="100.0" Stroke="#000000" Top="100.0" Width="200.0">
        <Script OnMouseOut="out" 0nMouseOver="over"><![CDATA [

        var intervalID;
        var rectangle;
        var originalColor;
        var animatedColor;
        function over(evt)
        {
            // Store the rectanle element and the original color
            rectangle = evt.getTarget();
            originalColor = rectangle.getAttribute("Fill");

            // Initialize the animated color and set this color
            animatedColor = "#FF0000";
            rectangle.setAttribute("Fill", animatedColor);

            // Start the timer
            intervallD = setlnterval(toggleColor, 500);
        }

        function out (evt)
        {
            // Stop the timer and restore the color
            clearlnterval(intervallD);
            rectangle.setAttribute("Fill", originalColor);
        }

        function toggleColor()
        {
            if(animatedColor == "#FF0000")
            animatedColor = "#0000FF";
            else
            animatedColor = "#FF00Q0";
            rectangle.setAttribute("Fill", animatedColor);
        }

        ] ]></Script>
    </Rectangle>
</TGML>
```

TGML JavaScript Functions - Example 4

This example displays an alert box when the cursor has been over the rectangle for one second. The timer is stopped when the cursor leaves the rectangle.

```
<TGML>
    <Rectangle
Fill="#FFFFFF" Height="100.0" Left="100.0" Stroke="#000000" Top="100.0" Width="200.0">

        <Script OnMouseOut="out" 0nMouseOver="over"><! [CDATA [

        var timeoutID;

        function over(evt)
        {
            // Start the timer.
            // Evaluate "alert('Hovering')" after 1 second
            timeoutID = setTimeout("alert('Hovering')", "1000")
        }

        function out(evt)
        {
            // Stop the timer (abort)
            clearTimeout(timeoutID);
        }

        ]]></Script>

    </Rectangle>
</TGML>
```

Basic TGML Elements

Basic TGML elements descriptions:

- TGML Document Type Element and Metadata
- TGML Grouping Elements
- TGML Basic Shapes
- TGML Segment Shapes
- TGML Curves and Paths
- TGML Raster Images
- TGML Text
- TGML Gradients

TGML Document Type Element and Metadata

Each TGML document contains the TGML root element. It also contains metadata created and interpreted by the TGML application.

Document Type Element

The root element of a TGML document is <TGML>. This element specifies that the document type is TGML.

Attribute	Type	Description
Background	Brush	The background color of the document canvas (the viewing area). **Default: "#FFFFFF"** **Inheritable:** No **Animatable:** Yes
DisablePanAndZoom	Boolean	Disables the normal zoom and pan commands in the viewer. For example useful in kiosk mode or headers/footers. **Default:** "False" **Inheritable:** No **Animatable:** No

Attribute	Type	Description
Height	Double	The height of the document. See Remarks. **Default:** "600" **Inheritable:** No **Animatable:** No
Stretch	Stretch	Specifies how the document shall be stretched initially within a viewer. See Remarks. **Default:** "None" **Inheritable:** No **Animatable:** No
UseGlobalScripts	Boolean	Enable a single execution context for scripts. For more information, see the TGML Script Context section. **Default:** "False" **Inheritable:** No **Animatable:** No
Width	Double	The width of the document. See Remarks. **Default:** "800" **Inheritable:** No **Animatable:** No

Remarks

The viewer uses the width, height, and stretch information to determine how the document initially is stretched.

The viewer can display a document where information about width and height is missing. When you view such a document no stretching is applied and any scroll bars are disabled.

Stretch="None": Preserve the original size. This usually means that scroll bars are enabled so the user can scroll to the right and the bottom of the document.

Stretch="Uniform": Resize the content, preserving the native aspect ratio. Scroll bars are initially disabled.

Stretch="Fill": The content is resized but aspect ratio is not preserved. Scroll bars are initially disabled.

Metadata

The <Metadata> element carries textual information about its parent element. It is up to the TGML application to create and interpret the metadata.

Applying metadata to the outermost TGML element is the same as applying the information to the TGML document.

Attribute	Type	Description
Name	String	A name that identifies the information. **Inheritable:** No **Animatable:** No
Value	String	The information. **Inheritable:** No **Animatable:** No

TGML Grouping Elements

With TGML there are several ways to collect elements in a common container element:

- TGML Grouping: <Group>
- TGML Components: <Component>
- TGML Layers: <Layer>

TGML Grouping: <Group>

The Group element is a container element, used for grouping elements together so they can, for example, be moved, copied and resized as if they were a single element.

Attribute	Type	Description
Opacity	Double	A value between 0.0 (transparent) and 1.0 (opaque). **Default:** "1.0" **Inheritable:** No **Animatable:** Yes
Visibility	Visibility	Specifies if the element shall be visible or not. **Default:** "Visible" **Inheritable:** No **Animatable:** Yes

Example:

```
<TGML>
    <Group Opacity="0.5">
        <Line X1="50.0" Y1="50.0" X2="150.0" Y2="150.0” />
        <Line X1="50.0" Y1="100.0" X2="150.0" Y2="200.0" />
    </Group>
</TGML>
```

TGML Components: <Component>

The Components element is a container element (similar to Group) which defines a reusable group of elements.

Top and Left of Component specifies the position of the component in the parent coordinate system. Component itself establishes a new coordinate system for the contained elements, which means that the upper left corner of Component is the origin (0,0) for the contained elements.

Width and Height specifies the "viewport" (size on screen) of the component. ContentWidth and ContentHeight specify the "viewbox", which is the boundary of the contained elements. If the viewport is different from the viewbox, a scale transformation is applied by the TGML implementation (similar to Stretch="Fill"). For more information, see TGML Document Type Element and Metadata. In other words, resizing a Component has the effect of scaling the content of the Component.

The Clip attribute specifies if the renderer shall clip elements that exceed (are drawn outside) the specified viewbox or not.

When a Component includes Bind or Link elements, the Name of Component will prefix the exposed bind names ("MyComponent.MyBind").

For more information, see TGML Components.

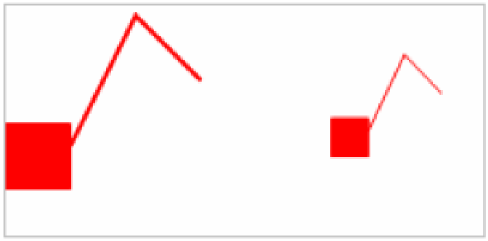
Attribute	Type	Description
Clip	Bool	Specifies if the content shall be clipped or not. **Default:** "True" **Inheritable:** No **Animatable:** No
ContentHeight	Double	Specifies the viewbox height (height of the content). **Inheritable:** No **Animatable:** No
ContentWidth	Double	Specifies the viewbox width (width of the content). **Inheritable:** No **Animatable:** No
Height	Double	The viewport height of the component (height on screen). **Inheritable:** No **Animatable:** Yes
Left	Double	The x coordinate of the component's upper left corner. **Default:** "0" **Inheritable:** No **Animatable:** Yes
Opacity	Double	A value between 0.0 (transparent) and 1.0 (opaque). **Default:** "1.0" **Inheritable:** No **Animatable:** Yes

Attribute	Type	Description
Top	Double	The y coordinate of the components upper left corner. **Default:** "0" **Inheritable:** No **Animatable:** Yes
Visibility	Visibility	Specifies if the element shall be visible or not. **Default:** "Visible" **Inheritable:** No **Animatable:** Yes
Width	Double	The viewport width of the component (width on screen). **Inheritable:** No **Animatable:** Yes

Example:

```
<TGML>
    <Rectangle
Left="0.0" Top="55" Width="30" Height="30" Fill="#FFOO00" Stroke="#FF0000" />
    <Polyline Points="30.0,65.0 60.0,5.0
90.0,35.0" Stroke="#FF0000" StrokeWidth="2.0"/>

    <Component
Left
="150.0"
 Top="20.0" Width="50.0" Height="50.0" ContentWidth="91" ContentHeight="86">
        <Rectangle
Left="0.0" Top="55" Width="30" Height="30" Fill="#FF0000" Stroke="#FF0000" />
        <Polyline Points="30.0,65.0 60.0,5.0
90.0,35.0" Stroke="#FFQOOO" StrokeWidth="2.0"/>
    </Component>
</TGML>
```

Example on screen:



TGML Layers: <Layer>

The Layer element is a container element used to create layered TGML graphics.

Only the TGML root element can contain Layer elements, which means that nested layers are not supported.

Layer elements cannot be transformed. Child transformation elements will have no effect on the Layer element.

Attribute	Type	Description
Opacity	Double	A value between 0.0 (transparent) and 1.0 (opaque). **Default:** "0" **Inheritable:** No **Animatable:** Yes
Visibility	Visibility	Specifies if the element is visible or not. **Default:** "Visible" **Inheritable:** No **Animatable:** Yes

Example:

```
<TGML Width="400" Height="250">

    <Layer Name="Background">
        <Image Left="0" Tcp="0" Width*"400" Height="250">
            < ! [CDATA[iVBORw0KGgoAAAANSUh...SWORK5CYII=] ]>
        </Image>
    </Layer>

    <Layer Name="Foreground">
        <Line Xl="75.0" Yl="50.0" X2="325.0" Y2="75.0” Stroke="#FF0000"
StrokeWidth="2.0"/>
        <Polyline Points="50.0,100.0 150.0,75.0 175.0,125.0 150.0,200.0
75.0,125.0" Stroke="#0000FF"/>
        <Polygon Points="200.0,125.0 350.0,100.0 325.0,150.0 350.0,200.0
250.0,175.0" Stroke="#00FF00" Fill="#FFFF00" />
    </Layer>

</TGML>
```

TGML Basic Shapes

The TGML specification contains a number of basic shapes:

- TGML Line
- TGML Polyline
- TGML Polygon
- TGML Rectangle
- TGML Ellipse

Shape Attributes

The following table describes the common attributes of the basic shapes.

Attribute	Type	Description
Fill	Brush	Specifies how the interior of the shape is painted. **Default: "None"** **Inheritable:** Yes **Animatable:** Yes
Opacity	Double	A value between "0.0" (transparent) and "1.0" (opaque) **Default:** "1.0" **Inheritable:** No **Animatable:** Yes
Stroke	Brush	Describes how the line is painted. **Default: "#000000"** **Inheritable:** No **Animatable:** Yes
StrokeDashArray	Array of Double	The pattern of dashes and gaps used to outline shapes: "<dash> [<gap> <dash> <gap>...]" If the array only specifies the first dash, the line is patterned as if a gap with the same length as the dash was specified. An array with only one dash set to "0" will result in a line without any pattern. **Default:** "1" **Inheritable:** Yes **Animatable:** Yes
StrokeWidth	Double	The width of the outline of a line. **Default:** "0" **Inheritable:** No **Animatable:** Yes
Visibility	Visibility	Specifies if the element shall be visible or not. **Default:** "Visible" **Inheritable:** No **Animatable:** Yes

TGML Line

The <Line> element describes a straight line between two points.

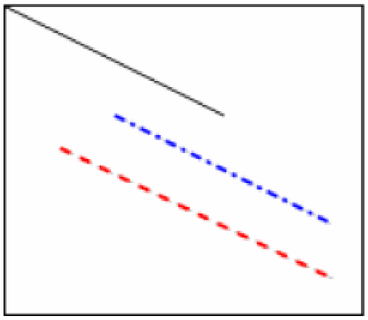
Attribute	Type	Description
Opacity	Double	A value between "0.0" (transparent) and "1.0" (opaque) **Default:** "1.0" **Inheritable:** No **Animatable:** Yes
Stroke	Brush	Describes how the line is painted. **Default: "#000000"** **Inheritable:** Yes **Animatable:** Yes
StrokeDashArray	Array of Double	The pattern of dashes and gaps used to outline shapes: "<dash> [<gap> <dash> <gap>...]" If the array only specifies the first dash, the line is patterned as if a gap with the same length as the dash was specified. An array with only one dash set to "0" will result in a line without any pattern. **Default:** "0" **Inheritable:** Yes **Animatable:** Yes
StrokeWidth	Double	The width of the outline of a line. **Default:** "1" **Inheritable:** Yes **Animatable:** Yes
Visibility	Visibility	Specifies if the element shall be visible or not. **Default:** "Visible" **Inheritable:** No **Animatable:** Yes
X1	Double	The x coordinate of the line start point. **Default:"0"** **Inheritable:** No **Animatable:** Yes
X2	Double	The x coordinate of the line end point. **Default:"0"** **Inheritable:** No **Animatable:** Yes

Attribute	Type	Description
Y1	Double	The y coordinate of the line start point. **Default:"0"** **Inheritable:** No **Animatable:** Yes
Y2	Double	The y coordinate of the line end point. **Default:"0"** **Inheritable:** No **Animatable:** Yes

Example:

```
<TGML>

    <Line X2="100.0" Y2="50.0"/>

    <Line
Xl
="50.0"

Yl
="50.0"
 X2="150.0" Y2="100.0" Stroke="#0000FF" StrokeWidth="2" StrokeDashArray="5.0 3.0
2.0 3.0"/>

    <Line Xl="25.0" Yl="65.0" X2="150.0" Y2=”125.0" Stroke="#FF0000"
StrokeWidth="2" StrokeDashArray="5.0"/>

</TGML>
```

Example on screen:



TGML Polyline

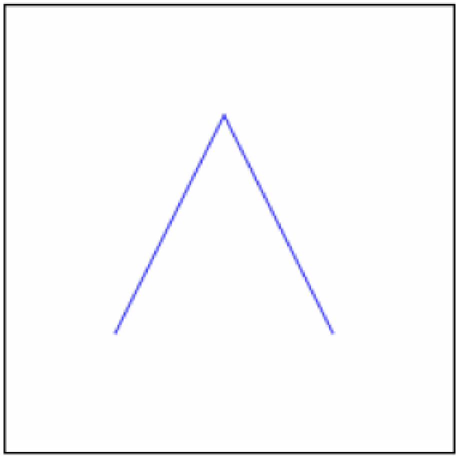
The <Polyline> element describes a series of connected straight lines.

Attribute	Type	Description
Fill	Brush	Specifies how the interior of the shape is painted. **Default: "None"** **Inheritable:** Yes **Animatable:** Yes
Opacity	Double	A value between "0.0" (transparent) and "1.0" (opaque) **Default:** "1.0" **Inheritable:** No **Animatable:** Yes
Points	Array of Point	The vertex points of the polyline: "<x1>,<y1> <x2>,<y2>..." **Inheritable:** No **Animatable:** No
Stroke	Brush	Describes how the line is painted. **Default: "#000000"** **Inheritable:** Yes **Animatable:** Yes
StrokeDashArray	Array of Double	The pattern of dashes and gaps used to outline shapes: "<dash> [<gap> <dash> <gap>...]" If the array only specifies the first dash, the line is patterned as if a gap with the same length as the dash was specified. An array with only one dash set to "0" will result in a line without any pattern. **Default:** "0" **Inheritable:** Yes **Animatable:** Yes
StrokeWidth	Double	The width of the outline of a line. **Default:** "1" **Inheritable:** Yes **Animatable:** Yes
Visibility	Visibility	Specifies if the element is visible or not. **Default:** "Visible" **Inheritable:** No **Animatable:** Yes

Example:

```
<TGML>
    <Polyline Points="50.0,150.0 100.0,50.0 150.0,150.0" Stroke="#0000FF"/>
</TGML>
```

Example on screen:



TGML Polygon

The <Polygon> element describes a polygon, which is a connected series of lines that forms a closed shape. The end point does not have to be specified. The polygon is closed automatically.

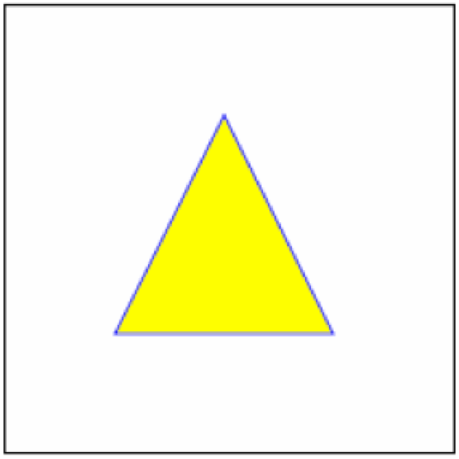
Attribute	Type	Description
Fill	Brush	Specifies how the interior of the shape is painted. **Default: "None"** **Inheritable:** Yes **Animatable:** Yes
Opacity	Double	A value between "0.0" (transparent) and "1.0" (opaque) **Default:** "1.0" **Inheritable:** No **Animatable:** Yes
Points	Array of Point	The vertex points of the polyline: "<x1>,<y1> <x2>,<y2>..." **Inheritable:** No **Animatable:** No
Stroke	Brush	Describes how the line is painted. **Default: "#000000"** **Inheritable:** Yes **Animatable:** Yes

Attribute	Type	Description
StrokeDashArray	Array of Double	The pattern of dashes and gaps used to outline shapes: "<dash> [<gap> <dash> <gap>...]" If the array only specifies the first dash, the line is patterned as if a gap with the same length as the dash was specified. An array with only one dash set to "0" will result in a line without any pattern. **Default:** "0" **Inheritable:** Yes **Animatable:** Yes
StrokeWidth	Double	The width of the outline of a line. **Default:** "1" **Inheritable:** Yes **Animatable:** Yes
Visibility	Visibility	Specifies if the element is visible or not. **Default:** "Visible" **Inheritable:** No **Animatable:** Yes

Example:

```
<TGML>
    <Polygon Points="50.0,150.0 100.0,50.0
150.0,150.0" Stroke="#0000FF" Fill="#FFFF00"/>
</TGML>
```

Example on screen:



TGML Rectangle

The <Rectangle> element defines a rectangle. You can create rounded rectangles by setting values for the attributes RadiusX and RadiusY.

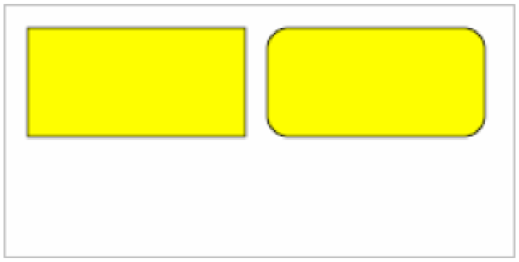
Attribute	Type	Description
Fill	Brush	Specifies how the interior of the shape is painted. **Default: "None"** **Inheritable:** Yes **Animatable:** Yes
Height	Double	The height of the rectangle. **Inheritable:** No **Animatable:** Yes
Left	Double	The x coordinate of the upper left corner of the rectangle. **Default:** "0" **Inheritable:** No **Animatable:** Yes
Opacity	Double	A value between "0.0" (transparent) and "1.0" (opaque) **Default:** "1.0" **Inheritable:** No **Animatable:** Yes
RadiusX	Double	For rounded rectangles. The x axis radius of the ellipse used to round off the corners of the rectangle. **Default:** "0" **Inheritable:** No **Animatable:** Yes
RadiusY	Double	For rounded rectangles. The y axis radius of the ellipse used to round off the corners of the rectangle. **Default:** "0" **Inheritable:** No **Animatable:** Yes
Stroke	Brush	Describes how the line is painted. **Default: "#000000"** **Inheritable:** Yes **Animatable:** Yes

Attribute	Type	Description
StrokeDashArray	Array of Double	The pattern of dashes and gaps used to outline shapes: "<dash> [<gap> <dash> <gap>...]" If the array only specifies the first dash, the line is patterned as if a gap with the same length as the dash was specified. An array with only one dash set to "0" will result in a line without any pattern. **Default:** "0" **Inheritable:** Yes **Animatable:** Yes
StrokeWidth	Double	The width of the outline of a line. **Default:** "1" **Inheritable:** Yes **Animatable:** Yes
Top	Double	The y coordinate of the upper left corner of the rectangle. **Default:** "0" **Inheritable:** No **Animatable:** Yes
Visibility	Visibility	Specifies if the element is visible or not. **Default:** "Visible" **Inheritable:** No **Animatable:** Yes
Width	Double	The width of the rectangle. **Inheritable:** No **Animatable:** Yes

Example:

```
<TGML>
    <Rectangle
Left="10" Top="10" Width="100" Height="50" Fill="#FFFF00" Stroke="#000000"/>
    <Rectangle
Left
="120"

Top
="10"

Width
="100" Height="50" Fill="#FFFF00" Stroke="#000000" RadiusX="10" RadiusY="10"/>
</TGML>
```

Example on screen:



TGML Ellipse

The <Ellipse> element defines an ellipse.

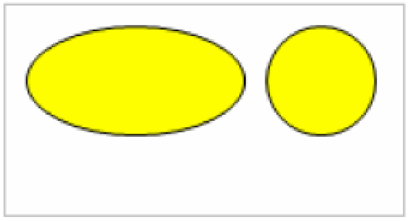
Attribute	Type	Description
Fill	Brush	Specifies how the interior of the shape is painted. **Default: "None"** **Inheritable:** Yes **Animatable:** Yes
Height	Double	The height of the ellipse. **Inheritable:** No **Animatable:** Yes
Left	Double	The x coordinate of the upper left corner of the ellipse. **Default:** "0" **Inheritable:** No **Animatable:** Yes
Opacity	Double	A value between "0.0" (transparent) and "1.0" (opaque) **Default:** "1.0" **Inheritable:** No **Animatable:** Yes
Stroke	Brush	Describes how the line is painted. **Default: "#000000"** **Inheritable:** Yes **Animatable:** Yes

Attribute	Type	Description
StrokeDashArray	Array of Double	The pattern of dashes and gaps used to outline shapes: "<dash> [<gap> <dash> <gap>...]" If the array only specifies the first dash, the line is patterned as if a gap with the same length as the dash was specified. An array with only one dash set to "0" will result in a line without any pattern. **Default:** "0" **Inheritable:** Yes **Animatable:** Yes
StrokeWidth	Double	The width of the outline of a line. **Default:** "1" **Inheritable:** Yes **Animatable:** Yes
Top	Double	The y coordinate of the upper left corner of the ellipse. **Default:** "0" **Inheritable:** No **Animatable:** Yes
Visibility	Visibility	Specifies if the element is visible or not. **Default:** "Visible" **Inheritable:** No **Animatable:** Yes
Width	Double	The width of the ellipse. **Inheritable:** No **Animatable:** Yes

Example:

```
<TGML>

    <Ellipse
Left="10" Top="10" Width="100" Height="50" Fill="#FFFF00" Stroke="#000000"/>

    <Ellipse
Left="120" Top="10" Width="50" Height="50" Fill="#FFFF00" Stroke="#000000"/>

</TGML>
```

Example on screen:



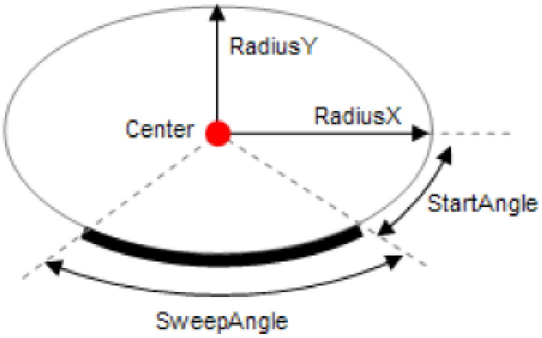
TGML Segment Shapes

The TGML specification defines a number of shapes that are segments of an ellipse:

- TGML Elliptical Arc: <Arc>
- TGML Elliptical Pie: <Pie>
- TGML Elliptical Chord: <Chord>

TGML Elliptical Arc: <Arc>

Arc defines an elliptical arc. The elliptical arc is part of an ellipse.



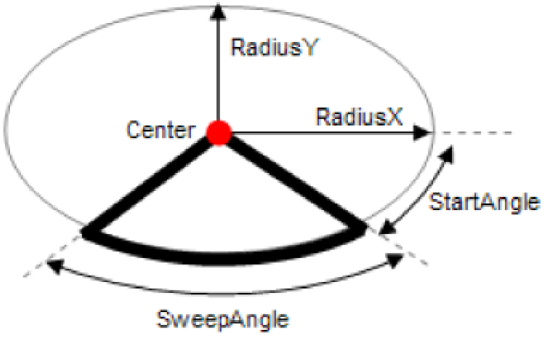
Attribute	Type	Description
Center	Point	The center point of the ellipse. **Default:** "0.0 , 0.0" **Inheritable:** No **Animatable:** Yes
Fill	Brush	Specifies how the interior of the shape is painted. **Default: "None"** **Inheritable:** Yes **Animatable:** Yes
Opacity	Double	A value between "0.0" (transparent) and "1.0" (opaque) **Default:** "1.0" **Inheritable:** No **Animatable:** Yes

Attribute	Type	Description
Left	Double	The x coordinate of the upper left corner of the ellipse. **Default:** "0" **Inheritable:** No **Animatable:** Yes
RadiusX	Double	The horizontal radius of the ellipse. **Inheritable:** No **Animatable:** Yes
RadiusY	Double	The vertical radius of the ellipse. **Inheritable:** No **Animatable:** Yes
StartAngle	Double	Specifies the angle between the x axis and the starting point of the arc. **Default:**"0.0" **Inheritable:** No **Animatable:** Yes
SweepAngle	Double	Specifies the angle between the starting and ending points of the arc. **Default:**"0.0" **Inheritable:** No **Animatable:** Yes
Stroke	Brush	Describes how the line is painted. **Default:"#000000"** **Inheritable:** Yes **Animatable:** Yes
StrokeDashArray	Array of Double	The pattern of dashes and gaps used to outline shapes: "<dash> [<gap> <dash> <gap>...]" If the array only specifies the first dash, the line is patterned as if a gap with the same length as the dash was specified. An array with only one dash set to "0" will result in a line without any pattern. **Default:** "0" **Inheritable:** Yes **Animatable:** Yes

Attribute	Type	Description
StrokeWidth	Double	The width of the outline of a line. **Default:** "1" **Inheritable:** Yes **Animatable:** Yes
Visibility	Visibility	Specifies if the element is visible or not. **Default:** "Visible" **Inheritable:** No **Animatable:** Yes

TGML Elliptical Pie: <Pie>

Pie defines an elliptical pie slice. Pie is similar to Arc.

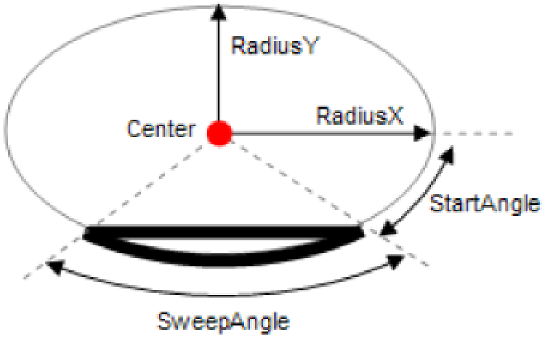


Attribute	Type	Description
Center	Point	The center point of the ellipse. **Default:** "0.0 , 0.0" **Inheritable:** No **Animatable:** Yes
Fill	Brush	Specifies how the interior of the shape is painted. **Default: "None"** **Inheritable:** Yes **Animatable:** Yes
Opacity	Double	A value between "0.0" (transparent) and "1.0" (opaque) **Default:** "1.0" **Inheritable:** No **Animatable:** Yes
RadiusX	Double	The horizontal radius of the ellipse. **Inheritable:** No **Animatable:** Yes

Attribute	Type	Description
RadiusY	Double	The vertical radius of the ellipse. **Inheritable:** No **Animatable:** Yes
StartAngle	Double	Specifies the angle between the x axis and the starting point of the arc. **Default:**"0.0" **Inheritable:** No **Animatable:** Yes
SweepAngle	Double	Specifies the angle between the starting and ending points of the arc. **Default:**"0.0" **Inheritable:** No **Animatable:** Yes
Stroke	Brush	Describes how the line is painted. **Default:"#000000"** **Inheritable:** Yes **Animatable:** Yes
StrokeDashArray	Array of Double	The pattern of dashes and gaps used to outline shapes: "<dash> [<gap> <dash> <gap>...]" If the array only specifies the first dash, the line is patterned as if a gap with the same length as the dash was specified. An array with only one dash set to "0" will result in a line without any pattern. **Default:** "0" **Inheritable:** Yes **Animatable:** Yes
StrokeWidth	Double	The width of the outline of a line. **Default:** "1" **Inheritable:** Yes **Animatable:** Yes
Visibility	Visibility	Specifies if the element is visible or not. **Default:** "Visible" **Inheritable:** No **Animatable:** Yes

TGML Elliptical Chord: <Chord>

Chord defines an elliptical chord. Chord is similar to Pie and Arc.



Attribute	Type	Description
Center	Point	The center point of the ellipse. **Default:** "0.0 , 0.0" **Inheritable:** No **Animatable:** Yes
Fill	Brush	Specifies how the interior of the shape is painted. **Default: "None"** **Inheritable:** Yes **Animatable:** Yes
Opacity	Double	A value between "0.0" (transparent) and "1.0" (opaque) **Default:** "1.0" **Inheritable:** No **Animatable:** Yes
RadiusX	Double	The horizontal radius of the ellipse. **Inheritable:** No **Animatable:** Yes
RadiusY	Double	The vertical radius of the ellipse. **Inheritable:** No **Animatable:** Yes
StartAngle	Double	Specifies the angle between the x axis and the starting point of the arc. **Default:**"0.0" **Inheritable:** No **Animatable:** Yes
SweepAngle	Double	Specifies the angle between the starting and ending points of the arc. **Default:**"0.0" **Inheritable:** No **Animatable:** Yes

Attribute	Type	Description
Stroke	Brush	Describes how the line is painted. **Default:"#000000"** **Inheritable:** Yes **Animatable:** Yes
StrokeDashArray	Array of Double	The pattern of dashes and gaps used to outline shapes: "<dash> [<gap> <dash> <gap>...]" If the array only specifies the first dash, the line is patterned as if a gap with the same length as the dash was specified. An array with only one dash set to "0" will result in a line without any pattern. **Default:** "0" **Inheritable:** Yes **Animatable:** Yes
StrokeWidth	Double	The width of the outline of a line. **Default:** "1" **Inheritable:** Yes **Animatable:** Yes
Visibility	Visibility	Specifies if the element is visible or not. **Default:** "Visible" **Inheritable:** No **Animatable:** Yes

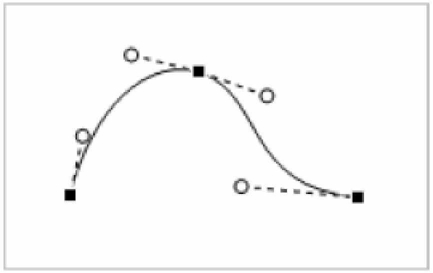
TGML Curves and Paths

TGML contains a definition for curve and path elements:

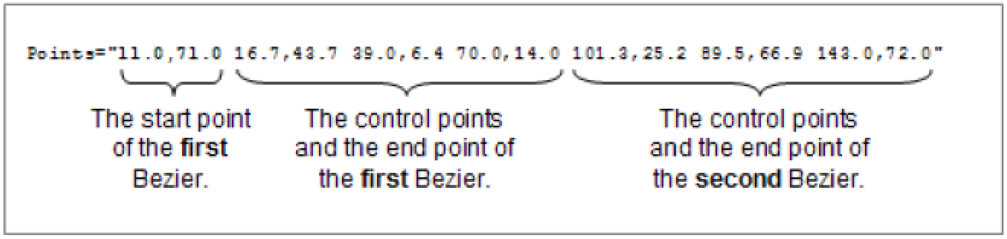
- TGML Cubic Bezier Curve: <Curve>
- TGML Path Element: <Path>

TGML Cubic Bezier Curve

Curve defines a cubic Bezier curve. The cubic Bezier curve has a start point, an end point, and two control points. The control points act as magnets, pulling the curve in certain directions to influence the way the Bezier curve bends.



Curve supports polybezier, which is a consecutive set of Bezier points. The end point of the preceding Bezier becomes the start point of the following Bezier.



```
Points="11.0,71.0 16.7,43.7 39.0,6.4 70.0,14.0 101.3,25.2 89.5,66.9 143.0,72.0"
```

The start point of the **first** Bezier.
The control points and the end point of the **first** Bezier.
The control points and the end point of the **second** Bezier.

Attribute	Type	Description
Closed	Bool	Describes if the curve is closed or not. That is, if Points data end with "z" or not. **Default:** "False" **Inheritable:** No **Animatable:** No
Fill	Brush	Specifies how the interior of the shape is painted. **Default: "None"** **Inheritable:** Yes **Animatable:** Yes
Opacity	Double	A value between "0.0" (transparent) and "1.0" (opaque) **Default:** "1.0" **Inheritable:** No **Animatable:** Yes
Points	Array of Point	The points: start point, control points, and end point, of the Bezier. Polybezier is supported. **Inheritable:** No **Animatable:** No
Stroke	Brush	Describes how the line is painted. **Default:"#000000"** **Inheritable:** Yes **Animatable:** Yes

Attribute	Type	Description
StrokeDashArray	Array of Double	The pattern of dashes and gaps used to outline shapes: "<dash> [<gap> <dash> <gap>...]" If the array only specifies the first dash, the line is patterned as if a gap with the same length as the dash was specified. An array with only one dash set to "0" will result in a line without any pattern. **Default:** "0" **Inheritable:** Yes **Animatable:** Yes
StrokeWidth	Double	The width of the outline of a line. **Default:** "1" **Inheritable:** Yes **Animatable:** Yes
Visibility	Visibility	Specifies if the element is visible or not. **Default:** "Visible" **Inheritable:** No **Animatable:** Yes

TGML Path Element

Path represents the outline of a shape.

The path is described by the PathData attribute, which can contain moveto, line, curve (both cubic and quadratic Beziers), arc, and closepath instructions.

The path element is an implementation of the SVG path data specification. For more information, see the Scalable Vector Graphics (SVG) 1.1 Specification.

Attribute	Type	Description
Fill	Brush	Specifies how the interior of the shape is painted. **Default: "None"** **Inheritable:** Yes **Animatable:** Yes
Opacity	Double	A value between "0.0" (transparent) and "1.0" (opaque) **Default:** "1.0" **Inheritable:** No **Animatable:** Yes
PathData	String	SVG path data. **Inheritable:** No **Animatable:** No

Attribute	Type	Description
Stroke	Brush	Describes how the line is painted. **Default:"#000000"** **Inheritable:** Yes **Animatable:** Yes
StrokeDashArray	Array of Double	The pattern of dashes and gaps used to outline shapes: "<dash> [<gap> <dash> <gap>...]" If the array only specifies the first dash, the line is patterned as if a gap with the same length as the dash was specified. An array with only one dash set to "0" will result in a line without any pattern. **Default:** "0" **Inheritable:** Yes **Animatable:** Yes
StrokeWidth	Double	The width of the outline of a line. **Default:** "1" **Inheritable:** Yes **Animatable:** Yes
Visibility	Visibility	Specifies if the element is visible or not. **Default:** "Visible" **Inheritable:** No **Animatable:** Yes

Remarks

Some of the path commands can be converted or replaced when you import SVG paths. For example, h, H (Horizontal lineto) and v, V (Vertical lineto) can be replaced with l and L (lineto).

TGML Raster Images

TGML defines raster images and an animated raster image:

- TGML Image Element: <Image>
- TGML Animated Images (GIF89a): <Animated Image>

TGML Image Element: <Image>

Image represents a raster image. Image supports JPEG and PNG images.

The image data is stored as a Base64 encoded string in the CDATA section of the Image element.

```
<Image Left="100" Top="100" Width="100" Height="100" ...>
    <![CDATA [iVBORwOKGgoAAAANSUh...SUVORK5CYII=]]>
</Image>
```

The image data is accessible through the Content attribute which means that it can be bound and dynamically updated in View mode.

Attribute	Type	Description
Content	String	The image data (Base64 encoded). **Inheritable:** No **Animatable:** Yes
Height	Double	The height of the image. **Inheritable:** No **Animatable:** Yes
Left	Double	The x coordinate of the upper left corner of the image. **Default:** "0" **Inheritable:** No **Animatable:** Yes
Opacity	Double	A value between "0.0" (transparent) and "1.0" (opaque) **Default:** "1.0" **Inheritable:** No **Animatable:** Yes
Top	Double	The y coordinate of the upper left corner of the image. **Default:** "0" **Inheritable:** No **Animatable:** Yes
Visibility	Visibility	Specifies if the element is visible or not. **Default:** "Visible" **Inheritable:** No **Animatable:** Yes
Width	Double	The width of the image. **Inheritable:** No **Animatable:** Yes

TGML Animated Images (GIF89a): <AnimatedImage>

AnimatedImage represents an animated raster image. AnimatedImage supports the GIF89a format.

The Animation attribute starts and stops the animation.

The image data is stored as a Base64-encoded string in the CDATA section of the AnimatedImage element. The image data is accessible through the Content attribute.

Attribute	Type	Description
Content	String	The image data (Base64 encoded). **Inheritable:** No **Animatable:** Yes
Height	Double	The height of the image. **Inheritable:** No **Animatable:** Yes
Left	Double	The x coordinate of the upper left corner of the image. **Default:** "0" **Inheritable:** No **Animatable:** Yes
Opacity	Double	A value between "0.0" (transparent) and "1.0" (opaque) **Default:** "1.0" **Inheritable:** No **Animatable:** Yes
Top	Double	The y coordinate of the upper left corner of the image. **Default:** "0" **Inheritable:** No **Animatable:** Yes
Visibility	Visibility	Specifies if the element is visible or not. **Default:** "Visible" **Inheritable:** No **Animatable:** Yes
Width	Double	The width of the image. **Inheritable:** No **Animatable:** Yes
Animation	Animation	Starts and stops the animation. **Default:** "Start" **Inheritable:** No **Animatable:** Yes

TGML Text

TGML supports two types of text elements:

- TGML Text Line: <Text>
- TGML Text Flow: <TextBox>

The character data (the text) is stored within the Text element as XML element content.

```
<Text ...>An example</text>
```

The character data is accessible through the Content attribute which means that it is possible to create dynamic text by animating or binding the Content attribute.

TGML Text Line: <Text>

Text defines a graphics element consisting of text. Each Text element causes a single string of text to be rendered. The Text element performs no automatic line break or word wrapping.

Attribute	Type	Description
Content	String	The character data. **Inheritable:** No **Animatable:** Yes
FontFamily	String	The name of the font or font family. **Inheritable:** Yes **Animatable:** Yes
FontSize	Double	The size of the font, in device independent pixels. For more information, see the TGML Coordinate System section. **Inheritable:** Yes **Animatable:** Yes
FontStyle	FontStyle	The style of the font, that is, **Normal** or **Italic**. **Default:** "Normal" **Inheritable:** Yes **Animatable:** Yes
FontWeight	FontWeight	The style of the font, that is, **Normal** or **Bold**. **Default:** "Normal" **Inheritable:** Yes **Animatable:** Yes
HorizontalAlign	HorizontalAlign	Describes the horizontal alignment of a text string: Text: Relative to the x coordinate Left TextBox: Relative to the specified box **Default:** "Left" **Inheritable:** Yes **Animatable:** Yes
Left	Double	The x coordinate of the upper left corner of the text area. **Default:** "0" **Inheritable:** No **Animatable:** Yes

Attribute	Type	Description
Opacity	Double	A value between "0.0" (transparent) and "1.0" (opaque) **Default:** "1.0" **Inheritable:** No **Animatable:** Yes
Stroke	Brush	Describes how the line is painted. **Default:"#000000"** **Inheritable:** Yes **Animatable:** Yes
TextDecoration	TextDecoration	Specifies decorations that are added to the text. **Default:** "None" **Inheritable:** Yes **Animatable:** Yes
Top	Double	The y coordinate of the upper left corner of the text area. **Default:** "0" **Inheritable:** No **Animatable:** Yes
VerticalAlign	VerticalAlign	Describes the vertical alignment of a text string: Text: Relative to the x coordinate Left TextBox: Relative to the specified box **Default:** "Top" **Inheritable:** Yes **Animatable:** Yes
Visibility	Visibility	Specifies if the text is visible or not. **Default:** "Visible" **Inheritable:** No **Animatable:** Yes

TGML Text Flow: <TextBox>

Text defines a graphics element consisting of text. TextBox wraps the text within the specified box. The TextBox element also supports manual line breaks (ASCII character 10).

The text is stored in the CDATA section of the TextBox element.

Attribute	Type	Description
Content	String	The character data. **Inheritable:** No **Animatable:** Yes

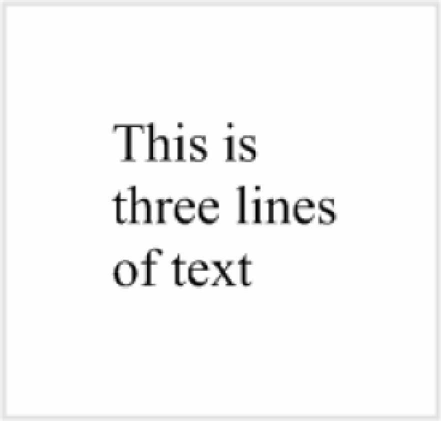
Attribute	Type	Description
FontFamily	String	The name of the font or font family. **Inheritable:** Yes **Animatable:** Yes
FontSize	Double	The size of the font, in device independent pixels. For more information, see the TGML Coordinate System section. **Inheritable:** Yes **Animatable:** Yes
FontStyle	FontStyle	The style of the font, that is, **Normal** or **Italic**. **Default:** "Normal" **Inheritable:** Yes **Animatable:** Yes
FontWeight	FontWeight	The style of the font, that is, **Normal** or **Bold**. **Default:** "Normal" **Inheritable:** Yes **Animatable:** Yes
HorizontalAlign	HorizontalAlign	Describes the horizontal alignment of a text string: Text: Relative to the x coordinate Left TextBox: Relative to the specified box **Default:** "Left" **Inheritable:** Yes **Animatable:** Yes
Left	Double	The x coordinate of the upper left corner of the text area. **Default:** "0" **Inheritable:** No **Animatable:** Yes
Opacity	Double	A value between "0.0" (transparent) and "1.0" (opaque) **Default:** "1.0" **Inheritable:** No **Animatable:** Yes
Stroke	Brush	Describes how the line is painted. **Default:"#000000"** **Inheritable:** Yes **Animatable:** Yes

Attribute	Type	Description
TextDecoration	TextDecoration	Specifies decorations that are added to the text. **Default:** "None" **Inheritable:** Yes **Animatable:** Yes
Top	Double	The y coordinate of the upper left corner of the text area. **Default:** "0" **Inheritable:** No **Animatable:** Yes
VerticalAlign	VerticalAlign	Describes the vertical alignment of a text string: Text: Relative to the x coordinate Left TextBox: Relative to the specified box **Default:** "Top" **Inheritable:** Yes **Animatable:** Yes
Visibility	Visibility	Specifies if the text is visible or not. **Default:** "Visible" **Inheritable:** No **Animatable:** Yes
Height	Double	The height of the text area. **Inheritable:** No **Animatable:** Yes
Width	Double	The width of the text area. **Inheritable:** No **Animatable:** Yes

Example:

```
<TextBox Left="50" Top="50 Width="200" Height="200">
<![CDATA [This is
three lines
of text]]>
</TextBox>
```

Example on screen:



This is
three lines
of text

TGML Gradients

Gradients consist of continuously smooth color transitions along a vector from one color to another. TGML provides for two types of gradients, linear gradients and radial gradients. The Gradient Stop describes the location and color of a transition point in a gradient.

- TGML Linear Gradient: <LinearGradient>
- TGML Radial Gradient: <RadialGradient>
- TGML Gradient Stop: <GradientStop>

TGML Linear Gradient

<LinearGradient> creates a linear gradient brush for the stroke or fill area of the immediate parent.

LinearGradient works in conjunction with gradient stops, which describe the location and color of transition points in gradients. For more information, see the TGML Gradient Stop: <GradientStop> section.

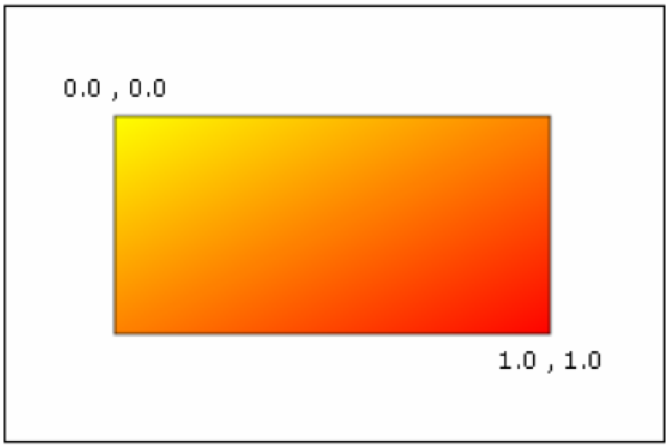
Attribute	Type	Description
Attribute	String	The brush attribute of the parent ("Fill" or "Stroke"). **Inheritable:** No **Animatable:** No
EndPoint	Point	The ending coordinates of the linear gradient. See Remarks. **Default:** "1.0 , 0.0" **Inheritable:** Yes **Animatable:** Yes
SpreadMethod	SpreadMethod	Specifies how the gradient should be drawn outside of the specified gradient vector or space. See Remarks. **Default:** "Pad" **Inheritable:** Yes **Animatable:** Yes

Attribute	Type	Description
StartPoint	StartPoint	The starting coordinates of the linear gradient. See Remarks. **Default:** "0.0 , 0.0" **Inheritable:** Yes **Animatable:** Yes

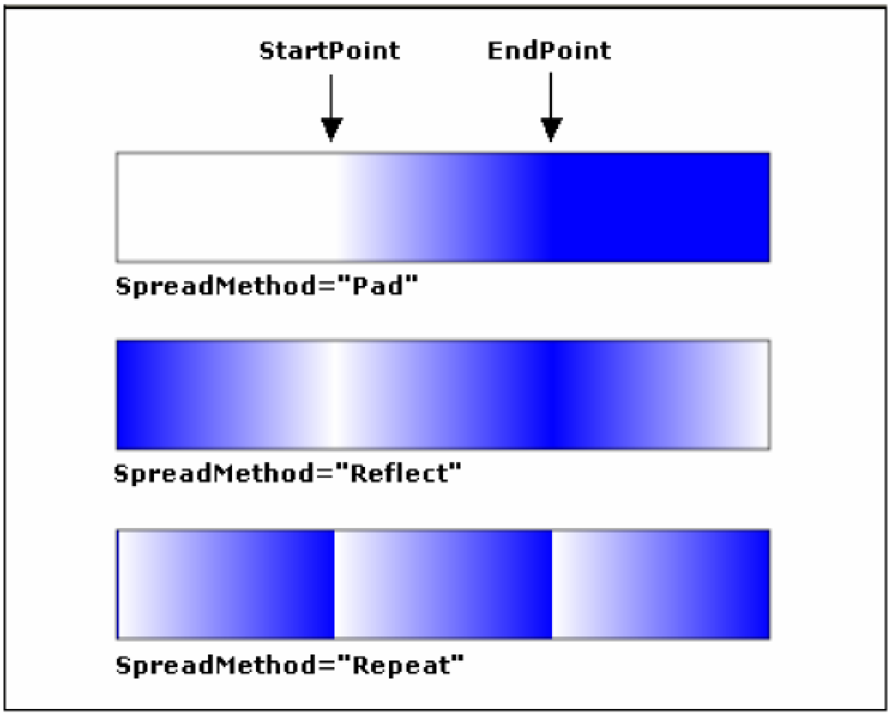
Remarks

StartPoint and EndPoint specify the starting and ending coordinates of the linear gradient. "0.0 , 0.0" represents the upper left corner of the element and "1.0 , 1.0" represents the lower right corner.

Example:



Different SpreadMethod values:



TGML Radial Gradient

<RadialGradient> defines a radial gradient brush for the stroke or fill area of the immediate parent.

<RadialGradient> works in conjunction with gradient stops, which describe the location and color of transition points in gradients. For more information, see the TGML Gradient Stop section.

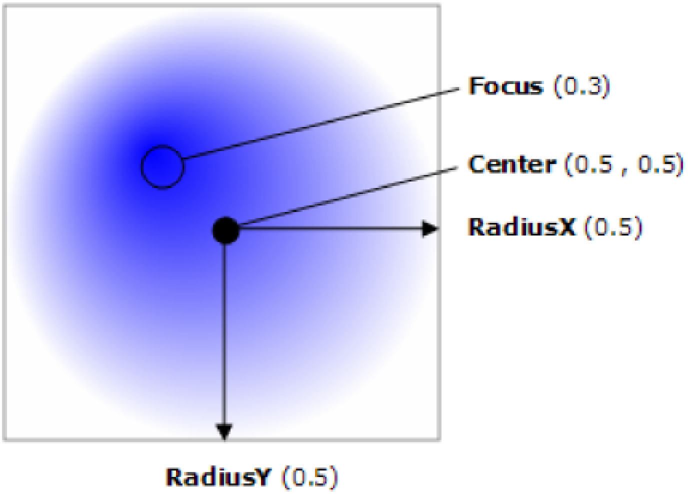
Attribute	Type	Description
Attribute	String	The brush attribute of the parent ("Fill" or "Stroke"). **Inheritable:** No **Animatable:** No
Center	Point	The center of the circle of the radial gradient. See Remarks. **Default:** "0.5 , 0.5" **Inheritable:** Yes **Animatable:** Yes
Focus	Point	The location of the focal point that defines the beginning of the gradient. See Remarks. **Default:** "0.5 , 0.5" **Inheritable:** Yes **Animatable:** Yes
RadiusX	Double	The horizontal radius of the circle of the radial gradient. See Remarks. **Default:** "0.5" **Inheritable:** Yes **Animatable:** Yes
RadiusY	Double	The vertical radius of the circle of a radial gradient. See Remarks. **Default:** "0.5" **Inheritable:** Yes **Animatable:** Yes
SpreadMethod	SpreadMethod	Specifies how the gradient should be drawn outside of the specified gradient vector or space. See Remarks. **Default:** "Pad" **Inheritable:** Yes **Animatable:** Yes

Remarks

The RadialGradient is similar in programming model to the LinearGradient. However, RadialGradient does not have start and end points, but a circle, along with a focal point, to define the gradient behavior. The focal point defines the beginning of the gradient, and the circle defines the end point of the gradient.

Radial gradient only supports the spread method Pad in TGML version 1.0.

Example:

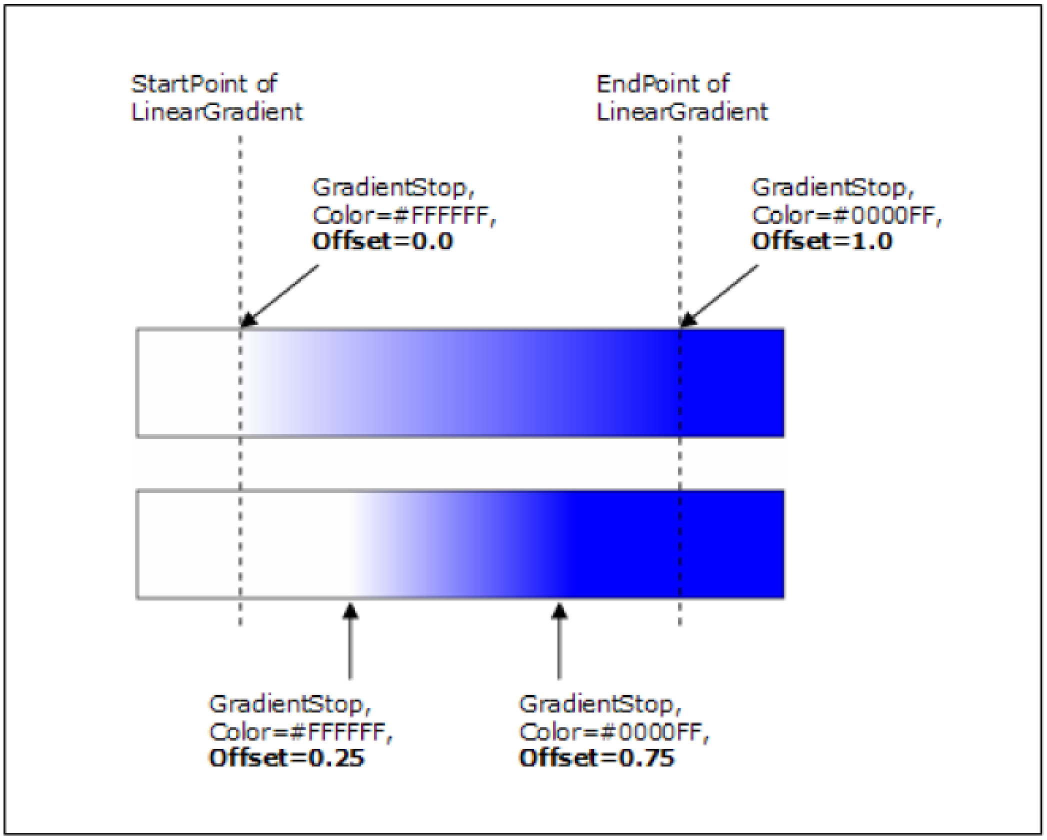


TGML Gradient Stop

<GradientStop> describes the location and color of a transition point in a gradient. GradientStop belongs to its immediate parent gradient element.

Attribute	Type	Description
Color	Color	The color of the gradient stop. **Inheritable:** Yes **Animatable:** Yes
Offset	Double	The location of the gradient stop within the gradient. **Inheritable:** Yes **Animatable:** Yes

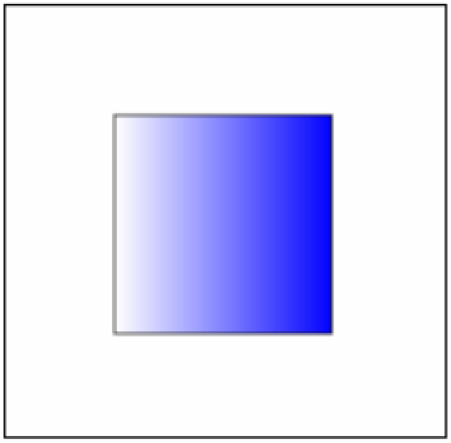
The relations between the StartPoint/EndPoint of the LinearGradient element and the Color/Offset of the GradientStop element:



TGML code containing LinearGradient with GradientStop elements:

```
<TGML>
    <Polygon Points="50.0,50.0 150.0,50.0 150.0,150.0
50.0,150.0" Stroke="#000000">
        <LinearGradient Attribute="Fill">
            <GradientStop Color="#FFFFFF" Offset="0.0"/>
            <GradientStop Color="#0000FF" Offset="1.0"/>
        </LinearGradient>
    </Polygon>
</TGML>
```

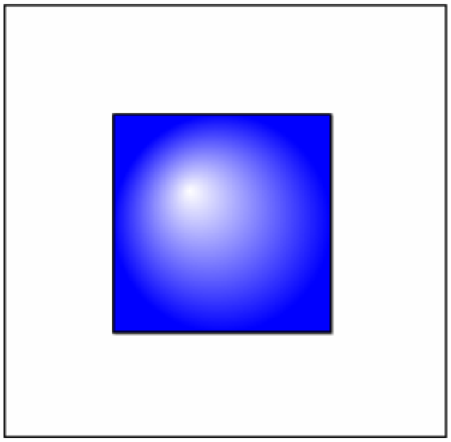
LinearGradient with GradientStop elements on screen:



TGML code containing RadialGradient with GradientStop elements:

```
<TGML>
    <Polygon Points="50.0,50.0 150.0,50.0 150.0,150.0
50.0,150.0" Stroke="#000000">
        <RadialGradient
Attribute="Fill" Center="0.5,0.5" Focus="0.35,0.35" RadiusX="0.5" RadiusY="0.5">
            <GradientStop Color="#FFFFFF" Offset="0.0"/>
            <GradientStop Color="#0000FF" Offset="1.0"/>
        </RadialGradient>
    </Polygon>
</TGML>
```

RadialGradient with GradientStop elements on screen:



Interactive TGML Elements

Interactive TGML elements descriptions:

- TGML Transformations
- TGML Link Element
- TGML Animations
- TGML Dynamics
- TGML Attribute Exposure
- TGML Scripting

TGML Transformations

Transformation elements control the size, position, rotation and skew of graphic objects. The transformation establishes a transformed coordinate system for the immediate parent element.

Transformations are applied in the same order as they are specified in the TGML file.

Transformations can be nested to any level. The effect of nested transformations is to post-multiply, that is, concatenate, the subsequent transformation matrices onto previously defined transformations.

- TGML Rotation: <Rotate>
- TGML Skewing: <SkewX> and <SkewY>

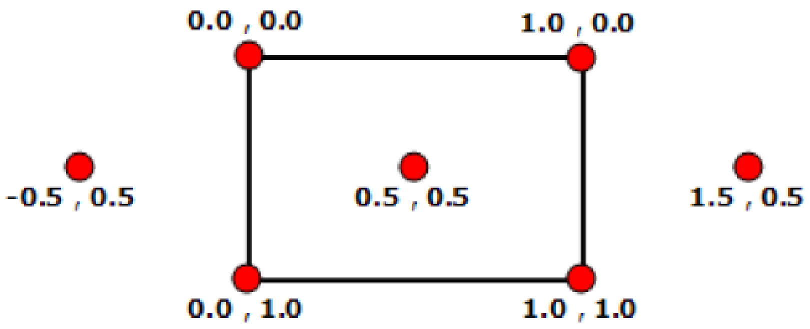
- TGML Scaling: <Scale>
- TGML Translations: <Translate>

TGML Rotation

<Rotate> rotates the coordinate system for the immediate parent element about a specified point.

Attribute	Type	Description
Angle	Double	The angle of the rotation, measured in degrees. A positive value implies clockwise rotation. A negative value implies counter-clockwise rotation. **Default:** "0.0" **Inheritable:** No **Animatable:** Yes
Center	Point	Describes the position of the center point ("X,Y") of the rotation. "0.0 , 0.0" represents the upper left corner of the element and "1.0 , 1.0" represents the lower right corner. See Remarks. **Default:** "0.5 , 0.5" **Inheritable:** No **Animatable:** Yes

Different Center values:



Center point coordinates:

$X_{COORDINATE} = Left_{ELEMENT} + X_{CENTER} * Width_{ELEMENT}$

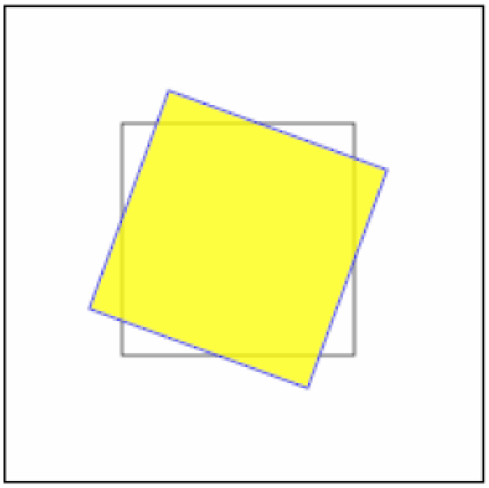
$Y_{COORDINATE} = Top_{ELEMENT} + Y_{CENTER} * Height_{ELEMENT}$

Example:

```
<TGML>
    <Polygon Points="50.0,50.0 150.0,50.0 150.0,150.0 50.0,150.0
Stroke="#000000" Fill="None"/>

    <Polygon Points="50.0,50.0 150.0,50.0 150.0,150.0 50.0,150.0
Stroke="#0000FF" Fill="#FFFF00" Opacity="0.75">
```

```
        <Rotate Angle="20" Center="0.5,0.5"/>
    </Polygon>
</TGML>
```

Example on screen:

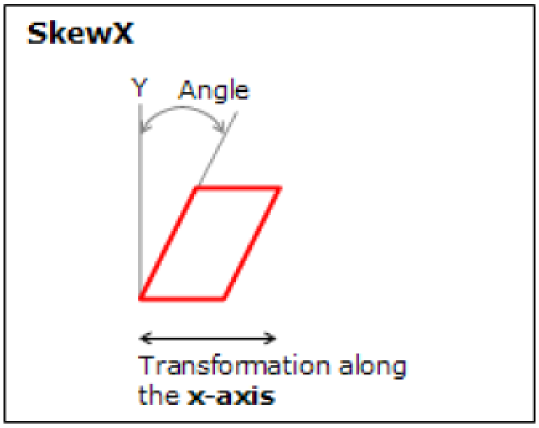


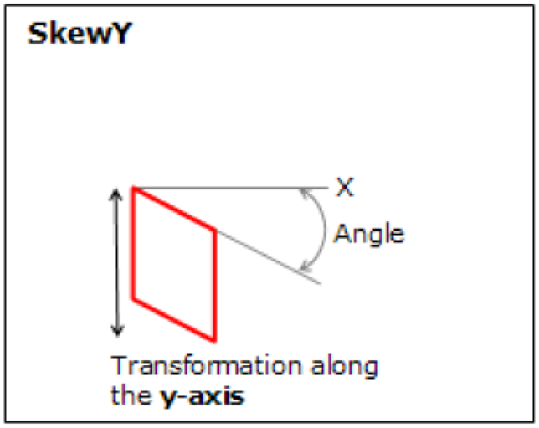
TGML Skewing: <SkewX> and <SkewY>

SkewX and SkewY skew (stretch) the coordinate system for the immediate parent element about a specified point.

Skew X specifies a skew transformation along the X axis. The skew angle is measured in degrees from the Y axis.

Skew Y specifies a skew transformation along the Y axis. The skew angle is measured in degrees from the X axis.



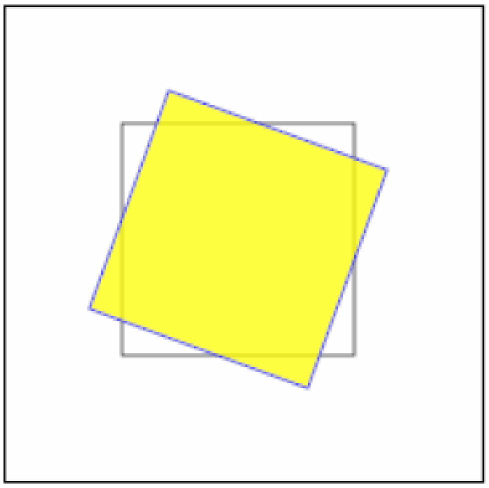


Attribute	Type	Description
Angle	Double	The skew angle, measured in degrees. A positive value implies counter-clockwise skew. A negative value implies clockwise skew. **Default:** "0.0" **Inheritable:** No **Animatable:** Yes
Center	Point	Describes the position of the center point of the skew. For more information, see the TGML Rotation <Rotate> section. **Default:** "0.5 , 0.5" **Inheritable:** No **Animatable:** Yes

Example:

```
<TGML>
    <Polygon Points="50.0,50.0 150.0,50.0 150.0,150.0
50.0,150.0" Stroke="#000000" Fill="None"/>
    <Polygon Points="50.0,50.0 150.0,50.0 150.0,150.0
50.0,150.0" Stroke="#0000FF" Fill="#FFFF00" Opacity="0.75">
        <Skew Angle="45" Center="0.5,0.5"/>
    </Polygon>

    <Polygon Points="250.0,50.0 350.0,50.0 350.0,150.0
250.0,150.0" Stroke="#000000" Fill="None"/>
    <Polygon Points="250.0,50.0 350.0,50.0 350.0,150.0
250.0,150.0" Stroke="#0000FF" Fill="#FF0000" Opacity="0.75">
        <Skew Angle="45" Center="0.5,0.5"/>
    </Polygon>
</TGML>
```

Example on screen:



TGML Scaling: <Scale>

Scale scales the coordinate system for the immediate parent element.

Attribute	Type	Description
Center	Point	Describes the position of the center point (origin) of the scale. For more information, see the TGML Rotation: <Rotate> section. **Default:** "0.5 , 0.5" **Inheritable:** No **Animatable:** Yes
ScaleX	Double	The horizontal scale factor. **Default:** "0.0" **Inheritable:** No **Animatable:** Yes
ScaleY	Double	The vertical scale factor. **Default:** "0.0" **Inheritable:** No **Animatable:** Yes

TGML Translations: <Translate>

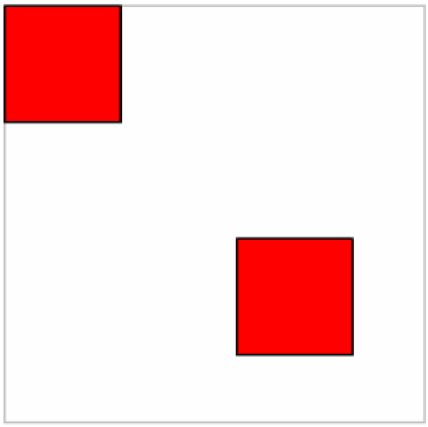
Translate translates (moves) the coordinate system for the immediate parent element.

Attribute	Type	Description
X	Double	Specifies the X direction. **Default:** "0.0" **Inheritable:** No **Animatable:** Yes
Y	Double	Specifies the Y direction. **Default:** "0.0" **Inheritable:** No **Animatable:** Yes

Example:

```
<TGML>
    <Rectangle Left="0" Top="0" Width="50" Height="50" Fill="#FF0000"/>

    <Rectangle Left="0" Top="0" Width="50" Height="50" Fill="#FF0000”>
        <Translate X="100" Y="100"/>
    </Rectangle>
</TGML>
```

Example on screen:



TGML Link Element

<Link> represents a hyperlink to another presentation stored in the database, or file system, of the connected server. Examples of presentation objects are: TGML graphics files, trend log views and on-line plots.

Link indicates that the immediate parent shape or container element is a hyperlink object, which the user can click to navigate to another presentation.

A Link is bound to a presentation object on the server with the same technique used for Dynamics. The Name attribute of the Link element is exposed as a connection point to which the presentation object is connected in the Graphics Editor. For more information, see the TGML Dynamics section.

The Link element has a Description attribute that you can use to add a description of the link. You can expose Description along with Name and present it to the user by binding it in the Graphics Editor.

Link makes the parent element an interactive element, and a TGML viewer is supposed to open the linked presentation object when the user clicks the element.

Attribute	Type	Description
Description	String	A user-defined description of the link. **Inheritable:** No **Animatable:** No
PreventDefault	Bool	Cancels the default action normally taken by the implementation, for example, the viewer. See Remarks. **Default:** "False" **Inheritable:** Yes **Animatable:** No

Remarks

The default action when the user clicks an element containing a Link element is to navigate to, or open, another presentation. When PreventDefault is set to "True" this action is canceled.

Example of text made into a link object:

```
<TGML>
    <Text...>
        Open Overview
        <Link Name="Overview" />
    </Text>
</TGML>
```

Only painted regions are clickable. Clicking a hollow shape, that is a shape with the Fill attribute set to "None", has no effect.

TGML Animations

An animation is a time-based modification of an element attribute. The animation defines a mapping of time to values for the target attribute.

The TGML implementation (Graphics Services) only runs animations in Dynamic mode. The value of an attribute that has been animated (changed) in Dynamic mode is not preserved in the TGML file. The animated value is only valid while running in Dynamic mode.

- TGML Animation: <Animate>
- TGML Sequences: <Sequence>

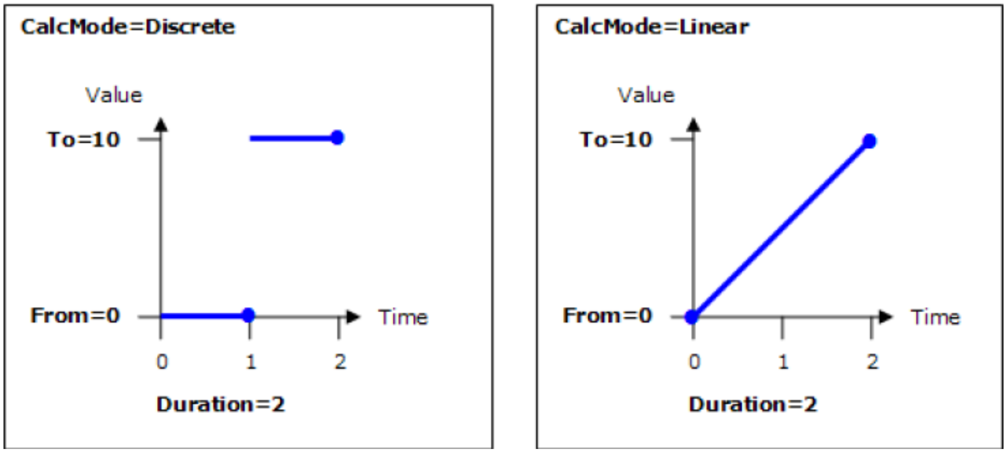
TGML Animation

<Animate> animates a specified attribute of the immediate parent element.

Attribute	Type	Description
Animation	Animation	Starts and stops the animation in Dynamic mode. **Default:** "Start" **Inheritable:** No **Animatable:** Yes
Attribute	String	The animated attribute of the parent element. **Default:** "0.5 , 0.5" **Inheritable:** No **Animatable:** No
AutoReverse	Bool	Indicates whether the timeline plays in reverse after it completes a forward iteration. **Default:** "False" **Inheritable:** No **Animatable:** Yes
CalcMode	CalcMode	Specifies how values are interpolated. See Remarks. **Default:** "Discrete" **Inheritable:** No **Animatable:** Yes

Attribute	Type	Description
Duration	Double	Specifies the "simple duration" of the animation measured in seconds. **Inheritable:** No **Animatable:** Yes
Freeze	Bool	Specifies if the animated attribute value is kept or not when the animation ends (end of "active duration" or stopped by setting Animation to Stop). **Default:** "False" **Inheritable:** No **Animatable:** Yes
From	String (untyped)	The starting value of an animation. The type is determined by the referenced Attribute. **Inheritable:** Yes **Animatable:** Yes
Repeat	Repeat	Describes the way the animation is repeated. **Default:** "Forever" **Inheritable:** No **Animatable:** Yes
To	String (untyped)	The ending value of the animation. The type is determined by the referenced Attribute. **Inheritable:** Yes **Animatable:** Yes

Different CalcMode values:



Example of TGML code containing an Animate element that performs a blink:

```
<TGML>
    <Polygon Points="50.0,50.0 150.0,50.0 150.0,150.0
50.0,150.0" Stroke="#000000" Fill="#FF0000">
```

```
        <Animate
Attribute
="Visibility" From="Visible" Tc="Hidden" Duration="1.0" AutoReverse="True" />
</TGML>
```

TGML Sequences: <Sequence>

Sequence plays a sequence of graphical elements, for example, shapes, text, images, as a 'movie'.

The Interval attribute controls the "frame rate". Interval specifies for how long, measured in seconds, each graphical element is visible. The sequence can be started and stopped by setting the Animation attribute to "Start" and "Stop".

Sequence belongs to the immediate parent container element and controls the visibility of the container's graphical elements.

Attribute	Type	Description
Animation	Animation	Starts and stops the animation in Dynamic mode. **Default:** "Start" **Inheritable:** No **Animatable:** Yes
Interval	Double	Specifies for how long, measured in seconds, each of the shapes in the container is to be visible. **Inheritable:** Yes **Animatable:** Yes

Example of TGML code containing a Sequence element:

```
<TGML>
    <Group>
        <Sequence Interval="1"/>
        <Rectangle
Top
="50.0"

Left
="50.0"
 Width="100" Height="100" Fill="#00FF00" Stroke="#000000" Visibility="Visible"/>
        <Rectangle
Top
="50.0"

Left
="50.0"
 Width="100" Height="100" Fill="#FFFF00" Stroke="#000000" Visibility="Hidden"/>
```

```
        <Rectangle
Top
="50.0"

Left
="50.0"
 Width="100" Height="100" Fill="#FF0000" Stroke="#000000" Visibility="Hidden"/>
    </Group>
</TGML>
```

The example plays a sequence of three rectangles and the frame rate is set to one second. The Visibility attributes of the second and third rectangles are set to "Hidden" to hide them in Edit mode.

TGML Dynamics

A dynamic graphic object is an object (TGML element) whose properties (TGML attributes) are bound to, and controlled by, server variables (signals).

The TGML implementation (Graphics Services) only runs the dynamics engine in Dynamic mode. That is, bound properties are not dynamically updated in Static (edit) mode.

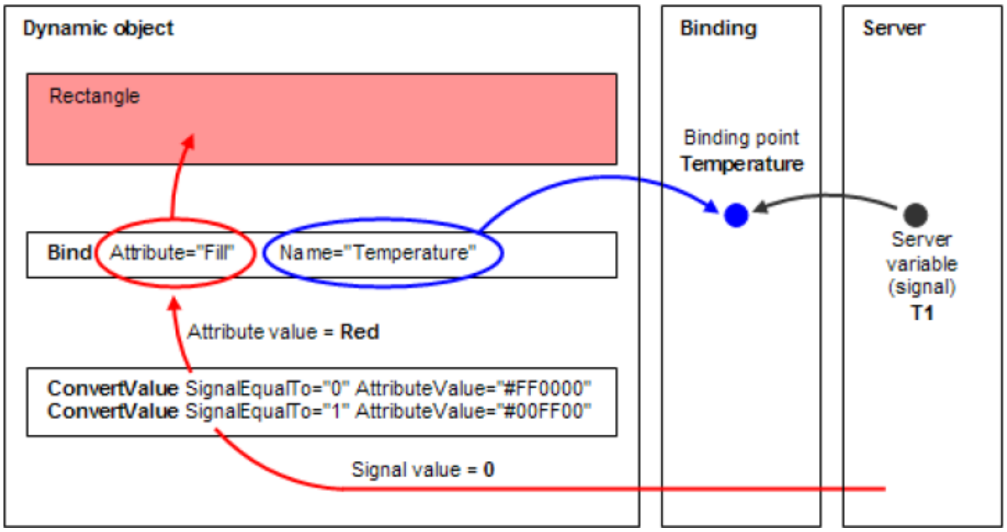
Server variables are connected to element attributes via Bind elements with associated rule elements. A Bind element belongs to the immediate parent element and specifies which attributes of the parent element are going to be dynamically updated (similar to animations). The rule elements belong to the immediate parent Bind element and they specify how a signal value is going to be converted to a TGML attribute value.

The rules are evaluated in sequence, in the same order as they are specified. A rule is executed (and the specified attribute is set) only if the specified conditions are fulfilled. If no matching rules are found, the bound attribute is left unaffected.

TGML supports different types of rule elements (ConvertValue, ConvertRange, ConvertText, ConvertStatus, and so on). Different types of rules can be combined in the same Bind element.

The Name attribute of the Bind element identifies the binding and is exposed to bind tools as a binding point to which the server variable is connected.

Overview of how an attribute in a graphic is bound to a server object, which in turn dynamically affects the graphic:

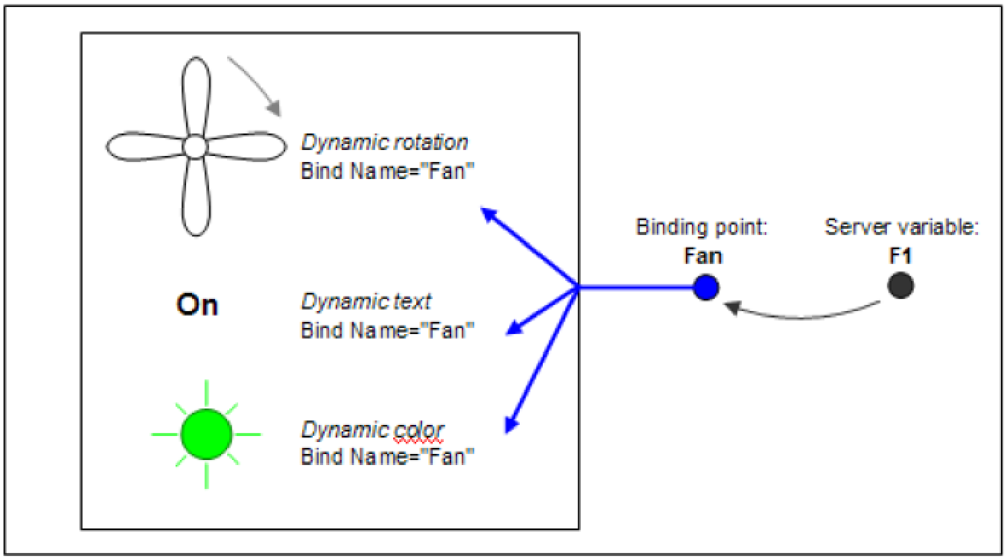


Bindings can be thought of as a "connection table" between the exposed Bind elements and the server variables. The TGML implementation (Graphics Services) only has knowledge about the Name of the Bind elements.

Subscriptions are set up using the names of the Bind elements. When the value of a bound signal is updated, it will be experienced as if the binding point was updated (the signal identity is "hidden" behind the binding point). It is up to the server, or the communication proxy depending on the binding implementation, to resolve the name.

A TGML document can contain multiple Bind elements with the same Name. However, the TGML implementation is only going to expose the Name once to the bind tool in the Graphics Editor (unique binding point). This makes it possible to have multiple presentations of a signal, using a single binding point.

Multiple Bind elements with the same Name:



For more information, see the following sections:

- TGML Signal Binding: <Bind>
- TGML Value Conversion: <ConvertValue>

- TGML Text Value Conversion: <ConvertText>
- TGML Value Range Conversion: <ConvertRange>
- TGML Custom Conversion: <ConvertCustom>
- TGML Status Conversion: <ConvertStatus>

TGML Signal Binding

<Bind> enables a dynamic (server/device controlled) update of an attribute of the immediate parent element.

The Name of the Bind element is exposed to the Graphics Editor bind tool as a binding point. For more information, see the TGML Common Attributes section.

The Bind element has a Description attribute that can be used to add a description of the binding. The Description can be exposed together with the Name and presented to the user in the bind tool.

Bind makes the parent element an "interactive" element. The TGML viewer is supposed to respond, for example, show a "change value" dialog box, when the user clicks the element.

Attribute	Type	Description
Attribute	String	The bound attribute of the parent element. **Inheritable:** No **Animatable:** No
Description	String	A user-defined description of the binding. **Inheritable:** No **Animatable:** No
Format	Format	Specifies the formatting of the subscribed data. See Remarks. **Default:** "None" **Inheritable:** No **Animatable:** No
PreventDefault	Bool	Cancels the default action normally taken by the implementation, for example, the viewer. See Remarks. **Default:** "False" **Inheritable:** Yes **Animatable:** No
DynamicUpdates	DynamicUpdates	Specifies if the Bind should be enabled or disabled. **Default:** "Enabled" **Inheritable:** Yes **Animatable:** No

Remarks

Format is an instruction to the server of how the subscribed value is to be formatted.

"None" means "deliver the data as is". The data type is preserved (integer, float, boolean, string, and so on). You typically use conversion elements to convert the server variable value to a TGML element attribute value.

"Presentation" is an instruction to the server: Deliver the text representation of the variable value, if any (for example, On/Off instead of 0/1). "Presentation" is typically used when the data is to be presented by a Text element without any value conversions.

The “default action” when a user clicks an element containing a Bind element, is usually to open an “edit value” dialog. This action is canceled when PreventDefault is set to 'True'.

PreventDefault is typically set to 'True' in components that mimic interactive controls such as check boxes and spin buttons. In such components, the value is set in a JavaScript using the setValue function.

DynamicUpdates can be used to, by scripting, turn off a group of Binds initially to load a picture faster. Subscriptions can in some systems be performance heavy, so perhaps only a select few Binds need to be active in a picture. By default this attribute is set to “Enabled”.

Example of dynamic text without any conversion elements using the presentation format:

```
<TGML>
    <Text ...>
        <Bind Name="Status" Attribute="TextContent" Format="Presentation" />
        This string is displayed as default.
    </Text>
</TGML>
```

Example showing a Bind that will be enabled/disabled on the OnMouseOver event:

```
<TGML>
    <Component
Clip
="False"

ContentHeight
="113.0"
 ContentWidth="166.0" Height="113.0" Left="73.0" Top="253.0" Width="166.0">
        <Text Left="65.0" Top="48.0">
            <Expose ExposedAttribute="Content" Name="EditModeText"/>
            <Bind Attribute="Content" Format="Presentation" Name="val">
            <Expose ExposedAttribute="Name"/>
            </Bind><! [CDATA[Text] ]>
        </Text>

    <Script OnMouseOver="over"><! [CDATA [
    function over(evt){
        var component = evt.getCurrentTarget();
        var bindings = component.getElementsByTagKame("Bind");
        var bind = null;

        for(var i = 0; i < bindings.getLength(); i++){
            bind = bindings.item(i);
            if (bind.getAttribute("DynamicUpdates") == "Disable"){
                bind.setAttribute("DynamicUpdates", "Enable");
            }
            else{
                bind.setAttribute("DynamicUpdates", "Disable");
            }
        }
    }
    ]]></Script>

    </Component>
</TGML>
```

TGML Value Conversion: <ConvertValue>

ConvertValue specifies how a server variable (signal) value is to be converted to a TGML element attribute value. A ConvertValue element belongs to the immediate parent Bind element.

A ConvertValue rule is only executed when each of the specified conditions is fulfilled.

Example of a ConvertValue element where the attribute is set to #00FF00 (green) when the signal value is more than 18 and less than or equal to 22:

```
<Bind Name="Temperature" Attribute="Fill">
    <ConvertValue AttributeValue="#00FF00"
        SignalMoreThan="18.0" SignalLessOrEqualTo="22.0"/>
</Bind>
```

An example rule that can never be fulfilled (and thus, never executed) since the signal value is both equal to 0 and more than 10:

```
<ConvertValue Name="Bad
Rule" AttributeValue="#FF0000" SignalEqualTo="0" SignalMoreThan="10"/>
```

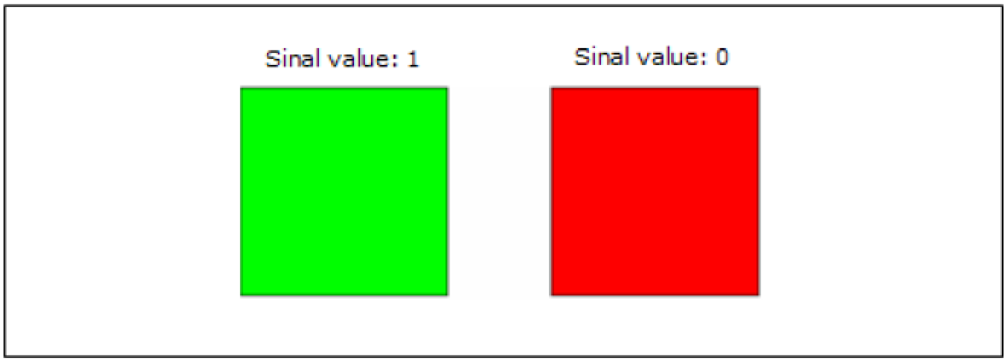
Attribute	Type	Description
AttributeValue	String (untyped)	The attribute value that is to be set (resulting value). The type is determined by the bound Attribute (referenced by the Bind element). **Inheritable:** No **Animatable:** No
SignalEqualTo	String (untyped)	Corresponds to "=" **Inheritable:** No **Animatable:** No
SignalMoreThan	String (untyped)	Corresponds to ">" **Inheritable:** No **Animatable:** No

Attribute	Type	Description
SignalMoreOrEqualTo	String (untyped)	Corresponds to ">=" **Inheritable:** No **Animatable:** No
SignalLessThan	String (untyped)	Corresponds to "<" **Inheritable:** No **Animatable:** No
SignalLessOrEqualTo	String (untyped)	Corresponds to "<=" **Inheritable:** No **Animatable:** No

Example of a SignalEqualTo ConvertValue element:

```
<TGML>
    <Polygon Points="50.0,50.0 150.0,50.0 150.0,150.0
50.0,150.0" Stroke="#000000" Fill="#000000">
        <Bind Name="State" Attribute="Fill">
            <ConvertValue AttributeValue="#00FF00" SignalEqualTo="1"/>
            <ConvertValue AttributeValue="#FF0000" SignalEgualTo="0"/>
        </Bind>
    </Polygon>
</TGML>
```

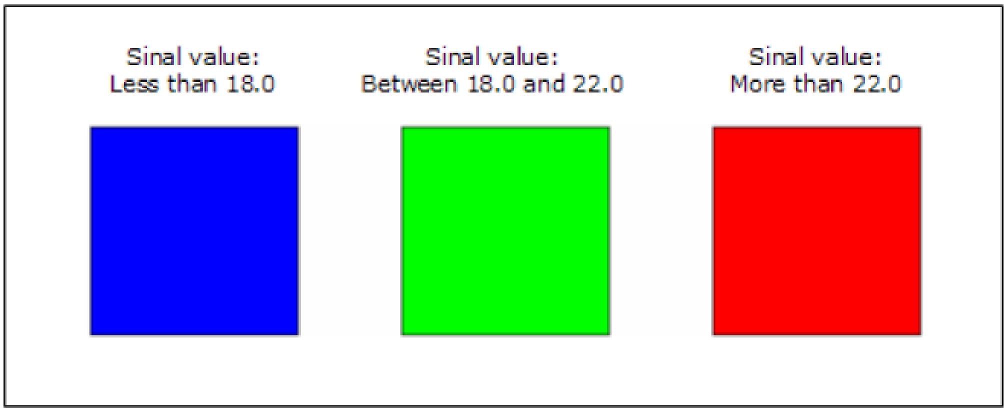
Example of a SignalEqualTo ConvertValue element on screen:



Example of other ConvertValue elements:

```
<TGML>
    <Polygon Points="50.0,50.0 150.0,50.0 150.0,150.0
50.0,150.0" Stroke="#000000" Fill="#000000">
        <Bind Name="Temperature" Attribute="Fill">
            <ConvertValue AttributeValue="#0000FF" SignalLessThan="18.0"/>
            <ConvertValue
AttributeValue="#00FF00" SignalMoreOrEqualTo="18.0" SignalLessOrEqualTo="22.0"/>
            <ConvertValue AttributeValue="#FF0000" SignalMoreThan="22.0"/>
        </Bind>
    </Polygon>
</TGML>
```

Example of other ConvertValue elements on screen:



TGML Text Value Conversion: <ConvertText>

ConvertText specifies how a server variable (signal) value is to be converted to a TGML element attribute value. ConvertText assumes that the signal value is a text. A ConvertText element belongs to the immediate parent Bind element.

A ConvertText rule is executed when the signal value matches the specified text (SignalEqualTo).

Attribute	Type	Description
AttributeValue	String (untyped)	The attribute value that is to be set (resulting value). The type is determined by the bound Attribute (referenced by the Bind element). **Inheritable:** No **Animatable:** No
SignalEqualTo	String (untyped)	Corresponds to "=" **Inheritable:** No **Animatable:** No

TGML Value Range Conversion: <ConvertRange>

ConvertRange specifies how a server variable (signal) value shall be converted to a TGML element attribute value using min and max values. A ConvertRange element belongs to the immediate parent Bind element.

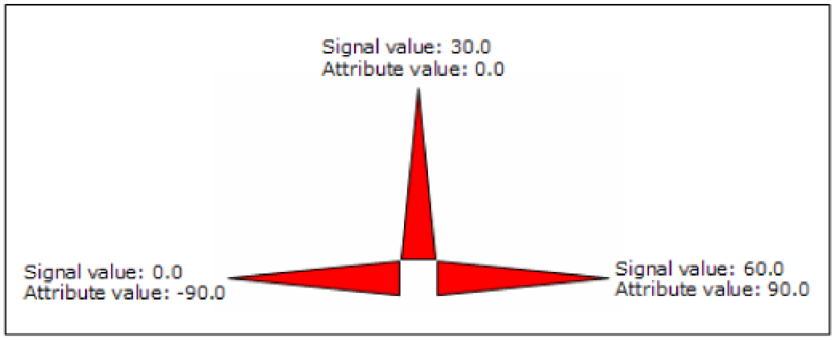
A ConvertRange rule is executed when the signal value is within the specified range.

Attribute	Type	Description
AttributeMaxValue	String (untyped)	The upper bound of the attribute value. The referenced attribute (referenced by the Bind element) will be set to this value when the signal value is more than or equal to SignalMaxValue. **Inheritable:** No **Animatable:** No
AttributeMinValue	String (untyped)	The lower bound of the attribute value. The referenced attribute (referenced by the Bind element) will be set to this value when the signal value is less than or equal to SignalMinValue. **Inheritable:** No **Animatable:** No
SignalMaxValue	String (untyped)	The upper bound of the signal value. **Inheritable:** No **Animatable:** No
SignalMinValue	String (untyped)	The lower bound of the signal value. **Inheritable:** No **Animatable:** No

Example:

```
<TGML>
    <Polygon Points-"150.0,50.0 160.0,150.0 140.0,150.0” Fill="#FF0000"
Stroke="#000000" StrokeWidth="l.0">
        <Rotate Angle="0.0" Center="0.5,1.1">
            <Bind Name="Flow" Attribute="Angle">
            <ConvertRange AttributeMinValue="-
90.0" SignalMinValue="0.0" AttributeMaxValue="90.0" SignalMaxValue="60.0" />
            </Bind>
        </Rotate>
    </Polygon>
</TGML>
```

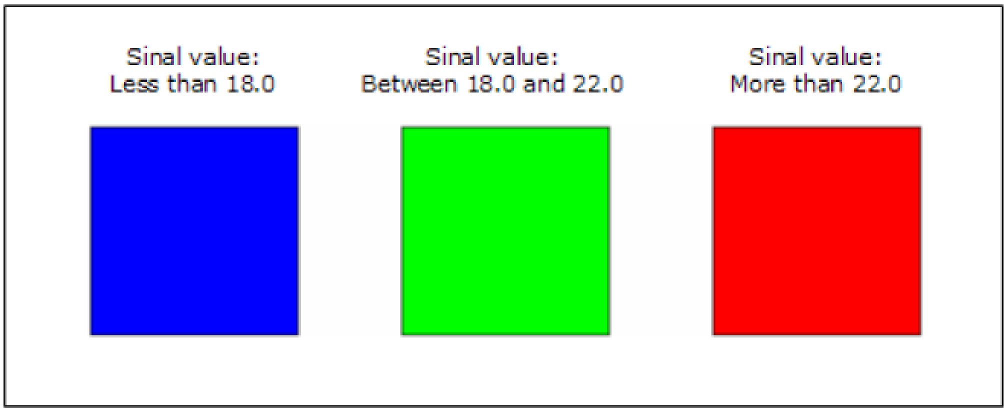
Example on screen:



Example of other ConvertValue elements:

```
<TGML>
    <Polygon Points="50.0,50.0 150.0,50.0 150.0,150.0
50.0,150.0" Stroke="#000000" Fill="#000000">
        <Bind Name="Temperature" Attribute="Fill" >
            <ConvertValue AttributeValue="#0000FF" SignalLessThan="18.0"/>
            <ConvertValue
AttribureValue="#00FF00" SignalMoreOrEqualTo="18.0" SignalLessOrEqualTo="22.0"/>
            <ConvertValue AttributeValue="#FF0000" SignalMoreThan="22.0"/>
        </Bind>
    </Polygon>
</TGML>
```

Example of other ConvertValue elements on screen:



TGML Custom Conversion: <ConvertCustom>

ConvertCustom defines a custom conversion of a signal value. The ConvertCustom element is a script element with a JavaScript function named 'convert'.

Example JavaScript convert function:

```
function convert(value)
{
    //To do: add conversion code here

    return value;
}
```

The function takes one parameter which is the signal value and returns a converted value.

The user can insert any valid JavaScript code in the function body to accomplish a value conversion. By default the function just returns the signal value as is.

Example ConvertCustom element:

```
<TGML>
    <Rectangle
Left="10" Top="10" Width="100" Height="50" Fill="#FFFF00" Stroke="#000000">
```

```
        <Bind Name="Temp" Attribute="Fill">
            <ConvertCustom> <![CDATA [
            function convert(value)
            {
                if(value >= 18 && value <= 22)
                    value = "#00FF00"
                else
                    value = "#FF0000"
                return value ;
            }
            ]]> </ConvertCustcm>
        </Bind>
    </Rectangle>
</TGML>
```

TGML Status Conversion: <ConvertStatus>

ConvertStatus specifies how the status of a server variable (signal) value is to be converted to a TGML element attribute value. A ConvertStatus element belongs to the immediate parent Bind element.

ConvertStatus is executed when the status of the value is different from “normal”, which means when the status value is 0, 1 or 3. For more information, see the TGML Appendix A: User-Defined Descriptions of Custom Attributes section.

ConvertStatus can be combined with other converters (for example, ConvertValue and ConvertRange) but the converters are executed in the same order as they are specified. The ConvertStatus should be placed last if it is combined with other converters.

Attribute	Type	Description
Error	String (untyped)	The attribute value that is to be set when the status is 0 (Error). The type is determined by the bound Attribute (referenced by the Bind element). **Inheritable:** No **Animatable:** No
Stored	String (untyped)	The attribute value that is to be set when the status is 1 (Stored value). The type is determined by the bound Attribute (referenced by the Bind element). **Inheritable:** No **Animatable:** No
Forced	String (untyped)	The attribute value that is to be set when the status is 3 (Forced value). The type is determined by the bound Attribute (referenced by the Bind element). **Inheritable:** No **Animatable:** No

TGML Attribute Exposure

In TGML, it is possible to indicate that an attribute is to be exposed by the TGML editor in such a way that it is easy to find and set the attribute value.

The primary purpose is to be able to create a "Component interface" to the user. A Component can be rather complex. The component developer can choose to expose some of the attributes of the Component (or the contained elements) to, for example, make it easy to change the appearance of the Component.

An Expose element is used to expose an attribute. The Name of the Expose element is presented to the user instead of the original attribute name.

The Expose element has a Description attribute that can be used to add a description of the exposed attribute. This description is presented to the user in the TGML editor.

The attribute exposure is only handled by the TGML editor. TGML viewers ignore the exposure.

TGML Expose Element: <Expose>

Expose indicates that an attribute of the immediate parent element is to be exposed by the TGML editor, using the Name of the Expose element instead of the original attribute name.

Attribute	Type	Description
ExposedAttribute	String	The exposed attribute of the parent element. **Inheritable:** No **Animatable:** No
Description	String	A user-defined description of the exposed attribute. **Inheritable:** No **Animatable:** No

Example:

```
<Component.. .>
    <Rectangle...>
        <Expose ExposedAttribute="Fill" Name="BackgroundColor"/>
    </Rectangle>
    <Ellipse...>
        <Bind ...>
            <ConvertRange...>
                <Expose
ExposedAttribute="SignalMinValue" Name="Min" Description="The minimum value of
the signal"/>
                <Expose
ExposedAttribute="SignalMaxValue" Name="Max" Description="The maximum value of
the signal"/>
            </ConvertRange>
    </Bind>
    </Ellipse>
</Component>
```

TGML Scripting

TGML supports the script language JavaScript 1.5. A JavaScript editor is available in the Graphics Editor. You can use the JavaScript editor to access the elements and their attributes in View mode. The script engine does not run in Edit mode.

The DOM (the elements and their attributes) is accessed using DOM methods such as `getCurrentTarget`, `getAttribute`, and `setAttribute`.

Apart from accessing the DOM, it is also possible to interact with the TGML Viewer, and thus the underlying system, using the EcoStruxure Building Operation specific JavaScript functions such as `setValue` and `execute`.

The execution of the scripts is event driven. Event attributes are used to specify the event and a function name. The function is executed in View mode when the specified event is raised.

For more information, see the TGML Appendix A: User-Defined Descriptions of Custom Attributes section, and the following sections:

- TGML Script Element: <Script>
- TGML Script Context
- TGML Target Area Element: <TargetArea>
- Global Scripts in TGML Graphics
- Invoke Function
- Panel Navigation

TGML Script Element: <Script>

Script defines a script block that belongs to the immediate parent element.

Specifying an event attribute on the Script element makes the parent element the target for the specified event. For example, if OnMouseClick and the function name "click" are specified, the click function is executed when the user clicks the parent element (assuming that the element is a visible graphical element).

If the parent element of Script is a container element, such as Component, the mouse event is sent when any of the contained graphical elements are hit. The hit element is the target, while the parent element, which handles the event, is the current target.

The functions are always defined with an in parameter. The parameter is a reference to the event object that can be used to get event specific information and to access the DOM, starting with getTarget or getCurrentTarget.

Each Script element (script block) creates a JavaScript context. Function calls between script blocks (contexts) are not supported, that is, no support for global script functions.

The script functions are stored in the CDATA section of the Script element. The CDATA section is accessible through the Content attribute.

Attribute	Type	Description
Content	String	The script. **Inheritable:** No **Animatable:** No

Events and event attributes:

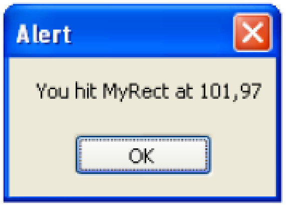
Event Attribute	Event Type	Target	Description
OnDocumentLoad	DocumentLoadEvent	Any element	The TGML document is uploaded (opened). **Cancelable:** No
OnMouseClick	MouseClickEvent	Painted element	A mouse button is clicked over an element. **Cancelable:** Yes
OnMouseDown	MouseDownEvent	Painted element	A mouse button is pressed over an element. **Cancelable:** Yes
OnMouseUp	MouseUpEvent	Painted element	A mouse button is released over an element. **Cancelable:** Yes
OnMouseOver	MouseOverEvent	Painted element	The pointer is moved onto an element. **Cancelable:** Yes
OnMouseMove	MouseMoveEvent	Bind element	The pointer is moved while it is over an element. **Cancelable:** Yes
OnMouseOut	MouseOutEvent	Painted element	The pointer is moved away from an element. **Cancelable:** Yes
OnSignalChange	SignalChangeEvent	Painted element	The value of the bound signal has been changed. The referenced attribute in the Bind element is updated before the event is sent. **Cancelable:** Yes

Example:

```
<TGML>
    <Rectangle
Name
="MyRect"
 Left="50" Top="50" Width="100" Height="100" Fill="#FFFFFF" Stroke="#000000" >
```

```
        <Script OnMouseClick="click"><! [CDATA [
            function click (evt)
            {
                var element = evt.getTarget();
                var name = element.getAttribute("Name");

                var mouseX = evt.getClientX();
                var mouseY = evt.getClientY();

                var message = "You hit " + name + " at " + mouseX +    "," + 
mouseY;

                alert(message);
            }
        ] ]></Script>
    </Rectangle>
</TGML>
```

Example on screen:



You hit MyRect at 101,97

TGML Script Context

A Script context can be seen as a sandbox in which the script is executed. Script contexts can be either Local (access within the current Script element) or Global (access across all Script Elements within the same TGML-file). This scope is regulated with the "UseGlobalScripts"-attribute on the TGML element.

By default, the "UseGlobalScripts" attribute is "False" and it is omitted from the TGML element. Adding it to the TGML element and explicitly setting it to "False" has the same result. The script can only access variables within the current Script element. All variables and function pointers that are declared in a Script element are only accessible within that context:

```
<TGML UseGlobalScripts="False"/>
```

Changing the "UseGlobalScripts" attribute to "True" makes all Script elements use one global JavaScript context. This enables variables and function pointers to be shared between all the Script elements.

To make a variable or a function pointer global and shared between different Script elements it needs to be declared without "var" and "UseGlobalScripts" needs to be set to "True":

```
<TGML UseGlobalScripts="True"/>
```

Remarks

This will decrease load time and memory usage greatly in the HTML5 graphics viewer. It is highly recommended to switch to True. You should modify your existing scripts if they unintentionally leak local values. The Statistics view enables you to see your current global variables and verify that they are not defined as a standard property in a Web browser by clicking on the Validate button.

TGML Target Area Element: <TargetArea>

TargetArea represents a clickable area in a graphic. Target area is not painted, that is, it is always invisible.

You do not have to use TargetArea to be able to handle mouse events. All of the graphical (painted) elements can be targets of mouse events. Use TargetArea when you need an invisible but clickable area.

Attribute	Type	Description
Height	Double	The height of the area. **Inheritable:** No **Animatable:** Yes
Length	Double	The x-coordinate of the area's upper left corner. **Default:** "0" **Inheritable:** No **Animatable:** Yes
Top	Double	The y-coordinate of the area's upper left corner. **Default:** "0" **Inheritable:** No **Animatable:** Yes
Width	Double	The width of the area. **Inheritable:** No **Animatable:** Yes

The TargetArea is used to create a "mouse sensitive" area that covers the whole Component.

Example:

```
<TGML>
    <Component
Left
="50.0"
 Top="50.0" Width="102" Height="102" ContentHeight="102" ContentWidth="102">
        <Script OnMouseOut="out" OnMouseOver="over"><![CDATA [

        function over(evt)
        {
            // Show blue edge
            var component = evt.getCurrentTarget();
            var rectangle = component.getChild("Hovering");
            rectangle.setAttribute("Visibility", "Visible");
        }

        function out(evt)
        {
            // Hide blue edge
            var component = evt.getCurrentTarget();
            var rectangle = component.getChild("Hovering");
            rectangle.setAttribute("Visibility", "Hidden");
        }
        ] ] ></Script>

        <Shapes .../>
            <Rectangle
Name
="Hovering"

Visibility
="Hidden"
 Left="1.0" Top="1.0" Width="100" Height="100" Fill="None" Stroke="#0000FF" />

        <TargetArea Left="0.0" Top="0.0" Width="102" Height="102"/>
    </Component>
</TGML>
```

Without the TargetArea, the `MouseOver` and `MouseOut` events are sent every time the cursor passes the contained elements since the Script is defined at the Component level in this example. The TargetArea has the effect of "hiding" the contained shapes and you only get one `MouseOver` and one `MouseOut` when the cursor passes the Component.

Example of a TGML code fragment containing an invisible link area that can be placed above other shapes:

```
<TargetArea Left="50.0" Top="50.0" Width="100.0" Height="100.0">
    <Link Name="AnotherGraphics" />
</TargetArea>
```

TGML Appendices

For additional TGML information, see the following sections:

- TGML Format Specifications
- TGML Element Summary
- TGML Limitations
- Displaying the TGML Version
- Global Scripts in TGML Graphics
- Invoke Function
- Panel Navigation

TGML Format Specifications

Additions to the TGML Format Specification:

TGML Appendix A: User-Defined Descriptions of Custom Attributes

In Graphics Editor, when an attribute is selected, a short description of the element attribute is displayed in the properties pane.

Custom attributes have no description since the TGML implementation has no knowledge of the attribute. However, it is possible to use Metadata to add a description of a custom attribute:

Example TGML code containing a user-defined description of a custom attribute. When the user selects MyAttribute in the Graphics Editor properties pane the description is displayed:

```
<TGML>
    <Component
Left
="100.0"

Top
="100.0"

Width
="30.0" Height="30.0" ContentWidth="30.0" ContentHeight="30.0" MyAttribute="0" >

        <Metadata Name="MyAttribute" Value="This is a description of
MyAttribute"/>
        .
        .
        .
    </Component>
</TGML>
```

The Name of the Metadata element has to be the same as the attribute name. Value contains the description.

TGML Appendix B: TGML View Object

Appendix B describes the TGML JavaScript Global Objects, the TGML View Object, and the TGML Console Object.

TGML JavaScript Global Objects

Global Object	Description
view	The view object
Console	The console object

TGML View Object

Members	Description
document	The document object
width	Width in normal pixels
height	Height in normal pixels
zoomLevel	Zoom level, a floating point value where1.0 is no zoom, <1 is zoomed out, >1 is zoomed in.
touchEnabled	Touch device (true/false)
addEventListener(type, listener[, useCapture])	Adds an event listener. Supported event: ‘resize’. See w3c standard interface EventTarget
removeEventListener(type, listener[, useCapture])	Remove an event listener. Supported event: ‘resize’. See w3c standard interface EventTarget

Example: The view properties are displayed in a textbox and updated when the viewer resizes or the zoom level changes:

```
<TGML>
    <TextBox
FontFamily
="Arial"

FontSize
="15"

FontStyle
="Normal"

FontWeight
="Normal"

Height
="30"

HorizontalAlign
="Center"

Left
="0"

Name
=""

Opacity
="1.0"

Stroke
="#000000" TextDecoration="None" Top="100" VerticalAlign="Middle" Width="200">

    <![CDATA [width/height@zoomLevel]]>

    <Script OnDocumentLoad="load"><! [CDATA [
        function load(evt)
        {
            function updateView()
            {
                evt.getCurrentTarget().setAttribute("Content", view.width + 'x'
+ view.height + '@'    + view.zoomLevel + (view.touchEnabled ? ' touch' : '
mouse'));
            }

            view.addEventLister('resize', function () {
                updateView ();
            });

            updateView();
        }

    ]]>
    </Script>

    </TextBox>
</TGML>
```

TGML Console Object

Members	Description
info(obj1 [, obj2, ..., objN])	Outputs a message to the Console view
warn(obj1 [, obj2, ..., objN])	Outputs a message to the Console view
error(obj1 [, obj2, ..., objN])	Outputs a message to the Console view
log(obj1 [, obj2, ..., objN])	Outputs a message to the Console view

Example: Outputs the message “test 9” to the console view:

```
<TGML>
    <Script OnDocumentLoad="load"><! [CDATA [
        function load(evt)
        {
            console.log('test', 9);
        }
    ]]></Script>
</TGML>
```

TGML Element Summary

TGML supports the following elements:

TGML Element	Type	Possible Parent	Description
<Animate>	Animation	Brush, Container, Image, Shape, Transform, <GradientStop>	Animate animates a specified attribute of the immediate parent element. For more information, see the TGML Animation: <Animate> section.
<AnimatedImage>	Image	Container, <Tgml>	Animated image represents an animated raster image. AnimatedImage supports the GIF89a format. For more information, see the TGML Animated Images (GIF89a): <AnimatedImage> section.
<Arc>	Shape	Container, <Tgml>	Arc defines an elliptical arc. The elliptical arc is part of an ellipse. For more information, see the TGML Elliptical Arc: <Arc> section.
<Bind>	Shape	Not in Rule, <Expose>, <Link>, <Metadata> or<Script>	Bind enables a dynamic (server/device controlled) update of an attribute of the immediate parent element. For more information, see the TGML Signal Binding: <Bind> section.

TGML Element	Type	Possible Parent	Description
<Component>	Container	Container, <Tgml>	The Components element is a container element (similar to Group) which defines a reusable group of elements. For more information, see the TGML Components: <Component> section.
<ComponentContent>	Container		
<ConvertCustom>	Rule	<Bind>	ConvertCustom defines a custom conversion of a signal value. The ConvertCustom element is a script element with a JavaScript function named 'convert'. For more information, see the TGML Custom Conversion: <ConvertCustom> section.
<ConvertRange>	Rule	<Bind>	ConvertRange specifies how a server variable (signal) value shall be converted to a TGML element attribute value using min and max values. A ConvertRange element belongs to the immediate parent Bind element. For more information, see the TGML Value Range Conversion: <ConvertRange> section.

TGML Element	Type	Possible Parent	Description
<ConvertStatus>	Rule	<Bind>	ConvertStatus specifies how the status of a server variable (signal) value is to be converted to a TGML element attribute value. A ConvertStatus element belongs to the immediate parent Bind element. For more information, see the TGML Status Conversion: <ConvertStatus> section.
<ConvertText>	Rule	<Bind>	ConvertText specifies how a server variable (signal) value is to be converted to a TGML element attribute value. ConvertText assumes that the signal value is a text. A ConvertText element belongs to the immediate parent Bind element. For more information, see the TGML Text Value Conversion: <ConvertText> section.

TGML Element	Type	Possible Parent	Description
<ConvertValue>	Rule	<Bind>	ConvertValue specifies how a server variable (signal) value is to be converted to a TGML element attribute value. A ConvertValue element belongs to the immediate parent Bind element. For more information, see the TGML Value Conversion: <ConvertValue> section.
<Chord>	Shape	Container, <Tgml>	Chord defines an elliptical chord. Chord is similar to Pie and Arc. For more information, see the TGML Elliptical Chord: <Chord> section.
<Curve>	Shape	Container, <Tgml>	Curve defines a cubic Bezier curve. The cubic Bezier curve has a start point, an end point, and two control points. The control points act as magnets, pulling the curve in certain directions to influence the way the Bezier curve bends. For more information, see the TGML Cubic Bezier Curve: <Curve> section.

TGML Element	Type	Possible Parent	Description
<Ellipse>	Shape	Container, <Tgml>	The Ellipse element defines an ellipse. For more information, see the TGML Ellipse: <Ellipse> section.
<Expose>		Not in <Script>	Expose indicates that an attribute of the immediate parent element is to be exposed by the TGML editor, using the Name of the Expose element instead of the original attribute name. For more information, see the TGML Expose Element: <Expose> section.
<GradientStop>	Brush		The GradientStop describes the location and color of a transition point in a gradient. GradientStop belongs to its immediate parent gradient element. For more information, see the TGML Gradient Stop: <GradientStop> section.

TGML Element	Type	Possible Parent	Description
<Group>	Container	Container, <Tgml>	The Group element is a container element, used for grouping elements together so they can, for example, be moved, copied and resized as if they were a single element. For more information, see the TGML Grouping: <Group> section.
<Image>	Image	Container, <Tgml>	Image represents a raster image. Image supports JPEG and PNG images. For more information, see the TGML Image Element: <Image> section.
<Layer>	Container	<Tgml>	The Layer element is a container element used to create layered TGML graphics. For more information, see the TGML Layers: <Layer> section.
<Line>	Shape	Container, <Tgml>	The Line element describes a straight line between two points. For more information, see the TGML Line: <Line> section.

TGML Element	Type	Possible Parent	Description
<LinearGradient>	Brush	Container, Shape	The LinearGradient creates a linear gradient brush for the stroke or fill area of the immediate parent. For more information, see the TGML Linear Gradient: <LinearGradient> section.
<Link>		Container, Image, Shape, <TargetArea>	Link represents a hyperlink to another presentation stored in the database, or file system, of the connected server. Examples of presentation objects are: TGML graphics files, trend log views and on-line plots. For more information, see the TGML Link element: <Link> section.
<Metadata>		Not in <Metadata> or <Script>	Each TGML document contains the TGML root element. It also contains metadata created and interpreted by the TGML application. For more information, see the TGML Document Type Element and Metadata section.
<Path>	Shape	Container, <Tgml>	Path represents the outline of a shape. For more information, see the TGML Path Element: <Path> section.

TGML Element	Type	Possible Parent	Description
<Pie>	Shape	Container, <Tgml>	Pie defines an elliptical pie slice. Pie is similar to Arc. For more information, see the TGML Elliptical Pie: <Pie> section.
<Polygon>	Shape	Container, <Tgml>	The Polygon element describes a polygon, which is a connected series of lines that forms a closed shape. The end point does not have to be specified. The polygon is closed automatically. For more information, see the TGML Polygon: <Polygon> section.
<Polyline>	Shape	Container, <Tgml>	The Polyline element describes a series of connected straight lines. For more information, see the TGML Polyline: <Polyline> section.
<RadialGradient>	Brush	Container, Shape	RadialGradient defines a radial gradient brush for the stroke or fill area of the immediate parent. For more information, see the TGML Radial Gradient: <RadialGradient> section.

TGML Element	Type	Possible Parent	Description
<Rectangle>	Shape	Container, <Tgml>	The Rectangle element defines a rectangle. You can create rounded rectangles by setting values for the attributes RadiusX and RadiusY. For more information, see the TGML Rectangle: <Rectangle> section.
<Rotate>	Transform	Container, Image, Shape	Rotate rotates the coordinate system for the immediate parent element about a specified point. For more information, see the TGML Rotation: <Rotate> section.
<Scale>	Transform	Container, Image, Shape	Scale scales the coordinate system for the immediate parent element. For more information, see the TGML Scaling: <Scale> section.
<Script>		Container, Image, Shape, <Bind>, <TargetArea>, <Tgml>	Script defines a script block that belongs to the immediate parent element. For more information, see the TGML Script Element: <Script> section.

TGML Element	Type	Possible Parent	Description
<Sequence>	Animation	Container	Sequence plays a sequence of graphical elements, for example, shapes, text, images, as a 'movie'. For more information, see the TGML Sequences: <Sequence> section.
<SkewX>	Transform	Container, Image, Shape	SkewX and SkewY skew (stretch) the coordinate system for the immediate parent element about a specified point. For more information, see the TGML Skewing: <SkewX> and <SkewY> section.
<SkewY>	Transform	Container, Image, Shape	SkewX and SkewY skew (stretch) the coordinate system for the immediate parent element about a specified point. For more information, see the TGML Skewing: <SkewX> and <SkewY> section.

TGML Element	Type	Possible Parent	Description
<Snippet>	Container		A snippet is a stored piece of TGML code. It can be used for reusing constructs such as preconfigured animations and gradients. For more information, see the TGML Code Snippets section.
<TargetArea>		Container, <Tgml>	TargetArea represents a clickable area in a graphic. Target area is not painted, that is, it is always invisible. For more information, see the TGML Target Area Element: <TargetArea> section.
<Text>	Shape	Container	Text defines a graphics element consisting of text. Each Text element causes a single string of text to be rendered. The Text element performs no automatic line break or word wrapping. For more information, see the TGML Text Line: <Text> section.

TGML Element	Type	Possible Parent	Description
<TextBox>	Shape	Container	Text defines a graphics element consisting of text. TextBox wraps the text within the specified box. The TextBox element also supports manual line breaks (ASCII character 10). For more information, see the TGML Text Flow: <TextBox> section.
<Tgml>			Each TGML document contains the TGML root element. It also contains metadata created and interpreted by the TGML application. For more information, see the TGML Document Type Element and Metadata section.
<Translate>	Transform	Container, Image, Shape	Translate translates (moves) the coordinate system for the immediate parent element. For more information, see the TGML Translations: <Translate> section.

TGML Limitations

In some implementations, for example, Microsoft GDI+, there are limitations to TGML attributes.

Positional and Size-Related Limitations

Positional attributes have a limitation of +/- 10.000 pixels. Size-related attributes have a limitation of 10.000 by 10.000 pixels. That is, no TGML figure can have a size that exceeds 10.000 by 10.000 pixels. This size limitation also applies for the canvas. The following TGML attributes are affected by the limitations:

- Top
- Left
- Height
- Width
- Length
- X1
- X2
- Y1
- Y2
- Points
- RadiusX
- RadiusY
- StartPoint
- EndPoint
- ContentHeight
- ContentWidth
- FontSize
- StrokeWidth

Displaying the TGML version

You can display the TGML version to find out which TGML version the Graphics Editor version supports.

For more information, see the TGML References Overview section.

To display the TGML version:

1. In Graphics Editor, click **Design**.
2. Click **File > Properties**. The TGML version is displayed in the TGML version box.

Global Scripts in TGML Graphics

By default, each Script element (script block) creates a JavaScript context. In this mode, function calls between script blocks (contexts) are not supported, that is, no support for global script functions.

It is possible to enable scripts to run in one single context for the whole graphic so that functions and variables can be shared between script blocks. This is done by setting the TGML element property ‘UseGlobalScripts’ in the Graphics Editor to ‘True’. By default, 'UseGlobalScripts' is ‘False’.

Using ‘UseGlobalScripts’ may have a very positive effect on graphics loading performance in some of the viewers, such as the HTML5-based viewer in Diagrams.

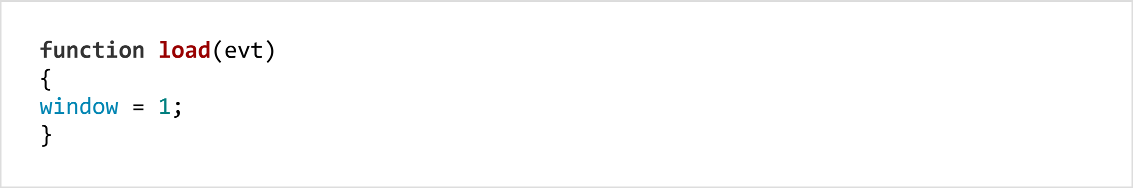
Global variables

In the HTML5 viewer some names are reserved for the Web Browser. It might not be apparent for users writing Java Scripts that using reserved names can cause a conflict when viewing the TGML graphic in the HTML5 viewer, as it works fine in the Diagrams viewer.

You can check whether or not there is a conflict by using the Global Variables tool in the Graphics Editor Statistics pane to analyze the scripts in the TGML graphic and look for global variables that could cause name conflicts.

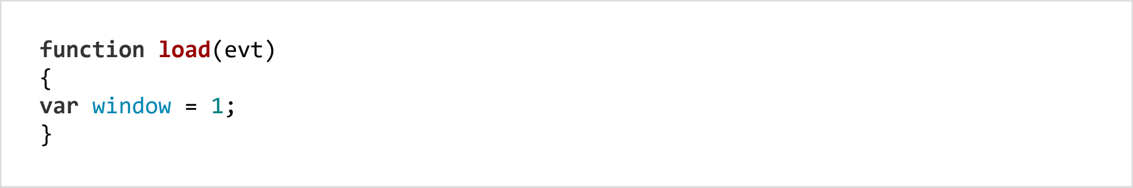
Example:

A script that contains a variable named ‘window’ without the var declaration might have been intended to be used as a local variable. However, not declaring it as ‘var’ makes it a global variable and since ‘window’ is a reserved word (an object in the Web Browser) this object is referenced instead, when the Web Browser executes the script.



```
function load(evt)
{
window = 1;
}
```

NOTE: The variable 'window' is visible in the list of Global script variables. To use reserved words locally in scripts as variables, make sure to declare them as ‘var’.



```
function load(evt)
{
var window = 1;
}
```

Invoke Function

Graphic components can be used to navigate within panels. The navigation function requires you add a script with an invoke function to the graphic component in Graphics Editor.

With invoke functions in the script, you can configure the graphic component to open a linked target object in a target location when you perform a specific action on the component. For example, a graphics component can be configured to open a trend chart in another pane of your graphic panel when you click the graphic component.

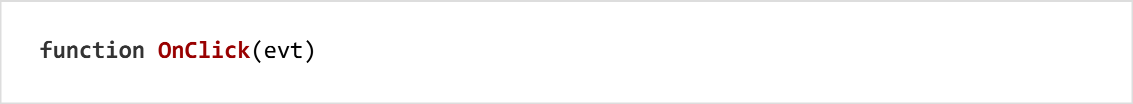
This is an example of the invoke command syntax:

invoke("BindingName", "Operation") where Operation is "OperationName Attribute 1 | Attribute2"

The operation in the invoke function defines where the target object is to be opened.

For more information, see the TGML Scripting section.

Example: The Invoke syntax in Graphics Editor



```
function OnClick(evt)
```

```
{
invoke("Link 1", "OpenInFloatingWindow ShowToolbar = Yes");
}
```

The following table lists the Graphics Editor Invoke commands:

Command	Description
invoke("Link 1", "OpenInFloatingWindow");	Opens the target object in a floating window.
invoke("Link 1", "OpenInNewWindow");	Opens the target object in a new window.
invoke("Link 1", "OpenInNewBrowserTab");	Opens the target object in a new browser window in Diagrams. Falls back to OpenInNewWindow in Diagrams.
invoke("Link 1", "OpenInParent");	The target object replaces the parent location, that is, the panel in which the graphic is contained.
invoke("Link 1", "OpenInSelf");	The target object replaces the graphic.
invoke("Link 1", "OpenInTarget");	Opens the target object in a specific pane.
invoke("Link 1", "OpenInTop");	The target object replaces the top panel, that is, the panel in which all other panels are contained.
invoke("Link 1", "OpenInWorkArea");	Opens the target object in the work area. This command replaces the obsolete 'View object' command.
invoke("", "HistoryForward");	Navigates forward to a view you have visited before in the selected window.
invoke("", "HistoryBack");	Navigates back to a view you have visited before in the selected window.
Invoke("", "LogOff");	Logs off the current user from Diagrams.

Invoke Function Attributes

You can include any number of attributes that are supported for the used command. The OpenInTarget command has to include the Target attribute in order to work.

For each invoke command you can set a number of attributes:

- **Width:** Sets the width of the window in pixels.
- **Height:** Sets the height of the window in pixels

- **Top:** Sets the top position of the window in number of pixels from the top left corner of the screen
- **Left:** Sets the left position of the window in number of pixels from the top left corner of the screen
- **Target:** Sets the target location where the link target is to be opened. You type the name of the pane. The pane and the graphic have to be contained in the same panel.
- **DisplayName:** Displays the name you have typed for the target pane. By default, the object name is the display name of the target pane.
- **ShowToolbar:** Displays the toolbar in the target pane when set to “Yes“.
- **SkipFallback:** When set to “Yes“, this attribute makes the invoke function return a "false" message if the target object does not exist. By using SkipFallback in your script, you make it possible for the script to control what to do if the target location does not exist. By default, the fallback handles a non-existing target location as follows: If the graphic is contained in a panel, the object opens in the workarea. If the graphic is stand-alone, the object replaces the graphic.

Graphics Editor Invoke Function Attributes:

Command	Supported Attributes
OpenInFloatingWindow	• WidthHeight • Top • Left • ShowToolBar
OpenInTarget	• Target • DisplayName • SkipFallback • ShowToolbar
OpenInSelf	• DisplayName • ShowToolbar
OpenInWorkArea	• SkipFallback • ShowToolbar
OpenInParent	• DisplayName • SkipFallback • ShowToolbar
OpenInTop	• SkipFallback • ShowToolbar
OpenInNewWindow	• ShowToolbar

Example: Invoke syntax if you want Trend Chart 1 to open in the pane named TopPane. If TopPane does not exist in the panel of the graphic, Trend Chart 1 is to be opened in a floating window. The display name is Graphic 1 and a toolbar is to be hidden.

```
function OnClick(evt)
{
    if (!invoke("Trend Chart 1", "OpenInTarget Target=TopPane |
DislpayName=Trend Chart 1 | ShowToolbar=No | SkipFallback=Yes")
        invoke("Trend Chart 1", "OpenInFloatingWindow ShowToolbar=No");
}
```

NOTE: The following attribute settings made in the graphic override the corresponding attributes of the previously displayed object in the panel:
DisplayName
ShowToolbar

Panel Navigation

When you use graphics in a panel, you can configure which target location is to be opened when you perform an action (e.g. a mouse click) on a target object linked to the graphic. You can configure the graphic in Graphics Editor by adding an invoke function script. In Diagrams, you link the graphic to the target object you want to open when you perform the action.

A target object can be opened from a graphic in any of the following target locations:

- Floating window
- New window
- Parent
- Self
- Target
- Top
- Work area

You can also use the invoke function script to give the user the ability to navigate back and forward to a previously visited view.

Graphics Editor References Overview

Use the Graphics Editor to create and edit graphics representing a site and the devices that make up the site.

For more information, see the following sections:

- Graphics Editor Overview
- Figures
- Attributes

- Binds and Links
- Graphics Editing Tools Overview
- Layers
- Groups
- Components
- Snippets
- Categories
- Graphic Menus
- TGML Appendices
- Graphics User Interface

Graphics Editor Overview

You can use Graphics Editor to create and edit graphics representing a site and the devices that make up the site.

In Graphics Editor, you can create advanced graphics. Graphics Editor contains tools to make geometrical figures, symbols, texts, flexible data conversions, animations, dynamics and interactivity. You can transform, move, align, arrange, and distribute graphics objects in a work area in several ways.

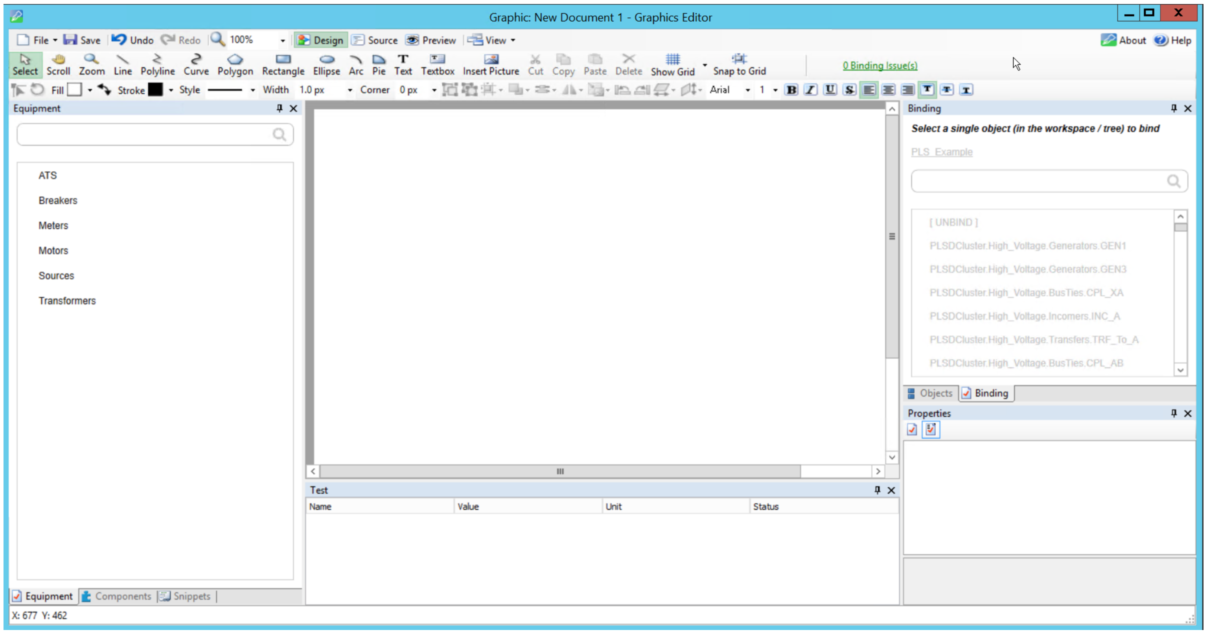
Often used functionality, standard symbols, and components representing common functions are available in libraries delivered with Graphics Editor. You can add to these libraries.

You can open and import graphics or photographs into Graphics Editor, paste graphics into other graphics, and export from Graphics Editor.

You can print the graphics on any printer supported by Microsoft Windows.

A graphic component is a predefined graphic that contains one or several other parts. Components are meant for reuse and typically represent a feature or a device in a live system. Components can be designed as symbols which can be used as building blocks and reused in several graphics. Components reside in dedicated libraries and are displayed in the Components pane. The analog watch is an example of a component.

When you design components, it is recommended that you set Graphics Editor Component mode to **Graphics**.



Select a single object (in the workspace / tree) to bind

When you create a new component, the default work area is 200x200 pixels (where a pixel is the smallest possible drawing unit). A standard graphic work area is 600x800. This smaller work area is usually sufficient to draw a fairly detailed component. When you use the component in the TGML graphic, however, the component is automatically scaled to one fifth of the graphic size. This default size of 40x40 pixels makes the component comparable in size to the ISO and DIN standard components.

For more information, see the TGML File Format section.

When you create a component, the root element, ComponentContent, is used (instead of TGML for a graphic object). When the component is stored in a library and used in a graphic, the ComponentContent element is replaced with the Component element.

The root element of a component always includes at least two metadata elements describing the component: Name and Description. These metadata elements automatically get their values when you save the component.

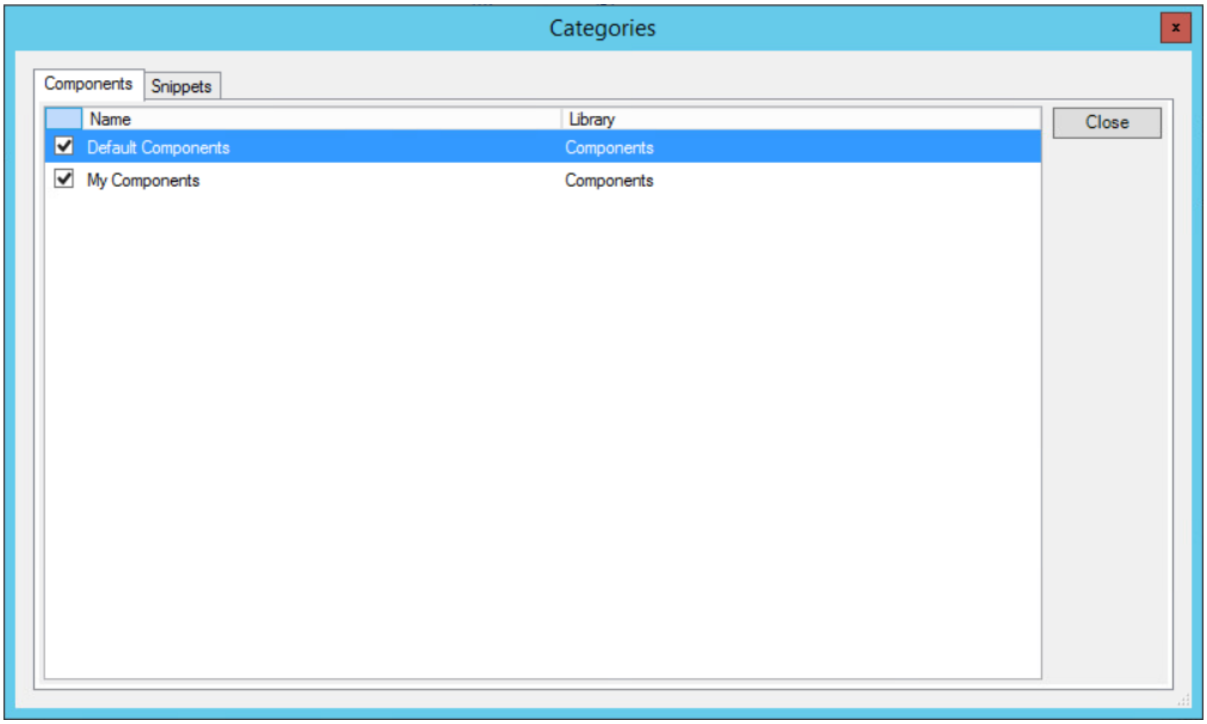
The ComponentContent has the following properties:

- Opacity
- Visibility
- Height
- Width

The height and width values are copied to the ComponentHeight and ComponentWidth properties of the Component element when you store the component. When you use the component in a TGML graphic, however, the height and width values are scaled to 20%, as mentioned above.

Thus, the component has one size when you use it in a graphic and another size (usually larger) when you create or edit the component.

Components are stored as separate files in subfolders of the Components folder. Typically, you would save different categories of components in different subfolders. The subfolder names are displayed as separate bars in the Components pane.



	Name	Library
☑	Default Components	Components
☑	My Components	Components

You can create and import components in several ways:

- Create a new component in Graphics Editor **Component** mode
- Group and save as a component in Graphics Editor **Graphic** mode
- Import components from an external source

You can create graphics that you edit in Graphics Editor. Graphics consist of one or several graphic objects. You can set properties for the graphic objects to define their appearance and behavior.

You can create the graphic objects by using the drawing tools, by copying objects from the work area, or by using instances of objects from the libraries delivered with Graphics Editor.

All drawn objects belong to one of the following two groups:

- Graphics: Free-form drawings
- Components: Standardized graphics for defined reuse

For more information, see the following sections:

- Graphics in Diagrams
- Figures
- Components
- Graphic Menus
- Zooming on Graphics
- Disabling Pan and Zoom
- Saving a Graphic
- Printing Graphics

- Graphics Editor Libraries
- Graphics Editor Keyboard Shortcuts
- Testing a Graphic
- Troubleshooting Console

Graphics in Diagrams

Graphics are representations of a site or parts of a site. Graphics can be made up of figures, text and imported pictures. They can also contain components and functions.

In Diagrams, graphics are represented by a graphics icon.

For more information, see the following sections:

- TGML File Format
- SVG Support

TGML File Format

TGML (TAC Graphics Markup Language) is a declarative XML-based language for dynamic 2D graphics.

Graphics created in Graphics Editor are saved as .TGML files.

When you create a graphic in Graphics Editor, a TGML root object is created at the bottom of the Objects tree in the objects pane. The TGML root object is also the default layer in the work area of Graphics Editor. The TGML object cannot be deleted.

For more information, see the following sections:

- Supported File Formats
- SVG Support

Supported File Formats

You can import other graphics or photographs in supported formats. The following formats are supported in the Graphics Editor:

Graphic Type	File Type
TGML	*.tgml
OGC	*.ogc
OGC	*.sym
OGC	*.sgr
OGC	*.ogx
SVG	*.svg
CAD	*.dxf
CAD	*.dwg

You can export graphics from Graphics Editor to supported formats:

Graphic Type	File Type
Bitmap	*.bmp
GIF	*.gif
JPEG	*.jpg
JPEG	*.jpeg
JPEG	*.jpe
PNG	*.png
TIFF	*.tiff
TIFF	*.tif

SVG Support

You can import SVG pictures; however, only the following SVG Elements and Properties are supported:

Supported SVG Elements

- SVG
- VERSION
- LINE
- POLYLINE
- POLYGON
- ELLIPSE
- CIRCLE
- RECT
- TEXT
- PATH
- GROUP
- DEFS
- LINEARGRADIENT
- RADIALGRADIENT
- STOP
- IMAGE

Supported SVG Attributes

- LEFT
- TOP

- WIDTH
- HEIGHT
- D
- R
- X
- X1
- X2
- CX
- RX
- Y
- Y1
- CY
- RY
- FX
- FY
- ID
- POINTS
- STYLE
- FILL
- STROKE
- STROKEWIDTH
- OPACITY
- OFFSET
- FONTFAMILY
- FONTSIZE
- VISIBILITY
- DISPLAY
- VISIBILITY_HIDDEN
- VISIBILITY_NONE
- VISIBILITY_COLLAPSE
- VISIBILITY_VISIBLE
- VISIBILITY_INLINE
- TRANSFORM
- TRANSLATE

- SCALE
- SCALEX
- SCALEY
- ROTATE
- SKEWX
- SKEWY
- STOP_COLOR
- SPREADMETHOD
- SPREADMETHOD_PAD
- SPREADMETHOD_REFLECT
- SPREADMETHOD_REPEAT
- STROKE_OPACITY
- FILL_OPACITY
- XLINK_HREF
- STROKEDASHARRAY
- GRADIENTSTOP

Adjusting the Graphic Work Area

You can adjust the graphic work area when you initially edit the graphic in order to ensure the work area settings, such as graphic size and background color, are appropriately defined for display in Diagrams.

For more information, see the Components Overview section.

To adjust the graphic work area:

1. In Graphics Editor, in the Objects pane, select the **Tgml** element.
2. In the Properties pane, in the Background box, select the background color.
3. In the Stretch box, select the behavior of the graphic when displayed in Diagrams.
4. In the Height box, type the value for the height of the work area.
5. In the Width box, type the value for the width of the work area.
6. On the File menu, click **Save**.

Zooming

You can zoom in to get a more detailed view of an object. You zoom out to get an overview of a design.

To zoom:

1. In Graphics Editor, on the Drawing toolbar, click **Zoom** 🔍.
2. On the Options toolbar, select the magnifying glass you want to use, then click **+** to zoom in

and **-** to zoom out.

3. In the work area, click the object you want to zoom in or out on.

NOTE: You zoom in/out by 10% each time you click the object.

To zoom to a specified value:

1. In Graphics Editor, on the menu bar, enter a percentage in the Zoom box.

NOTE: Zoom to fit zooms the drawing to fill the available presentation area. You can also enter any percentage value in the Zoom box.

Disabling Pan and Zoom

You can disable the pan and zoom function in your graphic so that the user cannot pan or zoom on the graphic in Diagrams.

To disable pan and zoom:

1. In Graphics Editor, in the Properties pane, select the **Tgml** element.
2. In the DisplayPanAndZoom box, select **True**.
3. On the File menu, click **Save**.

Saving a Graphic

When you have created a graphic you can save it to the database.

To save a graphic:

1. In Graphics Editor, on the File menu, click **Save**.

NOTE: You can only save a graphic in Preview mode. If you choose to save by using the **Save As** command, the link to the database is broken and you have to define the location where you want to save the TGML graphics file in the file system.

Printing Graphics

You print a graphic, for example, to present it to a customer or to get an overview.

To print a graphic:

1. In Graphics Editor, on the File menu, select **Print**, and then click **Print**.
2. In the Print dialog box, select the printer you want to use and set other print options.
3. Click **Print**.

You can set up a page for printing to print a graphic in a specified way.

To set up a page for printing:

1. In Graphics Editor, on the File menu, select **Print**, and then click Page settings.
2. In the Page Setup dialog box, enter paper size, orientation, margins and other properties.

NOTE: The print settings apply for all printouts until you change them.

You can preview a print to make sure the printed page will turn out the way you intended.

To preview a print:

1. In Graphics Editor, on the File menu, point to Print, and then click **Print preview**.
2. In the Print preview dialog box, set the number of pages you want the graphic to print on.
3. Click **Close**.

Graphics Editor Libraries

When Graphics Editor is installed, a library of brushes, components, and snippets are included.

The brushes, components, and snippets are stored in:

C:\ProgramData\Schneider Electric\Power SCADA Operation\v2020 R2\Applications\Graphics

Frequently used objects, animations, and behaviors are stored in the following sub folders:

Folder	Usage
\Brushes	The Colors and Gradients palettes are stored here.
\Components	Standard symbols with specific meaning, as well as some other common symbols, are stored as components in the Components library, available in one of the window panes. A component can be dragged and dropped directly into the Design pane. New components can be added to the library.
\Snippets	A snippet is a piece of TGML code whose purpose is to store a “behavior” for reuse. A number of common behaviors are stored in the Snippets pane. A snippet can be dragged and dropped on an object in the Objects pane. Objects can be copied and modified, even created, and then saved as new snippets in the library.

All subfolders in the Components library are displayed as clickable bars in the Components pane. All *.tgmlcomponent files in the Components subfolder are displayed as selectable components under the corresponding bar.

All subfolders in the Snippets library are displayed as clickable bars in the Snippets pane. All *.tgmlsnippet files in the Snippets subfolder are displayed as selectable snippets under the corresponding bar.

Graphics Editor Keyboard Shortcuts

You can access most of the Graphics Editor commands by using keyboard shortcuts.

Press	To
CTRL+N	Start a new graphic
CTRL+O	Open an existing graphic
CTRL+S	Save the current graphic

Press	To
CTRL+Shift+S	Save the current graphic in a specified location and with a specified file name
CTRL+Shift+P	Preview a print
CTRL+P	Print graphic
CTRL+F4	Close the current graphic
ALT+F4	Close the current graphic and exit the program
CTRL+Z	Undo the latest change
CTRL+Y	Revert the latest Undo command
F6	Set Design mode
F7	Set Source mode
F8	Set Preview mode
F11	Toggle between hiding and showing all panes
F12	Toggle between hiding and showing the Objects and Properties panes
F1	Access Help
F2	Rename
F5	Refresh
CTRL+Shift+F6	Go to a previous graphic
CTRL+F6	Go to a later graphic
CTRL+0	Zoom
CTRL+1	Select
CTRL+2	Use the Line tool
CTRL+3	Use the Polyline tool
CTRL+4	Use the Curve tool
CTRL+5	Use the Polygon tool
CTRL+6	Use the Rectangle tool
CTRL+7	Use the Ellipse tool
CTRL+8	Use the Arc tool

Press	To
CTRL+9	Use the Pie tool
CTRL+T	Use the Text tool
CTRL+Shift+T	Use the Textbox tool
CTRL+X	Cut
CTRL+C	Copy
CTRL+V	Paste
DEL	Delete
CTRL+F	Find
CTRL+H	Find and replace
CTRL+A	Select all
CTRL+D	Clear all
CTRL+Shift+Space	Toggle to show or hide the grid
CTRL+Space	Toggle to snap or unsnap to the grid
CTRL+G	Group figures
CTRL+Shift+G	Ungroup figures
CTRL+Mouse wheel up	Zoom in
CTRL+Mouse wheel down	Zoom out
+ (on numeric keyboard)	Zoom in
– (on numeric keyboard)	Zoom out
CTRL+* (on num. keyboard)	Restore to original size
Arrow keys	Move the selected figure to next grid point if Snap to Grid is activated
ALT+ Arrow keys	Move the selected figure one pixel

Testing a Graphic

You can test the behavior of graphics and components offline in **Preview** mode by setting test values in the Test pane.

Any graphic or component with an associated name and a Bind object is displayed in the Test pane.

You enter test values in the Value column and can set an optional Unit.

You can also test the behavior of certain signal status values. The Status column contains a drop-down menu where you can select four kinds of status:

- Error
- Database value
- Value from device
- Forced value

By default, Error status is handled by the graphic (the figure is crossed over in red). Other status types can be modified by user-written Java scripts.

You can test the animation, snippets, and other parts of a graphic to ensure it works the way it is intended to.

To test a graphic in Preview mode:

1. In Graphics Editor, on the menu bar, click **Preview** to open the graphic in preview mode.
2. On the menu bar, click **View**, and then click **Test**.
3. In the Test pane, in the Value column, type the value for the drawing object for which you want to test the behavior.
4. In the Status column select **Forced value**.
5. In the work area, check the behavior.

Graphics Editor Console

The console is used for testing and troubleshooting scripts in TGML graphics.

The console is optimized for developers and programmers. To be able to use the console, you must have knowledge of scripting.

To open the console:

1. On the menu bar, click **View** and then click **Console**.

Figures Overview

A figure is the smallest independent element of a graphic, for example, a circle. Figures are graphically represented TGML elements.

All figures in a graphic are displayed in a tree structure in the Objects pane. The TGML root object is always present in the tree structure and cannot be deleted. The TGML root object properties define the size and color of the work area. The position of the objects in the tree structure reflects the relationship between figures in the graphic. The closer in the tree structure a figure is to the TGML root object, the further back it is located in the work area. You can move the figures in the tree structure. When you move a figure in the tree structure, it is dynamically moved back or forward in the work area.

NOTE: Apart from the two surface coordinates, x and y, figures also have a hidden stacking order known as the z-coordinate or the z-order. The z order means that more recently added figures are put in the front and older ones in the back. Thus, more recent figures can cover previous figures in the graphic.

NOTE: To change the order of the figures, you select a figure in the Objects pane and click the Move up ⇧ or Move down ⇩ button.

When you select a figure in the work area, the figure and its elements are selected in the Objects pane. You can also select an element in the Objects pane.

NOTE: Before you can select a figure in the work area, you have to make sure that the layer where the figure is located is active.

To create certain elements on an object, you need to right-click on the object and then click the element in the Objects pane. Use this method to create the following elements on an object:

- Bind
- Link
- Animate
- TargetArea
- Metadata
- Chord
- AnimatedImage
- Expose
- Script

For more information, see the following sections:

- Drawing Tools
- Snippets
- Attributes
- Binds and Links
- Graphics Editing Tools
- Graphic Object Position

Drawing Tools Overview

Use the Graphics Editor drawing tools to add lines, polylines, curves, polygons, rectangles, ellipses, arcs, pies, texts, and textboxes to a graphic.

For more information, see the following sections:

- Drawing a Line
- Drawing a Polyline
- Drawing a Curve
- Editing a Curve
- Drawing a Polygon
- Drawing a Rectangle
- Drawing a Square

- Drawing an Ellipse
- Drawing a Circle
- Drawing an Arc or Pie
- Editing an Arc or Pie
- Adding Text and Textboxes
- Editing Text or Textboxes
- Inserting Pictures
- Adjusting a Picture
- Adding an Animated Picture

Drawing a Line

Use the Graphics Editor **Line** tool to draw a straight line, that is, a line between two points.

You can set stroke, style and width properties to change the line color, pattern and thickness.

You draw a line when you want to draw an extending one-dimensional figure that has no curvature.

To draw a line:

1. In Graphics Editor, in the Layers pane, select the layer where you want to draw the line.
2. On the Drawing toolbar, click **Line**.
3. In the work area, click where you want the line to start and drag to where you want it to end.
4. On the Drawing toolbar, click **Select**.
5. In the Properties pane, in the Name box, type the name of the line.

   NOTE: You only need to name the drawing object if you will be binding the object. Naming the object now will help you identify the object later.
6. On the Options toolbar, or in the Properties pane, adjust the appearance of the line.
7. On the File menu, click **Save**.

Drawing a Polyline

Use the Graphics Editor Polyline tool to draw a line with several nodes, that is, a line with angles.

You can set fill, stroke, style and line width to achieve a certain appearance of the polyline.

You draw a polyline to get a figure that consists of two or more connected line segments.

To draw a polyline:

1. In Graphics Editor, in the Layers pane, select the layer where you want to draw the polyline.
2. On the Drawing toolbar, click **Polyline**.
3. In the work area, drag from where you want the polyline to start and click for each new line segment you want to add.

NOTE: You have to add a new segment for every turn of the polyline.

4. Double-click to finish the polyline.
5. On the Drawing toolbar, click **Select**.
6. In the Properties pane, in the Name box, type the name of the polyline.

NOTE: You only need to name the drawing object if you will be binding the object. Naming the object now will help you identify the object later.

7. On the Options toolbar, or in the Properties pane, adjust the appearance of the curve.
8. On the File menu, click **Save**.

Drawing a Curve

Use the Graphics Editor Curve tool to draw a curve, that is, a line that is not straight.

Drawing perfect curves requires some understanding of the principles of curves, as well as some practical experience. When you draw a curve and click the key points of the curve, a number of curve segments are created. These segments are defined by three vertices:

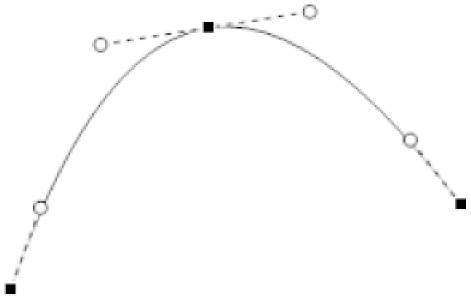
- The vertex, that is, the start point
- The highest/lowest point of a curve segment
- The end point

When a curve consists of more than one segment, the point connecting two curve segments is also a vertex. The curve passes through all of these vertex points.

Normally, the vertex points are not displayed, but to modify the curve you need to access these points.

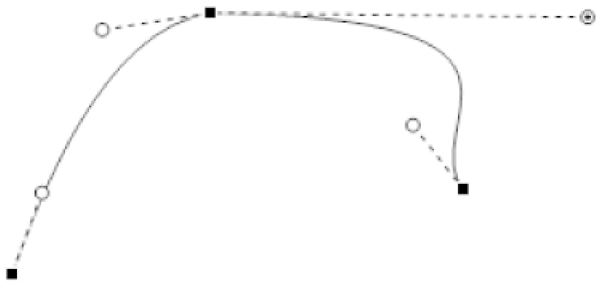
When you double-click the curve, the vertices are displayed. Two handles are associated with each vertex (except for the start and end points, which have only one handle).

Example: A simple curve with its three vertices and two plus two handles



Each handle controls the shape of the curve at its side of the vertex. The length and the angle of the handle determine the direction and curvature of that end of the segment. The handle can be regarded as a magnet attracting its part of the segment.

Example: By stretching the top right handle, that part of the curve segment is modified



If you want to modify a curve, you have the following options:

- Moving a Curve Vertex
- Adjusting a Curve Handle

NOTE: You can also nudge the vertex by using the arrow keys.

You draw a curve to get a line that is not straight and that consists of two or more segments.

To draw a curve:

1. In Graphics Editor, in the Layers pane, select the layer where you want to draw the curve.
2. On the Drawing toolbar, click **Curve**.
3. In the work area, drag from where you want the curve to start and click for each segment you want to add.

   NOTE: You have to add a new segment for each turn of the curved line.

4. Double-click to finish the curve.
5. On the Drawing toolbar, click **Select**.
6. In the Properties pane, in the Name box, type the name of the curve.

   NOTE: You only need to name the drawing object if you will be binding the object. Naming the object now will help you identify the object later.

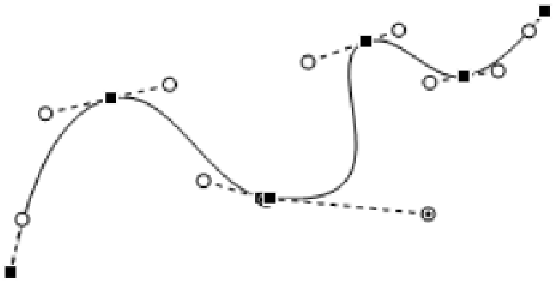
7. On the Options toolbar or in the Properties pane, adjust the appearance of the curve.
8. On the File menu, click **Save**.

Editing a Curve

You edit a curve to adjust any point of the curve.

To edit a curve:

1. In Graphics Editor, in the work area, double-click the curve to display its vertices.
2. Drag the vertex you want to change to a new position.



3. Click outside the curve to finish.

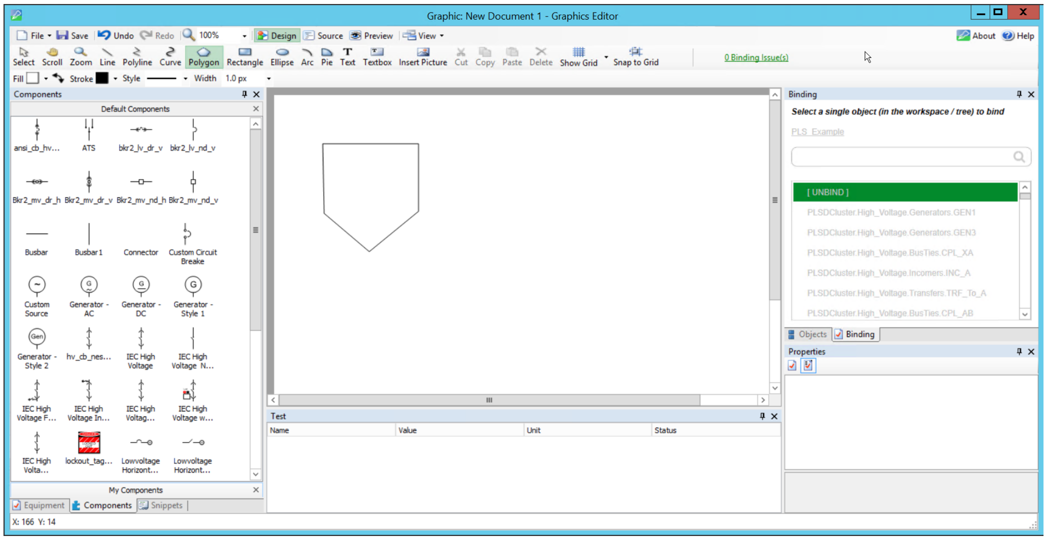
Drawing a Polygon

Use the Graphics Editor Polygon tool to draw a polygon, that is, a plane figure that is bounded by a closed path, composed of a finite sequence of straight line segments.

You draw a polygon when you need a closed multi-sided figure.

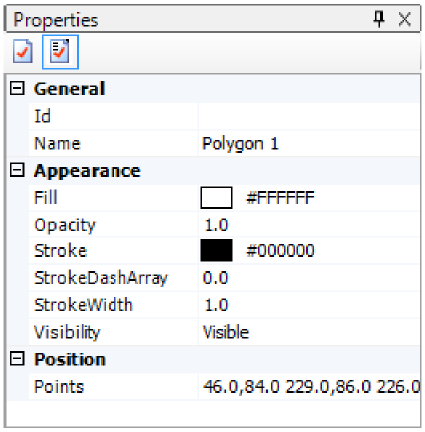
To draw a polygon:

1. In Graphics Editor, in the Layers pane, select the layer where you want to draw the polygon.
2. On the Drawing toolbar, click **Polygon**.
3. In the work area, click where you want to locate the corners of the polygon.



4. Double-click to close the polygon.
5. On the Drawing toolbar, click **Select**.
6. In the Properties pane, in the Name box, type the name of the polygon.

> **NOTE**: You only need to name the drawing object if you will be binding the object. Naming the object now will help you identify the object later.



Properties

⊟ **General**	
Id	
Name	Polygon 1
⊟ **Appearance**	
Fill	□ #FFFFFF
Opacity	1.0
Stroke	■ #000000
StrokeDashArray	0.0
StrokeWidth	1.0
Visibility	Visible
⊟ **Position**	
Points	46.0,84.0 229.0,86.0 226.0

7. On the Options toolbar or in the Properties pane, adjust the appearance of the polygon.
8. On the File menu, click Save.

Drawing a Rectangle

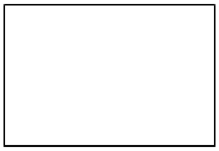
Use the Graphics Editor Rectangle tool to draw a simple rectangle, that is, a quadrilateral with four right angles.

You draw a rectangle when you need a four-sided figure with four 90° angles and there is no component that works for this situation.

For more information, see the Drawing Tools section.

To draw a rectangle:

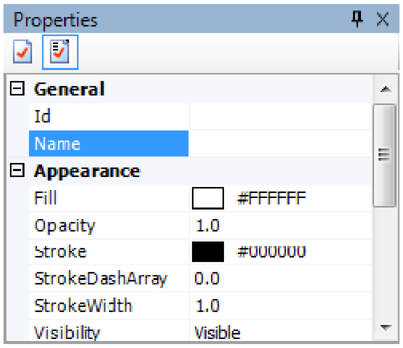
1. In Graphics Editor, in the Layers pane, select the layer you want to draw the rectangle on.
2. On the Drawing toolbar, click **Rectangle**.
3. In the work area, click where you want the rectangle to begin and drag the pointer to where you want it to end.



4. On the Drawing toolbar, click **Select**.

5. In the Properties pane, in the Name box, type the name of the rectangle.

> **NOTE**: You only need to name the drawing object if you will be binding the object. Naming the object now will help you identify the object later.



Properties

⊟ **General**	
Id	
Name	
⊟ **Appearance**	
Fill	□ #FFFFFF
Opacity	1.0
Stroke	■ #000000
StrokeDashArray	0.0
StrokeWidth	1.0
Visibility	Visible

6. On the Options toolbar or in the Properties pane, adjust the appearance of the rectangle.
7. On the File menu, click **Save**.

Drawing a Square

You draw a square when you need a figure with four equal sides and four 90° angles.

To draw a square:

1. In Graphics Editor, in the Layers pane, select the layer you want to draw the square.
2. On the Drawing toolbar, click **Rectangle**.
3. In the work area, press **Shift** and click where you want the square to begin and drag to where you want it to end.
4. On the Drawing toolbar, click **Select**.
5. In the Properties pane, in the Name box, type the name of the square.

   NOTE: You only need to name the drawing object if you will be binding the object. Naming the object now will help you identify the object later.

6. On the Options toolbar or in the Properties pane, adjust the appearance of the rectangle.
7. On the File menu, click **Save**.

Drawing an Ellipse

Use the Graphics Editor Ellipse tool to draw an ellipse, that is, a plane curve that results from the intersection of a cone by a plane in a way that produces a closed curve.

NOTE: To draw a circle, select **Ellipse** and press **Shift** while you draw.

You draw an ellipse when you need a conic section whose plane is not parallel to the axis, base, or generatrix of the intersected cone.

To draw an ellipse:

1. In Graphics Editor, in the Layers pane, select the layer where you want to draw the ellipse.
2. On the Drawing toolbar, click **Ellipse**.
3. In the work area, drag the pointer from where you want to start the ellipse to where you want it to end.
4. On the Drawing toolbar, click **Select**.

5. In the Properties pane, in the Name box, type the name of the ellipse.

   NOTE: You only need to name the drawing object if you will be binding the object. Naming the object now will help you identify the object later.

6. On the Options toolbar or in the Properties pane, adjust the appearance of the ellipse.
7. On the File menu, click **Save**.

Drawing a Circle

In Graphics Editor, use the Ellipse tool to draw a circle, that is, a line forming a closed loop, every point on which is a fixed distance from a center point. A circle is actually a special case of an ellipse. In an ellipse, if you make the major and minor axis the same length, the result is a circle, with both foci at the center.

You draw a circle when you need a figure forming a closed loop where every point is a fixed distance from the center point.

To draw a circle:

1. In Graphics Editor, in the Layers pane, select the layer where you want to draw the circle.
2. On the Drawing toolbar, click **Ellipse**.
3. In the work area, click **Shift** while dragging the pointer from where you want the circle to begin to where you want it to end.
4. On the Drawing toolbar, click **Select**.
5. In the Properties pane, in the Name box, type the name of the circle.

   NOTE: You only need to name the drawing object if you will be binding the object. Naming the object now will help you identify the object later.

6. On the Options toolbar or in the Properties pane, adjust the appearance of the circle.
7. On the File menu, click **Save**.

Drawing an Arc or Pie

Arc

Use the Graphics Editor Arc tool to draw an arc, that is, a part of the periphery of an ellipse, or a circle.

The arc is defined by a center point, a radius X, a radius Y, a start angle, and a sweep angle. The start angle is the angle between the X-axis and the start of the arc. The sweep angle can lie in the interval ±(0° – 360°).

Pie

Use the Graphics Editor Pie tool to draw a pie, that is, an area enclosed by two radii of a circle and their intercepted arc.

The pie is defined by a radius X, a radius Y, a start angle, and a sweep angle. The start angle is the angle between the X-axis and the start of the arc. The sweep angle can lie in the interval ±(0° – 360°).

A pie is similar to an arc, but includes the two radii and the area within.

Arc or Pie

Use Arc or Pie to draw a curve-like segment, but with specified start and end points. The initial sweep angle is always 90°. The orientation of the 90° arc or pie corresponds to the position of the end point, related to the start point.

To draw an arc or pie:

1. In Graphics Editor, in the Layers pane, select the layer where you want to draw the arc or pie.
2. On the Drawing toolbar, select **Arc**. The cursor will change into a crosshair pointer.
3. In the work area, drag the pointer from where you want to start the arc or pie to where you want it to end.
4. On the Drawing toolbar, click **Select**.
5. In the Properties pane, in the Name box, type the name of the arc or pie.

   NOTE: You only need to name the drawing object if you will be binding the object. Naming the object now will help you identify the object later.

6. On the Options toolbar, or in the Properties pane, adjust the appearance of the arc or pie.
7. On the File menu, click **Save**.

Editing an Arc or Pie

You edit the start and sweep vertexes of an arc or pie to change its angle.

To edit an arc or pie:

1. In Graphics Editor, in the work area, double-click anywhere on the arc to display the vertexes.
2. Select the vertex you want to edit and drag it to a new position.

   NOTE: You can extend the angle handle to increase the precision. This does not affect the arc in other ways. If you press the **Shift** key while moving the cursor, the angle changes in steps of 7.5°, somewhat depending on how much you extend the handle.

3. Click outside the arc to finish.

Adding Text and Textboxes

Use the Graphics Editor Text tool to write a single line of text with no wrapping. Use the Graphics Editor Textbox tool to write one or several lines of text that are wrapped within the specified box.

You can edit and format the text by using the standard formatting tools.

A text path is a free form curve of text. You use the text path to make the characters independent of any font library. The disadvantage of this is that you can no longer edit the characters as text.

When you create a path of text, you create a copy, which can be treated as an ordinary closed curve. You can set Stroke and Fill color for the text path.

The original text remains unchanged and if required you can delete it.

Text Tool

Text is typically used for adding labels or informative comments within your graphic. You add a single line of text using the Text tool. Textboxes are used when you need to wrap text.

To add text using the Text tool:

1. In Graphics Editor, in the Layers pane, select the layer you want to add the text on.
2. On the Drawing toolbar, click **Text**.
3. Click in the work area where you want the text to start.
4. Type the text you want to add to the graphic.
5. Press **Enter**.
6. On the Drawing toolbar, click **Select**.
7. In the Properties pane, in the Name box, type the name of the text.

   NOTE: You only need to name the drawing object if you will be binding the object. Naming the object now will help you identify the object later.

8. On the Options toolbar or in the Properties pane, adjust the appearance of the text.
9. On the File menu, click **Save**.

You can make text content dynamic so that the text changes according to the value of the variable it is bound to. This way you use only one text object to show different texts depending on the value of the variable.

To make text content dynamic:

1. In Graphics Editor, in the Layers pane, select the layer that contains the text you want to make dynamic.
2. In the work area, select the text you want to make dynamic.
3. In the Object pane, right-click **Text**, point to **New**, and then click **Bind**.
4. In the Properties pane, in the **Name** box, type the name of the Bind object.
5. In the Attribute box, select **Content**.
6. In the Objects pane, right-click **Bind**, point to **New**, and then click **ConvertValue**.
7. In the Properties pane, in the **Name** box, type a name for the ConvertValue object.
8. In the AttributeValue box, type the text that you want to display in the graphic.
9. In the SignalEqualTo box, type the value when the text is to be displayed.

10. Add more ConvertValue objects to the Bind object, one for each value of the variable that is to be displayed as text.
11. In the Objects pane, right-click the **Text**, point to **Group as**, and then click **Component**.
12. In the Properties pane, in the **Name** box, type the name of the component.
13. On the File menu, click **Save**.

Textbox Tool

You add text within a textbox when you want to add several lines of text with automatic line wrap within a defined area.

To add text using the Textbox tool:

1. In Graphics Editor, in the Layers pane, select the layer you want to add the text on.
2. On the Drawing toolbar, click **Textbox**.
3. In the work area, click where you want to locate the upper-left corner of the textbox.
4. Drag the pointer to where the lower-right corner of the textbox is to end.
5. Type the text you want to add to the graphic.
6. On the drawing toolbar, click **Select**.
7. Adjust the size of the textbox.
8. In the Properties pane, in the Name box, type the name of the textbox.

   NOTE: You only need to name the drawing object if you will be binding the object. Naming the object now will help you identify the object later.

9. On the Options toolbar or in the Properties pane, adjust the appearance of the text.
10. On the File menu, click **Save**.

Editing Text or Textboxes

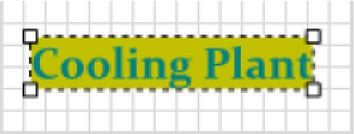
You can reformat single line texts and texts within textboxes.

To edit text:

1. In Graphics Editor, in the work area, select the text or textbox object containing the text you want to edit.

2. On the Options toolbar, expand Stroke and select a new text color from the Color palette.
3. On the Options toolbar, expand Fill and select a new background color for the textbox from the Color palette.
4. Double-click the text/textbox object and edit the text by adding or deleting text.

5. On the Options toolbar, set font, font size, and other text properties.



6. When finished, click outside the text/textbox.

Inserting Pictures

Use the Graphics Editor Insert Pictures tool to insert supported image files into your graphic.

The native Graphics Editor file formats are . *.gif, *.jpg, and *.png files. That is, pasted components and graphics will be saved as one of these file types.

You can insert the following file formats into a graphic:

Graphic Type	File Type
Bitmap	*.bmp
Bitmap	*.dib
GIF	*.gif
Icon	*.ico
JPEG	*.jpg
JPEG	*.jpeg
JPEG	*.jpe
JPEG	*.jfif
JPEG	*.exif
Metafile	*.wmf
Metafile	*.emf
PNG	*.png
TIFF	*.tiff
TIFF	*.tif

A raster image can be inserted anywhere in the work area. When you add an image, the actual image is saved with the graphic. That is, the image is not linked into the graphic.

NOTE: Raster images put a heavy load on graphics handling. To minimize system load, reduce size and color depth of the images before inserting them into a graphic.

Use an image editor to resize the image to the size required in the graphic. If you use the .jpg format, the image can be compressed to a quality of 60% without any adverse effects on the appearance.

If you want to use transparency, you should add it to the original image. This can be done if you use the .png format.

NOTE: You can paste any picture residing on the clipboard to the work area. You can also drag pictures to the work area.

To insert a picture

1. In Graphics Editor, in the Layers pane, select the layer where you want to add the picture.
2. On the Drawing toolbar, click **Insert Picture**.
3. In the work area, click where you want to locate the upper-left corner of the picture.
4. Select the picture you want to insert into the graphic.
5. On the Drawing toolbar, click **Select**.
6. In the Properties pane, in the Name box, type the name of the picture.

   NOTE: You only need to name the drawing object if you will be binding the object. Naming the object now will help you identify the object later.

7. On the Options toolbar or in the Properties pane, adjust the appearance of the picture.
8. On the File menu, click **Save**.

Adjusting a Picture

To adjust a picture

1. In Graphics Editor, in the work area, select the picture you want to adjust.
2. Drag the picture to reposition it.
3. Press **Shift** while dragging one of the corner handles to resize the picture but keep the aspect ratio.
4. To change the opacity or visibility for the picture, in the Properties pane, in the Appearance area, select the corresponding elements and enter new values.

   NOTE: For performance reasons, it is strongly recommended that you edit the picture before inserting it into the graphic.

5. On the File menu, click **Save**.

Adding an Animated Picture

You insert an animated picture when you want to add an animated image, such as a .gif file, into a graphic.

To add an animated picture:

1. In Graphics Editor, in the Layers pane, select the layer where you want to add an animated image.
2. On the Drawing toolbar, click **Insert Picture**.
3. In the work area, click where you want to locate the upper-left corner of the animated image.
4. Select the animated picture you want to insert into the graphic.
5. On the Drawing toolbar, click **Select**.

6. In the Properties pane, in the Name box, type the name of the animated image.

   NOTE: You only need to name the drawing object if you will be binding the object. Naming the object now will help you identify the object later.

7. On the Options toolbar or in the Properties pane, adjust the appearance of the animated image.
8. On the File menu, click **Save**.

You can make an animated image dynamic so that the animation can start and stop according to the value of the variable it is bound to. This way you use the animation in the image instead of using components.

To make an animated picture dynamic:

1. In Graphics Editor, in the Layers pane, select the layer where you want to make an animated image dynamic.
2. In the work area, select the picture you want to make dynamic.
3. In the Objects pane, right-click **AnimatedImage**, point to **New**, and then click **Bind**.
4. In the Properties pane, in the Name box, type the name of the Bind object.
5. In the Attribute box, select **Animation**.
6. In the Objects pane, right-click **Bind**, point to **New**, and then click **ConvertValue**.
7. In the Properties pane, in the Name box, type the name of the ConvertValue object.
8. In the AttributeValue box, select **Start**.
9. In the SignalEqualTo box, type the value that should start the animation.
10. In the Objects pane, right-click **Bind**, point to **New**, and then click **Convert Value**.
11. In the Properties pane, in the Name box, type the name of the ConvertValue.
12. In the AttributeValue box, select **Stop**.
13. In the SignalEqualTo box, type the value that should stop the animation.
14. On the File menu, click **Save**.

Graphic Object Position

You can check the position and size of a graphic object using the graphic object position bar. The size and position is displayed in pixels.

If the objects are grouped together the position and size for the whole group is displayed. If you multi-select objects the position and size is displayed for all selected objects.

This feature is only available in Design mode.

For more information, see the Graphics User Interface Overview section.

Attributes Overview

Each element in the Objects pane has a number of properties. The Graphics Editor element properties are referred to as attributes, in compliance with XML standards. The attributes are displayed in the Properties pane, where they can be edited. Attributes are used to give a complete description of a graphic element. Most of the attributes are automatically defined when the graphic element is created. By changing the attributes, you can change, for example, the appearance and behavior of a graphic element.

The Properties pane has two modes, where you can define which level of detail you want displayed:

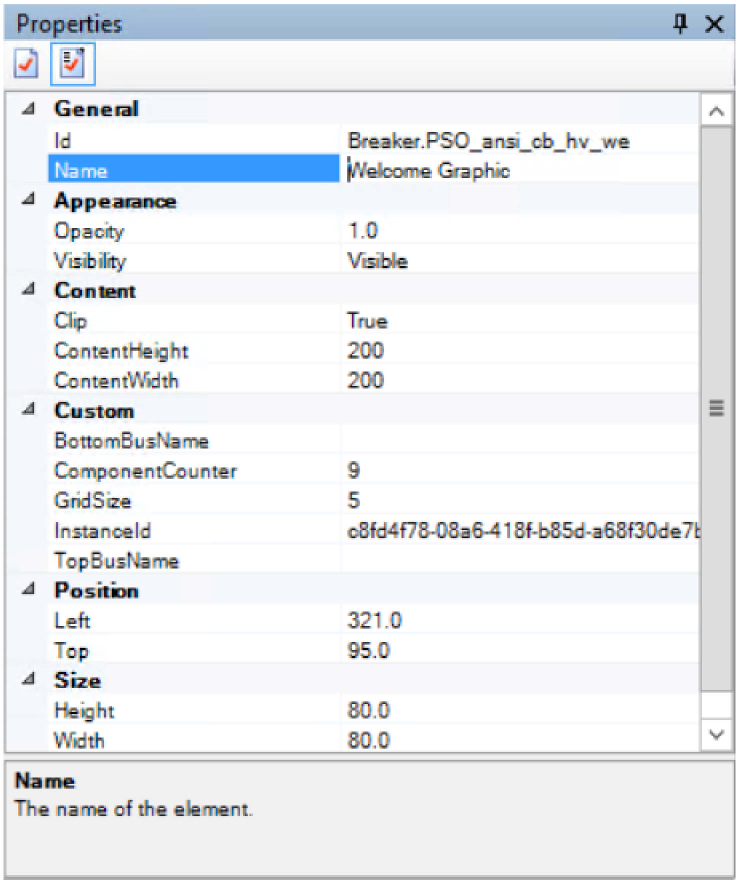
- Normal
- Detailed

Normal mode displays the most commonly edited attributes. Detailed mode displays all attributes.

You can add attribute values, for example, Name. Some attributes describe a dynamic behavior and the attributes become apparent only when the graphic is used in a dynamic environment, for example, as an online graphic accessible in the Diagrams viewer.

For some of the attributes in the Properties pane, a label is displayed at the bottom of the pane when you select the attribute.

Example: Attribute label in the Properties pane



Properties

General	
Id	Breaker.PSO_ansi_cb_hv_we
Name	Welcome Graphic
Appearance	
Opacity	1.0
Visibility	Visible
Content	
Clip	True
ContentHeight	200
ContentWidth	200
Custom	
BottomBusName	
ComponentCounter	9
GridSize	5
InstanceId	c8fd4f78-08a6-418f-b85d-a68f30de7
TopBusName	
Position	
Left	321.0
Top	95.0
Size	
Height	80.0
Width	80.0

Name
The name of the element.

Often an information text, with information on how the specific attribute gets its value, is displayed below the label:

(blank)

The value has been chosen when the element was created and applies to this element.

Default value

The value was set by default when the element was created and applies to this element.

Inherited value

The element is a part of a parent element and has inherited its value from the parent.

This information can be useful when you create more complex graphics, where attribute inheritance is used. For more information, see the Inherited Attributes section.

Different elements and items have different attributes, which are described with each item, but some general rules apply.

There are different categories of graphic element attributes:

- Generic Attributes
- Appearance, Position and Size attributes
- Behavior, Boundary and Target attributes

For more information, see the following topics:

- Graphic Object Attributes
- Inherited Attributes
- Exposed Attributes

Graphic Object Attributes

All graphic elements that you use in a graphic have attributes, that is, properties that describe the element. For example, shape, position, appearance, and dynamic behavior.

When you select a graphic element, all its element attributes are displayed in the Properties pane.

You can change most element attributes from the options bar and the associated menus. However, sometimes it is more convenient—or gives more precision—to enter the attribute values directly in the Properties pane.

For more information, see the following sections:

- Generic Attributes
- Inherited Attributes
- Exposed Attributes
- Modifying the Behavior of a Component

Generic Attributes

All items have two generic attributes:

Property	Type	Description
ID	String	The identity of the element. Reserved for scripts and other entities that need to use unique element identifiers to access specific elements.

Property	Type	Description
Name	String	The name of the element. The primary use is to identify exposed elements such as Bind.

For Bind elements, the Name is displayed in the Binds and Links pane. From there the Bind element cannot be connected to external signals. The actual binding is performed manually or from the Binding pane.

Inherited Attributes

When you design a graphic, you can establish an inheritance to apply an attribute of a parent or ancestor element on one or several child elements. A parent element can be, for example, a Group element. A child element can be, for example, a graphic element within a group.

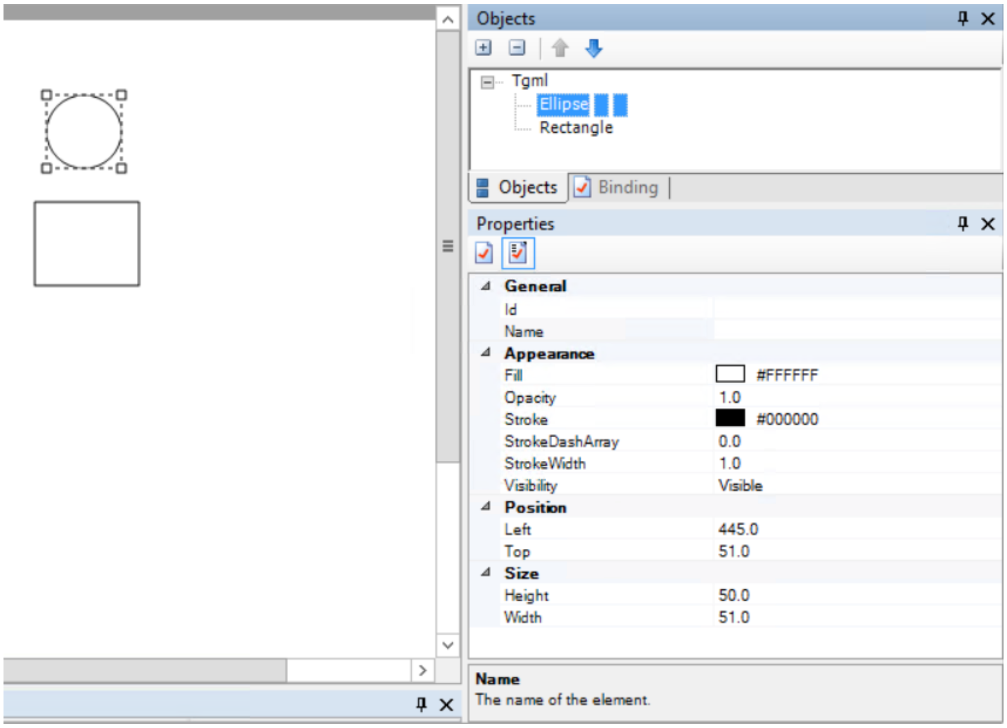
If you have set up an inheritance and have defined an attribute for a Group object, for example, Fill color, the fill color is applied on all the individual graphic elements in the group.

Inheritance only applies if you remove the corresponding attribute on the child element. Conversely, if you keep the attribute on the child element, it overrides the attribute of the parent element.

If an attribute is left undefined both in the child element and all its parents, a default attribute is used.

If you want to determine the origin of an attribute, you can click the attribute in the Properties pane and then read the text in the gray box beneath. For more information, see the Attributes Overview section.

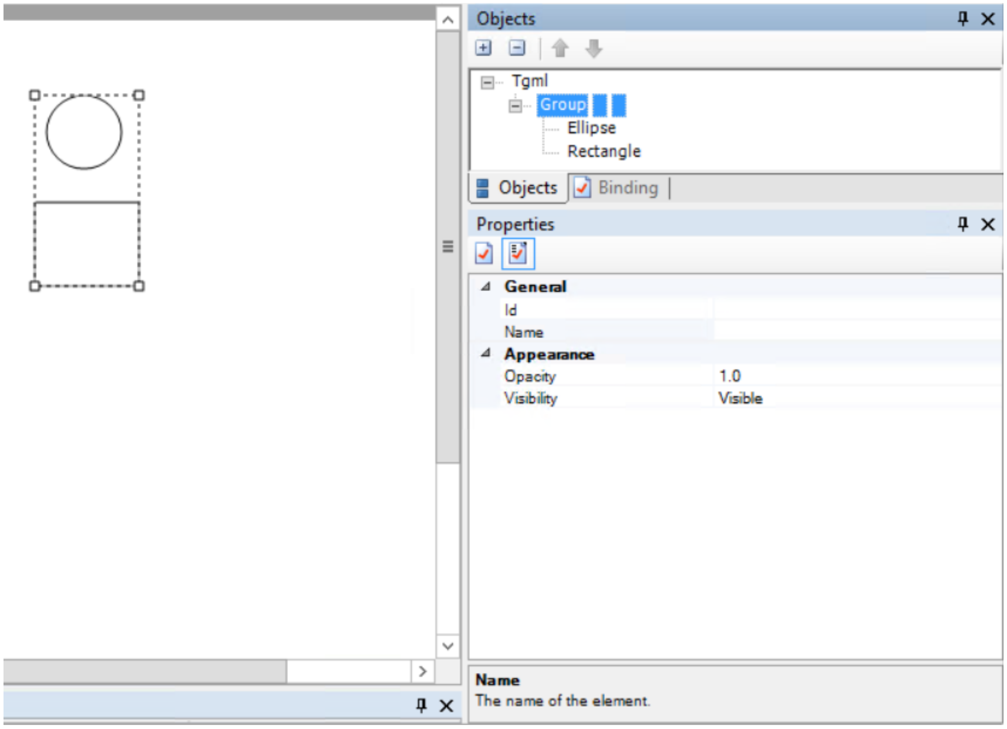
Example: If you create a rectangle and a circle (ellipse), both get the same Fill and Stroke attribute values from the values in the Options toolbar.



General	
Id	
Name	
Appearance	
Fill	□ #FFFFFF
Opacity	1.0
Stroke	■ #000000
StrokeDashArray	0.0
StrokeWidth	1.0
Visibility	Visible
Position	
Left	445.0
Top	51.0
Size	
Height	50.0
Width	51.0

Name
The name of the element.

Example: If you group the elements, the group only has two Appearance attributes: Opacity and Visibility, and both get default values.



General
Id
Name
Appearance
Opacity 1.0
Visibility Visible

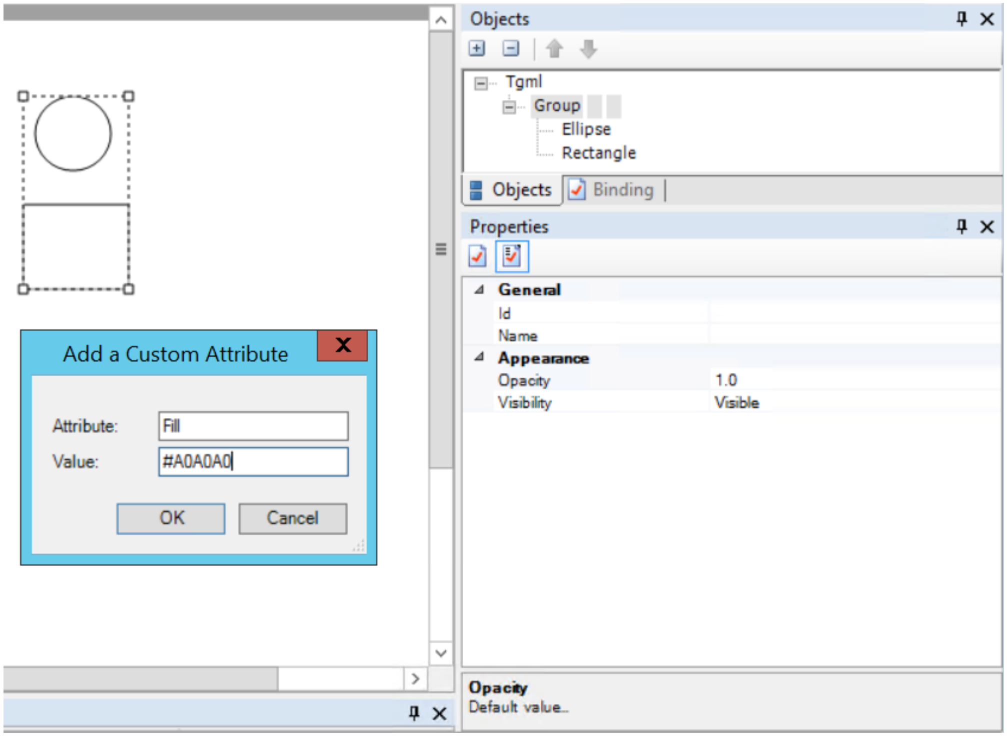
Name
The name of the element.

To define inheritance of Fill and Stroke from the Group element to the two constituting elements, two things have to be done:

- Create (or Add) the missing attributes in the Group element.
- Remove the Fill and Stroke attributes from the rectangle and ellipse.

Right-click the **Group** element in the Properties pane, select **Add** and enter the attributes; first **Fill** and then **Stroke**, and some suitable values.

Example: Attributes added to Group object.



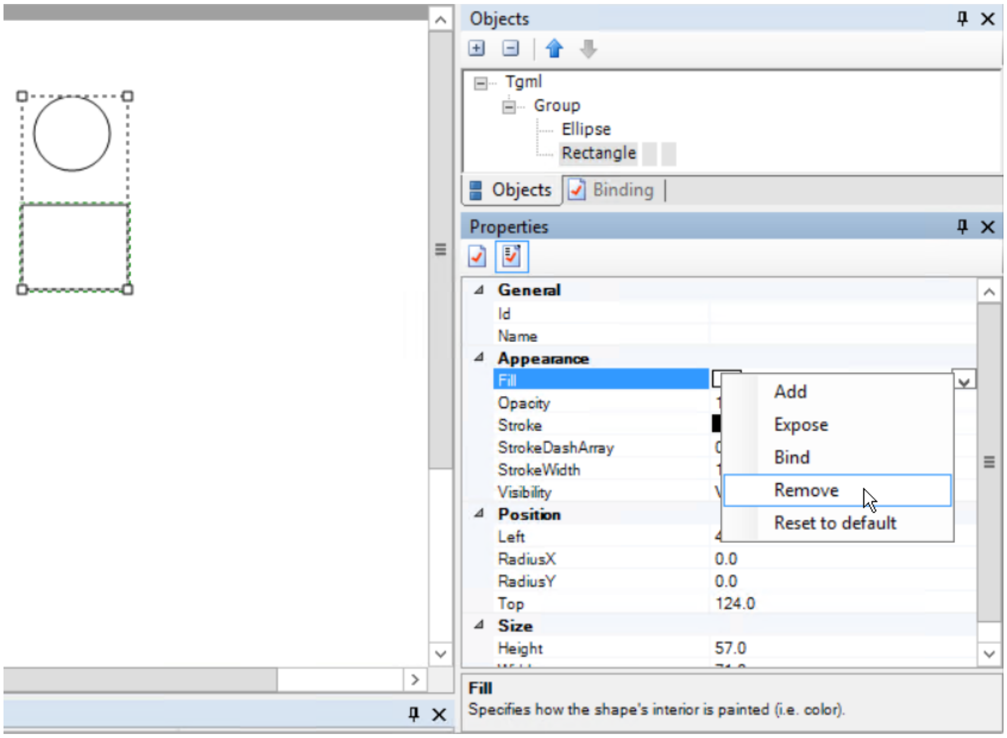
Attribute: Fill
Value: #A0A0A0

Opacity 1.0
Visibility Visible

Opacity
Default value.

In the tree structure in the Objects pane, select the rectangle, right-click the **Fill** attribute and select **Remove**. The rectangle immediately inherits the Fill attribute from the Group element.

Example: The graphic figure's own attribute is removed.



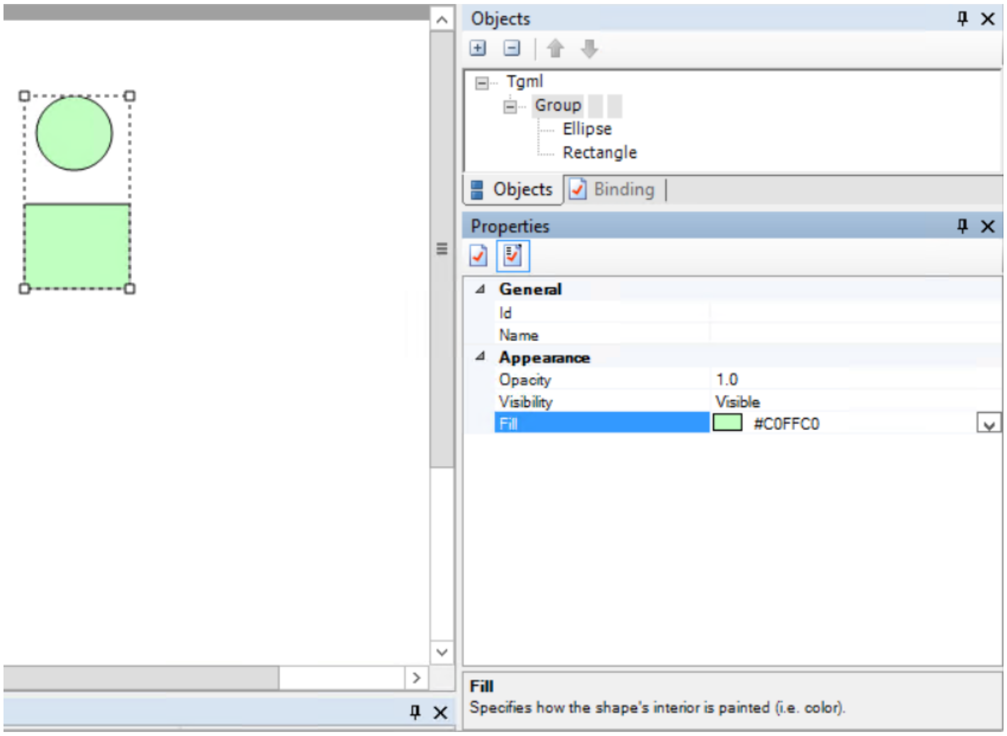
RadiusX 0.0
RadiusY 0.0
Top 124.0
Height 57.0

Fill
Specifies how the shape's interior is painted (i.e. color).

Repeat the procedure for the ellipse.

Now you can select the group and change its Fill attribute. The Fill value of the two elements in the group is also changed.

Example: The graphic figure in the group inherits the Fill attribute from the Group element.



NOTE: If you do not know the value (for example, a color code), you can leave the value box empty. In this case, when you add the attribute, the value box in the Properties pane indicates an invalid (empty) value. Use the drop-down menu to select a valid value.

For more information, see the following sections:

- Defining Inheritance
- Setting Up Inherited Attributes

Defining Inheritance

You define inheritance to enable attribute inheritance from an element containing other elements, for example, to control a specific attribute of an entire group of elements.

To define inheritance:

1. In Graphics Editor, in the Objects pane, select the parent element from which you want to use an attribute, for example, **Group**.
2. In the Properties pane, right-click anywhere and then click **Add**.
3. In the Add a Custom Attribute dialog box, type a name for the attribute that you want child elements to inherit, for example, **Fill**.
4. In the Objects pane, select the elements that are to inherit attributes from the container element to which they belong.
5. In the Properties pane, delete the attribute that you want the container element to control.
6. Click **Remove**.

NOTE: Make sure that the attribute that you add on the container element has a corresponding attribute on the inheriting elements. For example, Fill. Corresponding attributes must have identical names.

NOTE: You can override the inheritance from a container element by keeping the attribute on an element that belongs to the container element. This is useful when you want a group of elements in the same container element to inherit an attribute, but have one or a few elements in that container element, which should keep their individual attributes.

Setting Up Inherited Attributes

You set up inherited attributes to make sure that a component gets the same attributes as lead or ancestor components.

NOTE: If an attribute for a shadow object is removed, the object inherits the corresponding attribute from its lead object.

For more information, see the Designing Components section.

To set up inherited attributes:

1. In Graphics Editor, in the Objects pane, select **ComponentContent**.
2. In the Properties pane, right-click the attribute field, and then click **Add**.
3. In the Add a Custom Attribute dialog box, enter the name of the attribute that is to be inherited and then enter its initial value.

   NOTE: If you do not know the value (for example, a color code), you can leave the Value field blank. In this case, when you add the attribute, the Value field in the Properties pane indicates an invalid (empty) value. Use the drop-down menu to select a valid value.

4. In the Objects pane, select the object whose attribute should be inherited.
5. In the Properties pane, right-click the attribute or attributes that are to be inherited and then click **Remove**. The attribute does not disappear, but the text at the bottom of the Properties pane changes to Inherited value.
6. On the File menu, click **Save**. Inherent objects get a uniform appearance.

Exposed Attributes

You can make certain attributes of a component accessible from outside the component. For example, you can use the Fill attribute to let an external signal change the color of a component. You add Expose as a separate element to an object that has an attribute you want to make accessible in your component.

An exposed attribute is displayed among the attributes of its parent figure and ancestors all the way up to the root figure (Tgml) of a graphic, in the Exposed Properties part of the Properties pane.

The Expose element has two attributes:

- **Name:** The name of the exposed attribute.
- **ExposedAttribute:** The attribute that is exposed.

NOTE: When naming exposed attributes, note that if two or more exposed elements have the same name, this is considered to be intentional. It means that the named element is displayed only once in the Exposed Properties part of the Properties pane.

For more information, see the following topics:

- Adding an Expose Element
- Exposing an Attribute

Adding an Expose Element

You use exposed attributes to make certain attributes of a component accessible from outside Graphics Editor.

For more information, see the Exposed Attributes section.

To add an Expose object:

1. In Graphics Editor, in the Properties pane, right-click the object with the attribute you want to expose, point to **New**, and then click **Expose**.
2. In the Objects pane, in the component tree structure, select the new Expose element and change the name from My____ to a more descriptive name.
3. In the Properties pane, click the **ExposedAttribute** box, and select the attribute you want to expose from the drop-down list.

If you select the object containing the exposed attribute in the Objects pane, the exposed attribute is displayed in the Properties pane.

Exposing an Attribute

Use exposed attributes to make certain attributes of a component accessible from outside Graphics Editor.

NOTE: If an attribute for a low-level object is removed, the object inherits the corresponding attribute from its parent object.

For more information, see the Component Design section.

To expose an attribute:

1. In Graphics Editor, in the Objects pane, select the component containing the attribute you want to expose.
2. In the Properties pane, right-click the attribute you want to expose and then click **Expose**.

   NOTE: An ExposedAttribute element is added in the Properties pane.

3. In the Objects pane, select the new **Expose** element and change the name from **My____** to

a more descriptive name.

> **NOTE**: You can expose several attributes at a time.

If you select the object containing the exposed attribute in the Objects pane, the exposed attribute is displayed in the Properties pane.

Modifying the Behavior of a Component

You expose the properties of a component to modify its behavior.

For more information, see the Exposed Attributes section.

To modify the behavior of a component:

1. In Graphics Editor, on the File menu, point to **New** and then click **Component**.
2. In the Components pane, click the **My Components** category.
3. Select the required component, and drag it to the work area.
4. In the Properties pane, in the **Name** box, type the name of the modified component.

   > **NOTE**: You only need to name the drawing object if you will be binding the object. Naming the object now will help you identify the object later.

5. In the Properties pane, type values for the **Exposed Properties** object.
6. Select **Preview**.
7. In the Binds and Links pane, enter values to test the modified behavior.

   > **NOTE**: The bindings name consists of the component name, a dot, and the bind name 'Value'.

8. On the File menu, click **Save**.

By modifying exposed properties of a component, you can customize it when reusing the component in different applications.

Binding and Linking Overview

A dynamic graphic object can be bound to, and thus controlled by, Power SCADA server variables (signals). When the signal changes, the behavior or appearance of the object changes dynamically.

Similarly, you can link a graphic object to other database objects or Server pages by using a Link property to define a target.

Binds and links are created by using the Bind or Link elements of the graphic object. Binds and links defined for a graphic are displayed in the Binds and Links pane. Although you have to do the physical binding and linking in the Graphics Editor Binding pane, the Binds and Links pane gives you a useful overview.

When you select binds or links, the different panes highlight the selected elements, making them easy to locate. The binds and links remain selected even if you toggle between panes.

You can test bindings in the Graphics Editor in Preview mode, by manually entering test values for the binding to test the effect of the rules for a binding. For more information, see the Testing a Graphic section.

For more information on binding, see the following sections:

- Object Binding
- Object Linking
- Dynamic Updates

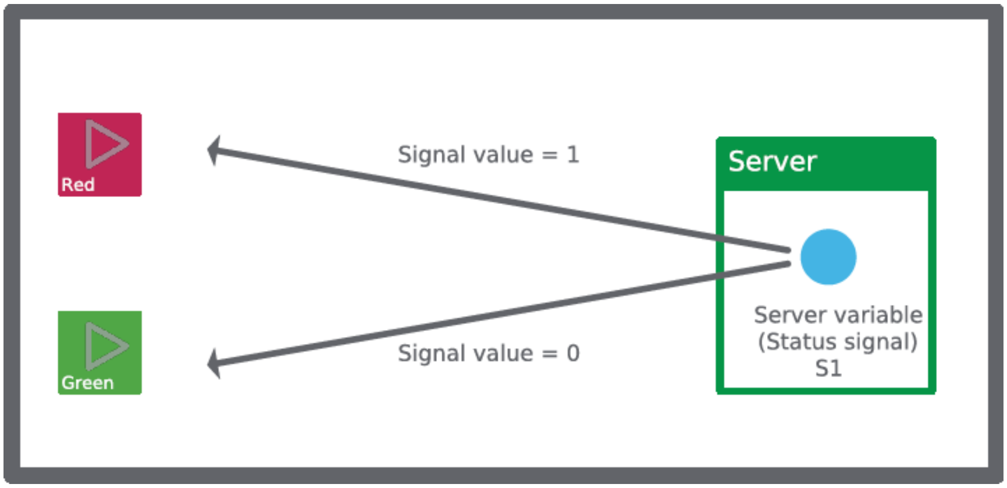
For more information on binding properties and more, see the Diagrams reference Library Components section.

Object Binding

A dynamic graphic object is an object whose appearance or behavior is controlled by variables from the server. This is done by binding elements of the graphic object to the variable.

A binding can simply reflect the variable value, for example, a symbol toggles between green when a status signal is 0 and red when the signal is 1.

Example: The desired dynamic behavior of a graphic object:

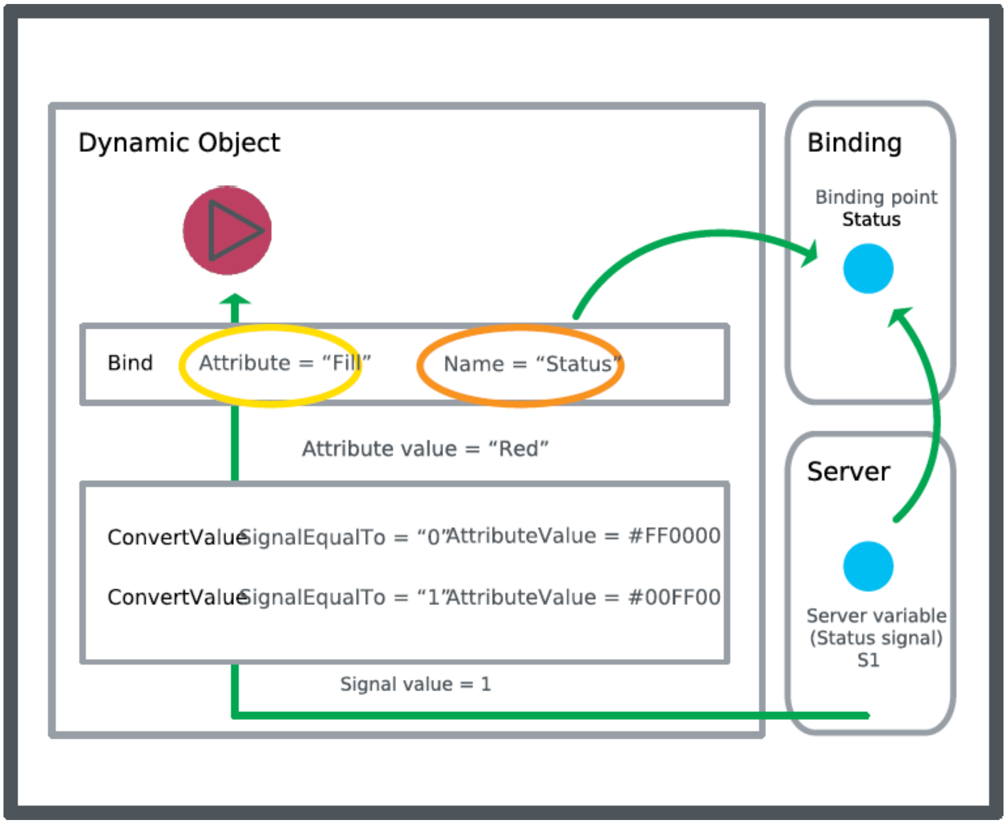


A binding can also contain converters (rules) that declare how the appearance or behavior of the graphic object should be affected if the variable value changes.

You can add bindings to, for example, lines, curves, and rectangles. You can also add bindings to transformations of objects, for example, rotate, scale, or translate elements.

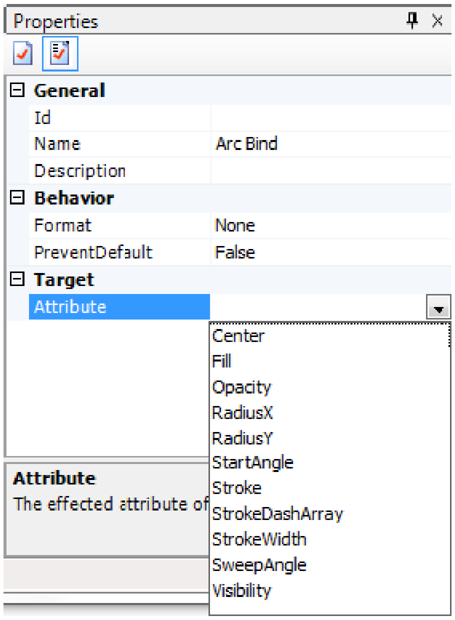
Bindings are made by adding a Bind element to the graphic object. The Bind element has a Target attribute that is changed as the incoming value from the Server changes.

Example: Graphics object properties bound to signals in the system:



You select the Target attribute from the Attribute drop-down list, which only shows the bindable properties of the parent object.

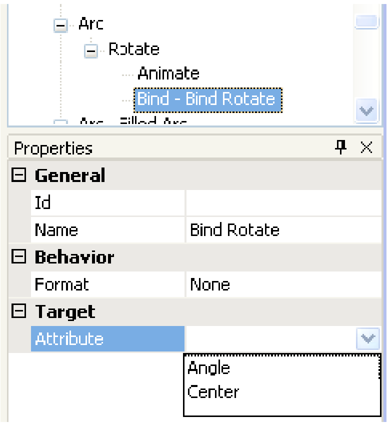
Example: When you add a Bind element to an arc, a certain number of arc attributes can be bound:


Properties

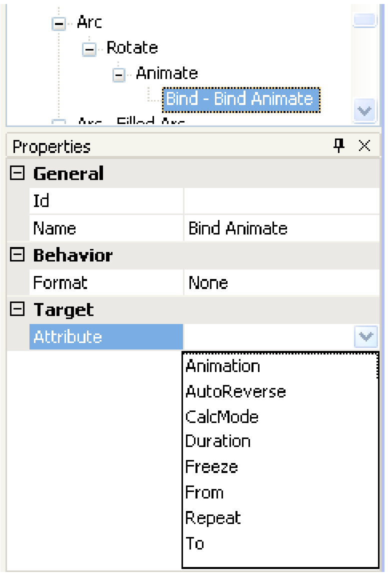
General	
Id	
Name	Arc Bind
Description	
Behavior	
Format	None
PreventDefault	False
Target	
Attribute	

Attribute
The effected attribute of

Example: When you add a Bind element to a Rotate element, only the rotation angle and center attributes can be bound:



Example: When you add a Bind element to an Animate element, only the animation attributes can be bound:



The remaining Bind property is Format. You can set it to **None** (deliver value as is) or Presentation. If you select Presentation, the value is converted to and presented as text.

For more information, see Adding a Bind.

Adding a bind

You add a bind when you want to bind an attribute of a dynamic graphic object to, and thus let it be controlled by, server variables (signals). When the signal changes, the behavior or appearance of the object changes dynamically.

For more information on binds, see Binds and Links Overview.

To add a bind:

1. In Graphics Editor, in the Objects pane, right-click the object to which you want to add the bind.
2. Point to **New** and then click **Bind**.

3. In the Properties pane, in the Attribute box, enter the object attribute you want to be affected by the server signal.

Before you can bind the graphic object attribute to a signal in Diagrams, you have to add, for example, a value converter to the bind.

Object Linking

If you want to dynamically link a graphic object to, for example, other graphic objects, trend log views, notes, and online plots, you add a Link element to the graphic object.

Linking is done by using the Link element of the graphic object. You have to set the Target property of the Link element to define the target that is to be displayed when the user clicks the link in the graphic object.

Adding a Link

You add links to a graphic to open Web sites, applications, or documents from within Diagrams.

For more information, see the Binds and Links Overview section.

To add a link:

1. In Graphics Editor, click the **Snippets** tab.
2. Select the **Basic Functions** category.
3. Drag the **Link** snippet to the drawing object that will be linked to a Web site, application, or document.
4. In the Objects pane, click the **Link** object.
5. In the Properties pane, in the **LinkName** box, type the name of the Link object.
6. Click **Save**.

The link is now added to the graphic and is available for binding.

Dynamic Updates

A graphic can contain numerous bindings to values that you want to be able to display in the graphic. When there are numerous bindings, reading the values can be time-consuming and can slow down the performance of the graphic. By setting the DynamicUpdates attributes in the Bind elements of the graphic to either disabled or enabled, you can control which values are updated. That is, which values the graphic subscribes to when the graphic is opened.

Disabling the updates can improve the performance when loading the graphic. By default, the dynamic updates are enabled, that is, the value connected to a Bind element in the graphic is updated when the graphic is opened.

To control how and when the values of bindings that are not dynamically updated are to be updated, you can add a script to the Script element of the Bind element.

For more information, see the following section:

- Activating a Binding with a Dynamic Update Attribute

Activating a Binding with a Dynamic Update Attribute

You edit the DynamicUpdate attribute to improve the performance when loading a graphic. By default, the dynamic updates are enabled, that is, the value connected to the Bind element of the graphic is updated when the graphic is opened.

For more information, see the Dynamic Updates section.

To activate a binding with a dynamic update attribute:

In Graphics Editor, in the Objects pane, select the Bind element for which you want to edit the DynamicUpdates attribute.

In the Properties pane, in the Behavior area, select the DynamicUpdates attribute and select Enable or Disable depending on whether or not you want the graphic to subscribe to the value connected to the Bind element when the graphic is opened.

Graphics Editing Tools Overview

You can edit all graphic objects, that is, modify their properties. For example, shape, size, and color.

You can edit all graphic objects individually. You can also edit some graphic objects simultaneously. You can edit the following properties of two or more graphic objects simultaneously:

- Fill color
- Stroke color
- Stroke style
- Stroke width
- Flip
- Rotate
- Skew

For text boxes, you can also edit the following properties for several text boxes simultaneously:

- Font
- Font size
- Font style
- Stroke width
- Horizontal text alignment in text box
- Vertical text alignment in text box

If you want a graphic object to be available for future use, you can save it as a component in the My Components library.

For more information, see the following sections:

- Layers
- Groups

- Organizing Objects
- Adding Custom Colors
- Gradients Overview
- Adding Animations
- Adding Paths
- Using the Grid

Organizing Objects

When organizing objects in a graphic there are many options. You can move, align, arrange, and distribute objects in the work area. You can use the tools on the toolbar or any of the features available from the panes.

For more information, see the following sections:

- Moving Objects
- Aligning Objects
- Arranging Objects
- Distributing Objects
- Resizing Objects
- Rotating Objects
- Skewing Objects
- Flipping Objects
- Copying an Object
- Editing Objects
- Deleting an Object

Moving Objects

You can move drawn objects individually, or collected in arbitrary groups.

You move objects to place them at a new position in a design.

To move objects:

1. In Graphics Editor, in the work area, select the object you want to move.
2. Drag the object to its new position.

NOTE: You can move multiple objects by selecting them all. You can also move the selected object(s) by using the arrow keys.

Aligning Objects

You can align two or more selected objects in seven different ways:

- **Left:** Horizontally along the left edge of the objects
- **Center:** Horizontally along the center of the objects

- **Right:** Horizontally along the right edge of the objects
- **Top:** Vertically along the top edge of the objects
- **Middle:** Vertically along the middle of the objects
- **Bottom:** Vertically along the bottom edge of the objects
- **Center Middle:** Horizontally along the center of the objects, and vertically along the top edge of the objects

NOTE: All alignments refer to the object considered to be the primary selection. Which object is regarded as the primary selection depends on how the objects are selected. For more information, see the Groups section.

NOTE: The Align drop-down menu options indicate how the objects can be positioned relative to the primary selection.

You align two or more objects to position them evenly.

To align objects:

1. In Graphics Editor, in the work area, select the objects you want to align.

   NOTE: Make sure the object that controls the alignment is the primary selection. The primary selection is enclosed within a blue, dashed rectangle.

2. On the Options toolbar, click the **Align** button and select the required alignment from the drop-down menu.

Arranging Objects

If an object more or less overlaps another object, you may want to arrange them. You can move an object so that it appears behind or in front of other objects.

You can also move a graphic object to the very back or the very front of the stack.

You can arrange one or more selected objects in four different ways:

- **Bring to Front:** Bring the object(s) to the top position
- **Bring Forward:** Bring the object(s) one position up
- **Bring Backward:** Bring the object(s) one position down
- **Bring to Back:** Bring the object(s) to the bottom position

NOTE: When you select more than one object, the selected objects keep their internal order during the arrangement procedure.

NOTE: The Arrange menu options list shows how the objects can be arranged.

You arrange objects that more or less overlap each other to put certain objects in front of or behind other objects.

To arrange objects:

1. In Graphics Editor, in the work area, select the object that you want to move backward or forward.
2. On the Options toolbar, click **Arrange**, and then click the required option.

Distributing Objects

You can distribute three or more selected objects in two directions:

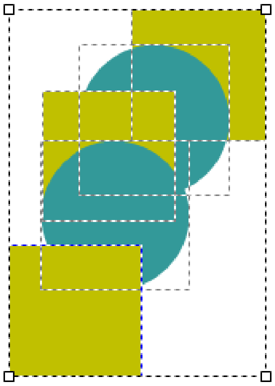
- **Horizontal:** Evenly distributed between the leftmost and the rightmost of the selected objects
- **Vertical:** Evenly distributed between the selected top object and bottom object

NOTE: The Distribute menu options list how the objects can be distributed..

You distribute three or more objects to position them evenly in a design, based on the center points of the objects.

To distribute objects:

1. In Graphics Editor, in the work area, select the objects you want to distribute.



2. On the Options toolbar, click the **Distribute** button and then click **Horizontal** or **Vertical** distribution.

NOTE: When the objects are distributed, the objects' center points, that is, the vertical or horizontal middle, are used. The result becomes most apparent when objects of different sizes are distributed..

For more information, see the following sections:

- Arranging a Table-Like Layout
- Duplicating Objects

Arranging a table like layout

You position objects in a table-like layout to get evenly spaced rows and columns.

To arrange a table-like layout:

1. In Graphics Editor, in the work area, select the objects for the top row, in order from left to right.
2. On the Options toolbar, click the **Align** button and then click **Top**. The selected objects align to the same horizontal height.

3. Select the objects that belong to the leftmost column, in order from top to bottom.
4. On the Options toolbar, click the **Align** button and then click **Left**. The selected objects align to the same left, vertical line.
5. Depending on the pattern and row/column distance you want, group the objects, row-by-row or column-by-column.
6. Position the top/bottom rows, or the leftmost/rightmost columns.
7. Select all groups, comprising either rows or columns.
8. On the Options toolbar, click the **Distribute** button and then click **Vertical** if the rows have been grouped, and **Horizontal** if the columns have been grouped.

The rows or columns are distributed evenly between the outermost rows/columns. If necessary, you continue to do adjustments by using the **Alig**n and **Distribute** commands on individual objects or groups.

Duplicating objects

You can copy any object or component to the same position as the original by working with the original object from Objects pane tree structure.

To duplicate an object to the same position:

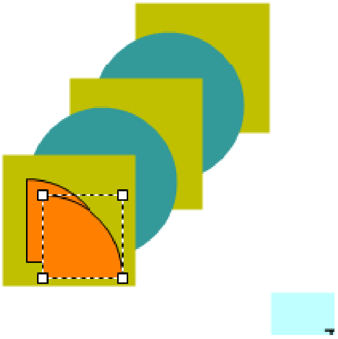
1. In Graphics Editor, in the Objects pane, select the object.
2. Press CTRL and drag the object to a new position in the tree structure.

A copy of the object is created directly on top of the original object in the drawing.

You can make equidistant copies of an object or component to get a certain conformity in a design. Specify the position of the first copy and use a special copy command to distribute the remaining copies as ‘extensions’ of the first copy operation.

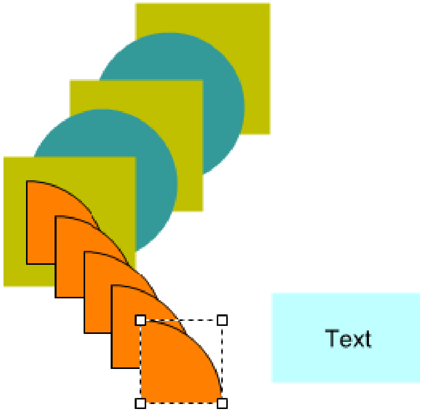
To duplicate an object with a controlled offset:

1. In Graphics Editor, in the Objects pane, select the object you want to copy.
2. On the Drawing toolbar, click **Copy**.
3. Press CTRL+V to paste a copy.



4. Press CTRL+SHFT and move the copy to its correct position, relative to the original.
5. Press CTRL+SHFT+V to paste another copy at the same distance relative to the previous copy.

6. Repeat for as many copies as you need.



A number of copies are created and they are all placed with the same offset from the previous object.

Resizing Objects

You can resize graphics objects, components and groups in the work area. There are two resizing methods:

- **General resizing:** Whatever is enclosed by a selection rectangle is resized
- **Resizing to same value:** The selected graphics object, component, or group is resized to the same size as another graphics object, component or group. The size can be compared in three different ways:
  - **Same width:** Resize the width of the graphics object, component, or group to the width of another graphics object, component, or group.
  - **Same height:** Resize the height of the graphics object, component, or group to the height of another graphics object, component, or group.
  - **Same width and height:** This action resizes the width and height of the graphics object, component, or group to the width and height of another graphics object, component, or group.

You resize an object to modify the size of the object.

To resize an object:

1. In Graphics Editor, in the work area, select the object you want to resize.
2. Click the border of the selection rectangle and drag until the object has the size you want.

NOTE: Pulling the “handles” at any of the corners of the rectangle affects the scaling both horizontally and vertically. Pressing **Shift** while resizing makes the scaling proportional, that is, resizes the object horizontally and vertically..

You can resize an object to the same value to transform the object to the same size as another object.

To resize an object to the same size:

1. In Graphics Editor, in the work area, select the objects you want to align.
2. Press **Ctrl** while selecting the object you want to use as a size template.

   NOTE: The last selected object is used as a size template. The size template is enclosed in a blue rectangle.

3. On the Options toolbar, click the **Resize** button and then click the resize option you want: **Same width**, **Same height**, or **Same width and height**.

Rotating Objects

You can rotate objects in different ways, by using the rotation tool or the two rotation commands. Rotating multiple objects is somewhat different from rotating a single object. When you rotate multiple objects they are rotated as if they were grouped. That is, with a common center point around which the objects are rotated.

You can rotate an object by using the **Rotate Selection** tool on the Options toolbar and dragging the rotation handle.
You can drag out the angle handle to increase the angle precision. This does not affect the rotation in other ways.

NOTE: The aspect ratio is automatically used when rotated objects are resized.

When you use the **Rotate Selection** tool, you can also change the center of rotation by clicking it and moving the entire rotation handle.

NOTE: Before changing the center of rotation, consider that the effect of future rotations can be difficult to anticipate.

If you move the center point of an object that has already been rotated, an additional rotation is added to the object and the moved center point applies to the new rotation that was added.

You can rotate a single object or multiple objects to change their orientation simultaneously. The rotate commands performs a 90° rotation each time the command is executed.

To rotate objects:

1. In Graphics Editor, in the work area, select the object(s) you want to rotate.
2. On the Options toolbar, click the **Rotate Left** button or the **Rotate Right** button.

Skewing Objects

Skewing an object means that you distort the shape in a horizontal or vertical direction by a number of degrees. You skew by selecting one or several objects and then applying a value from the horizontal or vertical skewing menu.

You skew horizontally to transform an object along the x-axis. The skew angle is measured in degrees from the y-axis. You skew vertically to transform an object along the y-axis. The skew angle is measured in degrees from the y-axis.

To skew horizontally or vertically:

1. In Graphics Editor, in the work area, select the object you want to skew.
2. On the Options toolbar, click the **Skew Horizontal** button or the **Skew Vertical** button to open the degree selection menu.
3. Click the required amount of degrees, –60° to +60°.

NOTE: A positive value implies counter-clockwise skew. A negative value implies clockwise skew.

Flipping Objects

Flipping an object means that you replace the object with a reflection of the original object on a horizontal or vertical axis.

You flip one or more objects to reflect them in a horizontal or vertical direction.

To flip an object:

1. In Graphics Editor, in the work area, select the object you want to flip.
2. On the Options toolbar, click the **Flip** button to open the flip axis menu.
3. Click the **Horizontal** button to reflect in a horizontal direction and the **Vertical** button to reflect in a vertical direction.

Copying an Object

You copy an object in the work area to reuse it.

To copy an object:

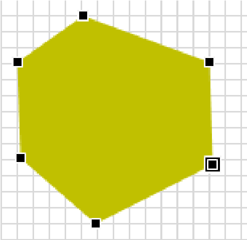
1. In Graphics Editor, in the work area, select the object you want to copy.
2. On the Drawing toolbar, click **Copy**.
3. On the Drawing toolbar, click **Paste**.
4. Move the copy to a new position.

Editing Objects

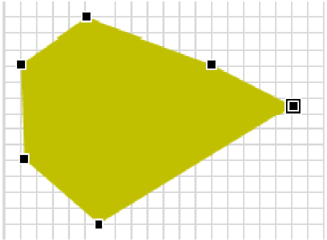
You can edit a point of an object to transform its appearance. You can edit the points of lines, polylines, curves, polygons, arcs, pies, or paths.

To edit a point of an object:

1. In Graphics Editor, in the work area, select the object whose point you want to edit.
2. Double-click the point you want to move.



3. Drag the selected point to a new position.



NOTE: When small objects are edited at an extreme zoom level, it can be difficult to select the points. This is due to unavoidable rounding errors, but the problem can be avoided if you draw and edit the object in a larger size and then down-scale it.
You can also use **Show Grid** (1 px) and **Snap to Grid** when you create and edit the object. Grid points are not numerically rounded.

You can edit object properties to change the object.

To edit object properties:

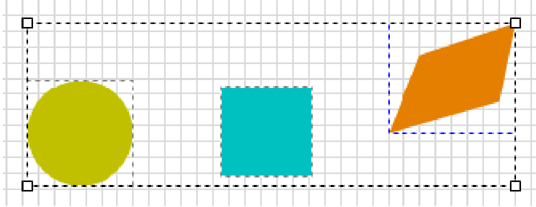
1. In Graphics Editor, in the work area, select the object whose properties you want to edit.
2. In the Properties pane, change, for example, the position properties of the object.
3. In the work area, click anywhere outside the object you have edited to clear the selection.
4. On the File menu, click **Save**.

You can edit a number of properties from the Properties pane.

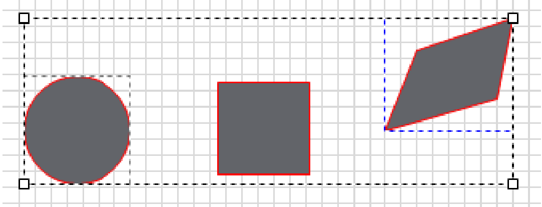
You can edit multiple objects to simultaneously modify, for example, their shape, size, and appearance.

To edit multiple objects:

1. In Graphics Editor, in the work area, select the objects you want to edit.



2. On the Options toolbar, click any of the Fill, Stroke (border) color, Style, or Width buttons and select appearance from the corresponding menu.

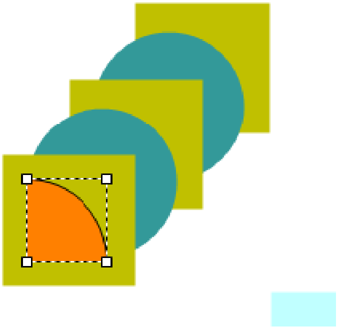


Deleting an Object

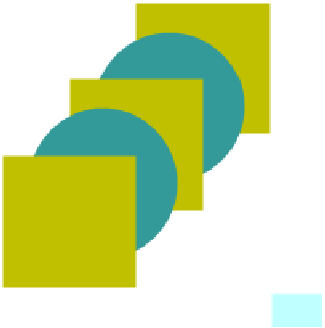
You delete objects in the work area that you do not want in your design.

To delete an object:

1. In Graphics Editor, in the work area, select the object you want to delete.



2. On the Drawing toolbar, click **Delete**.



The selected object is deleted, but you can undo the command by clicking **Undo** on the menu bar before you have edited any other objects.

Adding Custom Colors

You can apply color to most objects. Graphics Editor has a range of colors but you can also customize colors and save for future use. Objects with both stroke and fill can have different colors on stroke and fill.

You can define color hue from the Gradient palette.

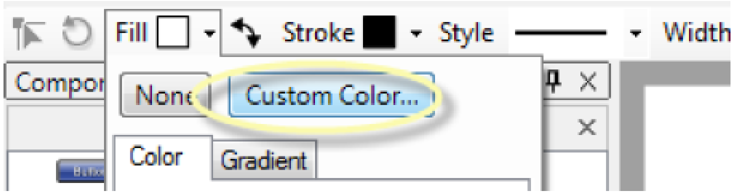
Object	Stroke Color	Stroke Style	Stroke Gradient	Fill Color	Fill Gradient
Line	Yes	Yes	Yes	-	-
Shape	Yes	Yes	Yes	Yes	Yes
Text	Yes	-	Yes	Yes[a]	-
Textbox	Yes	-	Yes	Yes[b]	Yes

a) Background behind text (default area)
b) Background behind text (textbox area)

You can add a custom color to the color palette to save it for quick access in the future.

To add a custom color:

1. In Graphics Editor, on the Options toolbar, click **Fill** (or **Stroke**) to open the color palette.
2. Select **Custom Color**.



3. In the Color dialog box, in the colored square, move the pointer to the color you want to add. If required, you can adjust the color by adjusting the numerical values for **Hue**, **Saturation**, **Red**, **Green**, **Blue**, and **Luminosity**.
4. Click **Add to Custom Colors**.

NOTE: **Fill** and **Stroke** use the same color palette and also the same custom colors.

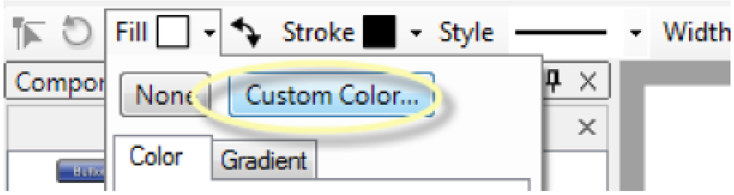
5. Click **OK**.

The color is displayed in one of the Custom Color boxes.

You can use custom colors when you want to use a specific color in your design.

To use a custom color:

1. In Graphics Editor, in the work area, select the object on which you want to use the custom color.
2. On the Options toolbar, click **Stroke** (or **Fill**) to open the color palette.
3. Select **Custom Color**.



4. Select the color in the Custom colors area.
5. Click **OK**.

Gradients Overview

In addition to the Color palette, Fill and Stroke have a Gradient palette. Gradients consist of smooth color transitions along a specified direction from one color to another. There are two types of gradients: linear and radial.

Gradients work with gradient stops where the two colors are specified and indicating where they start and stop.

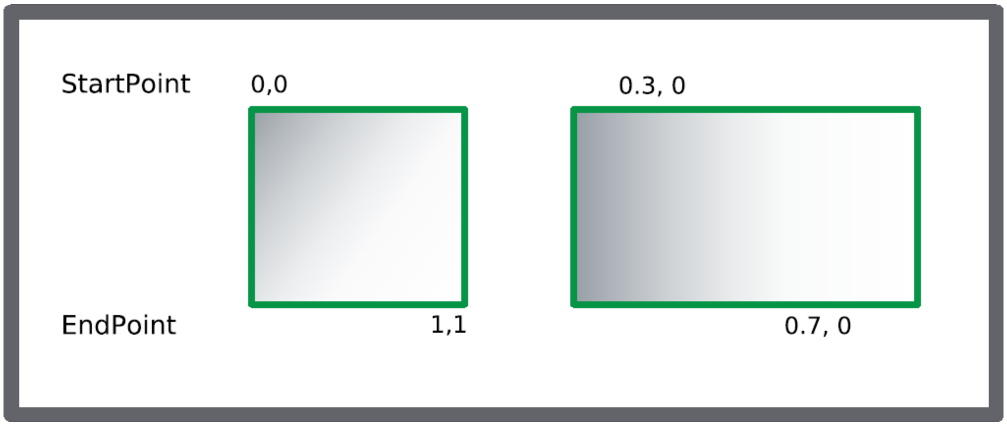
The Gradients and Gradient stops are properties, added to the object in the Objects pane. For more information, see the Figures section.

It is not difficult to change the gradient properties, but you have to have some understanding of the parameters involved. In most cases, it is sufficient to use the standard Gradient palette.

There are two types of gradients, linear and radial, each with somewhat different properties. These properties are displayed in objects under the corresponding shape object in the Objects tree. For more information, see the Figures section.

A linear gradient has a StartPoint and an EndPoint, local coordinates for the direction of the gradient.

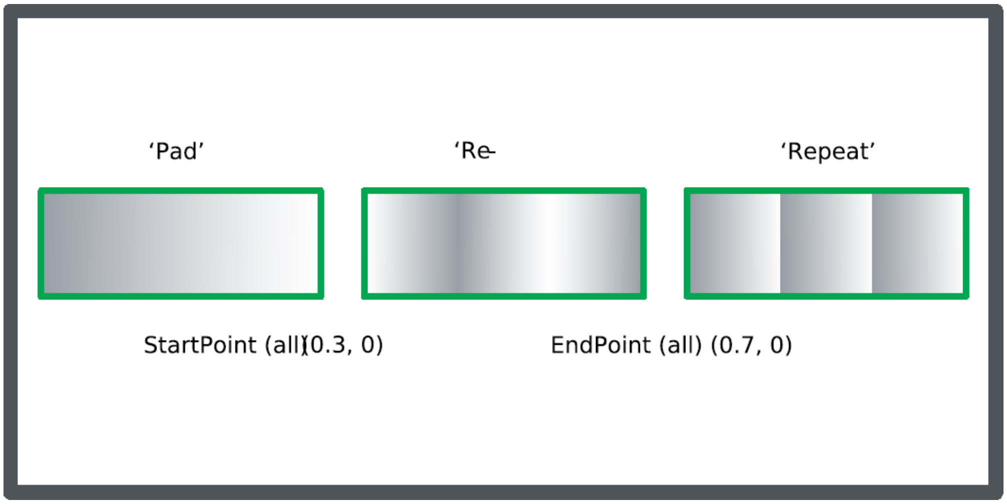
Example: A linear gradient, with a diagonal direction from 0, 0 to 1, 1; and a horizontal gradient from 0.3, 0 to 0.7, 0 (with SpreadMethod “Pad”):



A linear gradient also has a SpreadMethod, which tells how the areas outside the StartPoint and the EndPoint are to be treated. There are three methods:

- **Pad (default):** Extends the gradient end colors to the respective ends of the object.
- **Reflect:** Reflects the gradient like a mirror placed at the StartPoints and EndPoints.
- **Repeat:** Repeats the gradient pattern, as far as the “outside” areas stretches.

Example: The three different SpreadMethods used with the same StartPoints and EndPoints:

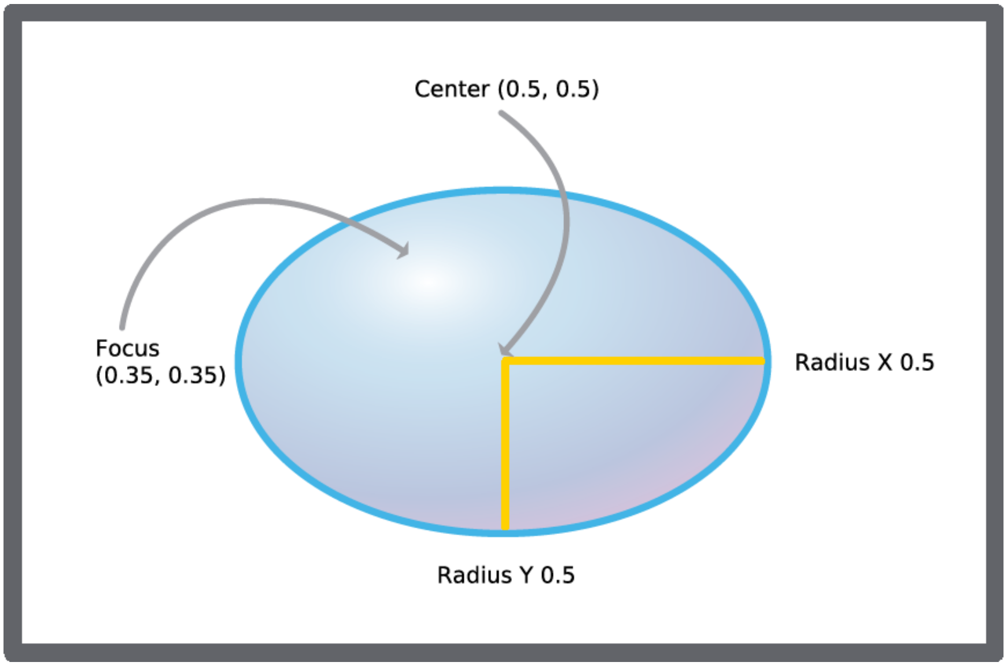


A radial gradient has the following properties:

- **Focus:** Point that defines where the radial gradient starts.
- **Center:** Center point for the circle (ellipse) that defines where the radial gradient ends.
- **RadiusX:** One of the two axes for the circle (ellipse) that define where the radial gradient ends.

- **RadiusY:** One of the two axes for the circle (ellipse) that define where the radial gradient ends.

Example: An ellipse, created with one of the standard circular gradient patterns from the Fill Gradient palette with an off-center focus, suggesting light from upper-left:

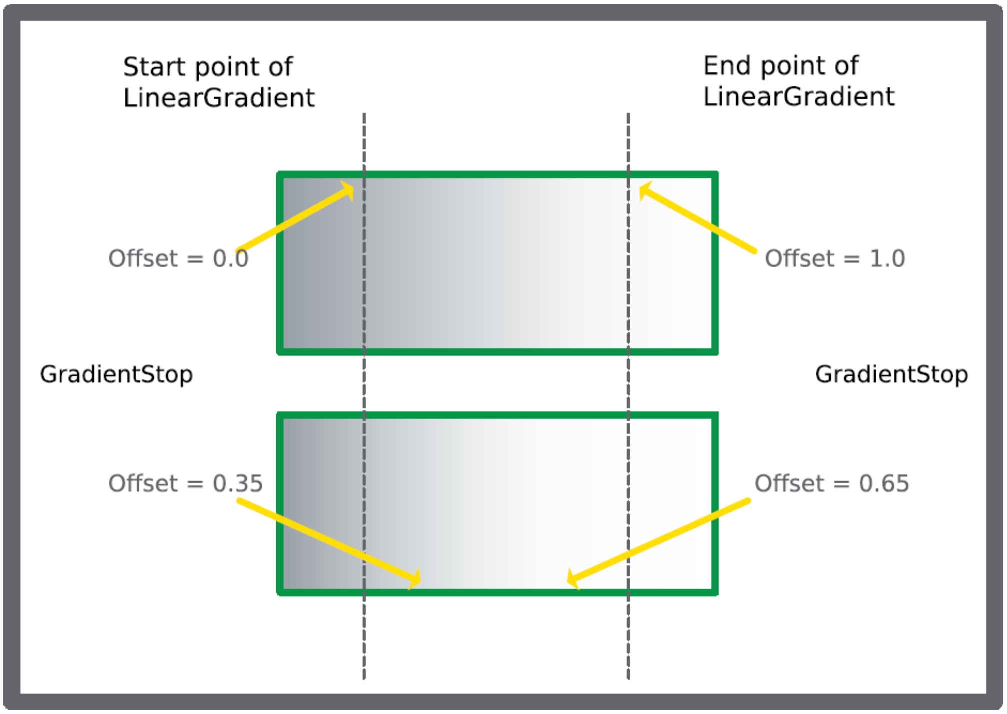


The SpreadMethod for a radial gradient is always “Pad”.

The colors at the StartPoint and EndPoint of a gradient (linear or radial) are specified in two GradientStop objects, belonging to the “parent” LinearGradient or RadialGradient object.

The GradientStop objects also have an Offset, which modifies where the gradient starts and stops. If the values are 0 and 1, the StartPoint and EndPoint are not modified.

Example: A rectangle with two different pairs of settings for the GradientStop, in the lower case modified by the GradientStop Offsets:

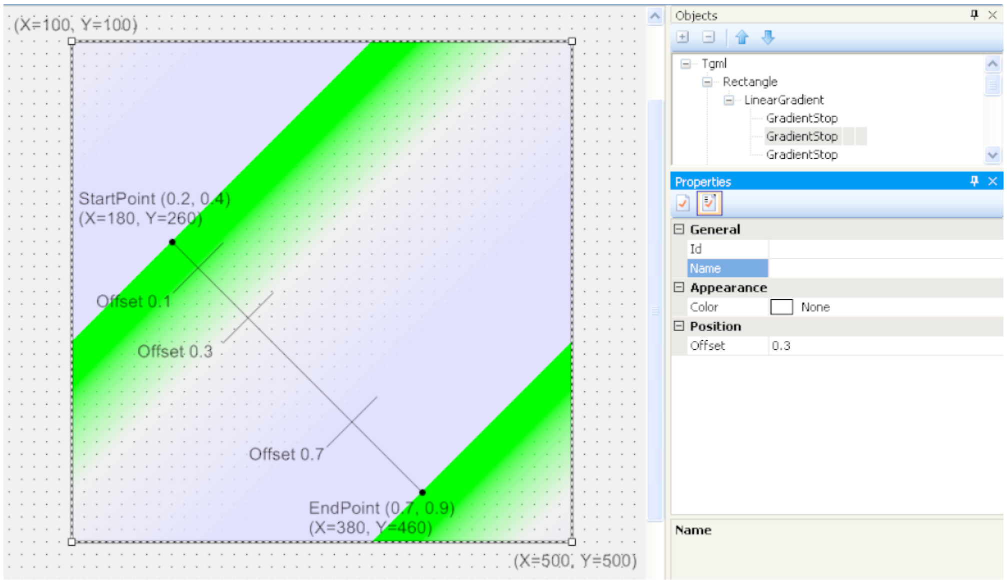


Gradients and gradient stops can be used in more complex ways.

For example, you can add more GradientStops to a Linear/RadialGradient to make the color change at each Offset distance. Below is an example with three GradientStops.

We use a square, 400x400, in which we want the gradients to run roughly in a diagonal direction, but described only within the inner part of the square. By setting the StartPoint to (0.2, 0.4) and the EndPoint to (0.7, 0.9), the gradients will run from the coordinates (180, 260) to (380, 460).

Example: A square with a limited linear gradient, three gradient stops, and two "SpreadMethod" areas outside the linear gradient definition:



General

Id	
Name	

Appearance

Color	□ None

Position

Offset	0.3

The three GradientStops have the following properties:

- Color 1 (green), Offset=0.1
- Color 2 (none), Offset=0.3

- Color 3 (blue), Offset=0.7

The area between Offset=0 and Offset=0.1 is padded with Color 1 (green).

The area between Offset=0.1 and Offset=0.3 gradually changes from Color 1 (green) to Color 2 (transparent).

The area between Offset=0.3 and Offset=0.7 gradually changes from Color 2 (transparent) to Color 3 (blue).

The area between Offset=0.7 and Offset=1 is padded with Color 3 (blue).

The two areas outside the LinearGradient definition use the same SpreadMethod, in this case Repeat, but Pad or Reflect could also have been used. With Repeat, the pattern is repeated according to the defined gradient. With Reflect, the pattern would have been mirrored, and with Pad the outermost colors (here 1 and 3) would have been extended.

NOTE: You have to add the GradientStops to the Objects pane in a strictly ascending Offset order (0 to 1), otherwise the result is undefined. When you add a GradientStop to the list, its Color and Offset are undefined, which means that the associated object are transparent until the parameters have been set.

For more information, see the following sections:

- Adding a Linear Gradient
- Adjusting a Linear Gradient
- Adding a Radial Gradient
- Adjusting a Radial Gradient

Adding a Linear Gradient

You add a linear gradient to give an object a smooth, varying hue from one point to another.

For more information, see section the Gradients Overview section.

To add a linear gradient:

1. In Graphics Editor, in the work area, select the object on which to apply the gradient.
2. On the Options toolbar, click **Fill** to open the Color/Gradient palette.
3. Click the **Gradient** tab.
4. Click the box with the gradient you want to use.

In the Objects pane, a LinearGradient element and two or more GradientStop elements are added to the object.

Adjusting a Linear Gradient

When you use the Fill – Gradient palette, you may need to adjust the color or some other aspect of the appearance.

Adjusting a linear gradient involves finding the specific object in the Objects pane and then changing the gradient parameters of the LinearGradient or the GradientStop elements.

You can adjust a linear gradient to customize it. For more information, see the Gradients Overview section.

To adjust a linear gradient:

1. In Graphics Editor, in the work area, select the object with the linear gradient you want to adjust.
2. In the Objects pane, locate the object.
3. Expand the object elements to display the LinearGradient element with its GradientStops.
4. Select the LinearGradient element.
5. To change the gradient target (the area or the stroke), in the Target area, in the Attribute box, switch between **Fill** and **Stroke**.
6. To change the gradient start and end points (also indicating the gradient direction), in the Position area, in the StartPoint or EndPoint box, type x and y-coordinates between (0,0), (0,1), (1,0) and (1,1). If the gradient is positioned well within the square outlined above, the areas “outside” the start and end points can be filled by one of three spread methods.
7. To change the gradient spread method, in the Behavior area, in the SpreadMethod box, switch between **Pad** (extending the end colors), **Reflect** (mirroring the gradient in the Start/End points), and **Repeat** (repeating the gradient as far as the “outside” area stretches).
8. To change any of the gradient colors (the “start” and “stop” colors), in the Appearance area, in the Color box, click a color in the Color palette, or type a hexadecimal color code.
9. To change the offset (where the “start” and “stop” colors will be positioned, relative to the LinearGradient Start/End points), in the Position area, in the Offset box, type a value between 0 and 1.

Adding a Radial Gradient

You add a radial gradient to give an object a smooth, varying hue from a center point to the periphery.

For more information, see the Gradients Overview section.

To add a radial gradient:

1. In Graphics Editor, in the work area, select the object to which you want to add the radial gradient.
2. On the Options toolbar, click **Fill** to open the Color/Gradient palette.
3. Click the **Gradient** tab.
4. Click the radial gradient you want to use.

In the Objects pane, a RadialGradient element with two or more GradientStops are added to the object.

Adjusting a Radial Gradient

When you use the Fill – Gradient palette, you may need to adjust the color or some other aspect of the appearance.

Adjusting a radial gradient involves first finding the specific object in the Objects pane and then changing the gradient parameters of the RadialGradient or the GradientStop elements.

You can adjust a radial gradient to suit current requirements. For more information, see the Gradients Overview section.

To adjust a radial gradient:

1. In Grahics Editor, in the work area, select the object with the radial gradient that has to be adjusted.
2. In the Objects pane, locate the object.
3. Expand the object elements to display the RadialGradient element with its GradientStops.
4. Select the **RadialGradient** element.
5. To change the gradient target (the area or the stroke), in the Target area, in the Attribute box, switch between **Fill** and **Stroke**.
6. To change the gradient focus (the beginning of the radial gradient), in the Position area, in the Focus box, type x and y-coordinates between (0,0) and (1,1). To change how far the gradient will reach (a kind of gradient end point periphery), a circle or ellipse is used.
7. In the Position area, click the periphery ellipse **Center**, and type x and y- coordinates between (0,0) and (1,1).
8. In the Position area, complete the circle/ellipse by selecting **RadiusX** or **RadiusY**, and type values between 0 and 1 to define the reach of the periphery.
9. To change any of the gradient colors (the “start” and “stop” colors), in the Appearance area, in the Color box, click a color in the Color palette, or type the a hexadecimal color code.
10. To change the offset (where the “start” and “stop” colors will be positioned, relative to the RadialGradient focal point), in the Position area, in the Offset box, type a value between 0 and 1.

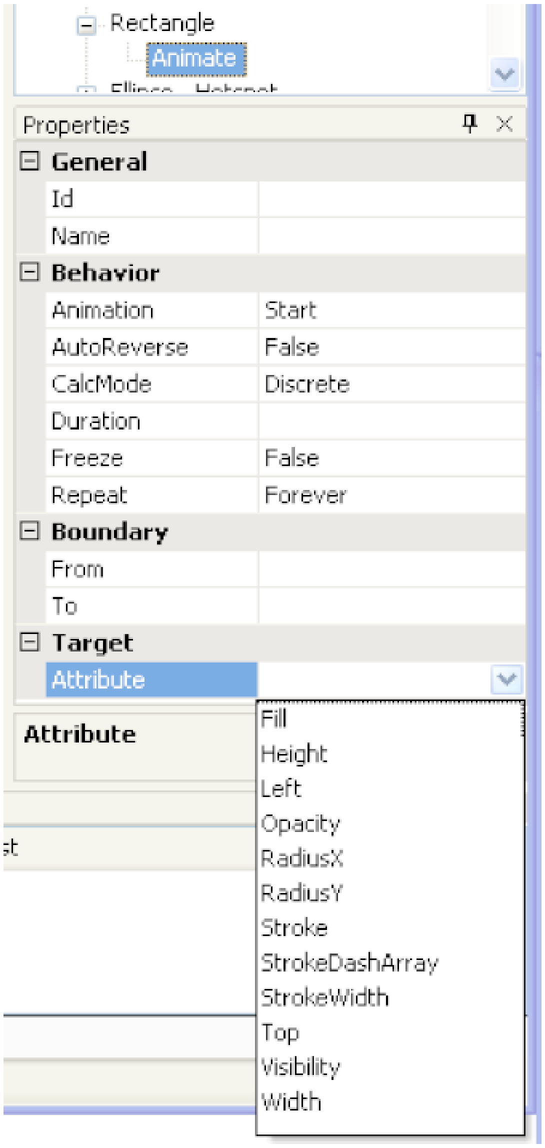
Adding Animations

You can add animations to a graphic object to create an illusion of movement. You can add animations to, for example, lines curves and rectangles. You can also add animations to transformations of objects, for example rotate, scale, translate and others. You can add the Animate property by adding it to the object you want to animate in the Objects pane.

The Animate element has a Target attribute, that is, the property in the parent object that you want the animation to change. For example, the Target attribute can be Visibility. The Boundary attribute of the Animate element decides how the Target attribute is affected. For example, a visible element can be hidden. The Behavior attribute of the Animate element decides how the parent element of the Animate element will go from visible to hidden. For example, Linear, in which case the parent element softly goes from visible to hidden. You select attribute from a drop-down list in the attribute box. Only relevant attributes are available in the list.

Two examples:

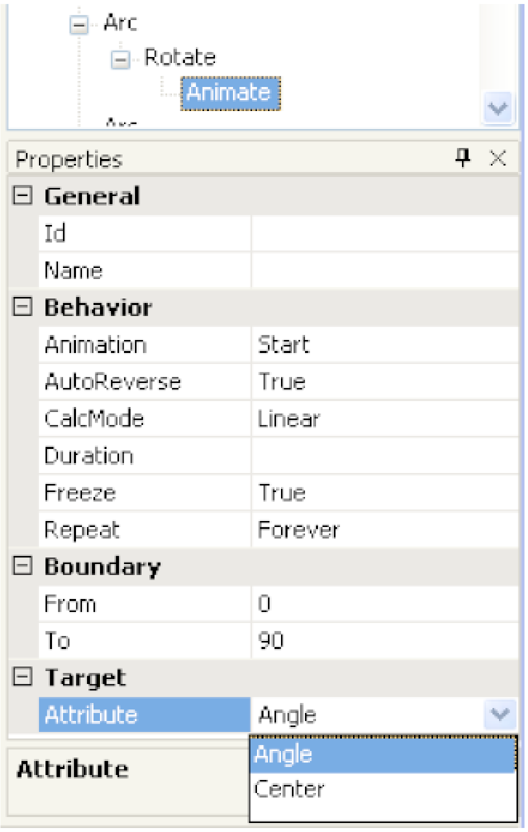
When you add an Animate element to a rectangle, you can animate a whole number of rectangle properties:



General	
Id	
Name	
Behavior	
Animation	Start
AutoReverse	False
CalcMode	Discrete
Duration	
Freeze	False
Repeat	Forever
Boundary	
From	
To	
Target	
Attribute	

Attribute

When you add an Animate element to a Rotate element, you can only animate the rotation angle and center properties:



General	
Id	
Name	
Behavior	
Animation	Start
AutoReverse	True
CalcMode	Linear
Duration	
Freeze	True
Repeat	Forever
Boundary	
From	0
To	90
Target	
Attribute	Angle

Attribute

The remaining Animate properties are used as follows:

- **Animation:** Start or Stop the dynamic behavior.
- **AutoReverse:** If True, the animation runs backwards at the end of the forward motion. If False, there is no backwards run at the end of the forward motion.
- **CalcMode:** If Discrete, the animation switches between the From and To Boundaries. If Linear, the animation progresses smoothly between the same Boundaries.

  Use discrete animations for:

  - Binary values (for example true/false, hidden/visible)
  - Enumerations (0, 1, 2)
  - Switching between two colors
  - Changing properties for a text (for example, Font, Size, Bold, or Italic)
  - Changing the size (height and width) of a shape
  - Options (for example, alignment and visibility)

  Use linear animations to:

  - Change properties that are numbers (double, analog values) to smoothly change the size
  - Position a shape
  - Set the angle of a rotated shape
  - Create a smooth blink, using the Visibility properties
  - Create smooth rotation animations

- **Duration:** The time in seconds for one forward animation.
- **Freeze:** If True, the current animation value is saved when the dynamic mode is exited. If False, the value is reset.
- **Repeat:** Either an entered number of runs or Forever, that is, endless repetition.
- **Boundary, From and To:** The end values for the animated property.

To add an animation:

1. In Graphics Editor, in the work area, select the object you want to animate.
2. In the Objects pane, right-click the object, click **New**, and then click **Animate**.
3. In the Properties pane, under Target, in the Attribute box, enter the property that you want the animation to affect.
4. In the Behavior field, in the Animation box, click **Start** to be able to specify what is to start the animation.
5. In the Properties pane, add other attributes to control how the attribute is to behave.
6. In the Objects pane, right-click the **Animate** element, point to **New**, and then click **Bind**.
7. In the Objects pane, right-click the **Bind** element, point to **New**, and then click a converter or another property that controls what the animation is to change.

You can now bind the animation to an actual signal in the Graphics Editor.

Adding Paths

When you create a path of one or several objects, you create a copy in the form of a path, where the original objects have been dissolved and replaced with corresponding lines and fills. For example, a rectangle dissolves into a path of four connected strokes.

By creating a path, the number of objects is reduced to one. That is, the resulting path.

The advantage of using a path is that it speeds up the drawing of the object, which can be important when using animations.

The original objects remain unchanged and can be deleted, if required.

For more information, see the following sections:

- Creating a Text Path
- Editing a Text Path

Creating a Text Path

You create a text path to protect the text so that it cannot be manipulated and to be able to use effects like gradients on the text. When you create a text path, the text in reality becomes a curve.

For more information, see the Adding Text and Textboxes section.

To create a text path:

1. In Graphics Editor, on the Options toolbar set the **Fill**, **Stroke**, **Style**, and **Width** properties for the text path you are going to create.
2. On the Drawing toolbar, click the **Text** T or **Textbox** T button.

3. In the work area, click the general area where you want to position the text path and write the text.
4. Right-click the text or textbox object and click **Create Path**.

The characters are transformed to curve paths, with the appearance you specified in the first step. You can edit the appearance.

NOTE: The original text remains untouched, but can be moved or deleted.

Editing a Text Path

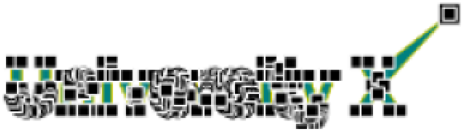
You edit a text path to change its appearance or other properties. The text path is in fact a curve and can be modified like any other curve, but you can no longer edit the text itself.

For more information, see the Adding Text and Textboxes section.

To edit a text path:

1. In Graphics Editor, in the work area, select the text path.

2. In the Properties pane, edit the properties (Fill, Opacity, Stroke, StrokeDashArray, StrokeWidth, or Visibility).
3. Double-click the path and drag the vertices to change the shape of the path.



4. Click outside the textpath to clear the selection.

Using the Grid

A grid is a Graphics Editor feature that can be of assistance when you draw and position graphic figures in a graphic. Graphic figures can be made to snap (attach) to the grid intersection points, which always align with the logical pixels.

There are two kinds of pixels:

- Logical pixels
- Screen pixels

The logical pixel is the unit of measurement in the graphic. Figure coordinates, stroke width, and so on, are all based on pixels. Although a line with Stroke width 1.0 px will look different (use few or many screen pixels) depending on the zoom level, the line will always have the width of 1.0 logical pixel.

The width of the grid lines can be set to: 1, 2, 5, 10, or 20 logical pixels.

The screen pixel is the smallest possible detail on a screen and its physical size depends on the screen resolution.

When you zoom out of a graphic, the grid pattern eventually becomes cluttered. In this case, the grid lines are removed so that the distance between two grid lines never gets shorter than five pixels.

You can customize the pixel space between the grid lines. This is useful when you draw small details.

The grid size is saved in the .tgml document.

When you zoom in on a graphic, additional, lighter grid lines are added to show the logical 1.0x1.0 pixel grid. This is the grid that is used when you nudge a figure.

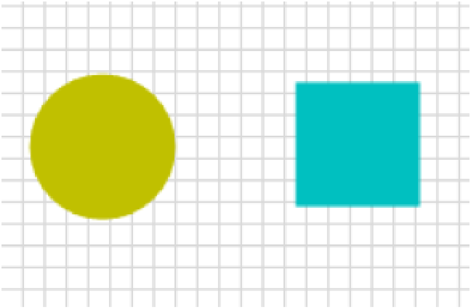
Showing/Hiding the Grid

You show the grid to get assistance when you position objects in the work area.

NOTE: By default, the grid is hidden. Even if the grid is hidden, **Snap to Grid** can be active.

To show/hide the grid:

1. In Graphics Editor, on the Drawing toolbar, click the **Show Grid/Hide Grid** button .



NOTE: **Show Grid** has a drop-down menu where you set the grid distance to 1, 2, 5, 10, or 20 pixels.

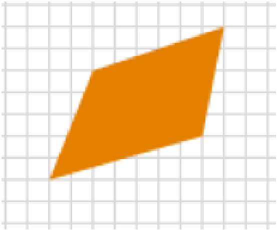
Show grid toggles between showing and hiding the grid.

Switching Snap to Grid On/Off

You use snap to grid to get a certain degree of alignment in a design.

To switch snap to grid on/off:

1. In Graphics Editor, on the Drawing toolbar, click the **Snap to Grid** button .



NOTE: You can snap to the grid even when the grid is hidden.

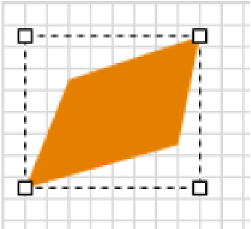
Snap to grid toggles between switching on and switching off the snap to grid.

Nudging an Object

You nudge an object to make a small adjustment to its position.

To nudge an object:

1. In Graphics Editor, in the work area, select the object you want to nudge.
2. If **Snap to Grid** is active, press ALT to temporarily disable the snap function during the nudging.



3. Use the arrow keys to move the object in the desired direction.

You move the object one pixel each time you press an arrow key.

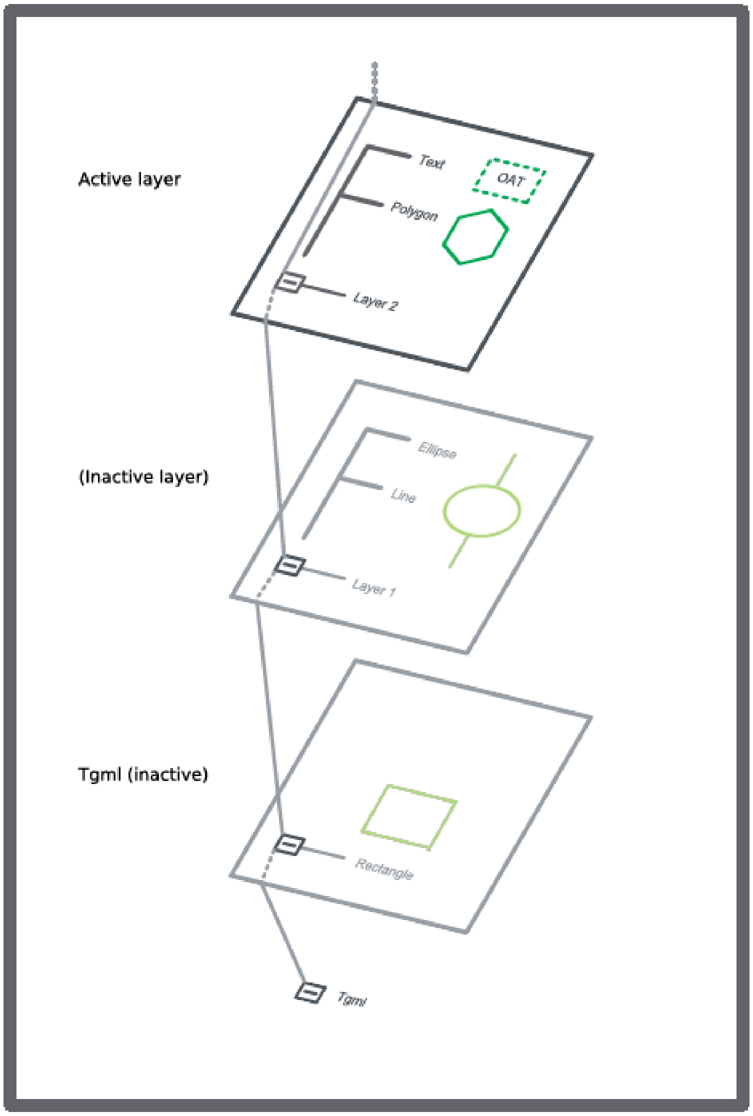
Layers Overview

When creating complex graphics, it can be a challenge to keep track of all the graphic figures in the work area. Graphic figures get hidden under one another and selecting becomes difficult. Layers provide a way to manage the graphic figures that make up your graphic. Layers can be regarded as folders that contain graphic figures.

The structure of layers in your document can be as simple or complex as you want it to be. By default, all graphic figures are organized in a single, root layer. This layer is named Tgml and you cannot rename it. The TGML layer is always visible.

Layers have the same properties as Group elements: Opacity and Visibility. You can control these layer properties from the Properties pane.

Example: TGML layer and two additional layers

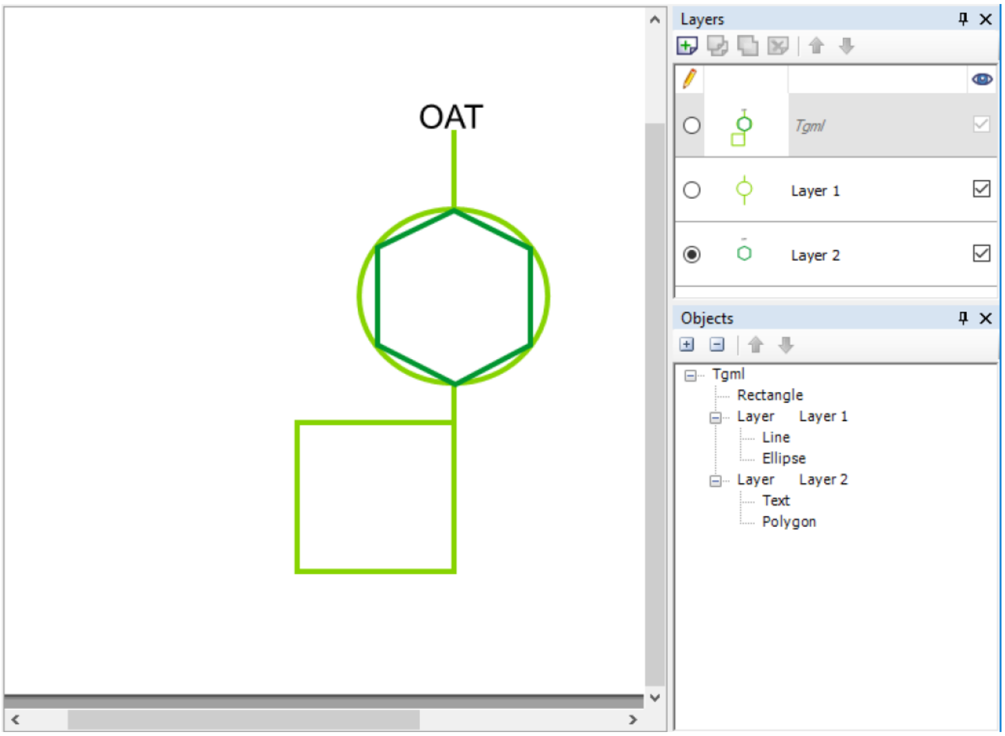


Layers are located on the TGML root level. New layers are added at the top of the work area (z-order, corresponding to the top of the tree), which means that the figures in the layer are displayed above (and possibly on top of) figures in previous layers. For more information, see the Figures section.

NOTE: To rearrange the order of the layers, you can select a layer in the Layers pane and click the Move up ⬆ and Move down ⬇ buttons to move the layer upward or downward in the tree.

You manage layers in the Layers pane. From the Layers pane you can create, select and merge layers. You can also show, hide, duplicate, move, rename and delete all layers except the Tgml layer. You can copy graphic figures from one layer to another. To edit a layer and access all its contents, you have to make sure that the layer is selected. When you select one layer, all other layers become inaccessible. If you move the layers up or down, you change the stacking order of the graphic figures in the work area.

Example: TGML and two additional layers, the last (Layer 2; topmost) active



The display area of the Layers pane has the following columns:

1. Buttons to select the active layer
2. Thumbnail, showing the contents of a layer. For TGML the merged contents are displayed.
3. The name of the layer, which can be edited (except for TGML)
4. Check box to show/hide a layer

NOTE: You can edit the name in the Name box in the Properties pane. You can also edit the name in the tree structure in the Objects pane. You can show/hide layers during the design process, but also in run time, for example, through a signal bound to the Visibility property.

You typically use layers to create a background image, which lies inert during the remaining design work, or to create layers with information that is to be displayed only under certain conditions.

Using Layers

You can add, copy, delete, rename, select, and hide layers. For more information, see the Layers Overview section.

Adding a Layer

You add layers to make it easier to select an individual element or groups of elements in the work area when you edit the graphic. Layers are especially useful when you use a background graphic in your work area.

To add a layer:

1. In Graphics Editor, on the View menu, click **Layers**.
2. In the Layers pane, on the Layers toolbar, click the **New Layer** button.

3. In the layer list, double click the layer name and type the name of the layer.
4. Press **Enter**.
5. On the File menu, click **Save**.

Copying a Layer

You copy a layer to reuse it.

To copy a layer

1. In Graphics Editor, on the View menu, click **Layers**.
2. In the Layers pane, right-click the layer you want to copy and then click **Duplicate**.

NOTE: The TGML layer cannot be copied.

In the Layers pane, the new layer is displayed at the bottom of the list with content identical to the content of the original layer. The new layer is automatically named **Copy of [name of the original layer]**. For example, **Copy of Text layer**.

Deleting a Layer

You delete a layer and all its contents if you no longer need it.

To delete a layer:

1. In Graphics Editor, on the View menu, click **Layers**.
2. In the Layers pane, right-click the layer you want to delete and then click **Delete**.

NOTE: The inherent TGML layer cannot be removed. Objects, components and groups in this layer are contents of the root level of the TGML object.

The selected layer is deleted from the layers list, and all its contents is removed from the work area.

Renaming a Layer

You can rename layers to make them easier to identify.

To rename a layer:

1. In Graphics Editor, on the View menu, click **Layers**.
2. In the Layers pane, click the layer name and then type a new name.

Selecting a Layer

You select a layer to perform an operation on it. You can only select and edit one layer at a time.

To select a layer:

1. In Graphics Editor, on the View menu, click **Layers**.
2. In the Layers pane, select the option button for the layer you want to edit.

NOTE: When you have selected a layer, all other layers will automatically be made inaccessible.

Hiding a Layer

You hide a layer to change the visibility property for the layer, and all its contents.

To hide a layer:

1. In Graphics Editor, on the View menu, click **Layers**.
2. In the Layers pane, clear the box in the eye icon column for the layer you want to hide.

NOTE: Selecting the box in the eye icon column displays the layer. It is not possible to hide the TGML layer.

Controlling Layer Visibility

You control the visibility of a layer in a graphic by binding a signal to the Visibility property, or by using a JavaScript.

For more information, see the Layers Overview section.

To control the visibility of a layer:

1. In Graphics Editor, in the Objects pane, select the layer that you want to control with a signal.
2. Right-click, point to **New**, and then click **Bind**.
3. In the Properties pane, under General, in the Name box, enter a layer name and add **".Value"**.
4. Right-click the **Bind** element and add two **ConvertValue** elements.
5. For the first ConvertValue element, in the Properties pane, under Behavior, in the AtributeValue box, type **Hidden.**
6. For the first ConvertValue element, in the Properties pane, under Behavior, in the SignalLessOrEqualTo box, enter **"0"**.
7. For the second ConvertValue element, in the Properties pane, under Behavior, in the AtributeValue box, type **Visible**.
8. For the second ConvertValue element, in the Properties pane, under Behavior, in the SignalMoreOrEqualTo box, enter **"1"**.

The visibility of the layer in the graphic can now be controlled by the signal values 0 and 1, bound to **[Layer name].Value**.

Groups Overview

A group is two or more objects or components that are combined as one entity. You can group graphical objects, such as figures, and non-graphical objects, such as metadata. Groups can have nested groups.

Grouping objects lets you perform an operation on all of them simultaneously. When you select a group, you can move, copy, and zoom in on the objects of the group using one command.

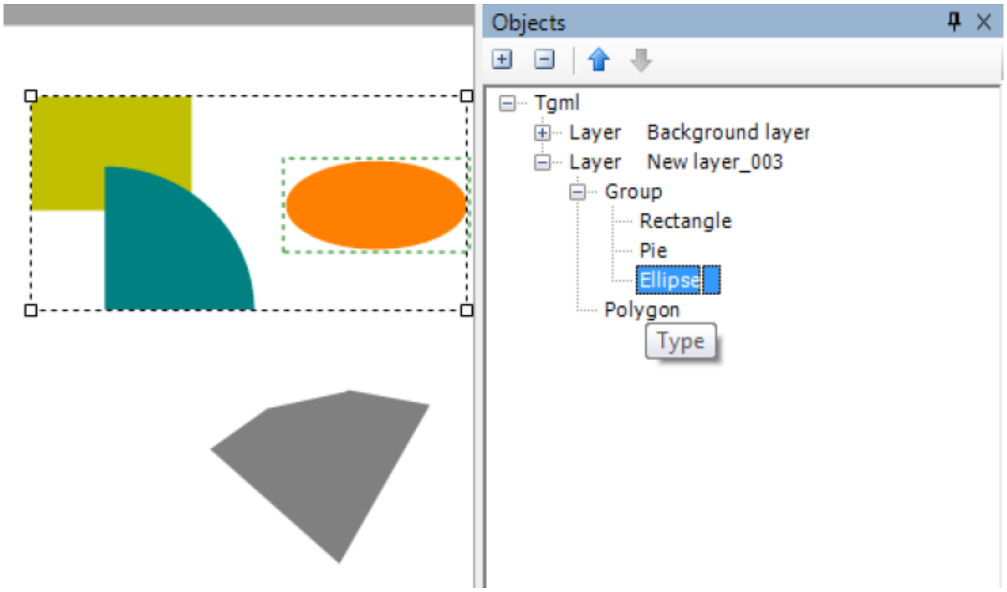
After you group objects, you cannot select individual objects within the group from the work area. You can still select one or several objects of the group from the Objects pane. The objects are then indicated as selected members of the group.

This can be useful, for example, if you want to see on which objects a specific element operates. The selected object is then indicated as a selected member of the group in the work area as well.

When you select one or more individual objects of a group, the selected object or objects are displayed with inverted text in the Objects pane, and are selected as Group member in the work area.

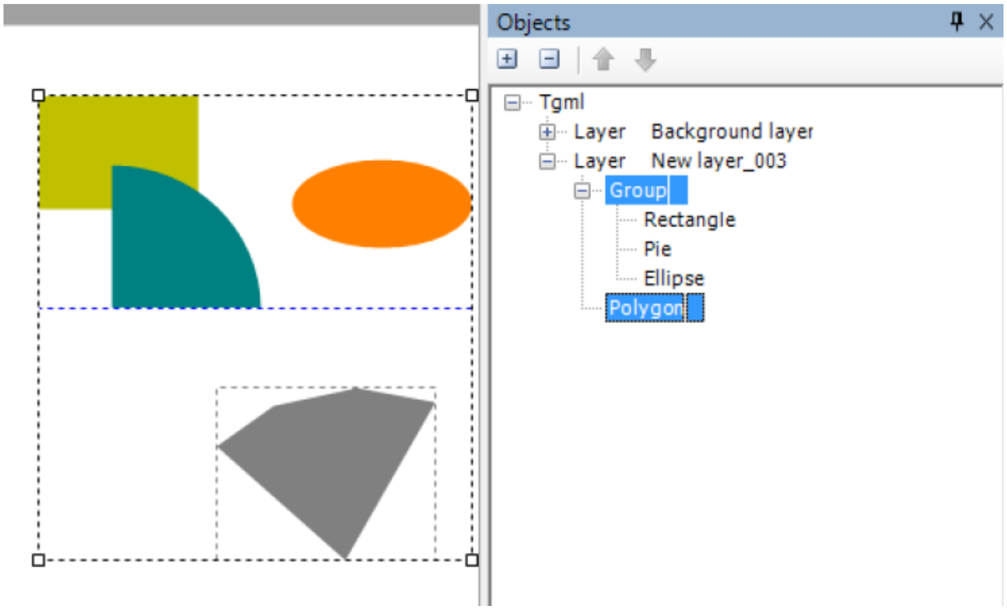
NOTE: A Group member selection is displayed in the work area as green dashed rectangle(s), enclosing the object(s).

Example: An individual member of a group has been selected, indicated with a green rectangle in the work area



You can add objects that do not belong to the group to a selection by pressing Ctrl and clicking in the Objects pane.

Example: A single curve is added to the selection and the ellipse is still surrounded by the green rectangle, but now the group, which is the primary selection, is surrounded by a blue rectangle



The usage of selection indicators in the work area, when you have selected one or several objects, can be summarized as follows.

- Primary selection is indicated with a dashed blue rectangle enclosing the object (which can also be a group)
- Group member selection is indicated with a dashed green rectangle enclosing the individual object

You have to select an object before editing it. There are several ways to select an object. You can select objects individually or simultaneously by clicking them or by drawing a marquee around them.

When you select two or more objects, one of them is regarded as the primary selection. The primary selection is displayed surrounded with a blue, dashed rectangle in the work area.

You can select two or more objects to align them. All the selected objects are aligned with the “primary selection” object. For more information, see Aligning Objects.

Selection Method	Primary Selection	Group Member Selection
Press Ctrl and click in the work area	Last selected object (group is also an object)	N/A
Selection box in the work area	Of the selected objects: the topmost (z-order) in the tree (group is also an object)	N/A
Press Ctrl and click non-grouped objects or groups in the Objects pane	First selected object (group is also an object)	N/A
Press Ctrl and click members of a group in the Objects pane	N/A	Each individually selected object in the group
Press Ctrl and click child elements of members of a group in the Objects pane	N/A	Parent object
Press Ctrl and click non-grouped objects and members of a group in the Objects pane (‘Mixed selection’)	First selected (group is also an object)	Each individually selected object in the group

Using Groups

You can group multiple objects, ungroup objects, select an object, select multiple objects, select all objects, clear the selection of multiple objects, select a group, and select an object within a group. For more information, see the Groups Overview section.

Grouping Multiple Objects

You group objects to be able to perform an operation on them all simultaneously.

To group multiple objects:

1. In Graphics Editor, in the work area, select the objects you want to include in the group.
2. On the Options toolbar, click **Group**.

The selected objects are now grouped and enclosed by the selection rectangle.

Ungrouping Objects

You ungroup objects to be able to perform operations on them individually. Nested groups are unfolded in the reverse order.

To ungroup objects:

1. In Graphics Editor, in the work area, select the group you want to ungroup.
2. On the Options toolbar, click **Ungroup**.

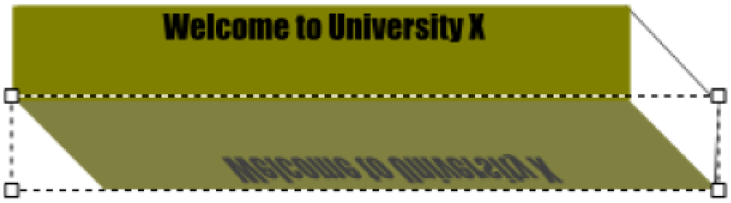
The selected objects are now ungrouped. The selection rectangle still encloses all the objects, but the objects also have separate selection indicators. All selections are cleared when you click outside the objects.

Selecting an Object

You can select a single object to perform an operation only on that object.

To select an object:

1. In Graphics Editor, in the work area, click the border or fill of the object you want to select.



NOTE: When an object is selected in the work area, it is highlighted in the Objects pane tree structure. The opposite also applies: when you click an object in the Objects pane, the corresponding object is selected in the work area.

Selecting Multiple Objects

You select multiple objects to perform an operation on them all simultaneously.

To select multiple objects:

1. In Graphics Editor, on the Drawing toolbar, click **Select**.
2. Press CTRL while clicking the border or fill of all the objects you want to include in the selection.

The last selected object is the primary selection.

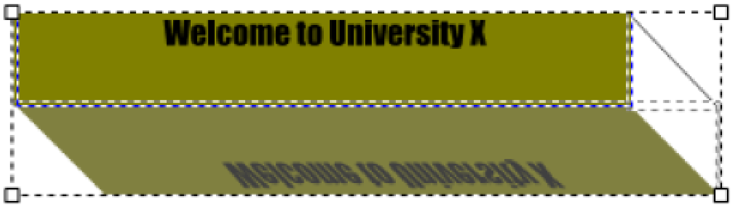
NOTE: You can add or remove objects by pressing CTRL while clicking the objects. Clicking the same object toggles between select and clear.

Selecting All Objects

You select all objects to perform an operation on them all simultaneously. This command selects all objects including invisible objects.

To select all objects:

1. In Graphics Editor, click anywhere in the work area and press CTRL+A.



The last selected object is the primary selection.

Clearing the Selection of Multiple Objects

You clear the selection of objects when you have completed an operation on them.

To clear the selection of multiple objects:

1. In Graphics Editor, in the work area, press CTRL+D.

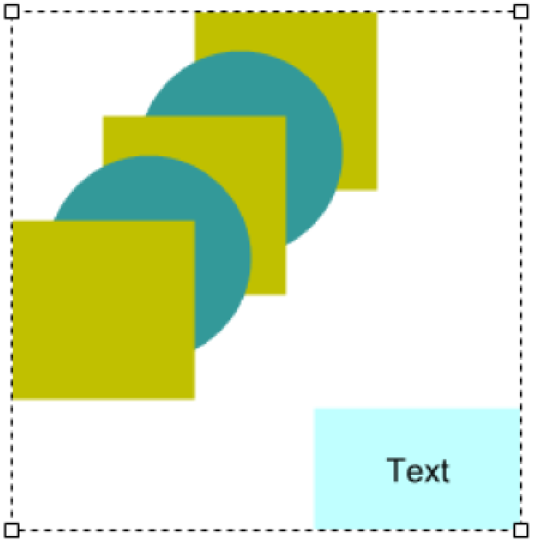
The selection of all objects in the drawing are cleared.

Selecting a Group

You select an entire group of objects to perform an operation on the group and all objects within the group simultaneously.

To select a group:

1. In Graphics Editor, in the work area, click the border, or any filled part, of any of the objects in the group.



Selecting an Object Within a Group

You select individual members within a group to perform an operation only on the selected objects.

To select an object within a group:

1. In Graphics Editor, in the Objects pane, click the object you want to select within the group.

The entire group as well as each individually selected object are selected in the work area. Any operation you perform only affects the individually selected objects, not the entire group.

Components and Snippets Overview

Components are standardized, predefined graphics for defined use. Snippets are standardized, predefined functions for defined use.

For more information, see the following topics:

- Components Overview
- Designing Components
- Categories Overview
- Snippets Overview
- Attributes Overview
- Binds and Links Overview
- Graphics Editing Tools Overview

Components Overview

Components are standardized, predefined graphics for defined use.

All drawn objects are either graphics, that is, free-form drawings, or components. A component contains one or several graphic figures. It can also have predefined functionality. Components typically represent a feature or a component in a live system. Components can be designed as symbols, which can be used as building blocks in graphics. Components are located in dedicated libraries and are displayed in the Components pane. An analog gauge is an example of a component.

For more information, see the following sections:

- Designing Components
- Documenting and Saving a Component
- Inhibiting Clipping
- Controlling the Appearance of the Component
- Grouping Drawing Objects as a Component
- Adding a Component
- Creating a New Component
- Editing a Component
- Saving as a Component

Designing Components

You create a component to make, for example, a symbol or a well-defined function available for future reuse.

The Design Process

You can create a component in two ways:

- Compose your graphics and group the elements as a Component. You then save the component in a category in the Components pane, or
- Open the component editor, draw your component and save it in a components category.

NOTE: When a component is edited in the components editor, the root element is called ComponentContent. The element has two metadata elements that contain the name of the component and the description. These elements are not present when the component is used in a graphic.

Component Documentation and Saving

Make components easier to understand and use by entering the following information in the Properties Dialog box before saving the component:

- A descriptive name
- A short, comprehensive description of the function
- Notes on the usage
- Exposed properties
- Links or Bindings, if any

You can edit the name and description of a component in the Component library by editing the properties of the component.

NOTE: To create a line break in the description, press **CTRL+ENTER**. Pressing **ENTER** is the same as clicking **OK**.

Documenting and Saving a Component

You enter a description to document a component.

NOTE: To make your components useful to others, you can document and save them in a standardized way.

For more information, see the Designing Components section.

To document and save a component:

1. In Graphics Editor, on the File menu, point to **Save As** and then click **Component**.
2. Select the category in which you want to save the component.
3. Click **OK**.
4. In the Name box, type the name of the component.
5. In the Description box, type a description using this structure:
   Short description of the function.
   Special notes, if any
   ==Bindings==, or ==Links==

Name of Binding/Link: Explanation of signal
==Exposed Properties==
Name of Property: Explanation of property

6. Click **OK**.

The component is now saved in the selected category and can be used in other graphics.

Inhibiting Clipping

You inhibit clipping to prevent borders of a component from partly disappearing when the component is used in a larger graphic.

For more information, see the Designing Components section.

To inhibit clipping:

1. In Graphics Editor, in the Objects pane, select **ComponentContent**.
2. In the Properties pane, right-click in any input field, and then click **Add**.
3. In the Add a Custom Attribute dialog box, in the Attribute box, type **'Clip'**. In the Value box, type **'False'**.
4. Click **OK**.

In the Properties pane, the Content attribute has been added and its Clip property is set to False.

Controlling the Appearance of the Component

You use the Bind object to control the appearance of a component.

NOTE: You only bind an exposed property. You can also use the Control element to control the appearance of a component.

For more information, see the Exposed Attributes section.

To control the appearance of a component:

1. In Graphics Editor, in the Objects pane, right-click the element you want the Bind element to control, point to **New**, and then click **Bind**.
2. In the Objects pane, right-click the bind element, point to **New**, and then click the required number of Convert elements.
3. In the Properties pane, set the behavior for the Convert element.
4. On the View menu, click **Test** to open the Test pane.
5. Click **Preview**.
6. In the Test pane, test the behavior by entering test values.

By using Inherit, Expose and Bind, you can design a standardized component whose appearance is easily controlled by an external signal.

Grouping Drawing Objects as a Component

You group multiple drawing objects within a graphic as a component for bind naming and graphics design efficiency. In order to save your work to a components library for reuse, the drawing elements must be grouped as a component.

To group drawing objects as a component:

1. In Graphics Editor, in the Layers pane, select the layer that contains the drawing objects you want to group.
2. On the work area, select all drawing objects that you want to group as a component.
3. Right-click the selected objects, point to **Group as**, and then click **Component**.
4. On the File menu, click **Save**.

Adding a Component

You add components to the graphic instead of drawing all the drawing objects yourself to simplify the graphics creation process.

For more information, see the Components Overview section.

To add a component:

1. In Graphics Editor, in the Layers pane, select the layer where you want to add the component.
2. In the Components pane, click the component category tab that contains the component you want to use.
3. Select the component you want to add.
4. Drag the component to the work area.
5. In the Properties pane, in the Name box, type the name of the component.

   NOTE: You only need to name the drawing object if you will be binding the object. Naming the object now will help you identify the object later.

6. On the Options toolbar or in the Properties pane, adjust the appearance of the component.
7. On the File menu, click **Save**.

Creating a new component

Create a new component when you want to create a design that represents a feature or a component in a live system and want this design to be available for reuse. For example, the component can be a button or a representation of a fan.

For more information, see the Components Overview section.

To create a new component:

1. In Graphics Editor, click **File > New > Component**.
2. In the work area, design the appearance of the component.
3. Click **File > Save As > Component**.

4. In the Components tree, select the category where you want to save the component.
5. Click **OK**.
6. In the Properties pane, in the Name box, enter the name of the component.
7. In the Description box, enter a description for the component.

   The description is displayed as a tooltip for the component in the Components pane.
8. In the Height and Width boxes, enter the size the component gets when used in a drawing.
9. Click **OK**.

NOTE: When you create a new component, a work area opens and a ComponentContent root element appears in the Objects pane.

The name you give the component is also the file name the component gets when it is saved, with the suffix `.tgmlcomponent`. The new component is now displayed in the selected category in the Components pane, and is ready to use in other graphics.

Editing a Component

You edit a component when you have an original component and want to reuse a number of its properties. You then save the edited version under a different name or in a different category.

For more information, see the Designing Components section.

To edit a component:

1. In Graphics Editor, in the Components pane, select the category containing the component you want to edit.
2. Right-click the component and click **Edit**.
3. In the Components work area, edit the component.
4. On the File menu, point to **Save As** and click **Component**.
5. Select the category where you want to save the edited component.
6. Click **OK**.
7. In the Properties dialog box, in the Name box, type the name of the component.
8. In the Description box, type a description to make the component easier to identify.
9. In the Height and Width boxes, enter the size of the component.
10. Click **OK**.

The edited component is saved in the selected category and under the specified name.

Saving as a Component

You save your components in the Components library so that they are available for future use.

For more information, see the Components Overview section.

To save as a component:

1. In Graphics Editor, on the work area, select the component you want to save. It is highlighted it in the Objects pane.
2. From the Objects pane, drag the component to the components category to which you want to add the component.

   NOTE: You have to drag the component from the Objects pane. You cannot drag the component from the work area.

3. In the Properties dialog box, in the Name box, type the name you want to display in the components category.
4. In the Description box, type the description you want to display as the tooltip in the components category.
5. In the Height box, type the height you want the component to have when it is added to the work area.
6. In the Width box, type the width you want the component to have when it is added to the work area.
7. Select **Use default scale (0x0)** to give the component the default size when it is added to the work area.
8. Click **OK**.

The component is now saved in the Components library for use in the current graphic and future graphics.

Snippets Overview

Snippets are standardized, predefined functions for defined use.

Snippets typically represent a feature in a live system. Snippets are located in dedicated libraries and are displayed in the Graphic Editor Snippets pane. Blink, which starts and stops a blink animation, is an example of a snippet.

By default, the Graphics Editor uses two dedicated folders, containing sub folders, for Components and Snippets. The folders are installed with the software. If your computer runs on Windows 7, Windows 8.1, or Windows 10, the components and snippets folders have the following paths:

C:\ProgramData\Schneider Electric\Power SCADA Operation\v2020 R2\Applications\Graphics\SnippetLibraries\Snippets\Basic

The sub folders are displayed as categories in the Components and Snippets panes.

Libraries can be located anywhere on the local disk.

You can show and hide categories from a selected library. You can add categories.

For more information, see the following topics:

- Adding a Snippet
- Saving as a Snippet

Adding a Snippet

You add snippets to a graphic to add pre-programmed behaviors, such as colors changing according to the state of a point or numerically displayed values.

To add a Snippet:

1. In Graphics Editor, in the Layers pane, select the layer where you want to add the snippet.
2. In the Snippets pane, select the snippet category tab that contains the snippet you want to use.
3. Select the snippet you want to add to the graphic.
4. Drag the snippet to the work area or to a drawing object.
5. On the Drawing toolbar, click **Select**.
6. On the Options toolbar or in the Properties pane, adjust the appearance of the snippet.
7. On the File menu, click **Save**.

For detailed information on configuring snippets, see "TGML snippet examples" on page 504.

Saving as a Snippet

Save your binded objects as snippets in the Snippets library for future reuse.

To save as a snippet:

1. In Graphics Editor, in the Objects Pane, select the Bind object that you want to save to the Snippets library.
2. Drag the Bind object to the snippet category to which you want to add the snippet.
3. In the Properties dialog box, in the Name box, type the name you want to display in the snippets category.
4. In the Description box, type the description that you want to display as the tooltip in the Snippets category.
5. Click **Ok**.

The snippet is now saved in the Snippets library for use in current and future graphics.

Categories Overview

Standard components and functions are categorized in logical groups. The following groups are delivered with Graphics Editor:

- **Basic Controls:** Control and sensor devices and buttons of different kinds
- **DIN Symbols (EN):** English standard ISO symbols
- **ISO Symbols:** Standard ISO symbols
- **My Components:** An empty folder where you can save components you want to make available for reuse
- **Basic Functions:** Functions of different kinds

- **My Snippets:** An empty folder where you can save functions you want to make available for reuse

For more information, see the Components Pane section.

NOTE: The categories listed above are read-only, and cannot be deleted.

The different categories are displayed as bars in the Components and Snippets panes. You can hide unused categories.

You can display hidden categories, or categories that reside in other libraries, in the Components pane.

You can store categories of components in other libraries. To make new libraries accessible from the Components pane, you have to add them to the list of available libraries.

When you import or export categories of components, or create new categories, the default library is used. You can set any components library folder as the default library.

For more information, see the following sections:

- Creating a Category
- Selecting a Category
- Renaming a Category
- Hiding a Category
- Displaying a Hidden Category
- Importing a Components Category
- Importing a Snippets Category
- Exporting a Category
- Deleting a Category

Creating a Category

You create a category when you want to organize your components in the Components pane or snippets in the Snippets pane.

For more information on categories, see Categories Overview.

To create a category:

1. In Graphics Editor, in the **Components** pane or in the **Snippets** pane, right- click and then click **New Category**.
2. In the New Component Category dialog box, type a name for the category.
3. Click **OK**.

The new category is displayed as a tab in the Components pane or Snippets pane.

Selecting a Category

You select a components or snippets category to display its content in the Graphics Editor, or before performing an operation on the category.

For more information on categories, see Categories Overview.

To select a category:

1. In Graphics Editor, in the **Components** pane or in the **Snippets** pane, click the tab of the category you want to select.

The content of the selected category is displayed in the pane.

Renaming a Category

You rename a components or snippets category to better reflect the contents.

For more information on categories, see Categories Overview.

To rename a category:

1. In the **Components** pane or **Snippets** pane, select the category you want to rename.
2. Right-click an empty space in the selected category and then click **Rename Category**.
3. In the Rename category dialog box, type a new name for the category.
4. Click **OK**.

The selected category is displayed with its new name in the Components pane or Snippets pane.

Hiding a Category

Hide a component or snippet category when you do not want to display it in the Components pane or Snippets pane.

For more information on categories, see Categories Overview.

To hide a category:

1. In the Components pane or Snippets pane, click the **X** to the right on the specific tab of the category you want to hide.

The selected category disappears from the Components pane or Snippets pane.

Displaying a Hidden Category

You display a component or snippet category that is hidden to make it available in the component or snippet library for use in Graphics Editor.

For more information on categories, see Categories Overview.

To display a hidden category:

1. In Graphics Editor, in the **Components** pane or **Snippets** pane, right-click and then click **Categories**. in the Components pane, right-click and then click **Categories**.
2. On the Components tab, select the category that you want to display.
3. Click **Close**.

Importing a Components Category

Import a components category into the components library to reuse the components between different projects.

For more information on categories, see Categories Overview.

To import a components category:

1. In Graphics Editor, in the **Components** pane, right-click and then click **Import**.
2. Select the .tgmlcomponentArchive file that contains the components you want to import.
3. Click **OK**.

Importing a Snippets Category

You import a snippets category into the snippets library to be able to reuse the snippets between different projects.

For more information on categories, see Categories Overview.

To import a snippets category:

1. In Graphics Editor, in the **Snippets** pane, right-click and then click **Import**.
2. Select the .tgmlsnippetArchive file that contains the snippets you want to import.
3. Click **OK**.

The imported snippets category is displayed as a tab in the Snippets pane.

Exporting a Category

You export component or snippets categories to create a component archive file or a snippet archive file, which in turn can be imported into the component or snippet library of Graphics Editor on other computers.

For more information creating components, see the Designing Components.

To export a category:

1. In Graphics Editor, right-click anywhere in the **Components** pane or **Snippets** pane and then click **Export**.
2. Enter the location where you want to save the export file.
3. In the File name box, verify or type a new name for the export file.
4. Click **Save**.

Deleting a Category

You delete a component or snippet category when you no longer need it.

For more information on categories, see Categories Overview.

To delete a category:

1. In the Components pane or Snippets pane, select the category you want to delete.
2. Right-click an empty space in the category and then click **Delete Category**.
3. Click **OK**.

The selected category is deleted from the database.

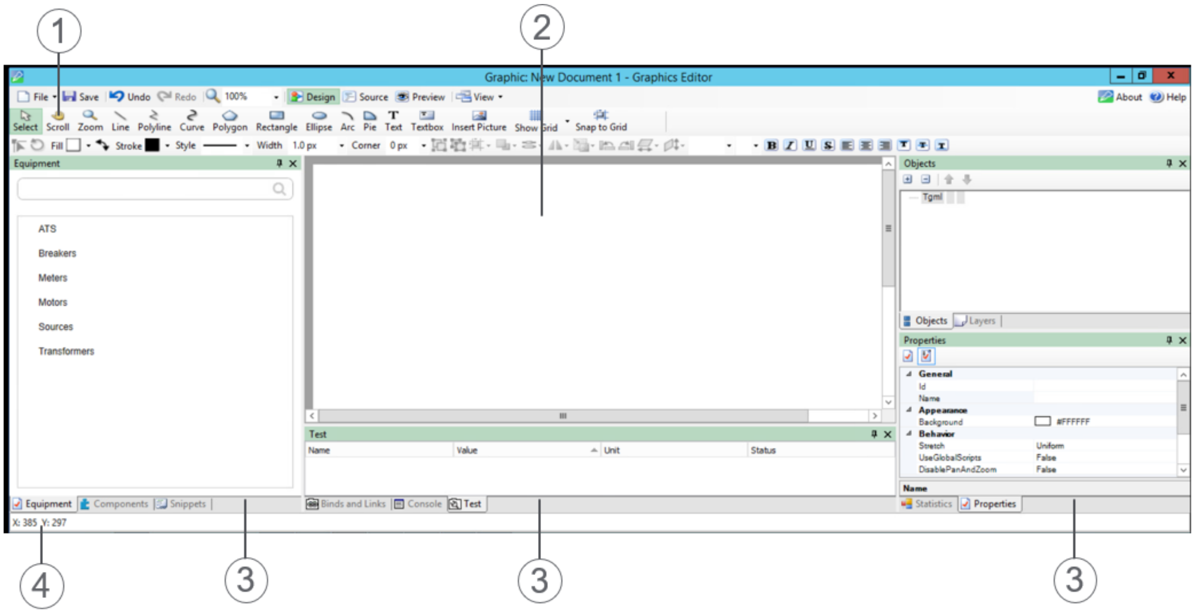
Graphics User Interface Overview

Graphics user interface overview information:

- [Graphics Editor]
- Graphics Editor Menu Bar
- Categories Context Menu
- Graphics Editor Drawing Toolbar
- Graphics Editor Options Toolbar
- Graphics Editor Panes
- Document Properties Dialog Box
- Unsupported Characters

Graphics Editor

Use the Graphics Editor main window to create, test, and save application graphics.



Number	Description
1.	**Toolbars** Contain tools used to create and edit TGML files and other objects.
2.	**Work area** You can drag elements from the panes to the work area. You can also draw free form objects by using the drawing tools.

Number	Description
3.	**Panes** Includes the following: • **Objects** pane with a tree structure of what is included in the graphic • **Properties** pane where you edit the properties of a selected object • **Layers** pane where you manage layers in a graphic • **Statistics** pane where you check the efficiency of a graphic • **Binds and Links** pane with information on the bind objects • **Test** pane where you test the behavior of a graphic • **Equipment** pane where you can access standard equipment libraries as well as equipment categories that you have imported or created on your own • **Components** pane where you can access standard components libraries as well as components categories that you have imported or created on your own • **Snippets** pane where you can access standard snippets libraries as well as snippets categories that you have imported or created on your own
4.	**Graphic Object Position Bar** Displays the position of the pointer and the position of a selected object. For more information, see the Figures Overview section.

Graphics Editor Menu Bar

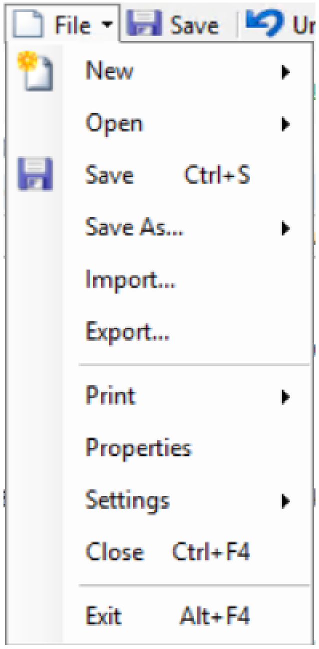
Use the menu bar to manage graphic files, and to toggle between different views of the current graphic.

Button	Description
🗋	**File** Click to open the File menu. For more information, see the Graphics Editor File Menu section.
💾	**Save** Click to open the standard Save As window.
↶	**Undo** Click to revert the graphic to the state it was in before the latest performed command was executed. Repeated use of Undo takes you back in the changes history, all the way to when the graphic file was opened.
↷	**Redo** Click to revert the graphic to the state it was in before the latest Undo command. Repeated use of Redo takes you forward in the changes history, all the way to the most recent change.

Button	Description
🔍	**Magnification** Click to enter the percentage of magnification of the work area.
	Design Click to open the current graphic's work area for drawing and editing. It also displays the tools on the drawing toolbar. Graphics Editor opens in Design mode.
	Source Click to open the current graphic for TGML text editing, by putting the cursor in the text. Common text editing tools are made available in the Options bar.
👁	**Preview** Click to open the current graphic to test animations, bindings, and links.
	View Click to open the View menu from which you can select the panes you want to use in Graphics Editor: Components, Snippets, Statistics, Layers, Objects, Properties, Binds and Links, and Test. You can also select Full Screen mode. For more information, see the Graphics Editor View Menu section.

Graphics Editor File Menu

Use the File menu to manage graphic files:

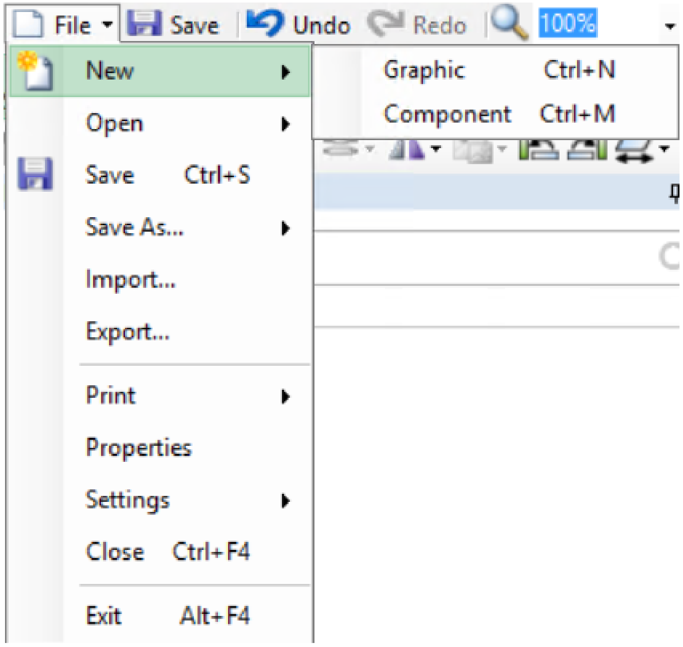


Command	Description
New	Opens a submenu where you can select if you want to create a graphic or a component.

Command	Description
Open	Opens a submenu where you select a.tgml graphic file or an .ogc file.
Save	Saves the graphic as a Power SCADA object in the database.
Save As	Opens a submenu where you select how the current graphic file is to be saved. When you use the Save As command to save a graphic as a file, the Power SCADA link is broken.
Import	Opens a dialog box where you can locate graphics files that you want to import into Graphics Editor.
Export	Exports a graphics file to the selected location, under the name you enter, and in the file format you specify.
Print	Opens a submenu where you can make a number of print preferences for printing the active graphic or component.
Properties	Opens the document properties pane. For more information, see the Graphics Editor Properties Pane section.
Settings	Opens a submenu from which you access user interface settings for Graphics Editor.
Close	Closes the current design and the editor in which it is open. If you have several instances of the editor open simultaneously, only the current editor is closed.
Exit	Exits all instances of the program.

Graphics Editor File Menu — New Submenu

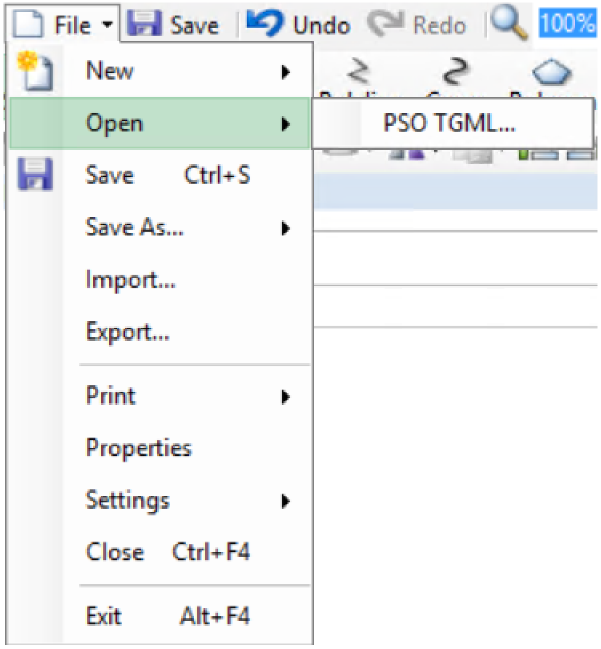
Use the New submenu to create a graphic or a component:



Command	Description
Graphic	Click to open a graphic workspace where you create a new graphic. For more information, see the Graphics Editor Overview section.
Component	Click to open a graphic workspace where you create a new graphic. For more information, see the Graphics Editor Overview section.

Graphics Editor File Menu — Open Submenu

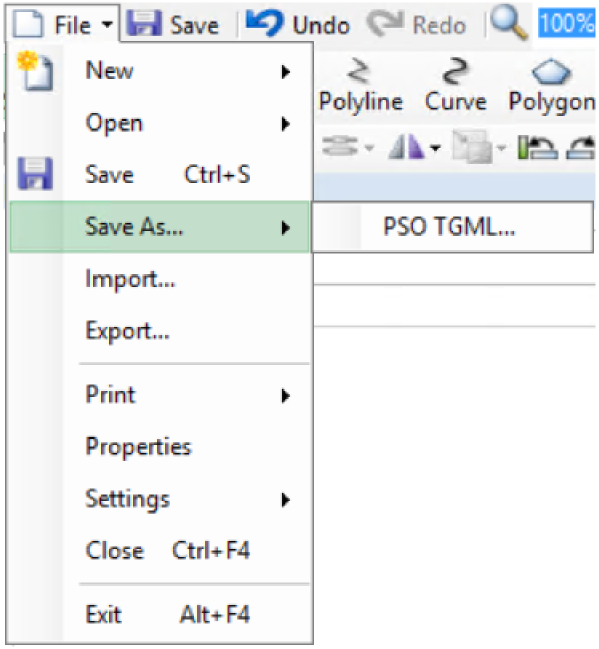
Use the Open submenu to open .tgml or .ogc files in Graphics Editor:



Command	Description
File	Click to open a graphics or components file in .tgml or .ogc format.

Graphics Editor File Menu — Save As Submenu

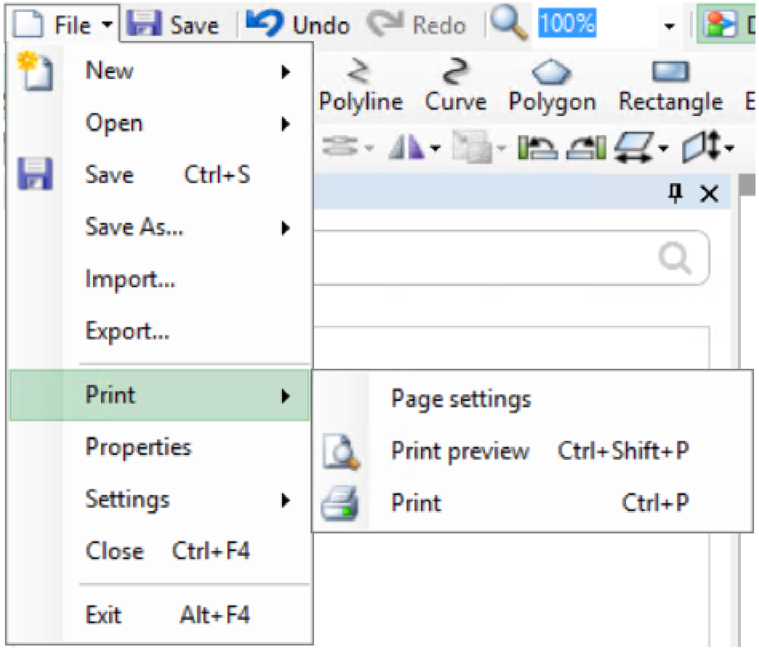
Use the Save As submenu to save a graphics or components file:



Command	Description
File	Click to save the graphic as a .tgml file.
Component	Click to save a component in a component category.

Graphics Editor File Menu — Print Submenu

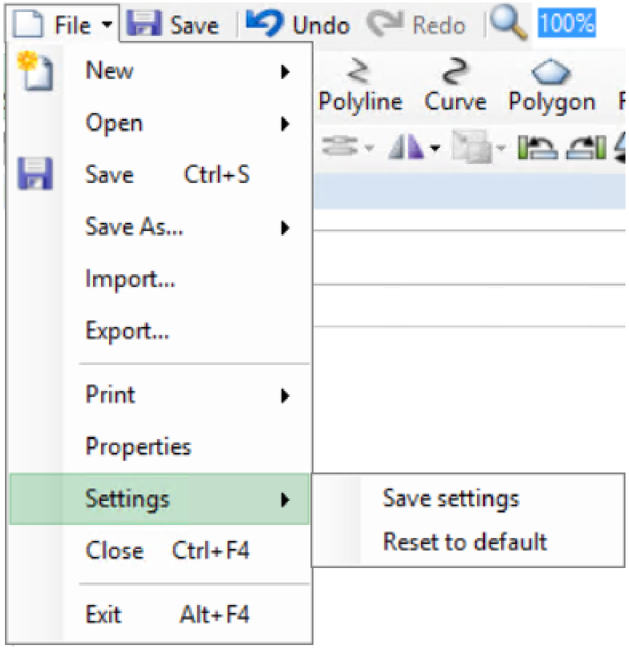
Use the Print submenu to make settings that affect how a graphic or component is printed:



Command	Description
Page Settings	Click to open a dialog box where you can set paper size, source, orientation and margins.
Print Preview	Click to display a print preview of the page, that is, a view of how the page appears when printed.
Print	Click to open the default Print dialog box where you can select a printer, set the page range, and set the number of copies you want to print.

Graphics Editor File Menu — Settings Submenu

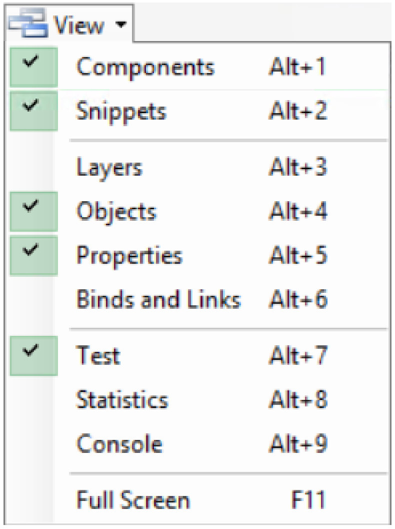
Use the Settings submenu to save settings you have made to the user interface or to reset the user interface to the default settings:



Command	Description
Save Settings	Click to save customized settings that define the size, position and visibility of the grid, panes, magnification, tools options, and columns. Snap to the grid is also saved.
Reset to default	Click to reset the user interface to the default settings.

Graphics Editor View Menu

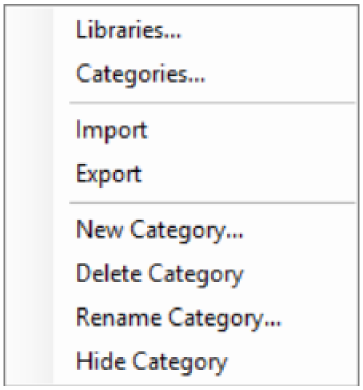
Use the View menu to show or hide the Graphics Editor tabs and panes. From the View menu, you can also toggle a graphic between full screen mode and displaying it in the Graphics Editor window:



Button	Description
Components	Click to show or hide the Components tab where you can access and manage components.
Snippets	Click to show or hide the Snippets tab where you can access and manage snippets.
Layers	Click to show or hide the Layers pane where you can manage the layers of your graphic.
Objects	Click to show or hide the Objects pane where you can view the object tree with all the elements included in the graphic.
Properties	Click to show or hide the Properties pane where you can view and set the properties of a graphic.
Binds and Links	Click to show or hide the Binds and Links pane where you manage the binds and links of the graphic.
Test	Click to show or hide the Test pane where you can test that your graphics functions behave as expected.
Statistics	Click to show or hide the Statistics pane where you can view the efficiency of the graphic.
Console	Click to show or hide the Console where you can troubleshoot and test scripts.
Full Screen	Click to view the graphic in full screen mode or to revert to the Graphics Editor window.

Categories Context Menu

Use the categories context menu to manage components and snippets categories:



Button	Description
Libraries	Opens the Libraries dialog box where you manage components and snippets libraries. For more information, see the Snippets Overview section.
Categories	Opens the Categories dialog box where you select or clear the components and snippets categories that you want to be displayed in the Components pane or Snippets pane.
Import	Opens an Explorer window where you can import a components or snippets archive file.
Export	Opens an Explorer window where you can save a components or snippets archive file.
New Category	Opens the New Component Category or the New Snippet Category dialog box where you type a name for the new category you want to create.
Delete Category	Opens the Delete Category dialog box where you confirm that you want to delete the selected components or snippets category.
Rename Category	Opens the Rename Category dialog box where you type a new name for the selected components or snippets category.
Hide Category	Hides the selected components or snippets category in Graphics Editor.

Graphics Editor Drawing Toolbar

Use the Drawing toolbar to access the tools you need to create and edit .tgml files and other objects.

Button	Description
↖	**Select** Click the border, or anywhere within a filled object, to select the object.
✋	**Scroll** Click and drag to adjust the work area in the pane.

Button	Description
🔍	**Zoom** Click to display three zoom tools in the Options bar: Restore original, Zoom In, and Zoom Out.
╲	**Line** Click-drag-release in the work area to draw a line between the two end points.
	Polyline Click a number of times in the work area to draw a polyline between the click points. Double-click to finish the polyline.
	Curve Click a number of times in the work area to draw a curve between the click points. Double-click to finish the curve.
⬠	**Polygon** Click a number of times in the work area to draw a polygon between the click points. Double-click to finish the polygon.
▭	**Rectangle** Click and drag in the work area to open up a rectangle between the two corner points. Simultaneously, press **Shift** to open up a square.
⬭	**Ellipse** Click and drag in the work area to open up an ellipse between the two size-determining points. Simultaneously, press **Shift** to open up a circle.
◝	**Arc** Click and drag in the work area to open up an arc between the two size-determining points. Simultaneously, press **Shift** to open up a quarter of a circle. For more information, see the Drawing an Arc or Pie section.
◣	**Pie** Click and drag in the work area to open up a pie (filled arc) between the two size- determining points. Simultaneously, press **Shift** to open up a quarter of a pie. For more information, see the Drawing an Arc or Pie section.
T	**Text** Click in the work area to position the start point of a text string. Type the text and press **ENTER.**

Button	Description
	Textbox Click and drag in the work area to open up a rectangular text box between the two corner points. Simultaneously, press **Shift** to open up a square. Type the text and press **ENTER**.
	Insert Picture Click in the work area to position the upper left corner of a picture insert. An Open window lets you browse to the desired picture file.
✂	**Cut** Click to remove the selected object from the graphic. The object is temporarily stored on the clipboard.
	Copy Click to save a copy of the selected object on the clipboard.
	Paste Click to create a copy of the object residing on the clipboard. The copy is displayed on the graphic slightly displaced from the original, or from any previous copy.
✕	**Delete** Click to delete the selected object from the graphic. The clipboard is not affected.
#	**Show Grid** Click to toggle between show and hide grid. Select a grid size value from the adjacent combo box.
	Snap to Grid Click to toggle between enabling and disabling the snap objects to the grid function.
🔗	**Auto Connect** Click to toggle between connecting the components automatically or manually.

Graphics Editor Options Toolbar

Use the Drawing toolbar to access the tools you need to create and edit .tgml files and other objects.

Button	Description
	Point Selection Click the border, or anywhere within a filled object, to select the object and display the curve points.

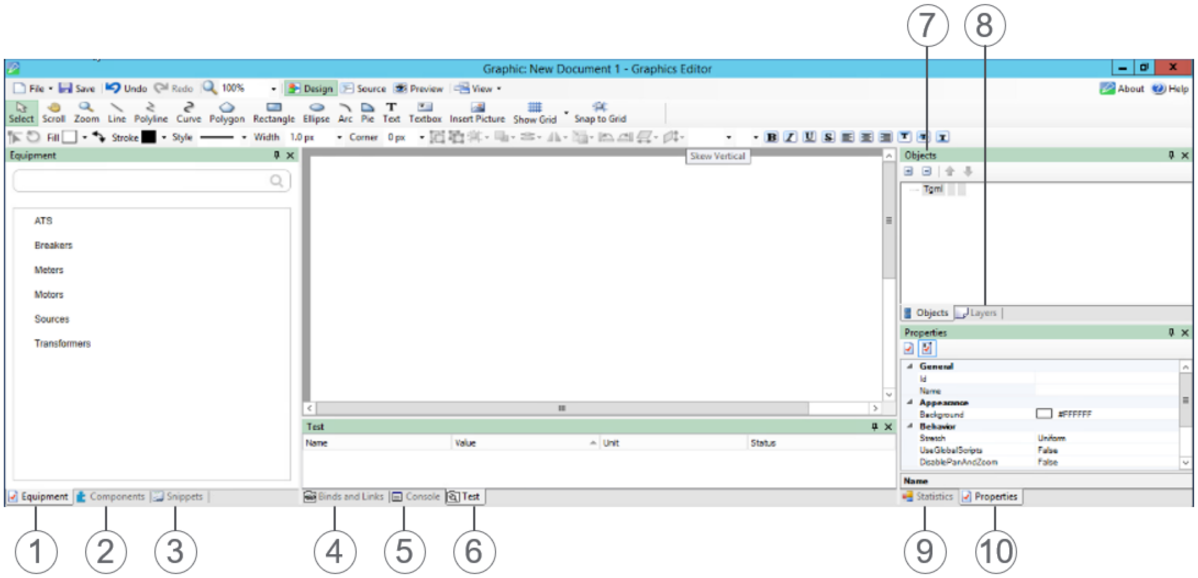
Button	Description
⟲	**Rotate Selection** Click to rotate the selected object by dragging the handle that is displayed at the top of the object.
Fill	Select the required fill color or gradient from the palette. For more information, see the Gradients Overview section.
↷	**Swap Colors** Click to swap the current colors of the stroke and the fill.
Stroke	Select the required stroke color or gradient from the palette. For more information, see the Gradients Overview section.
Style	Select the required stroke style from the list.
Width	Select the required stroke width, 0–30 pixels, from the list.
Corner	Select the required corner radius, 0–50 pixels, from the list.
	Group Click to group the selected objects in the work area.
	Ungroup Click to ungroup objects that were previously grouped.
	Arrange Select the required arrangement of an object relative to other objects, from the list.
	Distribute Select the required distribution of three or more objects, in the horizontal or vertical direction, from the list.
	Flip Select the required flip operation of the selected object, in the horizontal or vertical direction, from the list.
	Resize Select the required resize operation of two or more objects from the list: same width, height, or width and height.
	Rotate Left Click to rotate objects (single or group) 90° left.
	Rotate Right Click to rotate objects (single or group) 90° right.

Button	Description
	Skew Horizontal Click to skew the objects (single or group) horizontally to ±60°, ±45°, ±30°, or ±15°.
	Skew Vertical Click to skew the objects (single or group) vertically to ±60°, ±45°, ±30°, or ±15°.
Arial ▾	**Font** Select the required font family from the list.
1 ▾	**Size** Select the required font size, 8–100 pixels, from the list.
B	**Bold** Click to make the characters of the selected text string or text box bold.
I	**Italic** Click to make the characters of the selected text string or text box italic.
U	**Underline** Click to make the characters of the selected text string or text box underlined.
S	**Strikethrough** Click to display the characters of the selected text string or text box with strikethrough.
≡	**Text Align Left** Click to align the text to the left based on the insertion point of the text row. **Textbox Align Left** Click to align the text to the left in the text box.
≡	**Text Align Center** Click to center the text based on the insertion point of the text row. **Textbox Align Center** Click to center each text row in the textbox.
≡	**Text Align Right** Click to align the text to the left based on the insertion point of the text row. **Textbox Align Right** Click to align the text to the right in the text box.

Button	Description
T	**Text Align Top** Click to align the top of the text to the insertion point. **Textbox Align Top** Click to align the text to the top of the textbox
T	**Text Align Middle** Click to align the middle of the text to the insertion point. **Textbox Align Middle** Click to align the text in the middle of the texbox.
T	**Text Align Bottom** Click to align the bottom of the text to the insertion point. **Textbox Align Bottom** Click to align the text to the bottom of the textbox.

Graphics Editor Panes

Use the Graphics Editor panes to manage graphics and components:


7
8
Graphic: New Document 1 - Graphics Editor
File Save Undo Redo 100% Design Source Preview View
About Help
Select Scroll Zoom Line Polyline Curve Polygon Rectangle Ellipse Arc Pie Text Textbox Insert Picture Show Grid Snap to Grid
Fill Stroke Style Width 1.0 px Corner 0 px
Equipment
ATS
Breakers
Meters
Motors
Sources
Transformers
Skew Vertical
Objects
Tgml
Objects Layers
Properties
General
Id
Name
Appearance
Background #FFFFFF
Behavior
Stretch Uniform
UseGlobalScripts False
DisablePanAndZoom False
Name
Test
Name Value Unit Status
Equipment Components Snippets
Binds and Links Console Test
Statistics Properties
1
2
3
4
5
6
9
10

Number	Description
1.	**Equipment** Use the Equipment pane to access and manage equipment.
2.	**Components** Use the Components pane to access and manage components.
3.	**Snippets** Use the Snippets pane to access and manage functions.

Number	Description
4.	**Binds and Links** Use the Binds and Links pane to manage and test the binds and links of a graphic or a component.
5.	**Console** Use the Console pane to troubleshoot and test scripts.
6.	**Test** Use the Test pane to test the behavior of a graphic or a component.
7.	**Objects** Use the Objects pane tree structure to navigate among the objects that make up a graphic or a component.
8.	**Layers** Use the Layers pane to manage layers in a graphic.
9.	**Statistics** Use the Statistics pane to test the performance of a graphic.
10.	**Properties** Use the Properties pane to view and edit the properties of a graphic or a component.

Equipment Pane

Use the Equipment pane to access equipment and manage equipment categories:

Number	Description
Expanded equipment category	Select equipment and drag it to the work area.
Collapsed equipment category	Click the component category title to expand the equipment category.

Components Pane

Use the Components pane to access components and manage components categories:

Number	Description
Expanded component category	Select a component and drag it to the work area. For more information, see the Components Overview section.
Collapsed component category	Click the component category title to expand the component category. For more information, see the Components Overview section.

Snippets Pane

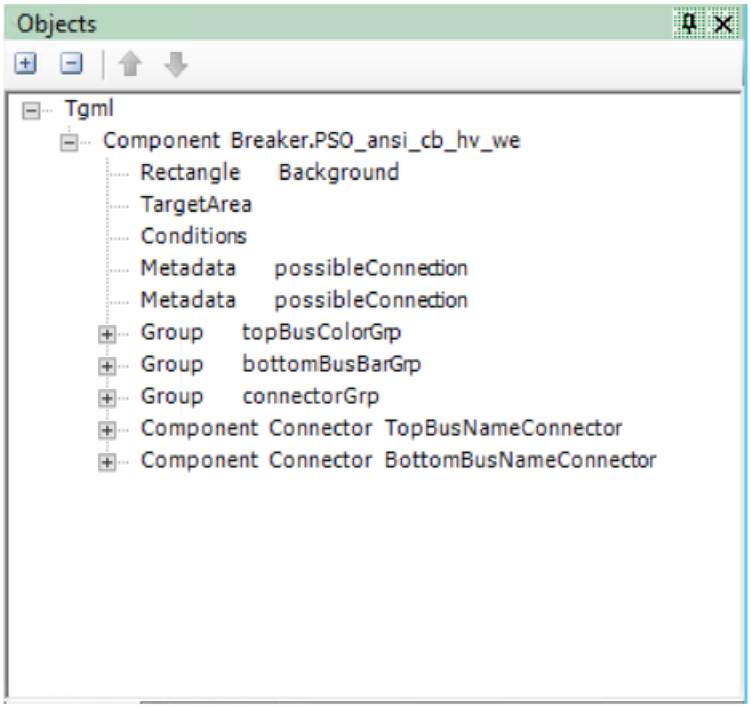
Use the Graphics Editor Snippets pane to manage function categories:

NOTE: To get a description of a snippet, right-click the snippet in the list, and then click **Properties**.

Component	Description
Basic Functions	This category contains standard functions delivered with the program. For more information, see the Snippets Overview section.
Global Bind	This category contains global bind snippets delivered with the program. For more information, see the Snippets Overview section.
Snippets Global Graphic Scripts	This category contains global graphic scripts delivered with the program. For more information, see the Snippets Overview section.
My Snippets	This category is where you save customized functions that you want to reuse in the future. For more information, see the Snippets Overview section.

Graphics Editor Objects Pane

Use the Objects pane to get an overview of the structure of a graphic and all its objects, graphical and non-graphical. You can also use the Objects pane to manage objects.



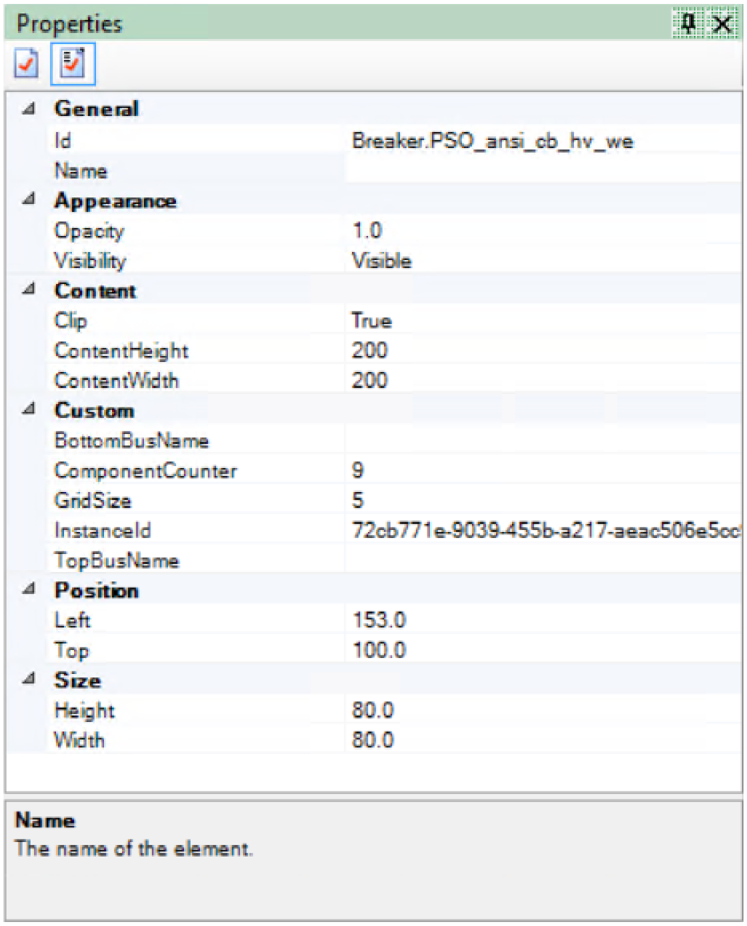
```
⊟ Tgml
  ⊟ Component Breaker.PSO_ansi_cb_hv_we
      Rectangle   Background
      TargetArea
      Conditions
      Metadata    possibleConnection
      Metadata    possibleConnection
    ⊞ Group   topBusColorGrp
    ⊞ Group   bottomBusBarGrp
    ⊞ Group   connectorGrp
    ⊞ Component Connector TopBusNameConnector
    ⊞ Component Connector BottomBusNameConnector
```

Button	Description
⊞	**Expand All** Click to show all branches in the **Tgml** tree.

Button	Description
⊟	**Collapse All** Opens the standard Save As window.
⬆	**Move Up** Moves the selected object upward in the **Tgml** tree.
⬇	**Move Down** Moves the selected object down in the **Tgml** tree.

Graphics Editor Properties Pane

Use the Properties pane to view and edit properties of the objects present in the work area.



General	
Id	Breaker.PSO_ansi_cb_hv_we
Name	
Appearance	
Opacity	1.0
Visibility	Visible
Content	
Clip	True
ContentHeight	200
ContentWidth	200
Custom	
BottomBusName	
ComponentCounter	9
GridSize	5
InstanceId	72cb771e-9039-455b-a217-aeac506e5cc
TopBusName	
Position	
Left	153.0
Top	100.0
Size	
Height	80.0
Width	80.0

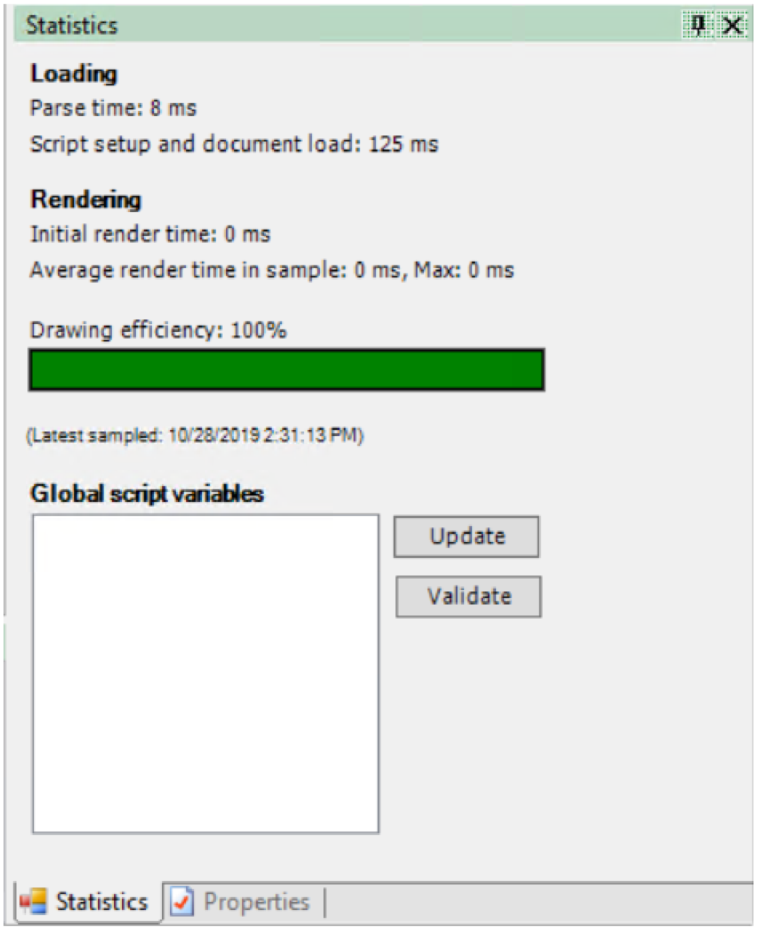
Name
The name of the element.

Button	Description
☑	**Normal** Click to display a selection of commonly used properties.
☑	**Detailed** Click to display all properties.

Which properties are displayed in the Properties pane depends on the objects included in the graphic. Graphics properties define the appearance, boundary, and behavior of the graphic.

Graphics Editor Statistics Pane

Use the Statistics pane to get information on the performance of your graphics and components:



Statistics

Loading

Parse time: 8 ms

Script setup and document load: 125 ms

Rendering

Initial render time: 0 ms

Average render time in sample: 0 ms, Max: 0 ms

Drawing efficiency: 100%

(Latest sampled: 10/28/2019 2:31:13 PM)

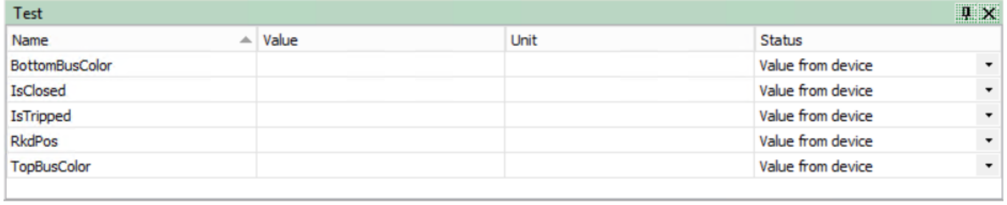
Global script variables

Component	Description
Loading	**Parse time** displays the time it takes to load the graphic file. Small files load quickly whereas large files take longer to load. If the loading process takes too long, consider reducing the number of figures included in the graphic. **Script setup and document load** display the amount of time it takes to load Script blocks, initiate and start the script engine, and execute the OnDocumentLoad scripts. If this takes too long, you can consider reducing the amount of OnDocumentLoad scripts and the total number of Script blocks in the graphic. Complex OnDocumentLoad scripts delay the opening of the graphic.

Component	Description
Rendering	**Initial render time** displays the time it takes to draw the graphics file the first time, that is, the time it takes for the system to draw all the graphic figures. The more complex the graphic is, the longer it takes to render. **Average render time in sample** displays the time it takes to update the graphic in run time. The result is updated every second. The average rendering time and the maximum rendering time are displayed.
Drawing efficiency	The bar indicates the performance of the graphic: • Green = Excellent performance • Yellow = Acceptable. Consider simplifying the graphic. • Red = The graphic could be perceived as slow. Simplify the graphic.
Update	Click to analyze all the scripts and refresh the view to show all global variables, that is, variables not declared as 'var'.
Validate	Click to analyze the found Global script variables and notify which variables are in conflict with reserved names.

Graphics Editor Test Pane

Use the Test pane to test the dynamic behavior in Preview mode.



Test

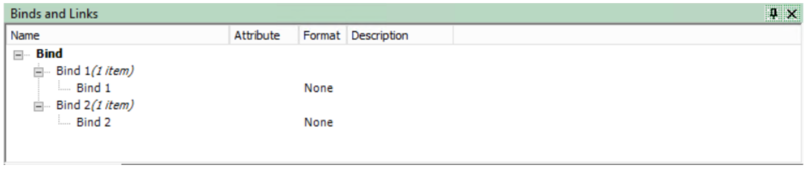
Name	Value	Unit	Status
BottomBusColor			Value from device
IsClosed			Value from device
IsTripped			Value from device
RkdPos			Value from device
TopBusColor			Value from device

Component	Description
Name	The name of the object you are testing.
Value	Tests a bind, that is, a dynamic behavior. Enter a value to start the test.
Unit Status	Tests the unit management under the condition that you have used `getUnit` in your scripts. Enter any text.
Status	Simulates signal status. "Value from device" is the normal status.

Graphics Editor Binds and Links Pane

Use the Binds and Links pane to get an overview of the binds and links used in a graphic and their properties.

NOTE: The actual binding and linking are performed in the Graphics Editor.



Name	Attribute	Format	Description
Bind			
Bind 1*(1 item)*			
Bind 1		None	
Bind 2*(1 item)*			
Bind 2		None	

Component	Description
Name	Displays the name of the bind/link, consisting of the object’s name and, for the binds, a suffix normally ‘Value’. Name also shows the number of bind values or link targets that use this name. (One signal can affect several properties.)
Attribute	Displays the property (if present) that is affected by the bound value.
Format	None or Presentation. As selected in the Properties pane, under Behavior - Format.
Description	Displays an optional descriptive text.

Graphics Editor Layers Pane

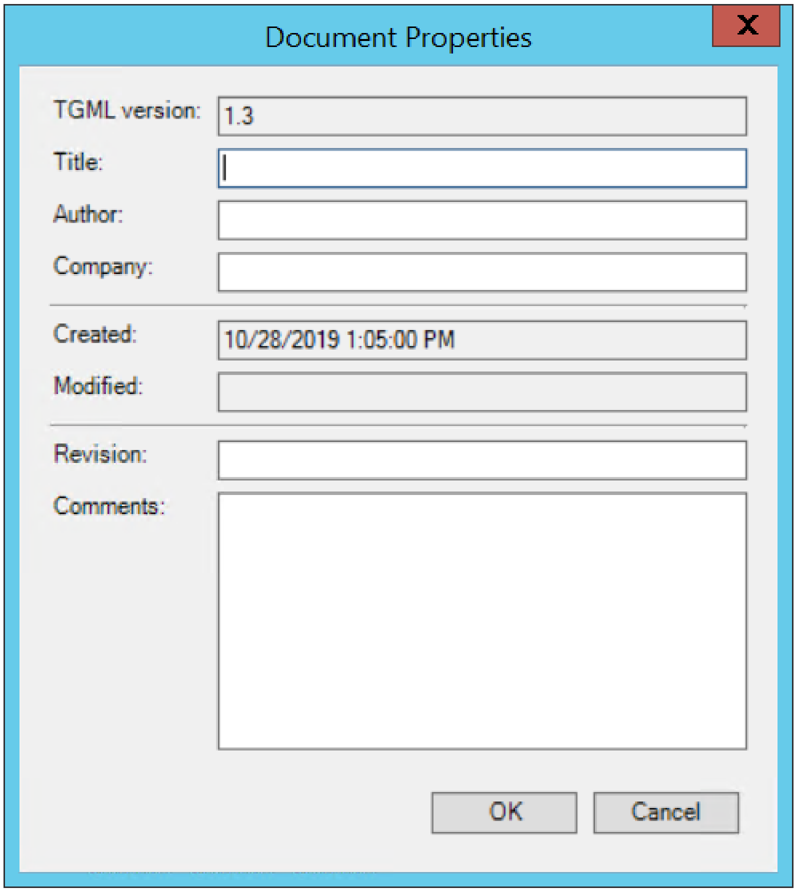
Use the Layers pane to manage layers in a graphic.

Button	Description
⊞	**New layer** Click to create a new layer.
⧉	**Duplicate layer** Click to duplicate the current layer.
❐	**Merge layer** Click to merge the selected layers.
☒	**Delete layer** Click to delete the current layer.
⬆	**Move up** Moves the selected layer upward in the layer pane.
⬇	**Move down** Moves the selected layer down in the layer pane.

Column	Description
1st Column	Click to select the active layer. For more information, see the Layers Overview section.
2nd Column	Displays the graphics contents of the layer.
3rd Column	Displays the name of the layer (the Name property).
4th Column	Select to make the layer visible.

Document Properties Dialog Box

Use the Document Properties dialog box to view, enter, or edit details on the creation of the TGML graphic.



TGML version: 1.3

Created: 10/28/2019 1:05:00 PM

Property	Description
TGML Version	Displays the TGML version of the TGML graphic.
Title	View, add, or edit the title of the TGML graphic.
Author	View, add, or edit the name of the author who created the TGML graphic.
Company	View, add, or edit the name of the company the author belongs to.
Created	Displays the date the TGML graphic was created.
Modified	Displays the date the TGML graphic was modified.
Revision	View, add, or edit the document revision number.
Comments	View, add, or edit the document comments.

Unsupported Characters

Object names cannot include any of the following characters: exclamation point (!), quotation mark (“), number sign (#), percent sign (%), ampersand (&amp;), apostrophe ('), left parenthesis ((), right parenthesis ()), asterisk (*), plus sign (+), comma (,), hyphen-minus (-), semicolon (;), less than sign (&lt;), greater than sign (&gt;), equals sign (=), question mark (?), backslash (\), or pipe symbol (|).

In addition, the following restrictions apply:

- Object names can contain spaces, however, leading and trailing spaces are not supported in objects names.
- Object names can contain full stops (.), however, leading and trailing full stops (.) are not supported in object names.
- Object names cannot be empty names.

Workflows

For more information, see the following workflows:

- "Binding and filtering alarm counts" on page 1206
- Configuring Arc Flash Graphics
- Control Operation
- Pop-Ups
- Graphics De-Cluttering

Binding and filtering alarm counts

You can bind alarm counts in TGML graphic pages and render the counts of different alarm types at the cluster and equipment level.

You can bind the following alarm type counts:

- TotalAlarmsCount
- UnackowledgeAlarmsCount
- ActiveAlarmsCount
- ActiveAndUnacknowledgedAlarm sCount
- ActiveOrUnacknowledgeAlarmsCount

The alarm filters are also applied on the count of a specific device type or on a cluster. Currently, the following alarm filters are supported:

- Priority
- Alarm Type
- IncidentId
- AlarmDefinitionId
- DateRange

Alarm filters can be applied in TGML components using the custom `GroupBy` property.

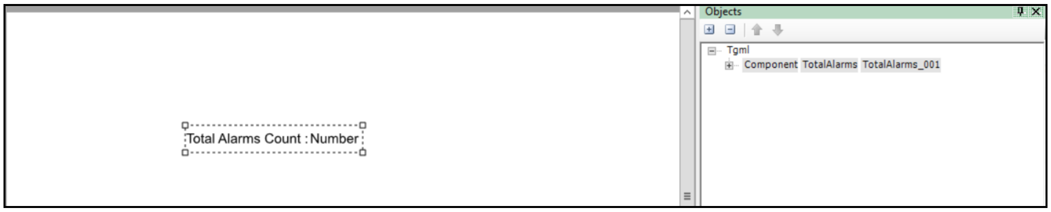
NOTE: For more information, see "Alarm count grouping" on page 568.

The following alarm count components are available by default in the alarm counts library:

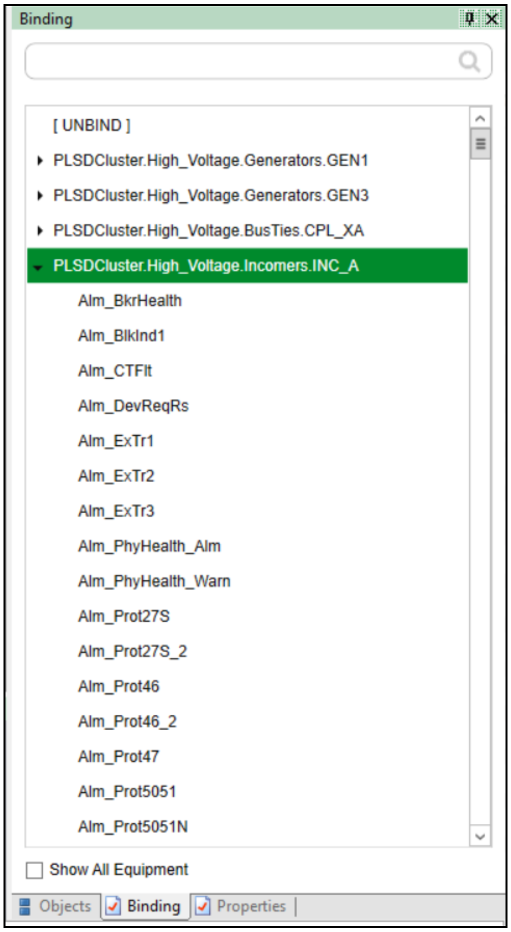
- Active alarms
- Active alarms with filter
- Active and unacknowledged alarms
- Active and unacknowledged alarms with filters
- Active or unacknowledged alarms
- Active or unacknowledged alarms with filters
- Total alarms
- Total alarms with filters
- Unacknowledged alarms
- Unacknowledged alarms with filters

The default alarm count components can be dragged and dropped to the workspace in the TGML Graphics editor.

For example, the **TotalAlarms** component is dragged to the workspace as shown in the below screen.


Total Alarms Count :Number
Objects
Tgml
Component TotalAlarms TotalAlarms_001

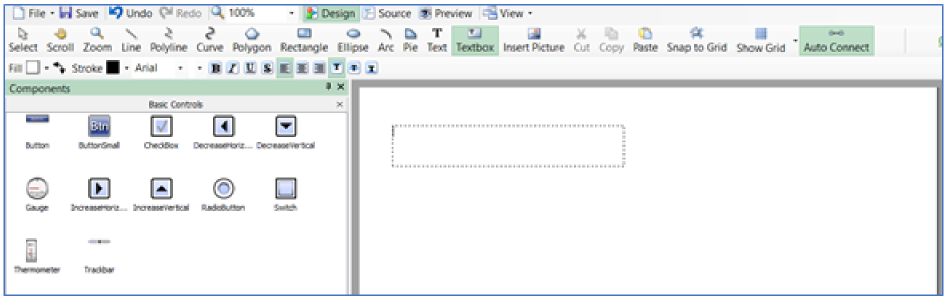
Assign the bind to the selected component and save the TGML graphic file.



NOTE: Inactive acknowledged alarms count will be excluded from TotalAlarms count.

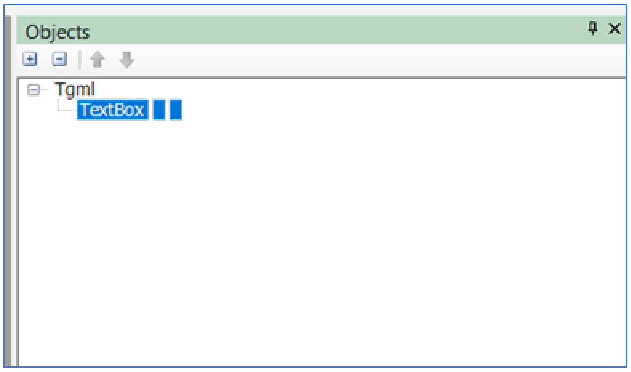
The following steps describe the compete workflow for creating alarm binds and using alarm filters:

1. In the Graphics Editor, click **New > Graphic**.
2. Select and then drag a **Textbox** to the workspace.

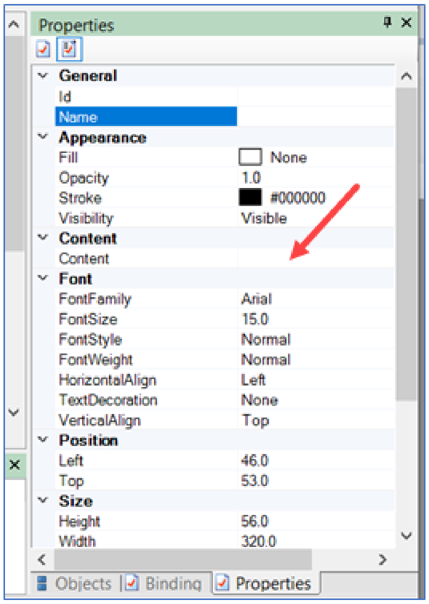


3. At the bottom right corner, click **Objects**.

   The following is displayed:

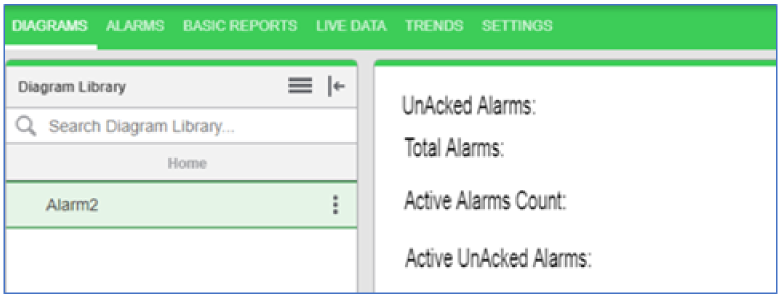


4. At the right bottom, click **Properties**.
5. In the **Content** section, enter the value of content as **UnAcked Alarms**.



General	
Id	
Name	
Appearance	
Fill	□ None
Opacity	1.0
Stroke	■ #000000
Visibility	Visible
Content	
Content	
Font	
FontFamily	Arial
FontSize	15.0
FontStyle	Normal
FontWeight	Normal
HorizontalAlign	Left
TextDecoration	None
VerticalAlign	Top
Position	
Left	46.0
Top	53.0
Size	
Height	56.0
Width	320.0

6. Repeat the steps from 2 through 5 for the other labels as shown in the following image:



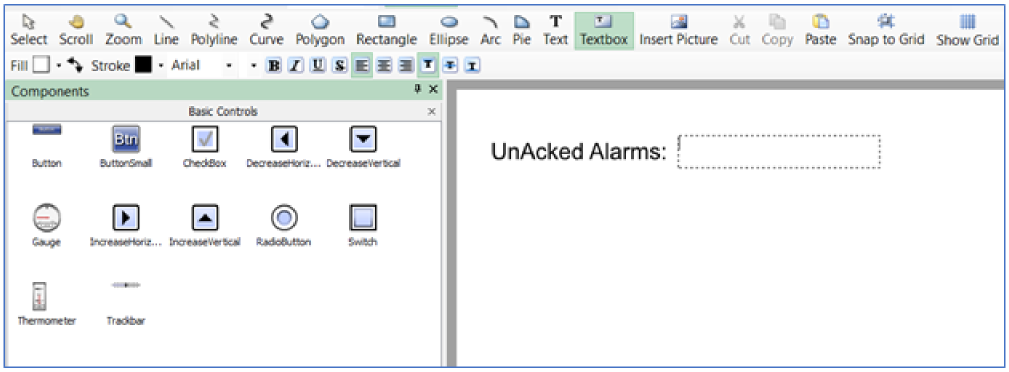
UnAcked Alarms:

Total Alarms:

Active Alarms Count:

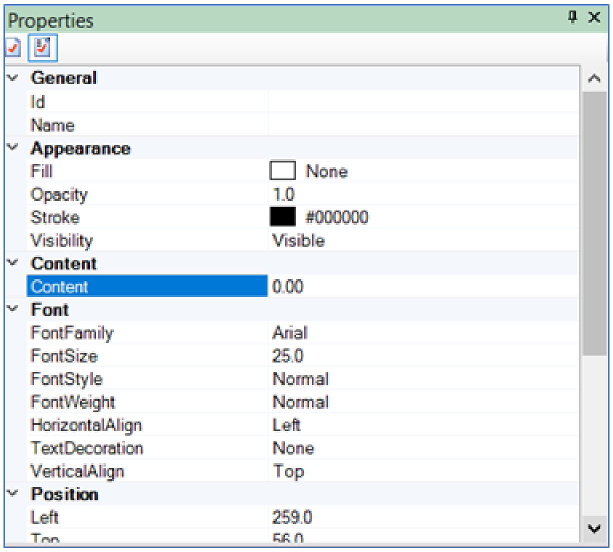
Active UnAcked Alarms:

7. Enter the Content values according to the labels that are created.
8. Select the **TextBox** and drop it next to the label.



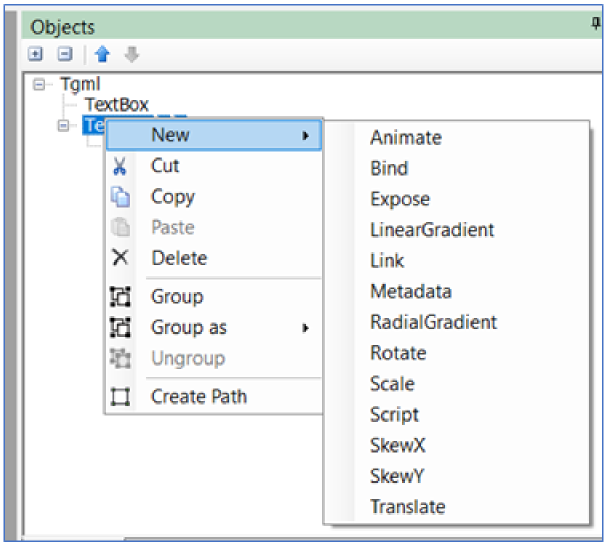
UnAcked Alarms:

9. Click the text box on the screen.
10. Click the **Properties** tab.
11. In the **Content** section, enter the value of content as **0.00**.



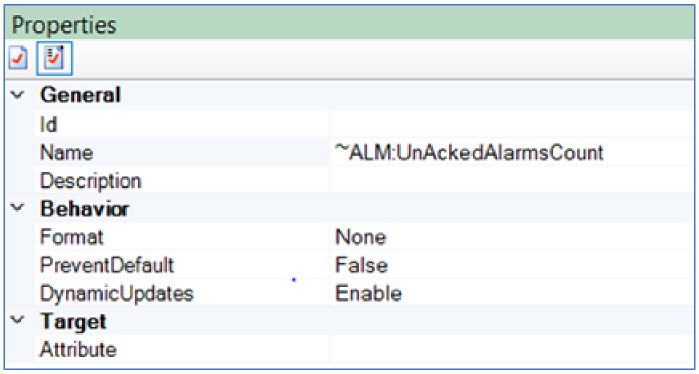
General	
Id	
Name	
Appearance	
Fill	□ None
Opacity	1.0
Stroke	■ #000000
Visibility	Visible
Content	
Content	0.00
Font	
FontFamily	Arial
FontSize	25.0
FontStyle	Normal
FontWeight	Normal
HorizontalAlign	Left
TextDecoration	None
VerticalAlign	Top
Position	
Left	259.0
Top	56.0

12. Click the **Object** tab.
13. Select **TextBox > New > Bind**.



14. Click the **Properties** tab.

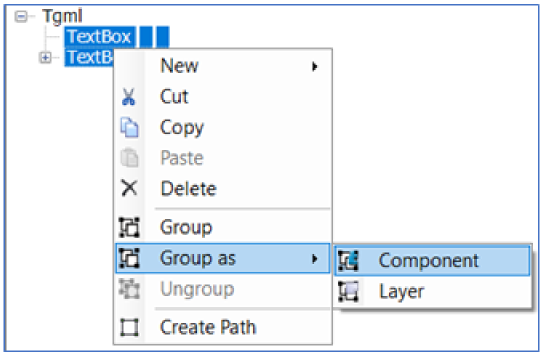
15. In the **Name** section, the bind name as **~ALM:UnAckedAlarmsCount**.


Properties

General	
Id	
Name	~ALM:UnAckedAlarmsCount
Description	
Behavior	
Format	None
PreventDefault	False
DynamicUpdates	Enable
Target	
Attribute	

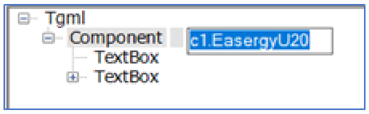
16. (Optional) To display the alarm counts of device type:

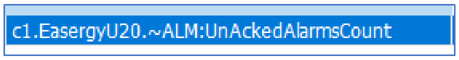
    a. Select both **TextBox** objects, and then click **Group as > Component**.



    b. Enter the value of **Component** as shown below:

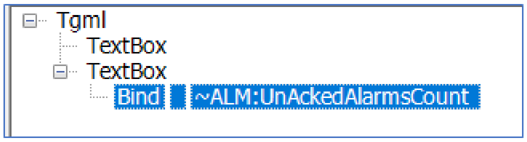
    Alarm counts for device type:



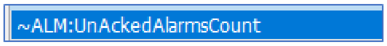


c1.EasergyU20.~ALM:UnAckedAlarmsCount

Alarm counts for cluster:



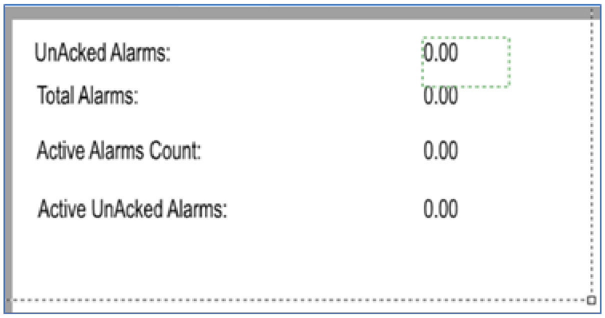
Tgml
TextBox
TextBox
Bind ~ALM:UnAckedAlarmsCount



~ALM:UnAckedAlarmsCount

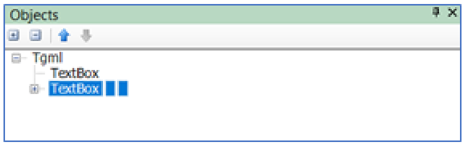
> **NOTE**: If the device type is specified, TGML shows the unacknowledged alarms count of device type. If the device type is not specified, TGML shows the unacknowledged alarms count of cluster.

17. Create custom property **GroupBy**.
18. Save the TGML file.
19. Repeat the steps from 8 through 12 to create the other alarm binds (**~ALM:ActiveAlarmsCount**, **~ALM:ActiveUnAckedAlarmsCount**, and **~ALM:TotalAlarms**) in TGML.



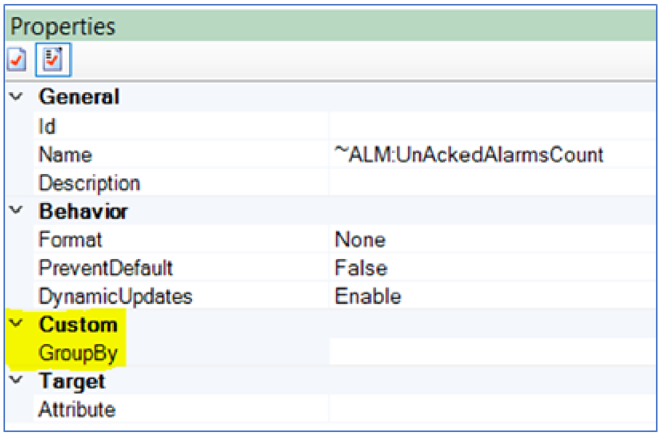
UnAcked Alarms: 0.00
Total Alarms: 0.00
Active Alarms Count: 0.00
Active UnAcked Alarms: 0.00

20. Click the **Object** tab.
21. Click **TextBox**.



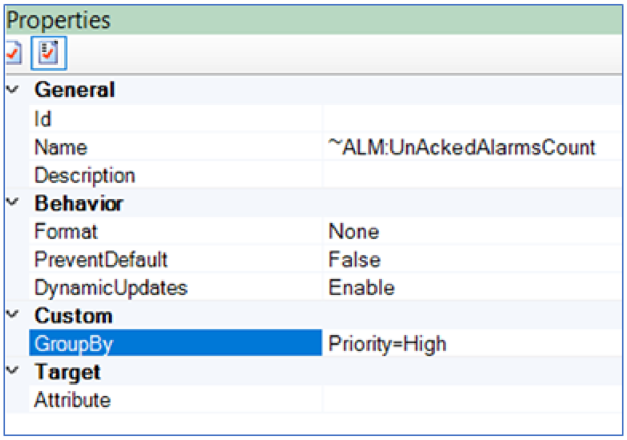
Tgml
TextBox
TextBox

22. If the custom property **GroupBy** does not exist, create a custom property with the same name.


Properties

General	
Id	
Name	~ALM:UnAckedAlarmsCount
Description	
Behavior	
Format	None
PreventDefault	False
DynamicUpdates	Enable
Custom	
GroupBy	
Target	
Attribute	

23. Apply the filters, type of filters, and its value in the custom property **GroupBy**.

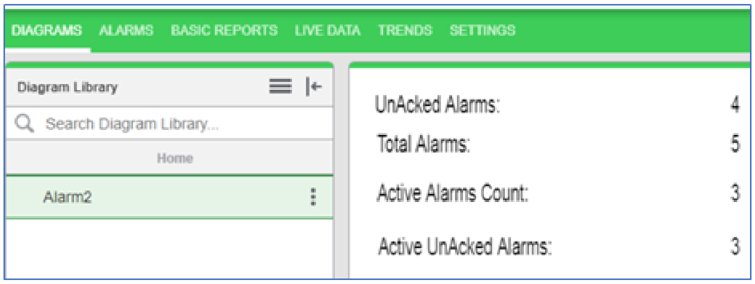

Properties

General	
Id	
Name	~ALM:UnAckedAlarmsCount
Description	
Behavior	
Format	None
PreventDefault	False
DynamicUpdates	Enable
Custom	
GroupBy	Priority=High
Target	
Attribute	

24. Save the TGML file.

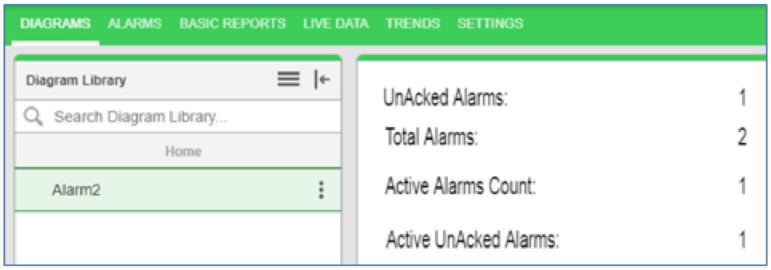
25. Load the saved TGML file in Web Applications to display the alarm counts.

Alarms TGML without filters:



UnAcked Alarms:	4
Total Alarms:	5
Active Alarms Count:	3
Active UnAcked Alarms:	3

Alarms TGML with filters (**Priority=High**):



UnAcked Alarms:	1
Total Alarms:	2
Active Alarms Count:	1
Active UnAcked Alarms:	1

26. If you want to apply multiple filters, each filter should be separated by the ampersand symbol **&**. For example:

    GroupBy: Priority=High&Type=OverCurrent&IncidentID=19&AlarmDefinitionId=Test&DateRange=13/03/2020 to 20/03/2020

Configuring Arc Flash Graphics

⚠ **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**

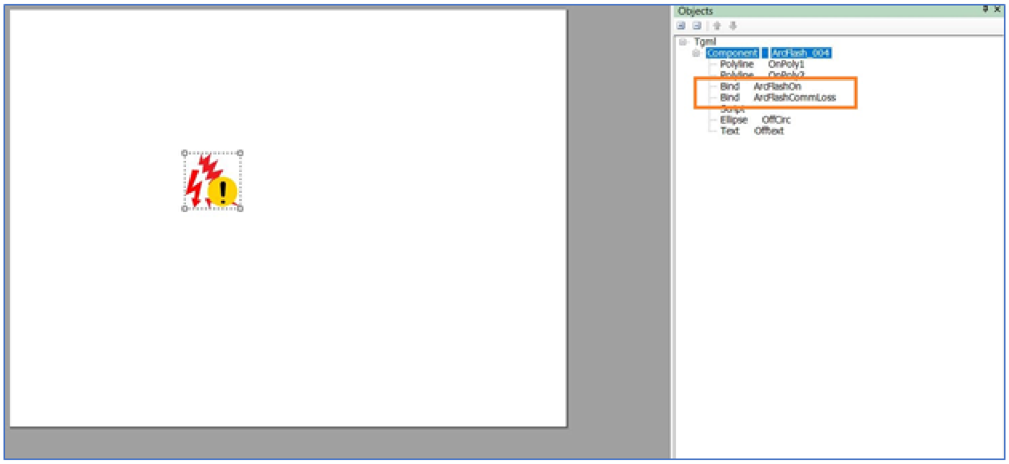
UNINTENDED EQUIPMENT OPERATION

- 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, or equipment damage.

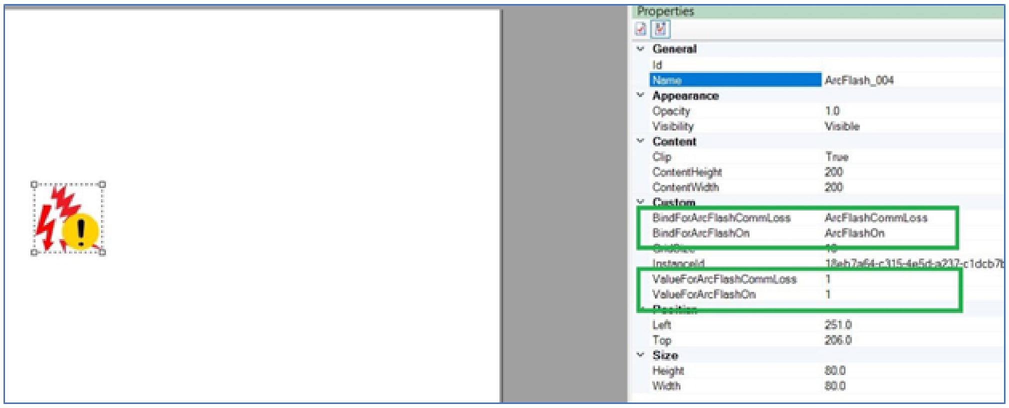
To configure an Arc Flash graphic:

1. Open the Graphics Editor.
2. Open a single line diagram or other TGML graphic.
3. Drag and drop the **ArcFlash** symbol.
4. Select the symbol and then select the **Objects** pane.
5. Edit the Bind values to appropriate names based on the project or field configuration and type of device.



```
Tgml
  Component    ArcFlash_004
    Polyline   OnPoly1
    Polyline   OnPoly2
    Bind   ArcFlashOn
    Bind   ArcFlashCommLoss
    Script
    Ellipse    OffCirc
    Text   Offtext
```

6. Select the **Properties** pane.
7. In the Custom section, in the BindForArcFlashCommLoss and BindForArcFlashOn fields, enter the same Bind name values you entered in step 5.
8. In the Custom section, enter the values for ValueForArcFlashCommLoss and ValueForArcFlashOn.



General	
Id	
Name	ArcFlash_004
Appearance	
Opacity	1.0
Visibility	Visible
Content	
Clip	True
ContentHeight	200
ContentWidth	200
Custom	
BindForArcFlashCommLoss	ArcFlashCommLoss
BindForArcFlashOn	ArcFlashOn
GridSize	10
InstanceId	18eb7a64-c315-4e5d-a237-c1dcb7b
ValueForArcFlashCommLoss	1
ValueForArcFlashOn	1
Position	
Left	251.0
Top	206.0
Size	
Height	80.0
Width	80.0

NOTE: To display the arc flash symbol states correctly, configure the device bind values ArcFlashCommLoss and ArcFlashOn to match the respective graphics properties ValueForArcFlashCommLoss and ValueForArcFlashOn values.

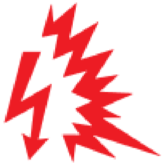
9. To bind the arc flash component to the respective arc flash field device, use the same binding process as when creating other graphics or advanced one-line components.

NOTE: For more information, see Adding a graphics page in the Graphics Editor or Creating a one-line on a graphics page.

10. Click **Save**.

Displaying Arc Flash States

In the event of an arc flash or an issue with the arc flash monitoring system, a symbol will display next to the device that detected the issue:

Symbol	State	Description
	Red arc flash symbol	Arc flash is detected.
	Gray arc flash symbol with yellow exclamation mark.	Any issue with the Arc Flash Monitoring System.

Control Operation

You can use the Control Operation snippet to control equipment, circuit breakers, and to change device states. Only authorized users can perform this control operation.

⚠ 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

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, or equipment damage.

To configure and operate controls:

1. Open the Graphics Editor.
2. Select the **Components** pane.
3. To configure the default Control component:

a. Select the **Control** component and drag and drop it on the workspace.

NOTE: The Control component contains the Control Operation snippet, which includes two additional properties: Link and Script.

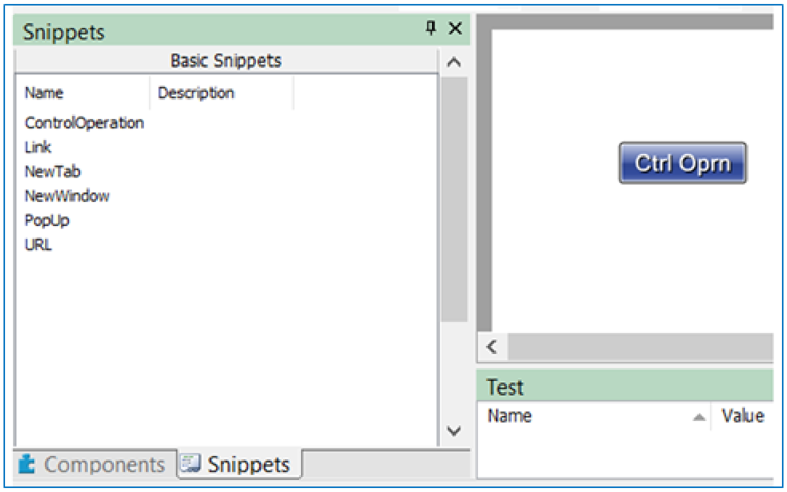
b. Select the **Properties** pane and enter the DataPoint value.

NOTE: The Control component contains the DataPoint attribute by default.

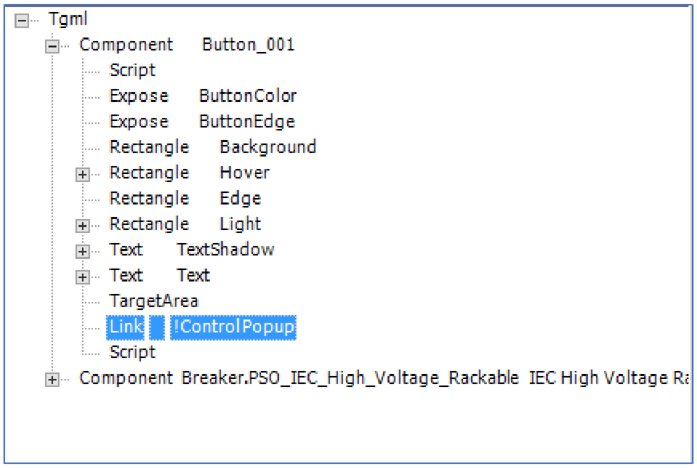
4. To build a custom Control component:

a. Select the required component.

b. Drag and drop the **Control Operation** snippet over the selected component in the workspace.

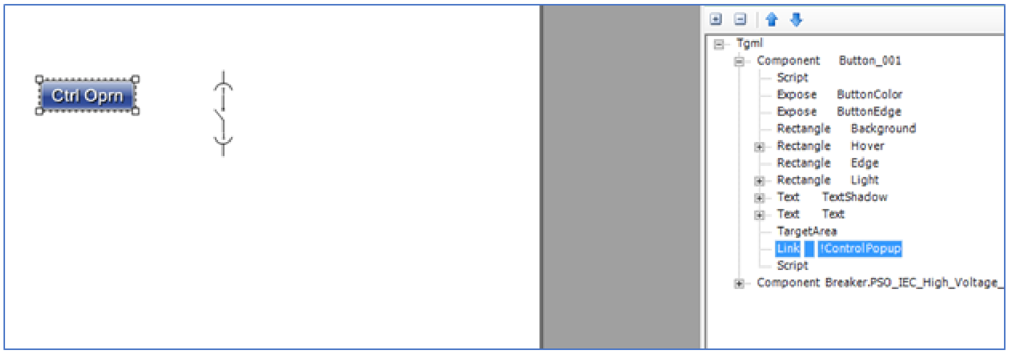


NOTE: The **Control Operation** snippet adds two additional properties: Link and Script.



```
Tgml
  Component   Button_001
    Script
    Expose    ButtonColor
    Expose    ButtonEdge
    Rectangle    Background
    Rectangle    Hover
    Rectangle    Edge
    Rectangle    Light
    Text    TextShadow
    Text    Text
    TargetArea
    Link    !ControlPopup
    Script
  Component  Breaker.PSO_IEC_High_Voltage_Rackable  IEC High Voltage Ra
```

- The Link default name is **Control Popup**.



- The Script creates the following syntax:

```
invoke(connectorName, "Type = PopUp | ComponentName=" +
componentName + " | InstanceID=" + instanceId + "  | DataPoint = "+
dataPoint +"| Title=" + title + " | Width=" + width + " | Height=" +
height + " | ShowTitleBar =" + showTitleBar + " | ShowUnamePwd =" +
showUnamePwd +" | UserCredBottom = "+usercredbottom +" |
UserCredLeft = "+ usercredleft+" | UserCredWidth = "+ usercredwidth
+" | UserCredHeight = "+usercredheight +" | UserCredBackColor =
"+usercredbackcolor+" |  UnamePwdWidth = "+unamepwdwidth+" |
UnamePwdColor = "+unamepwdcolor);
```

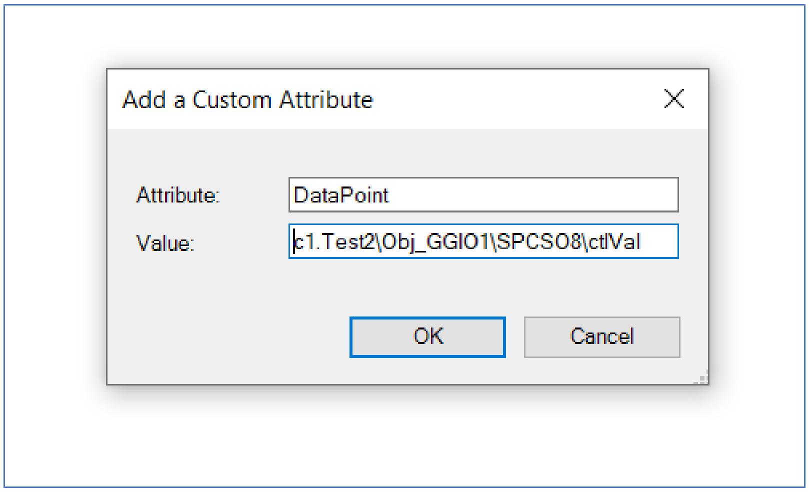
You can modify the following default parameters to configure the pop-up window:

Parameter	Default value
width	40
height	60
showTitleBar	"Yes"
showUnamePwd	"Yes"
usercredbottom	36
usercredleft	10
usercredbackcolor	"white"
usercredwidth	80
usercredheight	25
unamepwdwidth	50
unamepwdcolor	"#9FA0A4"

c. Select the **Properties** pane and click **Add**.

d. Enter the following details:

- **Attribute**: Datapoint
- **Value**: c1.Test2\Obj_GGI01\SPCS08\ctlVal



Add a Custom Attribute

Attribute: DataPoint

Value: c1.Test2\Obj_GGIO1\SPCSO8\ctlVal

   e. Click **OK**.

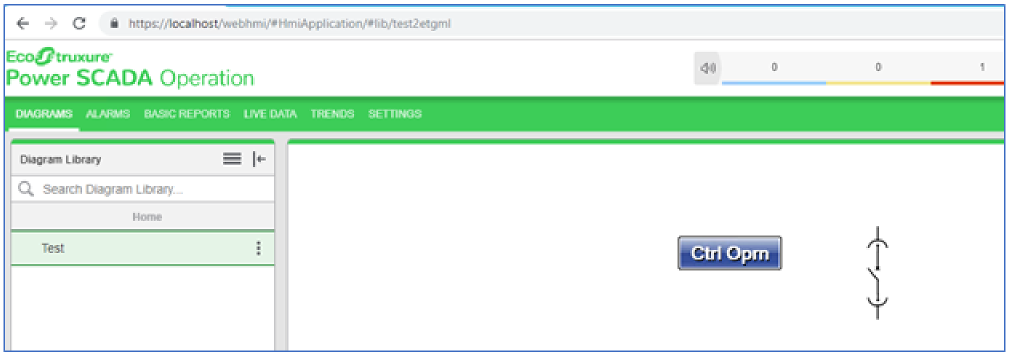
5. Navigate to the following path to save the file:

   `File > Save As > Project TGML`

6. Type the file name in the **File name** field.
7. Click **Save**.
8. Open a web browser.
9. Type `https://localhost/webhmi` in the address bar.

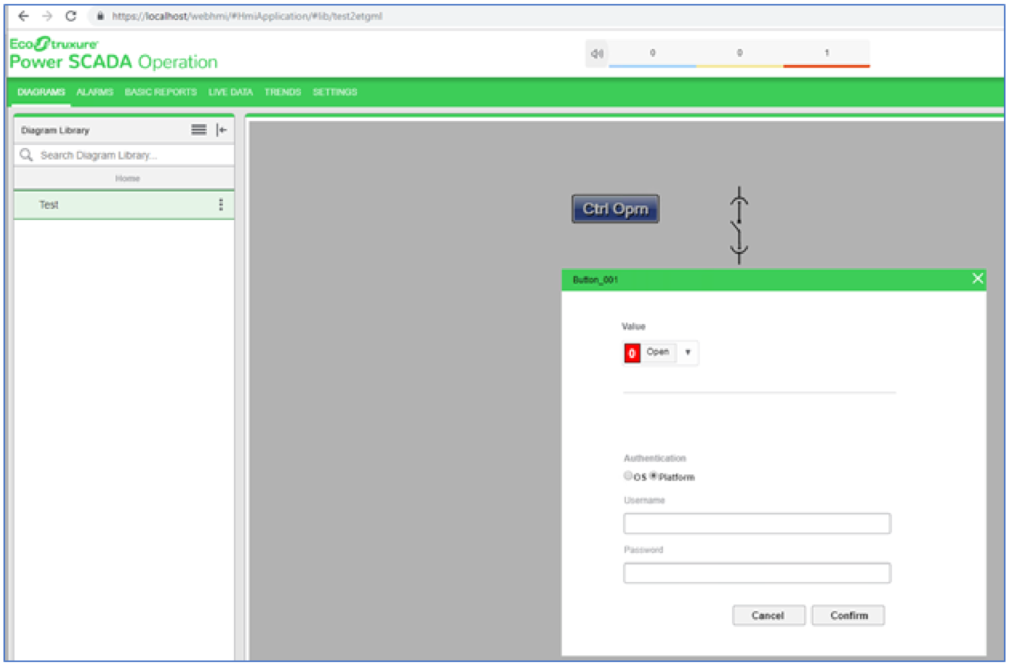
   NOTE: If Power SCADA Operation 2020 is installed on the remote server, type a valid URL in the address bar.

10. Click **Enter**. The Power SCADA Web Applications Home page appears.
11. Select the new TGML file.



https://localhost/webhmi/#HmiApplication/#lib/test2etgml

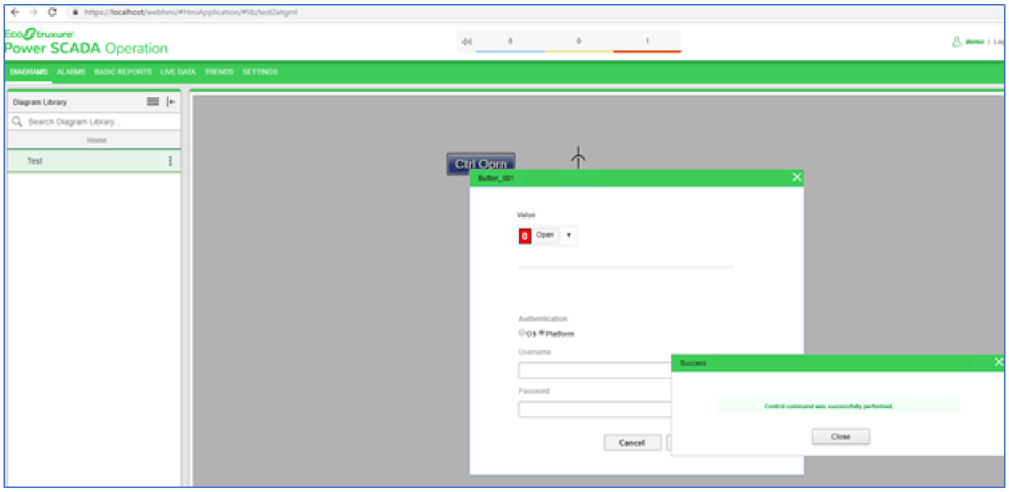
12. Click the **Control** component to open the control operation pop-up
13. Select a value (**Open** or **Close**) to perform the operation.



14. Enter **Username** and **Password**.

15. Click **Confirm**, and one of the following messages will appear:

- **Success:** Appears when the selected value (**Open**/**Close**) is updated on the device.

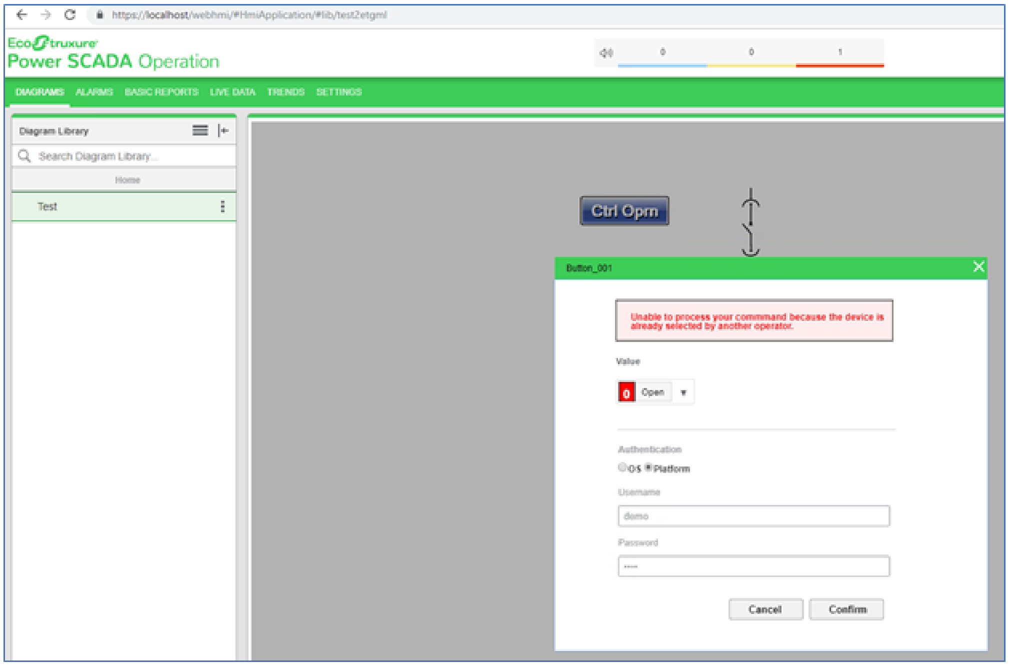


Control command was successfully performed

- **Unsuccessful:** Appears if the operation fails for one of several reasons, including:

  - The device is already in the selected state
  - The device selected by another user

- Another device problem or issue



Unable to process your commmand because the device is already selected by another operator.

Value

0 Open

Authentication

○ OS ● Platform

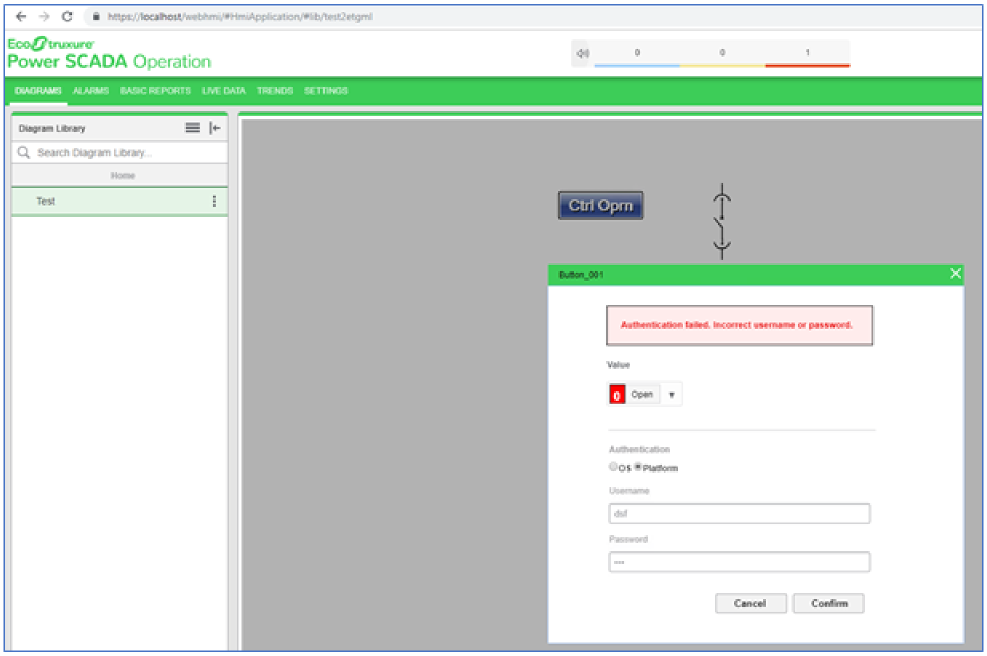
Username

demo

Password

••••

- **Authentication failed:** Appears if the provided credentials are not valid:



Authentication failed. Incorrect username or password.

Value

0 Open

Authentication

○ OS ● Platform

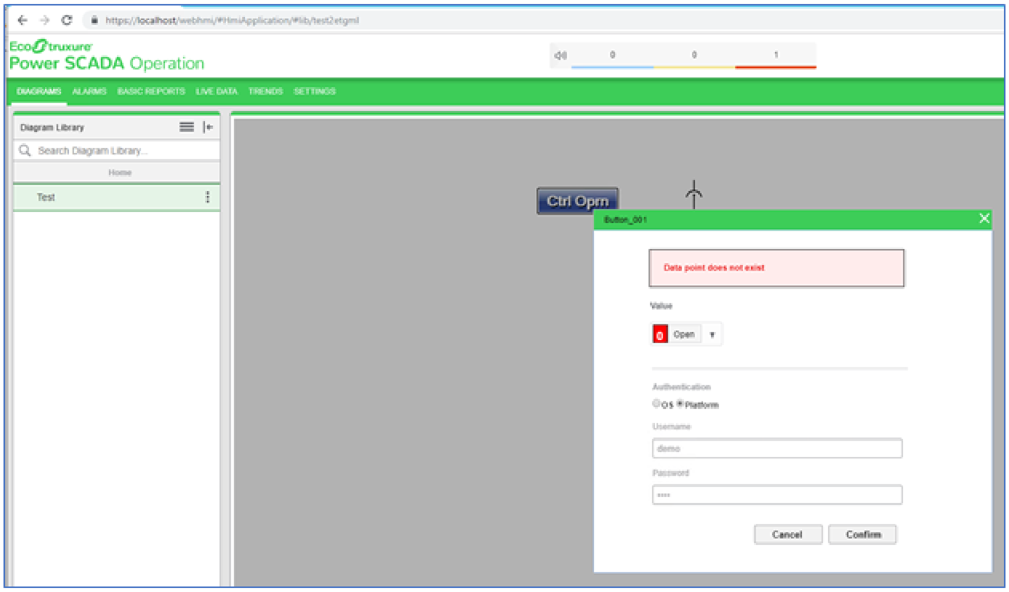
Username

dsf

Password

•••

- **Data point does not exist:** Appears if the provided tag names are not correct:



https://localhost/webhmi/#HmiApplication/#lib/test2etgml

Button_001

Data point does not exist

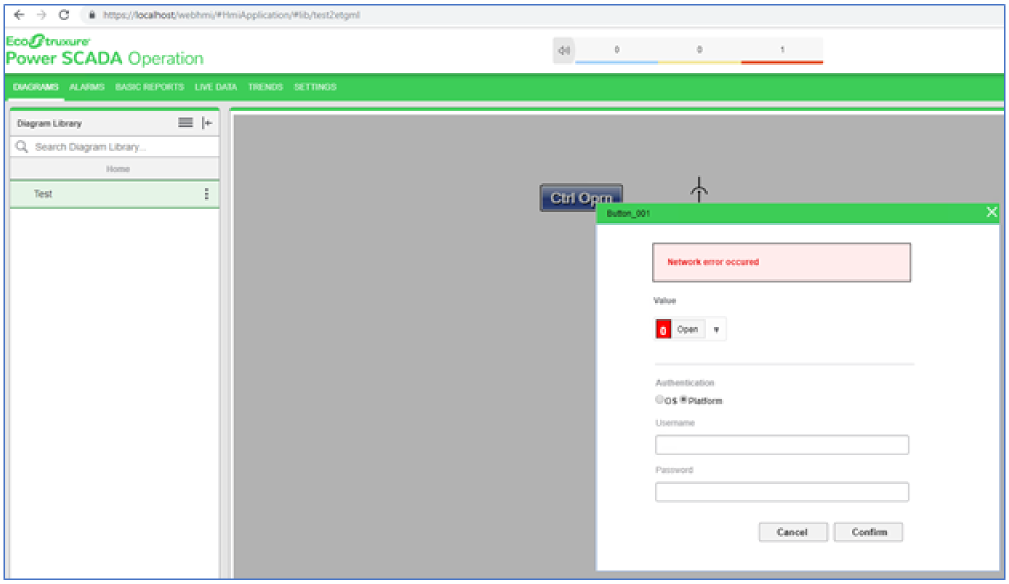
Value: 0 Open

Authentication: ○ OS ● Platform

Username: demo

Password: ****

- **Network error:** Appears if any network related issue occurs:



https://localhost/webhmi/#HmiApplication/#lib/test2etgml

Button_001

Network error occured

Value: 0 Open

Authentication: ○ OS ● Platform

Username

Password

Pop-Ups

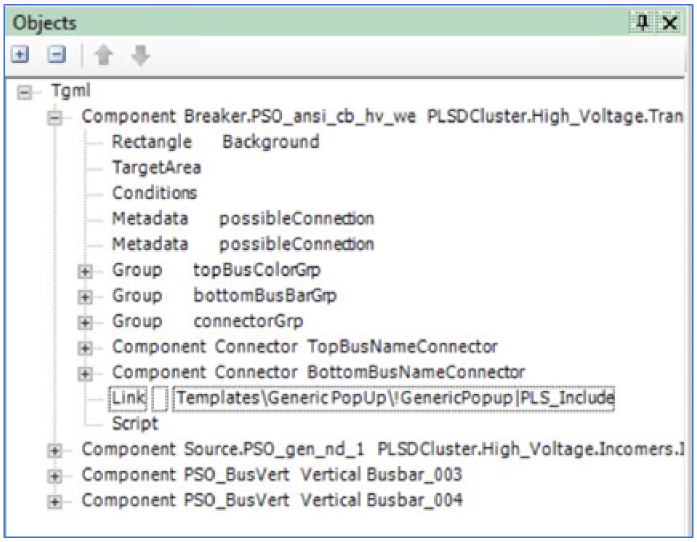
You can configure pop-ups to display real-time device readings.

To configure a pop-up:

1. Open the Graphics Editor.
2. In the Components pane, select a component and drag and drop it on the workspace.
3. In the Binding pane, select a component or device to bind to the selected component.
4. From the Snippets pane, drag and drop the **PopUp** snippet onto the component in the workspace. Two additional properties appear: Link and Script.

a. **Link**: Enter the page to be opened.

b. **Script**: Enter the display type: Link, New Tab, New Window, PopUp, or URL. Different snippets are available for these types.



```
Tgml
  Component Breaker.PSO_ansi_cb_hv_we  PLSDCluster.High_Voltage.Tran
    Rectangle    Background
    TargetArea
    Conditions
    Metadata    possibleConnection
    Metadata    possibleConnection
    Group    topBusColorGrp
    Group    bottomBusBarGrp
    Group    connectorGrp
    Component Connector  TopBusNameConnector
    Component Connector  BottomBusNameConnector
    Link    Templates\Generic PopUp\!GenericPopup|PLS_Include
    Script
  Component Source.PSO_gen_nd_1  PLSDCluster.High_Voltage.Incomers.I
  Component PSO_BusVert  Vertical Busbar_003
  Component PSO_BusVert  Vertical Busbar_004
```

NOTE: All the TGML templates, including the generic pop-up, are in: `C:\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020 R2\Applications\Services\Platform Server\PLS_Include\TGML\Templates`

- The generic pop-up file is in: `Templates\Generic PopUp\!GenericPopup|PLS_Include`
- For Diagrams to render a pop-up based on your configuration, copy the required TGML templates to: `C:\ProgramData\Schneider Electric\Power SCADA Operation\v2020 R2\User\Include\TGML\Templates`
- Create the TGML\Templates folder if it is not available in the location above.
- It is recommended that you keep the source and destination hierarchy similar to the default folder structure for seamless pop-up or link navigation. Any change in folder structure or file rename will require you to reconfigure the link property in the TGML graphic.

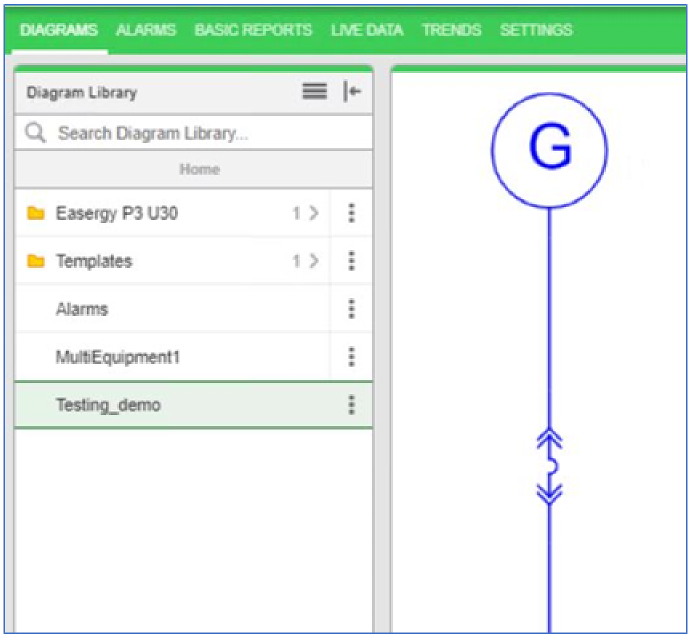
5. Navigate to the following path to save the file:

   `File > Save As > Project TGML`

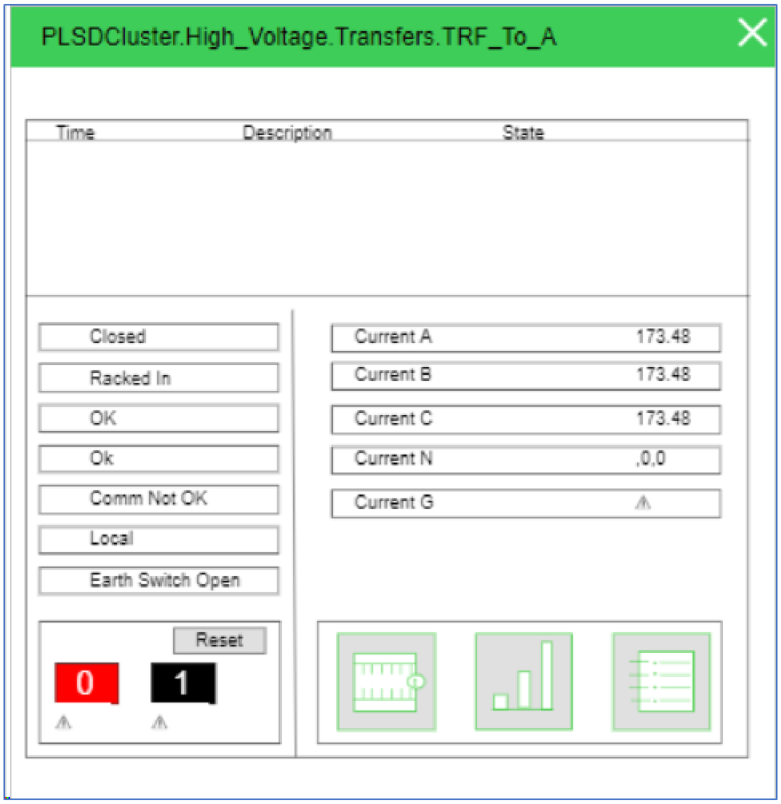
6. Type the file name in the **File name** field.
7. Click **Save**.
8. The **Connection Debugger** window appears. Select any required properties based on the one-line diagram.
9. Open a web browser.
10. Type `https://localhost/webhmi` in the address bar.

> **NOTE**: If Power SCADA Operation 2020 R2 is installed on the remote server, type a valid URL in the address bar.

11. Click **Enter**. The Power SCADA Web Applications Home page appears.
12. Select the new TGML file.



13. Click the graphic to open a pop-up displaying real time readings from the component.


PLSDCluster.High_Voltage.Transfers.TRF_To_A

Time	Description	State

Closed

Racked In

OK

Ok

Comm Not OK

Local

Earth Switch Open

Current A	173.48
Current B	173.48
Current C	173.48
Current N	,0,0
Current G	⚠

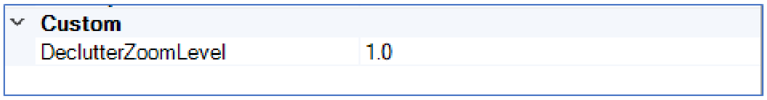
Graphics de-cluttering

Graphics de-cluttering allows you to show or hide elements of a detailed one-line diagram. When zooming in or out, you can display different levels of detail, and monitor high-level equipment.

To de-clutter graphics:

1. Design the page and, using layers, differentiate between components that will always be visible and components that will only be visible after zooming.
2. Add a custom property containing the hidden components to the second layer.

   NOTE: This layer provides information about the zoom level and the visibility of the graphics. Set the DeclutterZoomLevel value to **1.0** to make the hidden layer visible after zooming over 100%. Customize this value to meet your requirements.

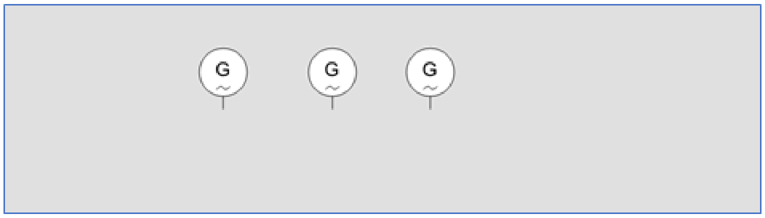
   Custom

   DeclutterZoomLevel 1.0

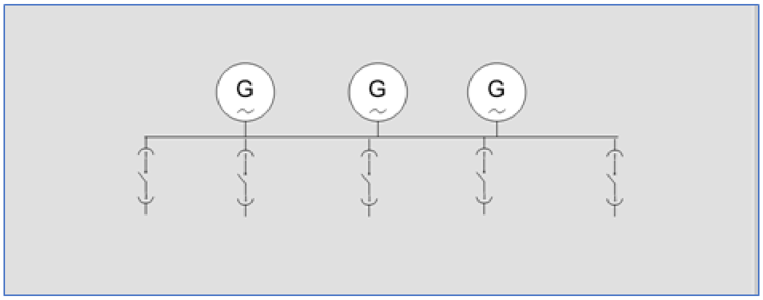
3. Use the custom property in the script load event function to hide or show the layer in the zoom event:

```
function declutter(element, declutterZoomLevel) {
  var currentVisibility = element.getAttribute("Visibility");
  var wantedVisibility = view.zoomLevel > declutterZoomLevel ? "Visible" :
"Hidden";
  if (currentVisibility != wantedVisibility) {
    element.setAttribute("Visibility", wantedVisibility);
  }
}

function load(evt)
{
  var layer = evt.getCurrentTarget();
  var declutterZoomLevel = parseFloat(layer.getAttribute
("DeclutterZoomLevel"));
  declutter(layer, declutterZoomLevel);
  view.addEventListener('resize', function () {
    declutter(layer, declutterZoomLevel);
  });
}
```

   - Normal TGML with 100% zoom:



- TGML with more than 100% zoom:



Legacy Graphics Builder References

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 can be created using the legacy Graphics Builder.

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 more information, see the following sections:

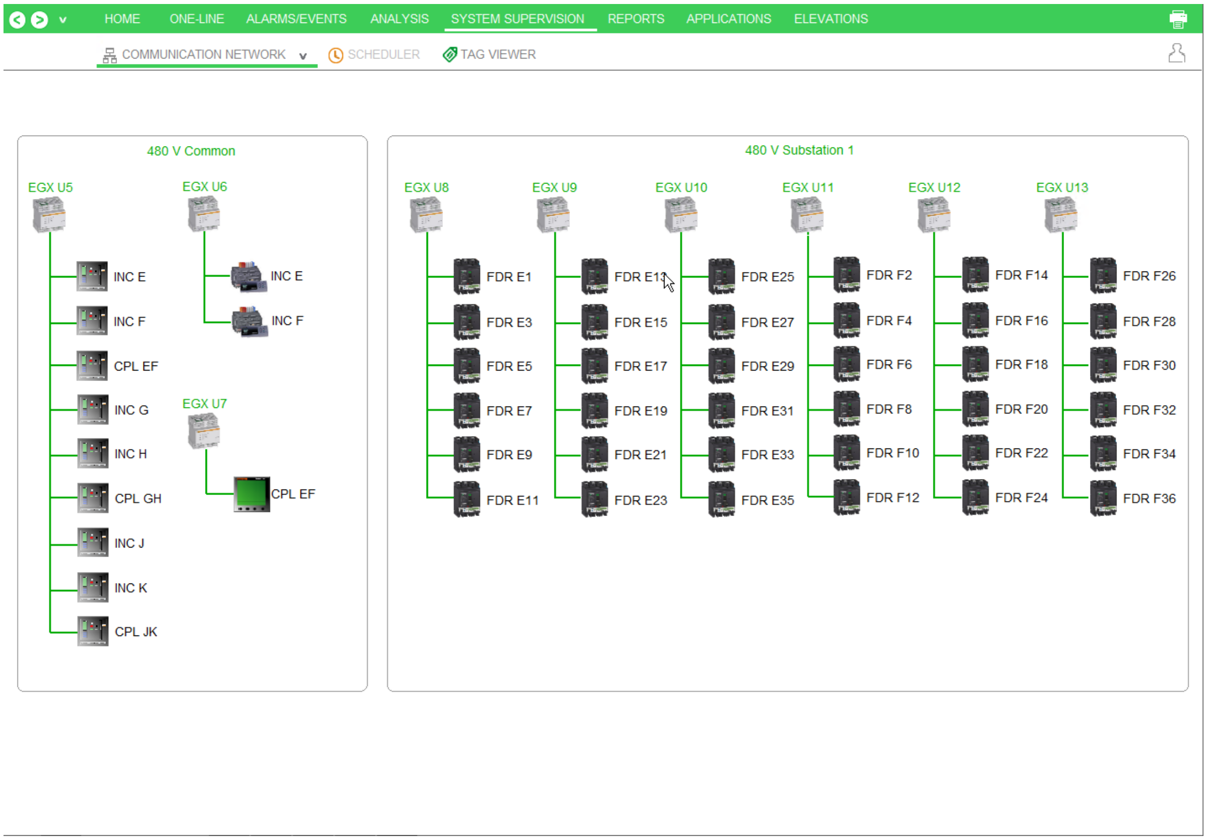
- Creating a Graphics page in Graphics Builder
- Legacy animated one-line diagrams

For more information on creating graphics pages in the Graphics Editor for viewing in the Diagrams web application, see the following sections:

- Diagrams configuration
  - Graphics pages
  - Animated one-line diagrams

Creating a Graphics page in Graphics Builder

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 940 for detailed information on the Power SCADA Operation genies.

This section includes the following topics:

- "Create a graphics page using a template" on page 1227
- "Add a new graphics page" on page 1228
- "Add custom images to graphics pages" on page 1229
- "Change the page background color" on page 1230
- "Change the genie color in project pages" on page 1230

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 313 for more information about changing the appearance of the graphics page.

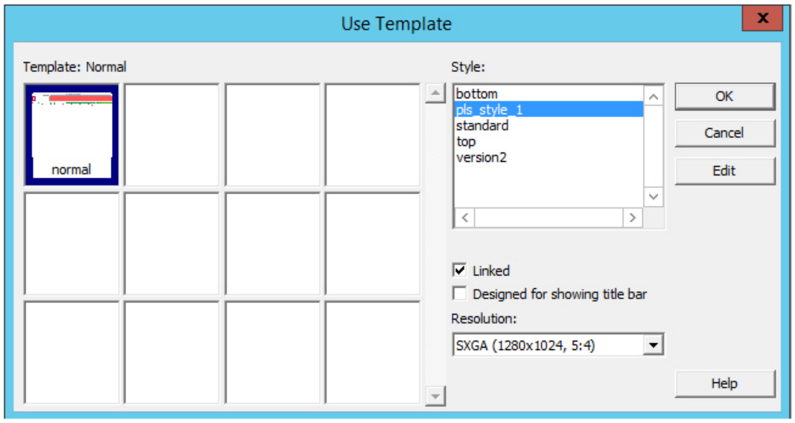
When creating a graphics page, these are the steps you will usually follow:

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:



Use Template

Template: Normal

normal

Style:

bottom
pls_style_1
standard
top
version2

☑ Linked

☐ Designed for showing title bar

Resolution:

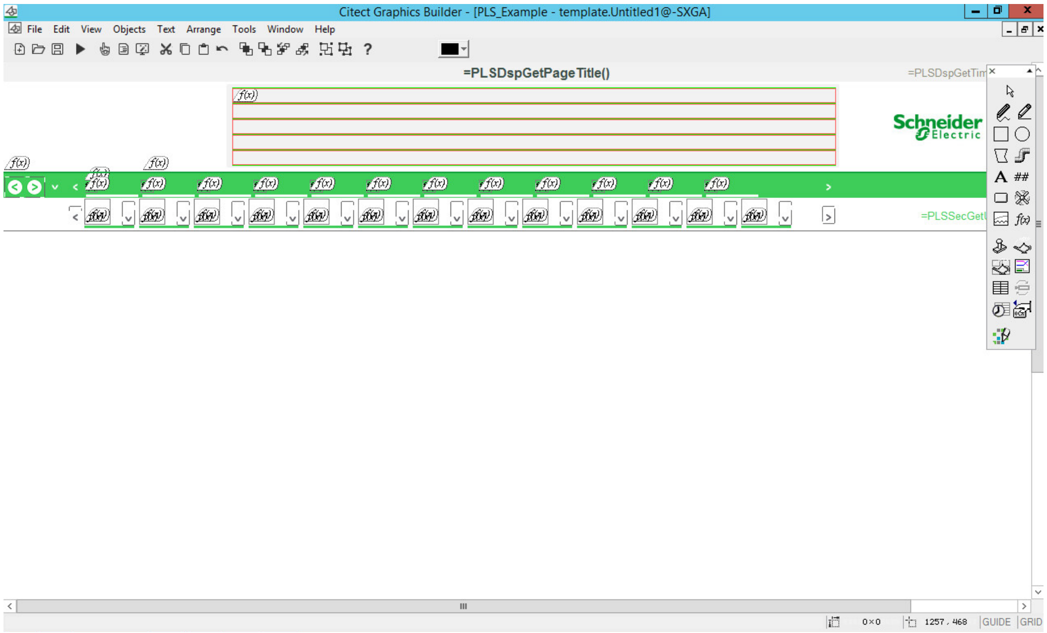
SXGA (1280x1024, 5:4)

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**.



Citect Graphics Builder - [PLS_Example - template.Untitled1@-SXGA]

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 1231 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.

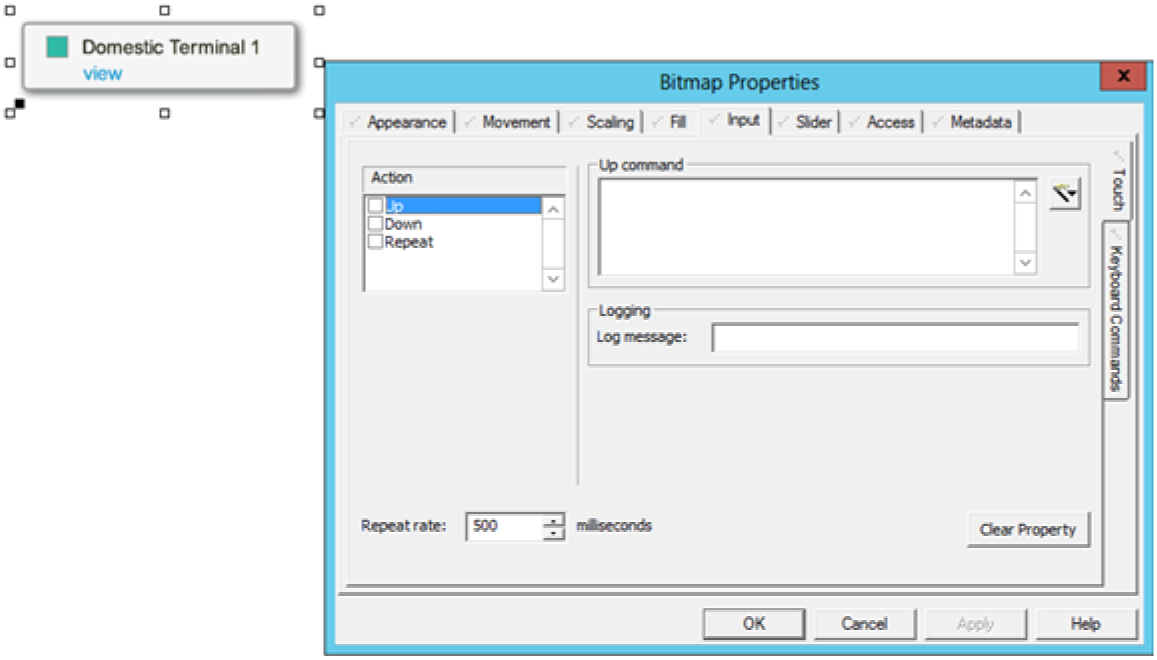
NOTE: Power SCADA Operation supports several image file types, however, bitmaps (BMP) tend to display better than other image file types.

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.



Repeat rate: 500 milliseconds

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 "Change the genie color in project pages" on page 1230.

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.

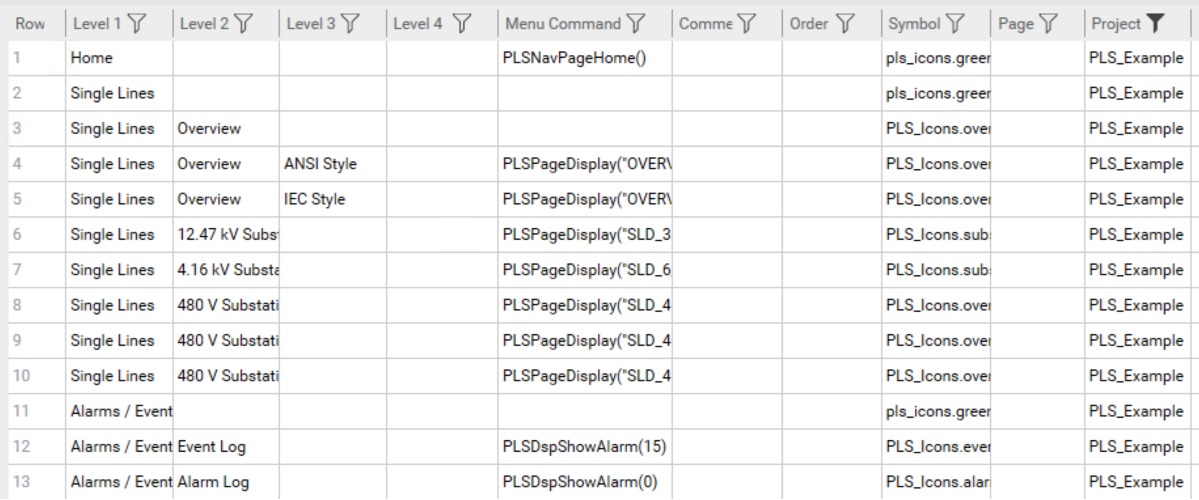
NOTE: If you use the "high contrast" starter project to create your project, this parameter is automatically added, as are all of the pages that you will use. For instructions on changing the background color of your project pages, see "Change the page background color" on page 1230.

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).



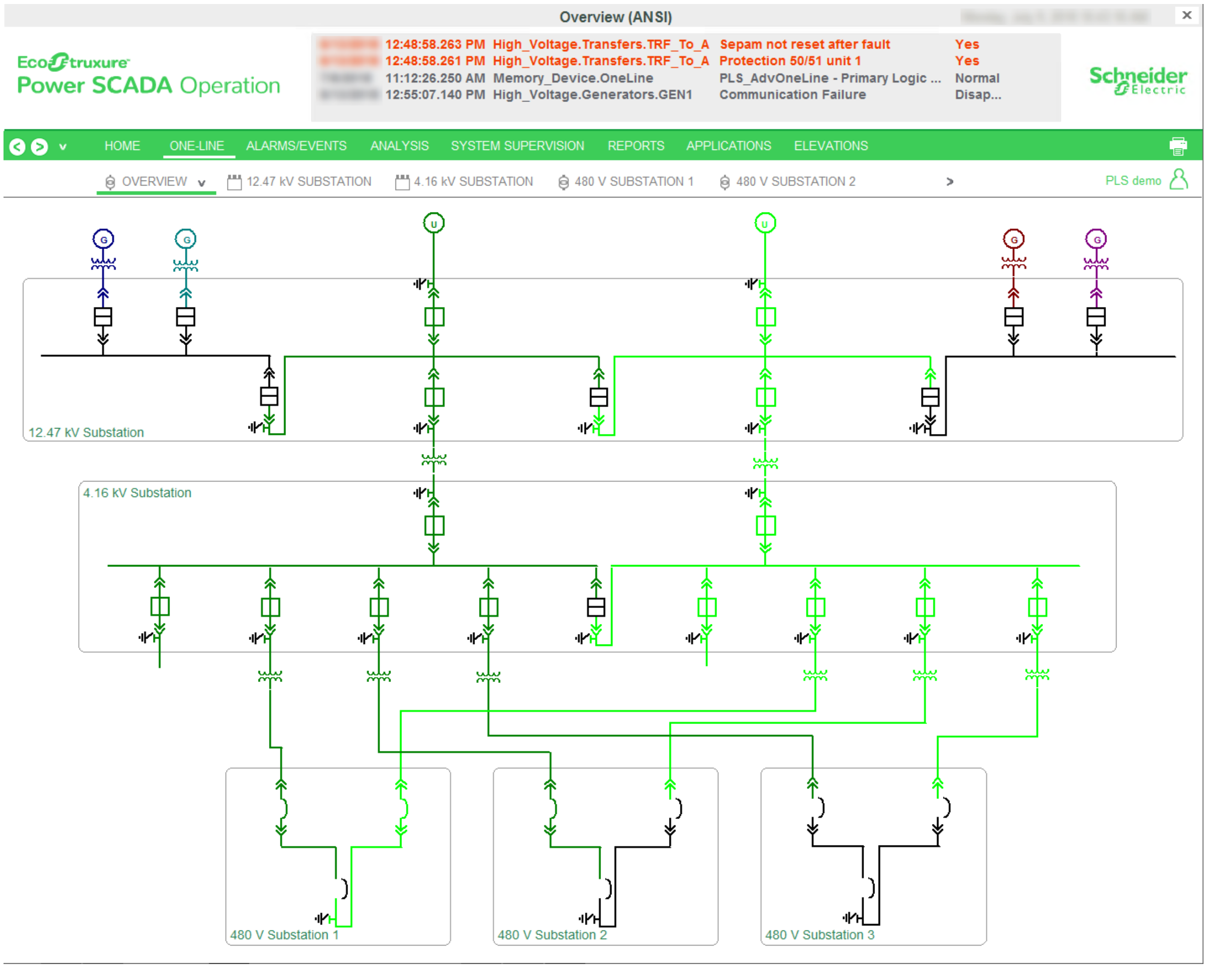
Row	Level 1	Level 2	Level 3	Level 4	Menu Command	Comme	Order	Symbol	Page	Project
1	Home				PLSNavPageHome()			pls_icons.greer		PLS_Example
2	Single Lines							pls_icons.greer		PLS_Example
3	Single Lines	Overview						PLS_Icons.ovei		PLS_Example
4	Single Lines	Overview	ANSI Style		PLSPageDisplay("OVERV			PLS_Icons.ovei		PLS_Example
5	Single Lines	Overview	IEC Style		PLSPageDisplay("OVERV			PLS_Icons.ovei		PLS_Example
6	Single Lines	12.47 kV Subs			PLSPageDisplay("SLD_3			PLS_Icons.sub		PLS_Example
7	Single Lines	4.16 kV Subst			PLSPageDisplay("SLD_6			PLS_Icons.sub		PLS_Example
8	Single Lines	480 V Substati			PLSPageDisplay("SLD_4			PLS_Icons.ovei		PLS_Example
9	Single Lines	480 V Substati			PLSPageDisplay("SLD_4			PLS_Icons.ovei		PLS_Example
10	Single Lines	480 V Substati			PLSPageDisplay("SLD_4			PLS_Icons.ovei		PLS_Example
11	Alarms / Event							pls_icons.greer		PLS_Example
12	Alarms / Event	Event Log			PLSDspShowAlarm(15)			PLS_Icons.evei		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.

Menu Item	Description
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).

Legacy 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:



Overview (ANSI) [redacted]

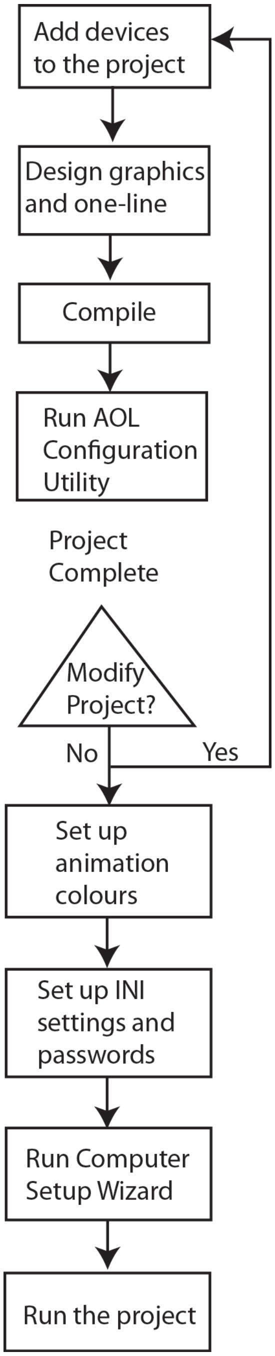
[redacted] 12:48:58.263 PM High_Voltage.Transfers.TRF_To_A Sepam not reset after fault Yes
[redacted] 12:48:58.261 PM High_Voltage.Transfers.TRF_To_A Protection 50/51 unit 1 Yes
[redacted] 11:12:26.250 AM Memory_Device.OneLine PLS_AdvOneLine - Primary Logic ... Normal
[redacted] 12:55:07.140 PM High_Voltage.Generators.GEN1 Communication Failure Disap...

This section includes the following topics:

- "One-line prerequisites " on page 1235
- "Creating a one-line on a graphics page" on page 1241
- "Create a new genie" on page 1246
- "Reviewing Genie Configurations" on page 1253
- "Repair one-line diagrams" on page 1251

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.


Add devices to the project
Design graphics and one-line
Compile
Run AOL Configuration Utility
Project Complete
Modify Project?
No
Yes
Set up animation colours
Set up INI settings and passwords
Run Computer Setup Wizard
Run the project

For detailed information on one-line diagrams see:

Running the Advanced One-Line Configuration Utility:

- "Reviewing Genie Configurations" on page 1253
- "Repair one-line diagrams" on page 1251

Setting Up Animation Colors:

- "Assign One-Line Colors" on page 1251

INI Settings and Passwords:

- "Add INI settings to AdvOneLine.ini.txt and Citect.ini" on page 1238

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 1238).

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 1253 for information about using this tab.

On the **Color Configuration tab**, you can assign colors to sources. see "Assign One-Line Colors" on page 1251 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 198
- "One-line memory device (zOL)" on page 1235
- "One-Line Engine configuration" on page 1236
- "Add INI settings to AdvOneLine.ini.txt and Citect.ini" on page 1238
- "Start and stop one-lines" on page 1241

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 1238. 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 1236): 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 \v2020 R2\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 1235.

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.

Creating 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\v2020 R2\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 1246 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 1243
   - "Configure a Source" on page 1243
   - "Configure a Circuit Breaker or Switch" on page 1244
   - "Configure a Busbar" on page 1244
   - "Configure a Transformer" on page 1245
   - "Configure an Automatic Transfer Switch (ATS)" on page 1245

> **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 1231 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.

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 1256 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 1257 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 1258 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 1259 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 1260 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 1246

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 940.

> **NOTE**: If you modify a genie, the modifications will affect 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.
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 2020. 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 1257
- "Busbar Information" on page 1258
- "Meter Information" on page 1255

- "Source information" on page 1256
- "Transformer Information" on page 1260

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.

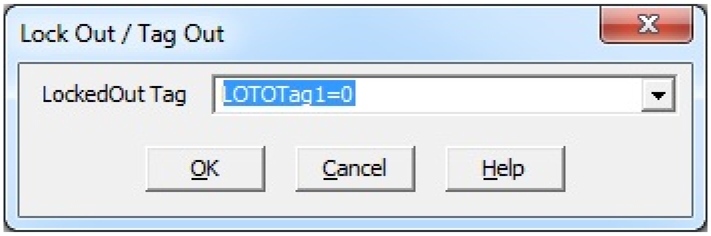
This is a read-only feature; but it does not prevent controls to the device or area. This feature is not available in PLS_Example.

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:



LockedOut Tag LOTOTag1=0

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 1245.

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 814

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 1253.
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 2020 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
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.

Field	Description
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 2020. 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 1257
- "Busbar Information" on page 1258
- "Meter Information" on page 1255
- "Source information" on page 1256
- "Transformer Information" on page 1260

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.
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. A source genie is required and must be properly configured.

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.
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.

Column	Description
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.

Operate references

The topics in this section contain detailed reference information about operating Power SCADA Operation.

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

Topic	70% width
"Alarms references" on page 1262	Detailed information on operating Alarms.
"Diagrams references" on page 1286	Detailed information on operating Diagrams.

Alarms references

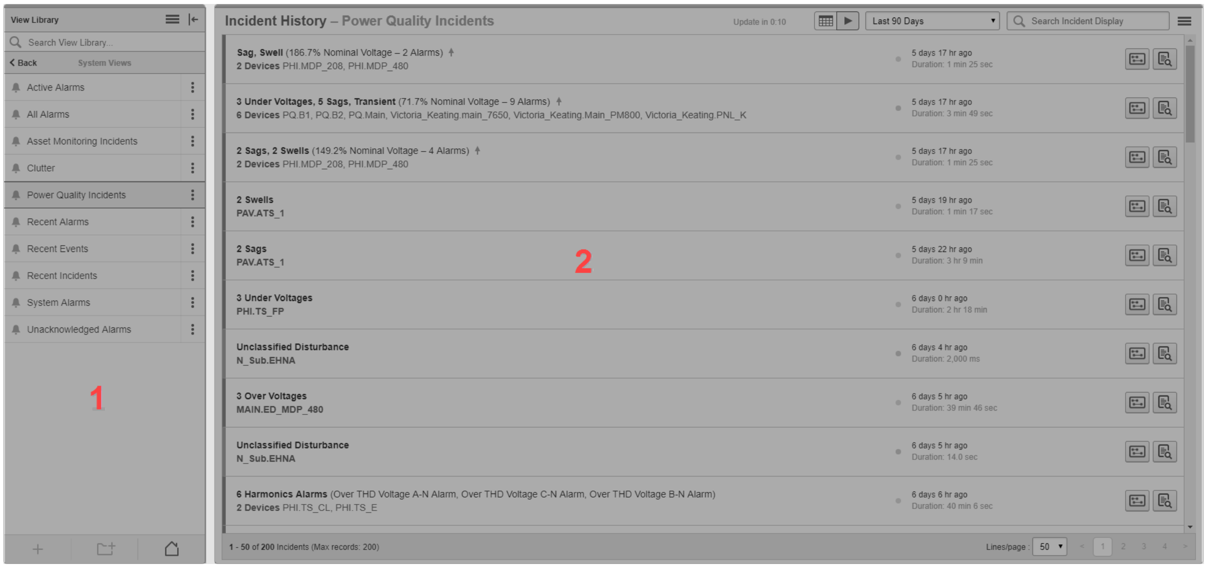
This section contains reference information related to using Alarms.

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

Alarms UI
Timeline Analysis UI
Waveforms UI
Alarm to Incident Mapping
Alarms terminology

Alarms UI

1 Main UI



Incident History – Power Quality Incidents

Sag, Swell (186.7% Nominal Voltage – 2 Alarms)
2 Devices PHI.MDP_208, PHI.MDP_480
5 days 17 hr ago
Duration: 1 min 25 sec

3 Under Voltages, 5 Sags, Transient (71.7% Nominal Voltage – 9 Alarms)
6 Devices PQ.B1, PQ.B2, PQ.Main, Victoria_Keating.main_7650, Victoria_Keating.Main_PM800, Victoria_Keating.PNL_K
5 days 17 hr ago
Duration: 3 min 49 sec

2 Sags, 2 Swells (149.2% Nominal Voltage – 4 Alarms)
2 Devices PHI.MDP_208, PHI.MDP_480
5 days 17 hr ago
Duration: 1 min 25 sec

2 Swells
PAV.ATS_1
5 days 19 hr ago
Duration: 1 min 17 sec

2 Sags
PAV.ATS_1
5 days 22 hr ago
Duration: 3 hr 9 min

3 Under Voltages
PHI.TS_FP
6 days 0 hr ago
Duration: 2 hr 18 min

Unclassified Disturbance
N_Sub.EHNA
6 days 4 hr ago
Duration: 2,000 ms

3 Over Voltages
MAIN.ED_MDP_480
6 days 5 hr ago
Duration: 39 min 46 sec

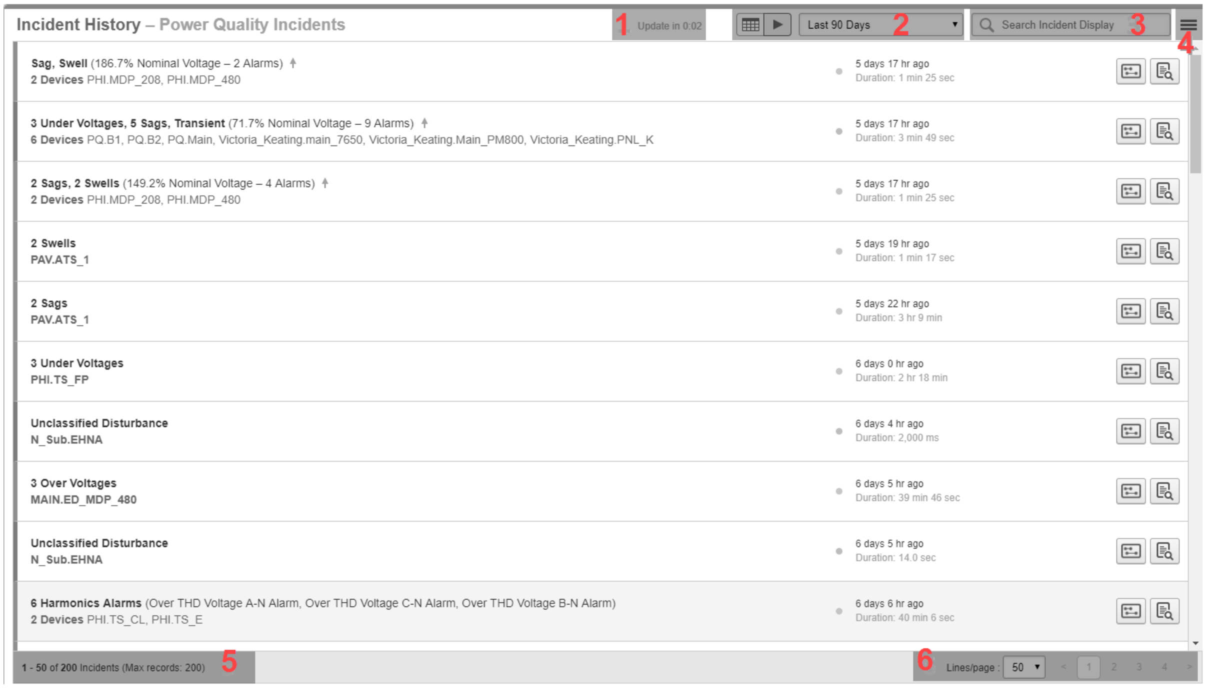
Unclassified Disturbance
N_Sub.EHNA
6 days 5 hr ago
Duration: 14.0 sec

6 Harmonics Alarms (Over THD Voltage A-N Alarm, Over THD Voltage C-N Alarm, Over THD Voltage B-N Alarm)
2 Devices PHI.TS_CL, PHI.TS_E
6 days 6 hr ago
Duration: 40 min 6 sec

1 - 50 of 200 Incidents (Max records: 200)

1	**View Library** The view library contains all the alarm views that are configured in the system. Alarm views can be listed individually or they can be organized within folders. **TIP**: To hide the library, click the Hide Library icon (|← or →|) in the top right corner of the library. To show the library, click the Show Library icon (→▯ or ▯←) at the top of the library ribbon, or click anywhere in the minimized library ribbon.
2	**Alarms Display** The alarms display pane shows the alarm view selected in the view library.

2 Alarms display UI



Incident History – Power Quality Incidents

Sag, Swell (186.7% Nominal Voltage – 2 Alarms)
2 Devices PHI.MDP_208, PHI.MDP_480
5 days 17 hr ago
Duration: 1 min 25 sec

3 Under Voltages, 5 Sags, Transient (71.7% Nominal Voltage – 9 Alarms)
6 Devices PQ.B1, PQ.B2, PQ.Main, Victoria_Keating.main_7650, Victoria_Keating.Main_PM800, Victoria_Keating.PNL_K
5 days 17 hr ago
Duration: 3 min 49 sec

2 Sags, 2 Swells (149.2% Nominal Voltage – 4 Alarms)
2 Devices PHI.MDP_208, PHI.MDP_480
5 days 17 hr ago
Duration: 1 min 25 sec

2 Swells
PAV.ATS_1
5 days 19 hr ago
Duration: 1 min 17 sec

2 Sags
PAV.ATS_1
5 days 22 hr ago
Duration: 3 hr 9 min

3 Under Voltages
PHI.TS_FP
6 days 0 hr ago
Duration: 2 hr 18 min

Unclassified Disturbance
N_Sub.EHNA
6 days 4 hr ago
Duration: 2,000 ms

3 Over Voltages
MAIN.ED_MDP_480
6 days 5 hr ago
Duration: 39 min 46 sec

Unclassified Disturbance
N_Sub.EHNA
6 days 5 hr ago
Duration: 14.0 sec

6 Harmonics Alarms (Over THD Voltage A-N Alarm, Over THD Voltage C-N Alarm, Over THD Voltage B-N Alarm)
2 Devices PHI.TS_CL, PHI.TS_E
6 days 6 hr ago
Duration: 40 min 6 sec

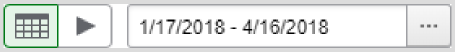
1 - 50 of 200 Incidents (Max records: 200)

1 **Update timer**

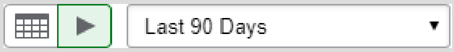
The update timer shows the time until the next display refresh.

2 **Update mode**

Use the update mode to switch between Date Filter mode and Auto-Update mode.



Date Filter mode: View alarms within a certain date range.



Auto-Update mode: View the latest alarms.

NOTE: This element is only available for history views, not for status views.

3 **Search filter**

Enter text into the search filter to search and filter the items displayed in the alarms display pane.

4 **Options menu**

The Options menu contains options relevant to the content displayed in the alarms display pane.

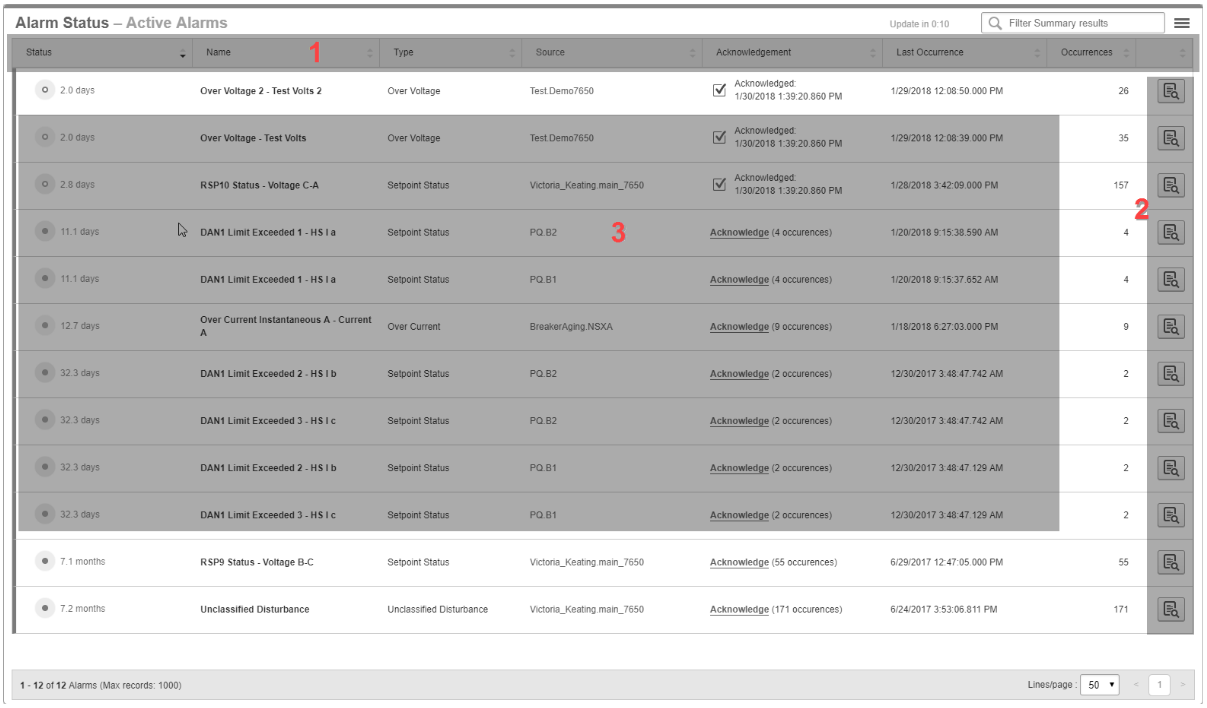
5 **Number of displayed items**

Shows the number of items visible on this page, and the total number in this view.

6 **Page selector**

Use the page selector to navigate between pages. Set the number of items that are displayed on a page.

3 Alarm status UI



Alarm Status – Active Alarms

Update in 0:10 Filter Summary results

Status	Name	Type	Source	Acknowledgement	Last Occurrence	Occurrences	
○ 2.0 days	**Over Voltage 2 - Test Volts 2**	Over Voltage	Test.Demo7650	☑ Acknowledged: 1/30/2018 1:39:20.860 PM	1/29/2018 12:08:50.000 PM	26	
○ 2.0 days	**Over Voltage - Test Volts**	Over Voltage	Test.Demo7650	☑ Acknowledged: 1/30/2018 1:39:20.860 PM	1/29/2018 12:08:39.000 PM	35	
○ 2.8 days	**RSP10 Status - Voltage C-A**	Setpoint Status	Victoria_Keating.main_7650	☑ Acknowledged: 1/30/2018 1:39:20.860 PM	1/28/2018 3:42:09.000 PM	157	
● 11.1 days	**DAN1 Limit Exceeded 1 - HS I a**	Setpoint Status	PQ.B2	Acknowledge (4 occurences)	1/20/2018 9:15:38.590 AM	4	
● 11.1 days	**DAN1 Limit Exceeded 1 - HS I a**	Setpoint Status	PQ.B1	Acknowledge (4 occurences)	1/20/2018 9:15:37.652 AM	4	
● 12.7 days	**Over Current Instantaneous A - Current A**	Over Current	BreakerAging.NSXA	Acknowledge (9 occurences)	1/18/2018 6:27:03.000 PM	9	
● 32.3 days	**DAN1 Limit Exceeded 2 - HS I b**	Setpoint Status	PQ.B2	Acknowledge (2 occurences)	12/30/2017 3:48:47.742 AM	2	
● 32.3 days	**DAN1 Limit Exceeded 3 - HS I c**	Setpoint Status	PQ.B2	Acknowledge (2 occurences)	12/30/2017 3:48:47.742 AM	2	
● 32.3 days	**DAN1 Limit Exceeded 2 - HS I b**	Setpoint Status	PQ.B1	Acknowledge (2 occurences)	12/30/2017 3:48:47.129 AM	2	
● 32.3 days	**DAN1 Limit Exceeded 3 - HS I c**	Setpoint Status	PQ.B1	Acknowledge (2 occurences)	12/30/2017 3:48:47.129 AM	2	
● 7.1 months	**RSP9 Status - Voltage B-C**	Setpoint Status	Victoria_Keating.main_7650	Acknowledge (55 occurences)	6/29/2017 12:47:05.000 PM	55	
● 7.2 months	**Unclassified Disturbance**	Unclassified Disturbance	Victoria_Keating.main_7650	Acknowledge (171 occurences)	6/24/2017 3:53:06.811 PM	171	

1 - 12 of **12** Alarms (Max records: 1000)

Lines/page : 50 ▾ < 1 >

Alarm status table columns

Click on any of the column headers to sort by that column. Use the **Show/Hide Columns** option in the alarms display pane Options menu to customize which columns are visible. The following columns are available:

	ID	Unique numeric alarm identifier.
	Priority	Alarm priority number from 0 - 255.
	State	Graphic display of active or inactive status. Also shows the amount of time since the alarm went last active.
	Active	Active or Inactive status.
1	Name	Alarm name.
	Type	Alarm type, for example Over Voltage.
	Source	Origin of the alarm.
	Unacknowledged	Number of unacknowledged alarm activations.
	Acknowledgement	A link to acknowledge the alarm.
	Last Occurrence	Datetime of latest alarm activation, in browser local time.
	Last Occurrence UTC	Datetime of latest alarm activation, in UTC time.
	First Occurrence	Datetime of first alarm activation, in browser local time.
	Occurrences	Total number of alarm activations.

2 **Details button**

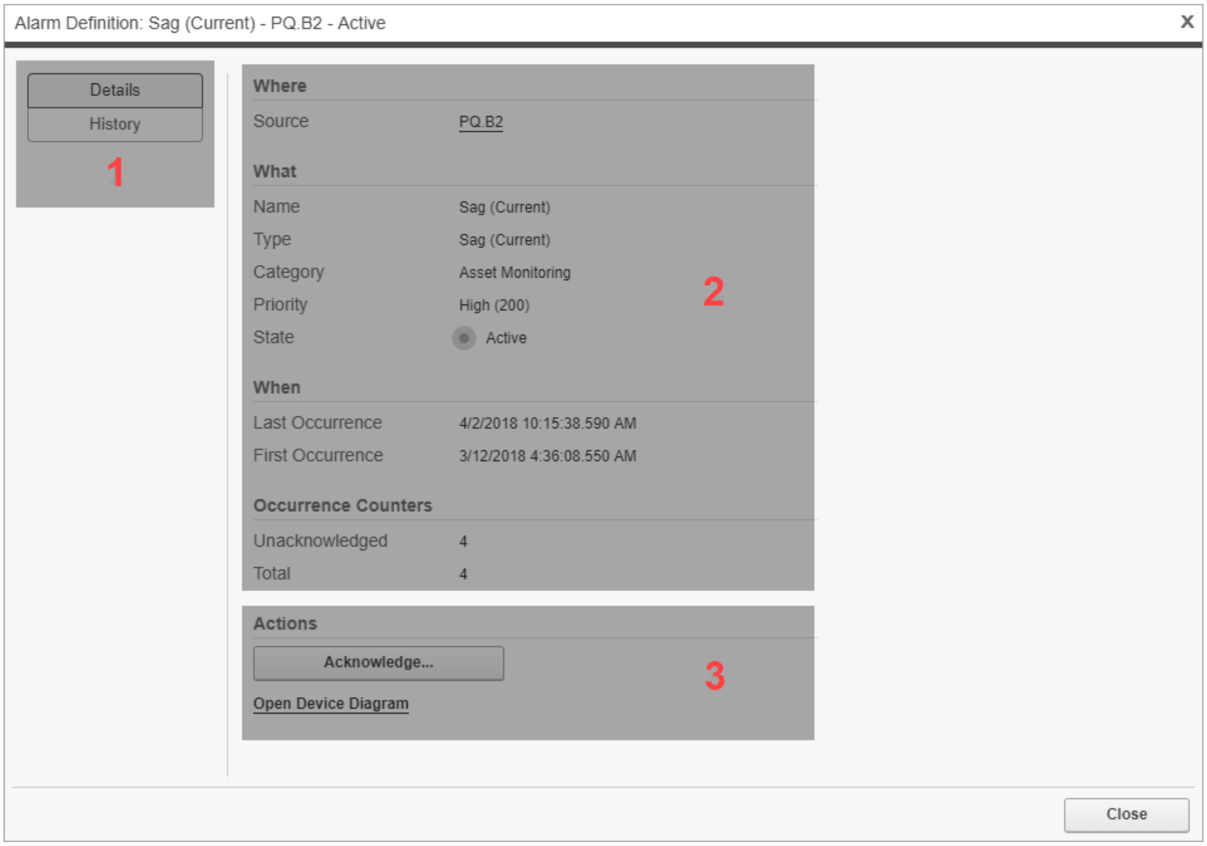
Click Details to see more information related to an alarm. (See below for more information.)

3 **Alarm status table rows**

Each row in the table shows an alarm definition that exists in the system. The filter settings in the view library control which alarm definitions are included in a view.

3-1 Alarm definition details

TIP: Click **Details** for an alarm definition or double-click an alarm definition row in the table to open the alarm details.



Alarm Definition: Sag (Current) - PQ.B2 - Active

Where

Source PQ.B2

What

Name Sag (Current)

Type Sag (Current)

Category Asset Monitoring

Priority High (200)

State ● Active

When

Last Occurrence 4/2/2018 10:15:38.590 AM

First Occurrence 3/12/2018 4:36:08.550 AM

Occurrence Counters

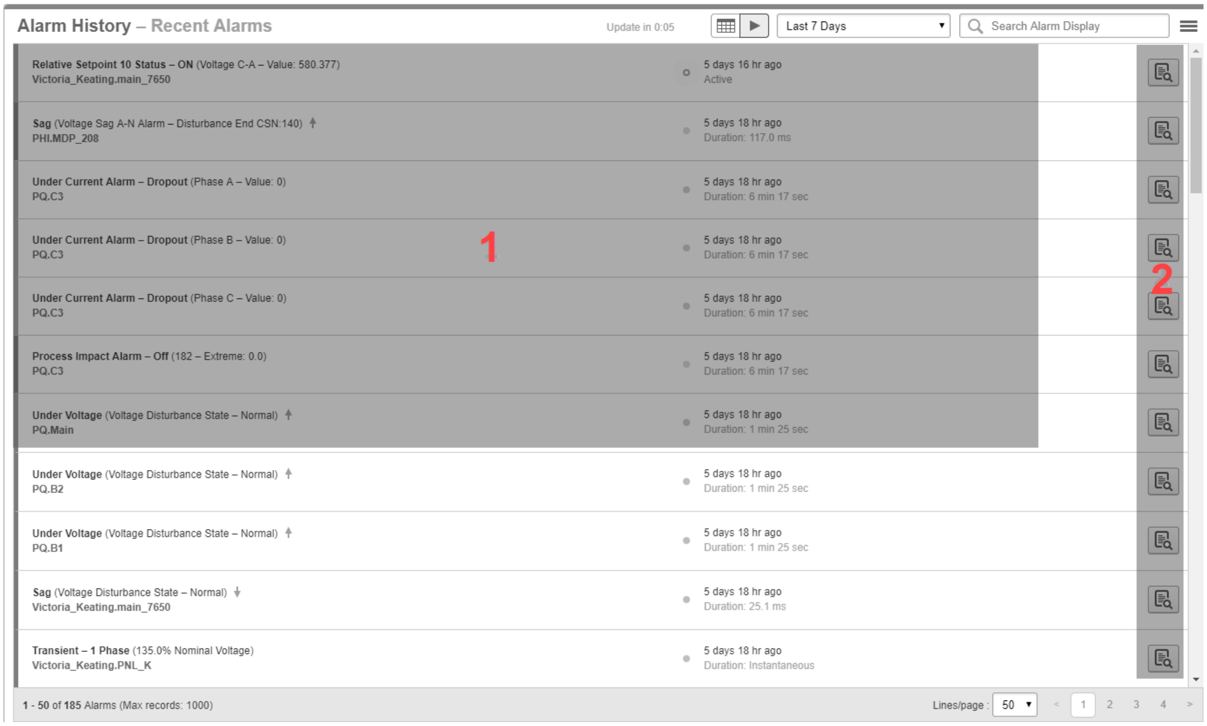
Unacknowledged 4

Total 4

Actions

1	**Display selector** Select Details to see information about the alarm definition. Select History to see past instances of this alarm.
2	**Alarm Definition Details information** See detailed information about this alarm definition.
3	**Actions** Click Acknowledge to open the Acknowledge Alarms window. Click Open Device Diagram to open the device diagram for the source this alarm is associated with.

4 Alarm history UI



Alarm History – Recent Alarms

Relative Setpoint 10 Status – ON (Voltage C-A – Value: 580.377) Victoria_Keating.main_7650	5 days 16 hr ago Active
Sag (Voltage Sag A-N Alarm – Disturbance End CSN:140) PHI.MDP_208	5 days 18 hr ago Duration: 117.0 ms
Under Current Alarm – Dropout (Phase A – Value: 0) PQ.C3	5 days 18 hr ago Duration: 6 min 17 sec
Under Current Alarm – Dropout (Phase B – Value: 0) PQ.C3	5 days 18 hr ago Duration: 6 min 17 sec
Under Current Alarm – Dropout (Phase C – Value: 0) PQ.C3	5 days 18 hr ago Duration: 6 min 17 sec
Process Impact Alarm – Off (182 – Extreme: 0.0) PQ.C3	5 days 18 hr ago Duration: 6 min 17 sec
Under Voltage (Voltage Disturbance State – Normal) PQ.Main	5 days 18 hr ago Duration: 1 min 25 sec
Under Voltage (Voltage Disturbance State – Normal) PQ.B2	5 days 18 hr ago Duration: 1 min 25 sec
Under Voltage (Voltage Disturbance State – Normal) PQ.B1	5 days 18 hr ago Duration: 1 min 25 sec
Sag (Voltage Disturbance State – Normal) Victoria_Keating.main_7650	5 days 18 hr ago Duration: 25.1 ms
Transient – 1 Phase (135.0% Nominal Voltage) Victoria_Keating.PNL_K	5 days 18 hr ago Duration: Instantaneous

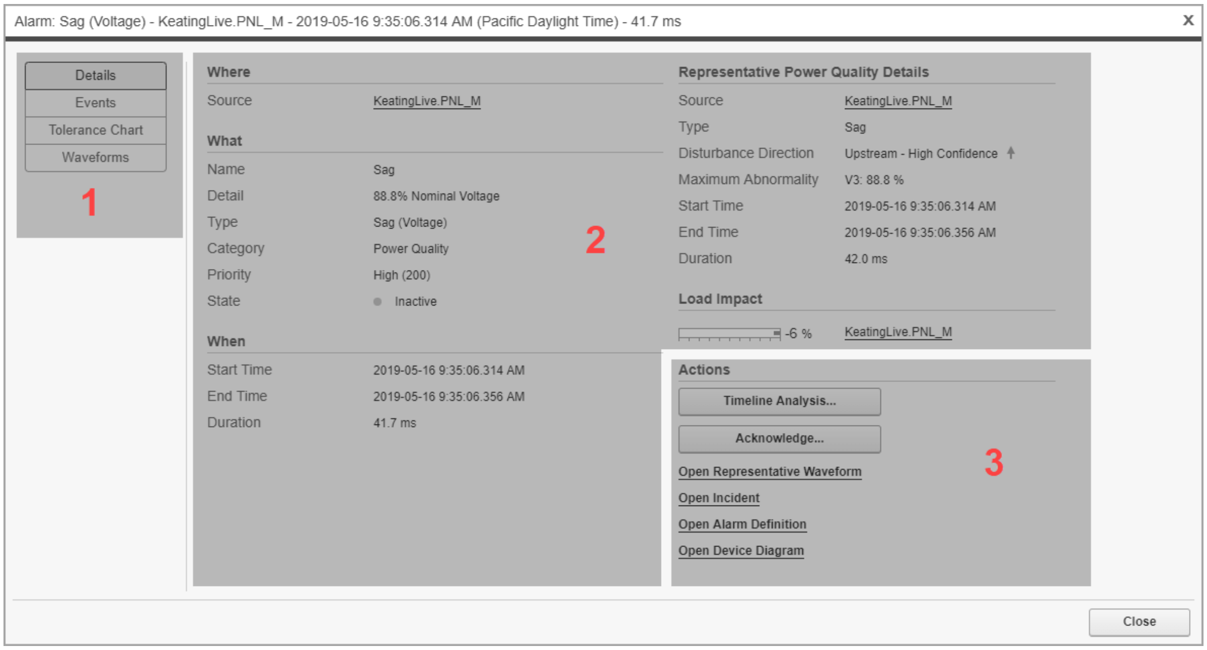
1 - 50 of **185** Alarms (Max records: 1000)

Lines/page : 50

1	**Alarm history table rows** Each row in the table shows an alarm instance that occurred. The filter settings in the view library control which instances are included in a view.
2	**Details button** Click Details to see more information related to the alarm instance. (See below for more information.)

4-1 Alarm instance details

TIP: Click Details for an alarm instance or double-click an alarm instance row in the table to open the alarm details.



Alarm: Sag (Voltage) - KeatingLive.PNL_M - 2019-05-16 9:35:06.314 AM (Pacific Daylight Time) - 41.7 ms

Where

Source KeatingLive.PNL_M

What

Name Sag
Detail 88.8% Nominal Voltage
Type Sag (Voltage)
Category Power Quality
Priority High (200)
State ● Inactive

When

Start Time 2019-05-16 9:35:06.314 AM
End Time 2019-05-16 9:35:06.356 AM
Duration 41.7 ms

Representative Power Quality Details

Source KeatingLive.PNL_M
Type Sag
Disturbance Direction Upstream - High Confidence
Maximum Abnormality V3: 88.8 %
Start Time 2019-05-16 9:35:06.314 AM
End Time 2019-05-16 9:35:06.356 AM
Duration 42.0 ms

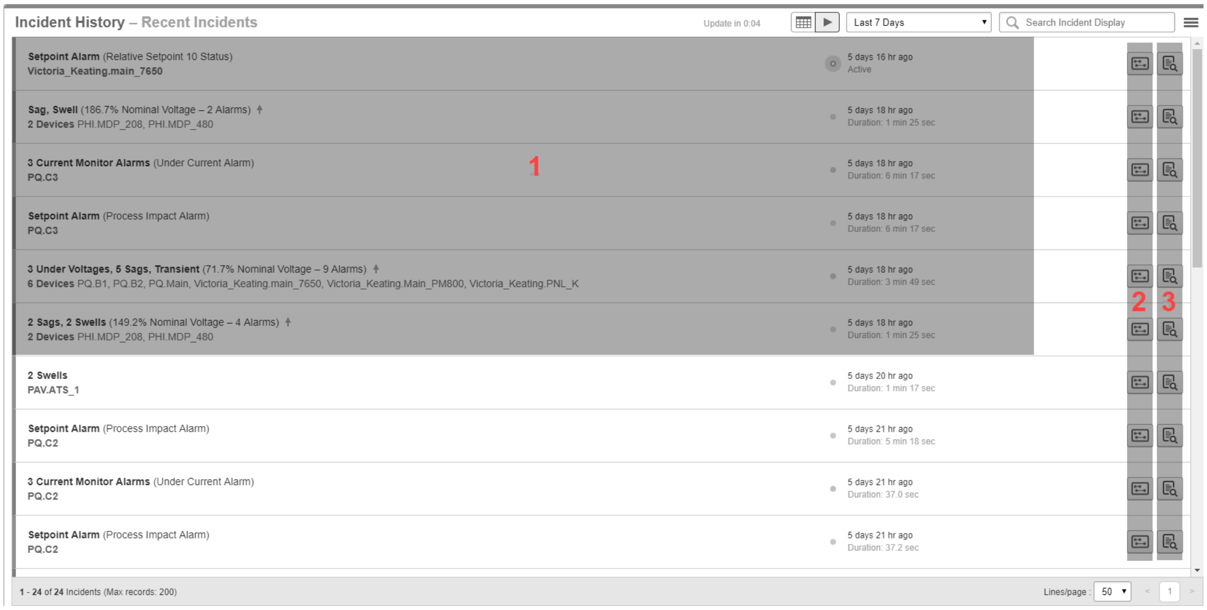
Load Impact

-6 % KeatingLive.PNL_M

Actions

1	**Display selector** Select Details to see information about this alarm instance. Select Events to see the events that are associated with this alarm instance. Select Tolerance Chart to see an ITIC/CBEMA or SEMI F47-0706 plot for the alarm instance. Note: This only applies to voltage disturbance alarms. Select Waveforms to see all the waveform that are associated with this alarm instance.
2	**Alarm instance details information** See detailed information about this alarm instance.
3	**Actions** Click Timeline Analysis to open the Timeline window. Click Acknowledge to open the Acknowledge Alarms window. Click Open Representative Waveform to see the waveform of the worst disturbance that is associated with this alarm instance. Click Open Incident to see information on the incident that is associated with this alarm instance. Click Open Alarm Definition to see information on the alarm definition for this alarm. Click Open Device Diagram to see the device diagram for the source that is associated with this alarm.

5 Incident history UI



Incident History – Recent Incidents

Update in 0:04

Last 7 Days

Search Incident Display

Setpoint Alarm (Relative Setpoint 10 Status)
Victoria_Keating.main_7650
5 days 16 hr ago
Active

Sag, Swell (186.7% Nominal Voltage – 2 Alarms)
2 Devices PHI.MDP_208, PHI.MDP_480
5 days 18 hr ago
Duration: 1 min 25 sec

3 Current Monitor Alarms (Under Current Alarm)
PQ.C3
5 days 18 hr ago
Duration: 6 min 17 sec

Setpoint Alarm (Process Impact Alarm)
PQ.C3
5 days 18 hr ago
Duration: 6 min 17 sec

3 Under Voltages, 5 Sags, Transient (71.7% Nominal Voltage – 9 Alarms)
6 Devices PQ.B1, PQ.B2, PQ.Main, Victoria_Keating.main_7650, Victoria_Keating.Main_PM800, Victoria_Keating.PNL_K
5 days 18 hr ago
Duration: 3 min 49 sec

2 Sags, 2 Swells (149.2% Nominal Voltage – 4 Alarms)
2 Devices PHI.MDP_208, PHI.MDP_480
5 days 18 hr ago
Duration: 1 min 25 sec

2 Swells
PAV.ATS_1
5 days 20 hr ago
Duration: 1 min 17 sec

Setpoint Alarm (Process Impact Alarm)
PQ.C2
5 days 21 hr ago
Duration: 5 min 18 sec

3 Current Monitor Alarms (Under Current Alarm)
PQ.C2
5 days 21 hr ago
Duration: 37.0 sec

Setpoint Alarm (Process Impact Alarm)
PQ.C2
5 days 21 hr ago
Duration: 37.2 sec

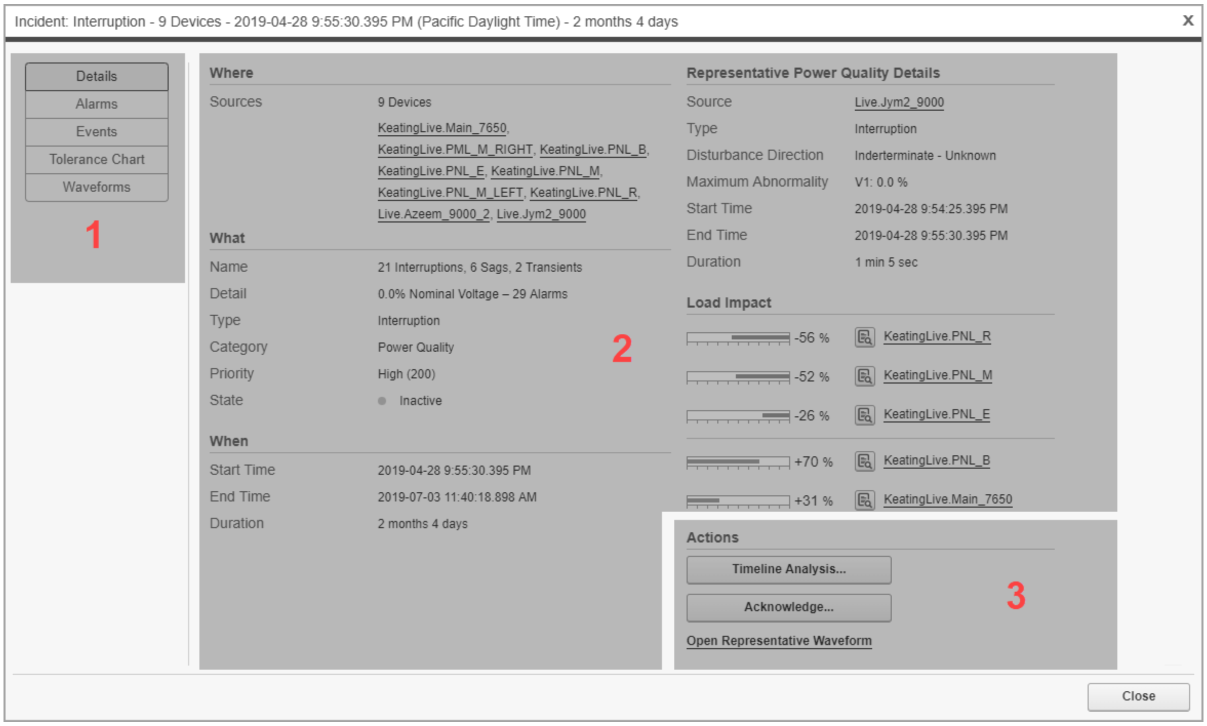
1 - **24** of **24** Incidents (Max records: 200)

Lines/page : 50

1	**Incident history table rows** Each row in the table shows an incident that occurred. The filter settings in the view library control which incidents are included in a view.
2	**Analysis button** Click Open Timeline Analysis to open the timeline analysis window for the incident. **TIP**: To analyze multiple Incidents together, select the Incidents in the table and then choose **Open Timeline Analysis on selection** from the Options menu in the top right corner of the alarms display pane. **TIP**: For multi-selection, use `Ctrl+Click` to select individual Incidents, use `Shift+click` to select a block of Incidents.
3	**Details button** Click Details to see more information related to the incident. (See below for more information.)

5-1 Incident details

TIP: Click Details for an incident or double-click an incident row in the table to open the incident details.



Incident: Interruption - 9 Devices - 2019-04-28 9:55:30.395 PM (Pacific Daylight Time) - 2 months 4 days

Where

Sources	9 Devices KeatingLive.Main_7650, KeatingLive.PML_M_RIGHT, KeatingLive.PNL_B, KeatingLive.PNL_E, KeatingLive.PNL_M, KeatingLive.PNL_M_LEFT, KeatingLive.PNL_R, Live.Azeem_9000_2, Live.Jym2_9000

What

Name	21 Interruptions, 6 Sags, 2 Transients
Detail	0.0% Nominal Voltage – 29 Alarms
Type	Interruption
Category	Power Quality
Priority	High (200)
State	● Inactive

When

Start Time	2019-04-28 9:55:30.395 PM
End Time	2019-07-03 11:40:18.898 AM
Duration	2 months 4 days

Representative Power Quality Details

Source	Live.Jym2_9000
Type	Interruption
Disturbance Direction	Inderterminate - Unknown
Maximum Abnormality	V1: 0.0 %
Start Time	2019-04-28 9:54:25.395 PM
End Time	2019-04-28 9:55:30.395 PM
Duration	1 min 5 sec

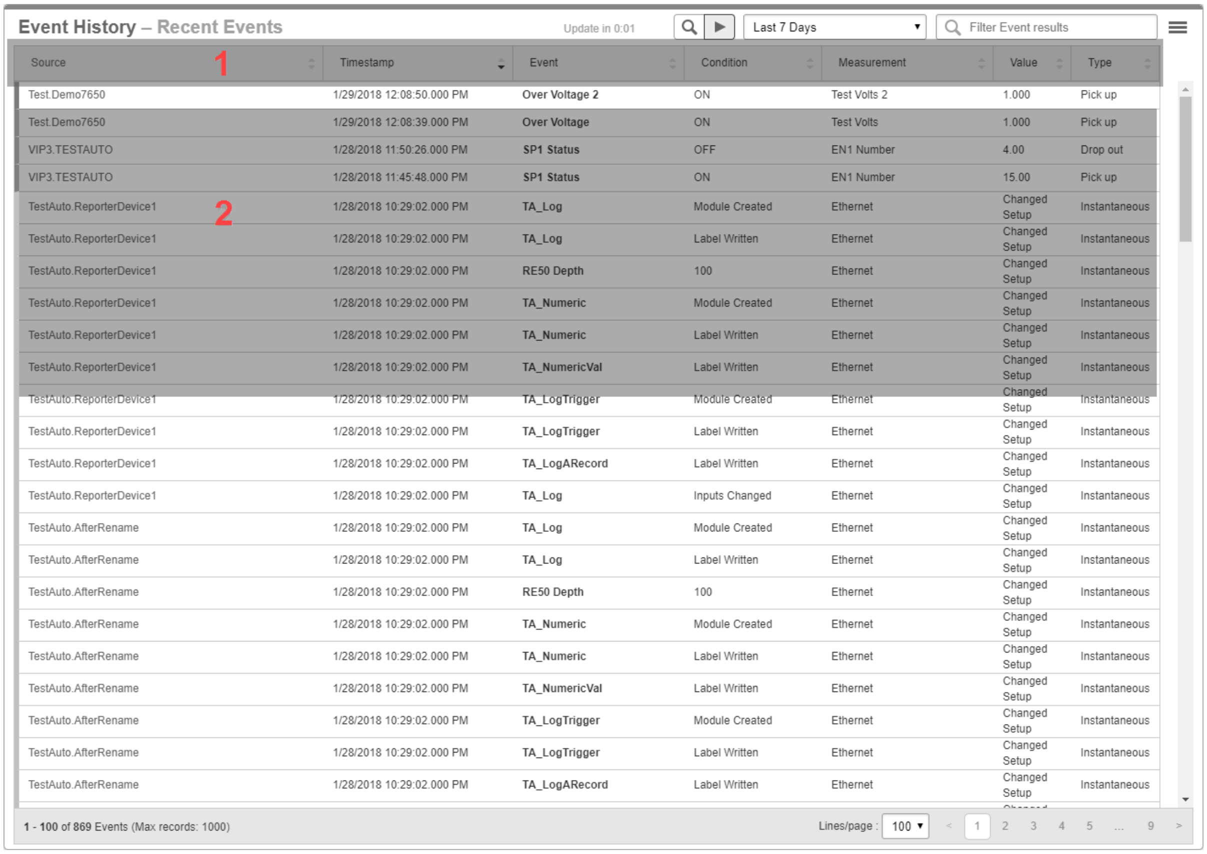
Load Impact

-56 %	KeatingLive.PNL_R
-52 %	KeatingLive.PNL_M
-26 %	KeatingLive.PNL_E
+70 %	KeatingLive.PNL_B
+31 %	KeatingLive.Main_7650

Actions

1	**Display selector** Select Details to see information about this incident. Select Alarms to see the alarm instances that are associated with this incident. Select Events to see the events that are associated with this incident. Select Tolerance Chart to see an ITIC/CBEMA or SEMI F47-0706 plot for the incident. Note: This only applies to voltage disturbances. Select Waveforms to see all the waveform that are associated with this incident.
2	**Incident Details information** See detailed information about this incident.
3	**Actions** Click Timeline Analysis to see the timeline analysis of the incident. Click Acknowledge to open the acknowledge alarms window. Click Open Representative Waveform to see the waveform of the worst disturbance that is associated with this incident.

6 Event history UI



Event History – Recent Events

Source	Timestamp	Event	Condition	Measurement	Value	Type
Test.Demo7650	1/29/2018 12:08:50.000 PM	Over Voltage 2	ON	Test Volts 2	1.000	Pick up
Test.Demo7650	1/29/2018 12:08:39.000 PM	Over Voltage	ON	Test Volts	1.000	Pick up
VIP3.TESTAUTO	1/28/2018 11:50:26.000 PM	SP1 Status	OFF	EN1 Number	4.00	Drop out
VIP3.TESTAUTO	1/28/2018 11:45:48.000 PM	SP1 Status	ON	EN1 Number	15.00	Pick up
TestAuto.ReporterDevice1	1/28/2018 10:29:02.000 PM	TA_Log	Module Created	Ethernet	Changed Setup	Instantaneous
TestAuto.ReporterDevice1	1/28/2018 10:29:02.000 PM	TA_Log	Label Written	Ethernet	Changed Setup	Instantaneous
TestAuto.ReporterDevice1	1/28/2018 10:29:02.000 PM	RE50 Depth	100	Ethernet	Changed Setup	Instantaneous
TestAuto.ReporterDevice1	1/28/2018 10:29:02.000 PM	TA_Numeric	Module Created	Ethernet	Changed Setup	Instantaneous
TestAuto.ReporterDevice1	1/28/2018 10:29:02.000 PM	TA_Numeric	Label Written	Ethernet	Changed Setup	Instantaneous
TestAuto.ReporterDevice1	1/28/2018 10:29:02.000 PM	TA_NumericVal	Label Written	Ethernet	Changed Setup	Instantaneous
TestAuto.ReporterDevice1	1/28/2018 10:29:02.000 PM	TA_LogTrigger	Module Created	Ethernet	Changed Setup	Instantaneous
TestAuto.ReporterDevice1	1/28/2018 10:29:02.000 PM	TA_LogTrigger	Label Written	Ethernet	Changed Setup	Instantaneous
TestAuto.ReporterDevice1	1/28/2018 10:29:02.000 PM	TA_LogARecord	Label Written	Ethernet	Changed Setup	Instantaneous
TestAuto.ReporterDevice1	1/28/2018 10:29:02.000 PM	TA_Log	Inputs Changed	Ethernet	Changed Setup	Instantaneous
TestAuto.AfterRename	1/28/2018 10:29:02.000 PM	TA_Log	Module Created	Ethernet	Changed Setup	Instantaneous
TestAuto.AfterRename	1/28/2018 10:29:02.000 PM	TA_Log	Label Written	Ethernet	Changed Setup	Instantaneous
TestAuto.AfterRename	1/28/2018 10:29:02.000 PM	RE50 Depth	100	Ethernet	Changed Setup	Instantaneous
TestAuto.AfterRename	1/28/2018 10:29:02.000 PM	TA_Numeric	Module Created	Ethernet	Changed Setup	Instantaneous
TestAuto.AfterRename	1/28/2018 10:29:02.000 PM	TA_Numeric	Label Written	Ethernet	Changed Setup	Instantaneous
TestAuto.AfterRename	1/28/2018 10:29:02.000 PM	TA_NumericVal	Label Written	Ethernet	Changed Setup	Instantaneous
TestAuto.AfterRename	1/28/2018 10:29:02.000 PM	TA_LogTrigger	Module Created	Ethernet	Changed Setup	Instantaneous
TestAuto.AfterRename	1/28/2018 10:29:02.000 PM	TA_LogTrigger	Label Written	Ethernet	Changed Setup	Instantaneous
TestAuto.AfterRename	1/28/2018 10:29:02.000 PM	TA_LogARecord	Label Written	Ethernet	Changed Setup	Instantaneous

1 - 100 of **869** Events (Max records: 1000)

Event history table columns

Use the Show/Hide Columns option in the alarms display pane Options menu to customize which columns are visible. The following columns are available:

	ID	Unique numeric event identifier.
	Source	Origin of the event.
	Timestamp	Datetime when the event was recorded, in browser local time.
1	Timestamp UTC	Datetime when the event was recorded, in UTC time.
	Event	Event string, for example RSP10 Status.
	Condition	Threshold value of the event trigger at the time the event was recorded.
	Measurement	Measurement that triggered the event.
	Value	Measured value at the time the event was triggered.
	Type	Event trigger type, Pick up, Drop out, or Instantaneous.
	Priority	Event priority number from 0 - 255.

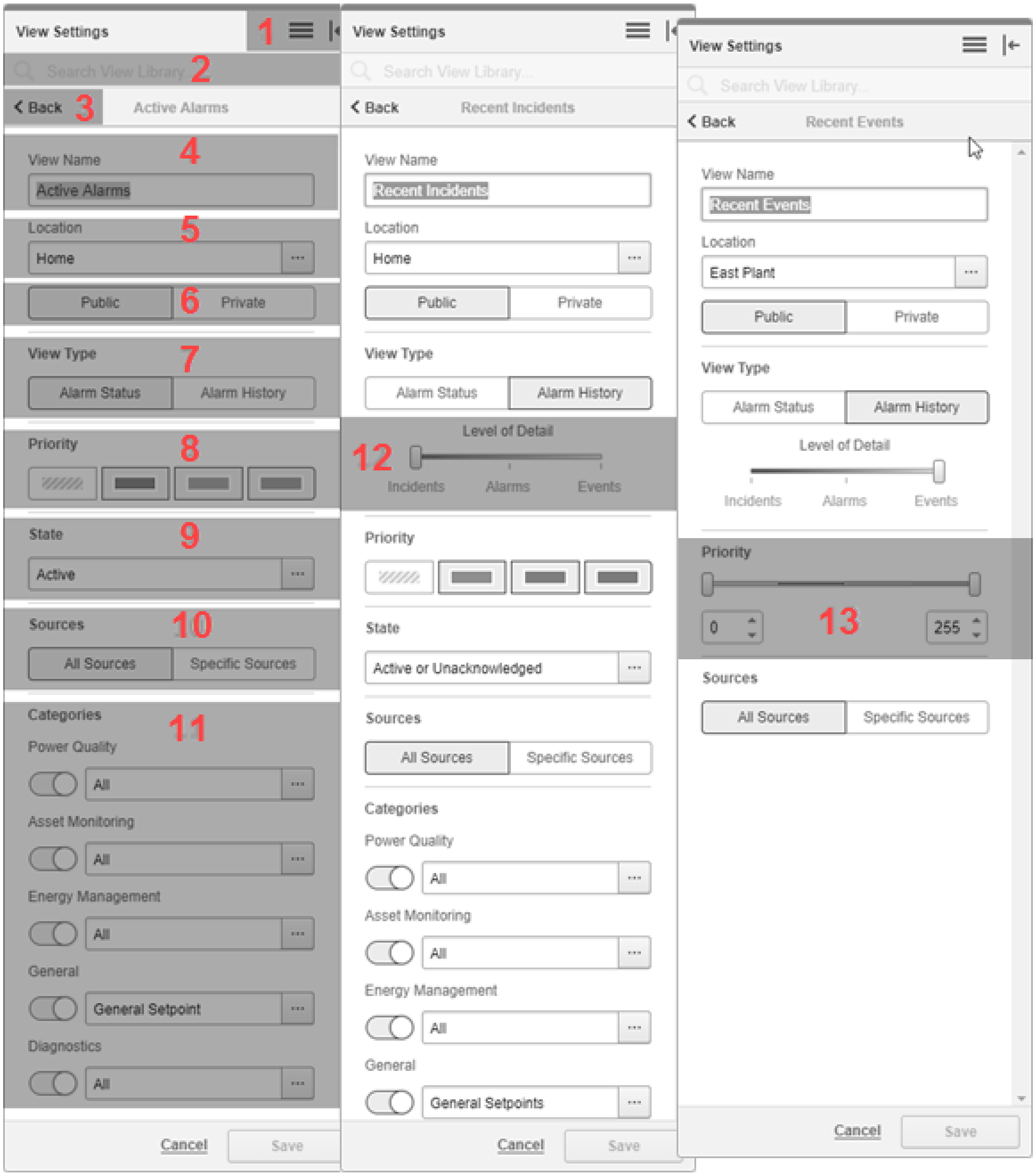
Event history table rows

2 Each row in the table shows an event that occurred. The filter settings in the view library control which events are included in a view.

TIP: Double-click an event row in the table to open the alarm instance details for the alarm that is associated with this event.

7 View settings

Alarm Status Alarm and Incident History Event History



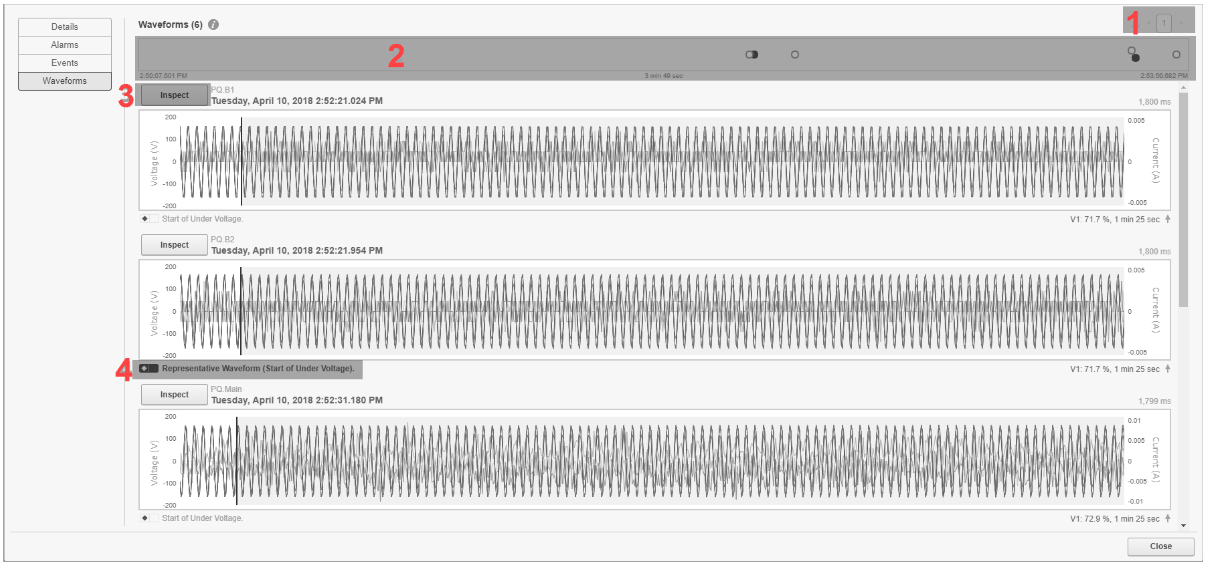
1	**Options menu ≡ and Hide Library icon ⇤.** The Options menu contains options relevant to the view library. The following options are available: Add View Add Folder
2	**Search filter** Enter text into the search filter to search and filter the views displayed in the library.

3	**Back button** Use the Back button to exit the view settings and go back to the library.
4	**View Name** Set the name of the view in the library.
5	**Location** Determine where the view is stored in the library.
6	**View Access Permissions selector** Select Public to make this view public. Select Private to make this view private. **NOTE**: A public item is visible to all users in your user group. A private item is visible to you and any user in your user group with Edit permissions on this item type. See "Assign and control user privileges" on page 686 for details.
7	**View Type selector** Select Alarm Status to create an alarm status view. Select Alarm History to create an alarm history view.
8	**Priority filter** Click the priority buttons to include or exclude alarms with that priority. The priorities are, from left to right: No, Low, Medium, High.
9	**State selector** Select which alarm states to include. The following options are available: Active or Unacknowledged Active and Unacknowledged Unacknowledged Active All
10	**Sources selector** Include all sources, or select specific sources.
11	**Category selector** Include or exclude certain categories of alarms and choose specific types within each category. The following categories are available: Power Quality (includes filter settings for Load Impact, and Disturbance Direction) Asset Monitoring Energy Management General Diagnostics See Alarms for a list of available types in each category.
12	**Level of Detail selector** Select to see incidents, alarms, or events. **NOTE**: This setting is only available for history views, not for alarm status views.

13	**Priority filter** Select which priority events to include or exclude. This filter allows more precise priority filtering than the other priority filter. **NOTE**: This selector is only available for event history views, not for alarm status or incident and alarm history views.

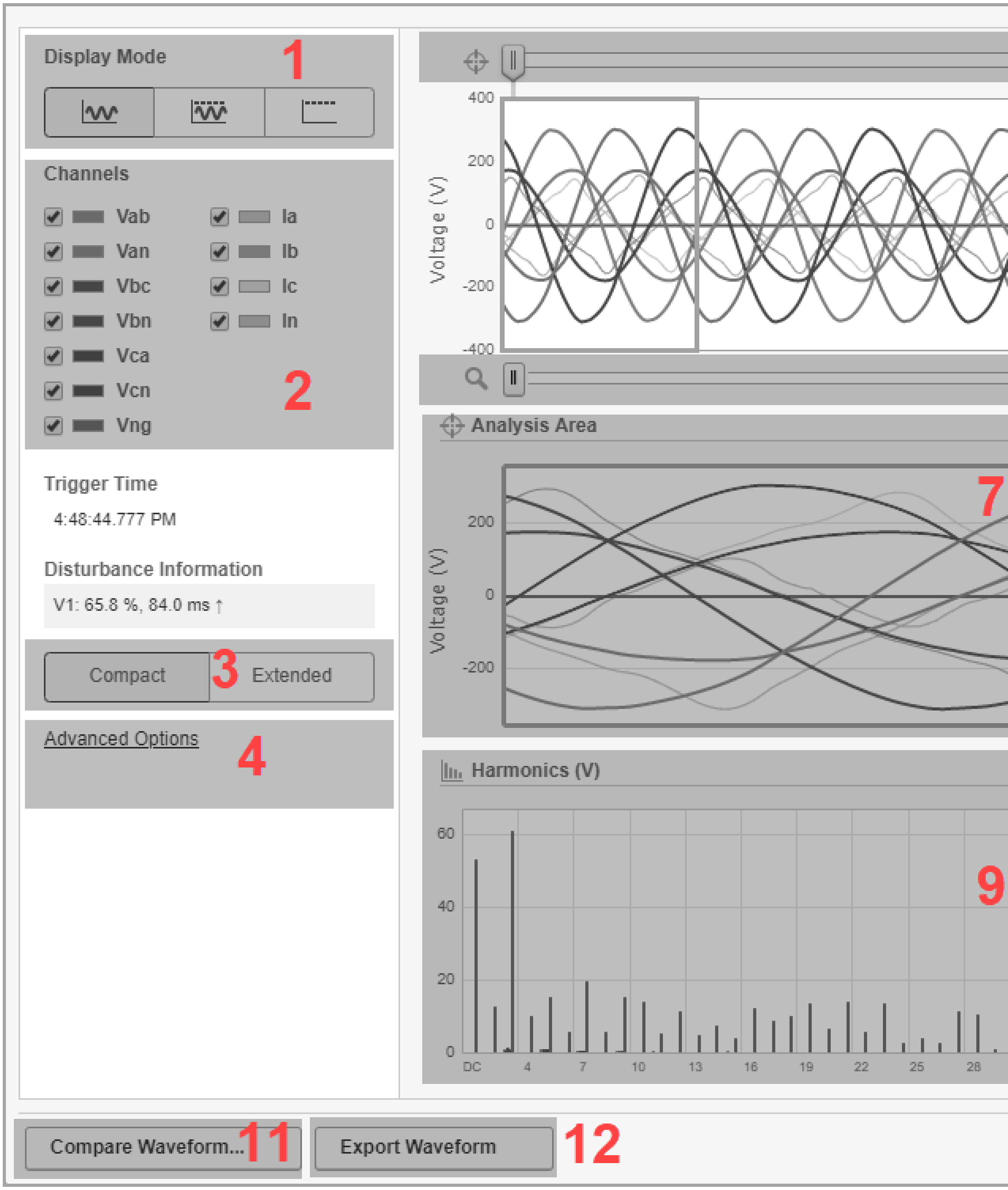
Waveforms UI

Incident and alarm instance waveforms UI



1	**Page selector.** Navigate between pages.
2	**Waveforms timeline.** The timeline shows at what point in time the waveforms that are associated with this incident or alarm instance were captured. Each waveform capture is represented by a dot. The representative waveforms for this incident or alarm instance are shown with black dots.
3	**Inspect button** Click the button to open the waveform inspection window for this waveform.
4	**Representative waveform** The black marking identifies the representative waveform for this incident or alarm instance. The representative waveform is the waveform for the worst disturbance in the incident or alarm instance.

Waveform inspection UI



Display Mode

Channels

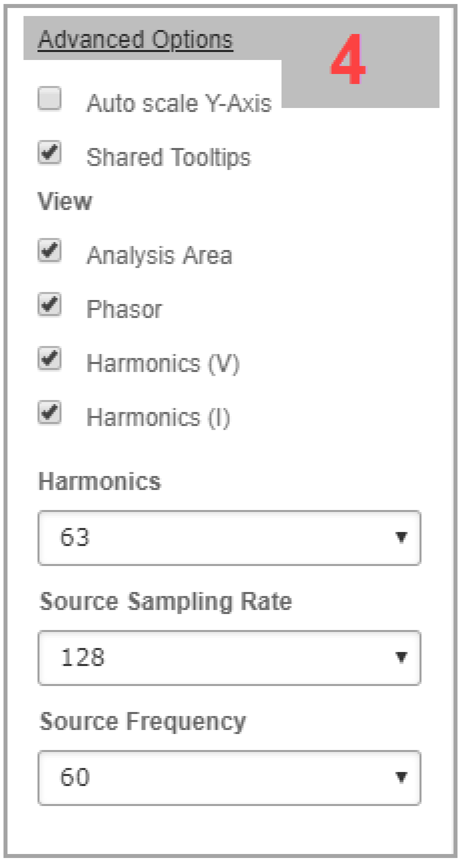
☑ Vab ☑ Ia
☑ Van ☑ Ib
☑ Vbc ☑ Ic
☑ Vbn ☑ In
☑ Vca
☑ Vcn
☑ Vng

Trigger Time

4:48:44.777 PM

Disturbance Information

V1: 65.8 %, 84.0 ms ↑



Advanced Options

☐ Auto scale Y-Axis

☑ Shared Tooltips

View

☑ Analysis Area

☑ Phasor

☑ Harmonics (V)

☑ Harmonics (I)

Harmonics: 63

Source Sampling Rate: 128

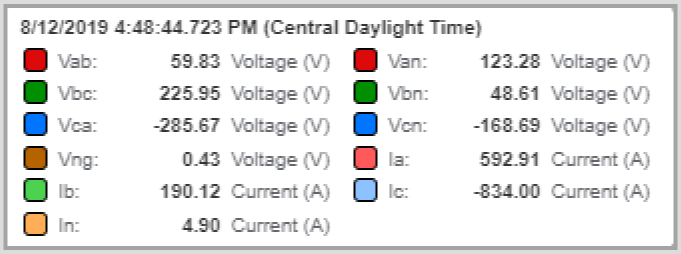
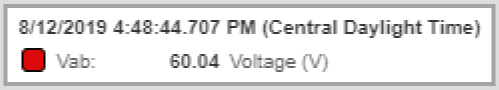
Source Frequency: 60

1	**Display Mode** Select one of the following display modes for the waveform chart: Waveform, Waveform and RMS, RMS. **NOTE**: The display mode selector is not available for high speed transient waveform captures.
2	**Channels** Select which channels (V1, V2, V3, I1, I2, I3) to include or exclude from the waveform chart.
3	**View type selector** Use the view type selector to switch between a Compact View and an Extended View. The Compact View groups the analysis charts together to fit the window size. The Extended View shows the charts below each other with a larger display area for each chart. **NOTE**: The view type selector is not available for high speed transient waveform captures.

4

Advanced Options

TIP: The Advanced Options are hidden by default. Click the Advanced Options label to show or hide these settings.

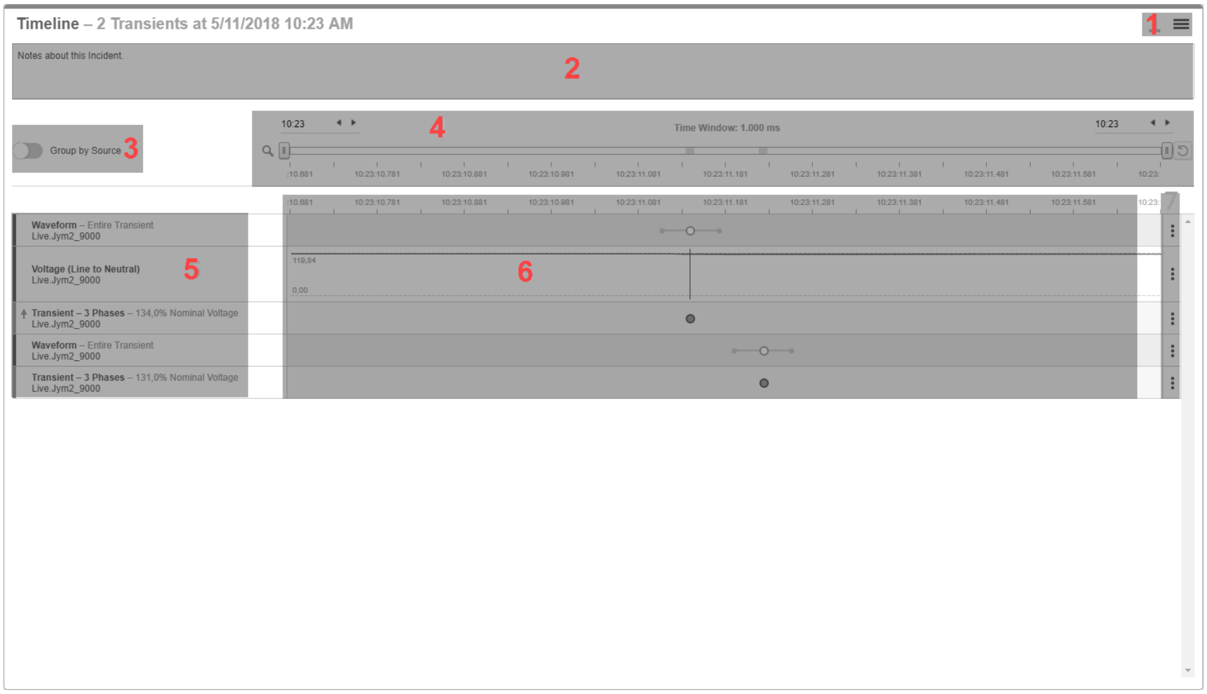
Auto scale Y-Axis	Auto scale adjusts the y-axis automatically as you zoom or pan the waveform plot.
Shared Tooltips	Shared tooltips display measurement details for all voltage and current phases as you move the pointer over the waveform plot. Non shared tooltips only display details for the voltage or current the pointer is hovering over. Example: Hover the pointer over the Vab voltage waveform plot. Shared Tooltips (shows all details)  8/12/2019 4:48:44.723 PM (Central Daylight Time) 🟥 Vab: 59.83 Voltage (V) 🟥 Van: 123.28 Voltage (V) 🟩 Vbc: 225.95 Voltage (V) 🟩 Vbn: 48.61 Voltage (V) 🟦 Vca: -285.67 Voltage (V) 🟦 Vcn: -168.69 Voltage (V) 🟫 Vng: 0.43 Voltage (V) 🟥 Ia: 592.91 Current (A) 🟩 Ib: 190.12 Current (A) 🟦 Ic: -834.00 Current (A) 🟧 In: 4.90 Current (A) Non Shared Tooltips (only shows Vab details)  8/12/2019 4:48:44.707 PM (Central Daylight Time) 🟥 Vab: 60.04 Voltage (V)
View	Select which charts (Analysis Area, Phasors, Harmonics (V), Harmonics (I)) are shown in the analysis pane. **NOTE**: The View option is not available for high speed transient waveform captures.
Harmonics	Set the number of harmonics to display in the harmonics column chart. **NOTE**: The harmonics setting is not available for high speed transient waveform captures.
Source Sampling Rate	Select the sampling rate at which the waveform was captured. The sampling rate is detected automatically. Use this control to make adjustments if the sampling rate setting is incorrect. The sampling rate is set correctly when the analysis region covers one cycle of waveform capture. **NOTE**: The sampling rate setting is not available for high speed transient waveform captures.

	Source Frequency Select the source frequency. The frequency is detected automatically. Use this control to make adjustments if the frequency setting is incorrect. **NOTE**: The frequency setting is not available for high speed transient waveform captures.
5	**Analysis area selector** Use the slider to select an analysis area in the waveform chart. **NOTE**: The analysis area selector is not available for high speed transient waveform captures.
6	**Zoom** Use the left and right sliders to zoom in and out of the waveform chart. You can also click and drag the pointer on the plot to zoom. To pan while zoomed in, click and drag the area between the sliders. Click ↺ to the right of the sliders to zoom out to the original size.
7	**Analysis area chart** This chart shows the waveform signature of the section of the waveform that has been selected by the analysis area selector (see 5). The phasor and harmonics calculations are based on the waveform data from the analysis area. The y-axis is automatically scaled. **NOTE**: The analysis area chart is not available for high speed transient waveform captures.
8	**Phasor chart** This chart shows the phasor analysis of the section of the waveform that has been selected by the analysis area selector (see 5). Phasor details are shown in a polar diagram and a data table. **NOTE**: The phasor chart is not available for high speed transient waveform captures.
9	**Voltage harmonics chart** This chart shows the voltage harmonic analysis of the section of the waveform that has been selected by the analysis area selector (see 5). Harmonic details are shown in a column chart. **NOTE**: The voltage harmonics chart is not available for high speed transient waveform captures.
10	**Current harmonics chart** This chart shows the current harmonic analysis of the section of the waveform that has been selected by the analysis area selector (see 5). Harmonic details are shown in a column chart. **NOTE**: The current harmonics chart is not available for high speed transient waveform captures.

11	**Compare Waveforms** Use this option to open this waveform in a new, Compare Waveforms tab in the browser. You can then select other waveforms to open in the same window. If a Compare Waveforms tab is already open, then the present waveform is added to that window.
12	**Export Waveform** Use this option to download the waveform data of the present waveform in .csv file format. The file is downloaded to your local Windows Downloads folder.

Timeline Analysis UI

Analysis UI

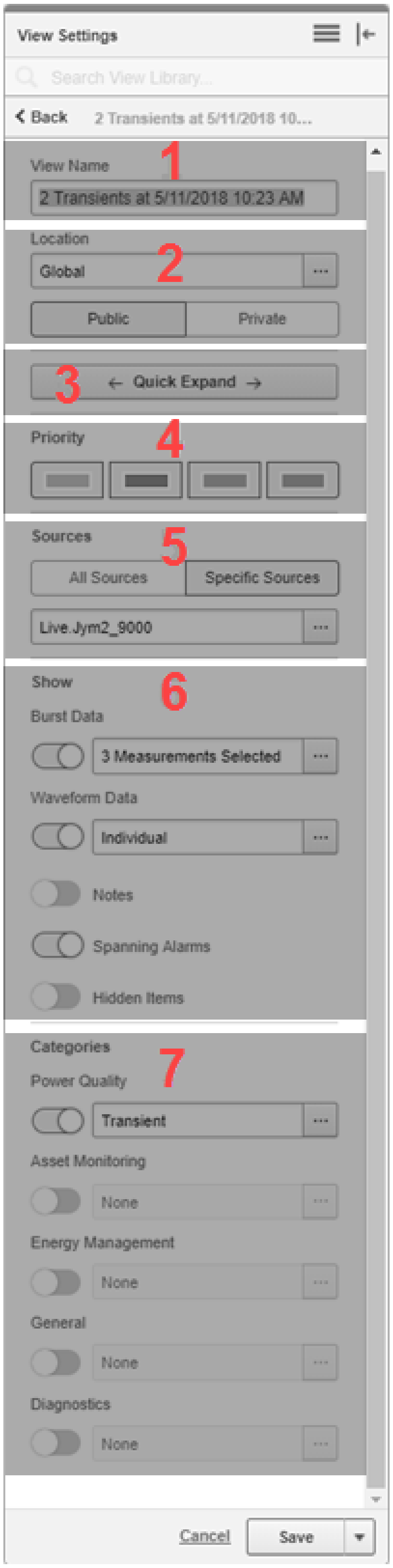


NOTE: Alarms and data measurements during an incident occur in very short time intervals. To show the correct sequence of events in the timeline analysis, the timestamps must be accurate. Consider using monitoring devices with Precision Time Protocol (PTP) or GPS time synchronization for accurate time stamping.

1	**Options menu** Contains options relevant to the content displayed in the Analysis UI.
2	**Notes area** (Optional) Enter notes related to the Analysis.
3	**Grouping control** Choose to group the items in the Analysis by time or by source.

4	**Zoom and Heatmap** Use the sliders or the time controls to zoom in or out of the analysis time window. Use the button on the right of the slider to zoom out to the original size. The colored areas act as a heatmap, showing you where the analysis items are located on the time window timeline.
5	**Analysis items** These are the alarms, waveforms, and bursts that are associated with this timeline. The color bars to the left of the items indicate the item priority. Arrows, pointing up or down, to the left of some of the items indicate Disturbance Direction Detection measurements. Hover the pointer over the arrows to get specific disturbance direction information. **TIP**: Click the item name to open a details view for the item.
6	**Timeline** Each analysis item is represented by a dot on the timeline or a burst data display. The color of the dot indicates the priority of the item. Alarms with a start and end event are shown with two dots, connected by a line. Waveforms are shown with a white dot. Zoom in to see the waveforms timeline. Click a waveform dot to open the waveform viewer.
7	**Analysis item Options** Hide an item from view or choose to open a details view for an item.

Timeline analysis view settings UI



1	**View Name** Shows the name of the timeline view.

2	**Location and sharing** Determines where the view is stored in the library and who can access it. **NOTE**: A public item is visible to all users in your user group. A private item is visible to you and any user in your user group with Edit permissions on this item type. See "Assign and control user privileges" on page 686 for details.
3	**Quick Expand** Click this option to extend the time window of the view and adds all devices, and all categories.
4	**Priority filter** Click the priority buttons to include or exclude alarms with that priority. The priorities are, from left to right: No, Low, Medium, High.
5	**Sources selector** Include all sources, or select specific sources.
6	**Show control** Show or hide burst data, waveform data, the notes area, spanning alarms, hidden items. **NOTE**: Spanning alarms are alarms that started before the time window. Hidden items are analysis items that are marked as hidden through the item Options menu. Hidden items appear dimmed when shown.
7	**Category selectors** Include or exclude certain categories of alarms from the analysis and choose specific types within each category. The following categories are available: Power Quality Asset Monitoring Energy Management General Diagnostics See Alarms for a list of available types in each category.

Alarm to Incident Mapping

The following table shows the mapping of alarm types to Incidents:

Incident Category	Incident Type	Alarm Types
Power Quality	Interruption	Interruption
	Over Voltage	Over Voltage
	Under Voltage	Under Voltage
	Unclassified Disturbance	Unclassified Disturbance
	Sag	Sag (Voltage)
	Swell	Swell (Voltage)
	Transient	Transient
	Flicker	Flicker
	Frequency Variation	Frequency Variation
	Harmonics	Harmonics
		Harmonics (Current)
		Harmonics (Power)
		Harmonics (Voltage)
	Unbalance	Unbalance
		Unbalance (Current)
		Unbalance (Voltage)
Diagnostics	Communication Status	Communication Status
	Device Status	Device Status
	System Status	System Status
Energy Management	Air	Air
	Demand	Demand
	Electricity	Electricity
	Gas	Gas
	Power Factor	Power Factor
	Steam	Steam
	Water	Water
Asset Monitoring	Arc Flash	Arc Flash
	Protection	Protection
	Backup Power	Backup Power
	Current Monitor	Over Current
		Sag (Current)
		Swell (Current)
		Under Current
	Thermal Monitor	Thermal Monitor

Incident Category	Incident Type	Alarm Types
General	Clutter	General Event
		Clock / Time
		Device Settings
		Unassociated Dropout
	General Setpoints	General Setpoint

Alarms terminology

The following is a list of commonly used terms related to Alarms in Power SCADA Operation.

Alarm

The term Alarm is commonly used to describe both, an alarm definition and an alarm instance. Which one it represents in any particular application must be derived from the context in which it is used. It is better to use the terms alarm definition and alarm instance to avoid ambiguity.

Alarm definition

An alarm definition is the specification of defined condition for a particular measurement from a particular source. When the condition is met, the alarm goes active. When the condition is no longer met, the alarm goes inactive. Example: An Overcurrent alarm that goes active when the measured current for a load goes above a defined limit. The alarm definition includes the alarm name, the source and measurement, the alarm limits, and any other conditions that are relevant for the alarm.

Alarm instance

An alarm instance is a record of an occurrence where a monitored load exceeds the limits set in the alarm definition. An alarm instance starts when the alarm state goes active and ends when it goes inactive. An alarm Instance has a start and end date.

Alarm occurrence count

The alarm occurrence count is the number of alarm instances that have happened for an alarm definition.

Alarm state

The alarm state shows if the monitored load presently meets the conditions defined in the alarm definition or not. If it meets the conditions, the alarm state is Active. If it does not meet the conditions, the alarm state is Inactive.

Alarm acknowledgment

An alarm acknowledgment is a way to indicate in the software that you have seen the alarm and that it is being managed. When you acknowledge an alarm, the date and time of the acknowledgment is recorded together with an optional note that you can enter in the acknowledge window.

An alarm can be acknowledged after it has gone active. An alarm stays unacknowledged until you acknowledge it. After you have acknowledged an alarm, it stays acknowledged until the next time it goes active. At that point it is reset to unacknowledged and is waiting for you to acknowledge it again.

NOTE: You can acknowledge alarms in status views and history views. If you acknowledge alarms through an incident history view, all alarms that are part of this Incident will be acknowledged. Whenever you acknowledge an alarm, from any of these locations, you are acknowledging the alarm definition itself, not a particular instance of it. That means acknowledging an alarm marks it as acknowledged for all instances and resets the unacknowledged occurrence counter.

Burst data

Burst data is pre- and post-event data that is logged during an alarm instance. The recording of the data is triggered by an alarm going active. Devices that support burst data keep a continuous buffer of data logs in memory, until a burst data capture is triggered. At that point, the data is recorded and uploaded to the software, together with the trigger time. By showing both pre- and post-event data, burst data is a very effective analysis tool. Burst data can be high speed data, such as half-cycle RMS measurements of voltages, currents, and other quantities, or it can be 1 second measurements for slowly changing parameters.

Event

Events are records of activity or conditions in the monitoring system. Events are generated by devices and the software and are logged and displayed as they happen in the system without any processing or aggregation. The system uses event records to determine alarm types and states.

History view

A history view in the Alarms application shows instances of incidents, alarms, or events that have occurred in the system.

Incident

An incident combines alarms, waveforms, and burst data from many sources in the system. The elements are combined based on the proximity in time when the data was recorded and based on an analysis of the type of data. The goal is to create a single representation of a real world power event that shows the impact of this event on the power system as a whole.

Representative power quality details (representative disturbance)

The representative power quality details describe the representative disturbance for an alarm or incident. The representative disturbance is used to categorize and quantify the alarm or incident. For an alarm the representative disturbance is the one that triggered the alarm. For an incident, which can include multiple alarms, the representative disturbance is the one with the highest severity in the incident. The representative power quality details include the source, type, maximum abnormality, start time, end time, and duration of the disturbance.

Example representative power quality details:

- Source: Campus.Residence Hall
- Type: Sag
- Maximum Abnormality: V3: 88.5%
- Start Time: 2019-07-26 9:08:49.330 PM
- End Time: 2019-07-26 9:08:49.530 PM
- Duration: 200.0 ms

Representative waveform

The representative waveform is the waveform that is related to the representative disturbance for an alarm or an incident. If multiple waveforms are associated with the representative disturbance, then the representative waveform is selected based on the following priorities:

1. The waveform covers full disturbance
2. The waveform covers the start of the disturbance
3. The waveform covers the end of the disturbance
4. The waveform is inside the disturbance

Status view

A status view in the Alarms application shows alarm definitions in the system, their present state, how often they occurred, their priority, and other relevant information.

Diagrams references

This section contains reference information related to using Diagrams.

Use the link below to find the content you are looking for:

- Library Components

Library Components

This section provides information on the library components, which are used for monitoring the power system.

The various library components include:

- Circuit Breaker
- Motor
- Switch
- Automatic Transfer Switch
- Lockout/Tagout
- Generator

- Transformer
- Utilities
- Busbar

There are two prevailing standards for electrical equipment:

- ANSI
- IEC

NOTE: The color codes shown in the graphic representations of bind properties for various library components depend on the Advanced One Line (AOL) configuration.

Circuit Breaker

A circuit breaker is an automatically-operated electrical switch, which is designed to protect an electrical circuit from damage that is caused by overload or short circuit. Its basic function is to interrupt the current flow when a fault is detected.

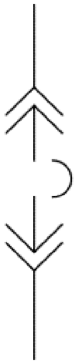
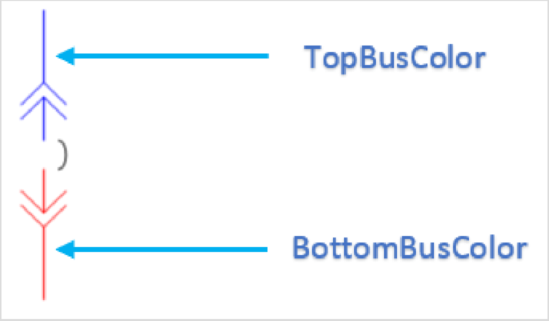
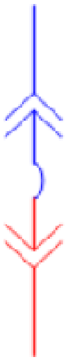
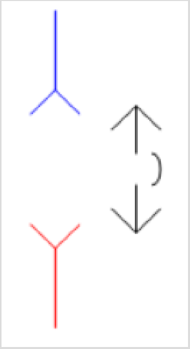
The different types of circuit breakers that are supported in the Power SCADA Operation are:

Type	Description
ANSI/IEC type	Circuit Breaker with Earth
	Circuit Breaker Without Earth
	Circuit Breaker with Control
	Circuit Breaker with No Control
	Circuit Breaker with Rackable
	Circuit Breaker with Non-rackable
	Circuit Breaker with Feeder
	Circuit Breaker with Incomer
ANSI type	Circuit Breaker with HV
IEC type	Circuit Breaker with LV

The bind properties available for circuit breakers are:

- TopBusColor
- IsTripped
- IsClosed
- RkdPos
- BottomBusColor
- EarthSwitchClosed

The graphic representation of the default position and the bind properties is explained in the below table:

Bind Property	Description	Graphic
NA	Circuit breaker is in default position.	
IsTripped	Circuit breaker is tripped due to high current flow (<range> for LV and HV)	
IsClosed	Circuit breaker is closed and there is adequate current flow.	
RkdPos	Circuit breaker is racked out of the compartment.	

Motor

A motor is an electrical device, which converts electrical energy into mechanical energy, using the principles of electricity and magnetism.

The different types of motors that are supported in the Power SCADA Operation are:

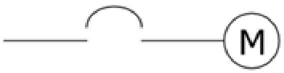
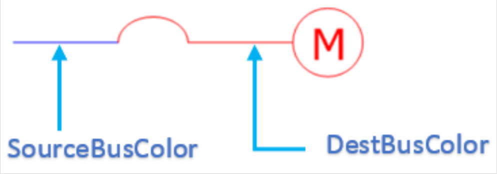
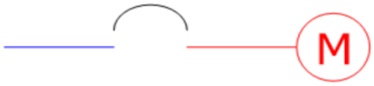
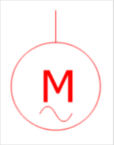
- ANSI motor
- IEC motor
- AC motor
- DC motor

The bind properties available for ANSI/IEC motors are:

- DestBusColor
- IsMotorOn
- SourceBusColor

AC/DC motors has only one bind property; BusColor.

The graphic representation of the default position and the bind property is explained in the below table:

Type	Bind Property	Description	Graphic
All	NA	Motor is in default position.	
ANSI/IEC motors	IsMotorOn	Motor is On.	
		If the motor is Off, the circuit representation looks as shown.	
AC/DC motors	BusColor	Sets the bus color.	

Switch

A switch in an electronic device, which is used to interrupt the flow of electricity or electric current.

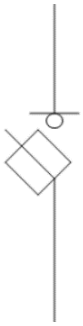
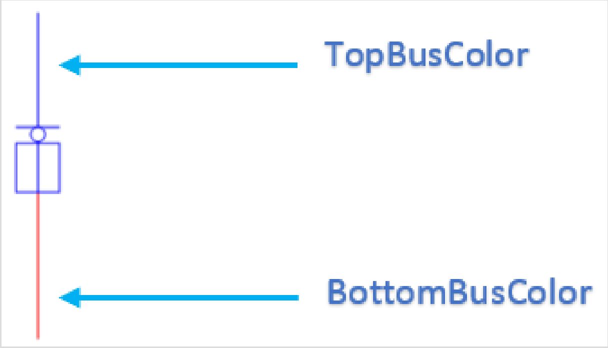
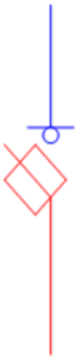
The different types of switches that are supported in the Power SCADA Operation are:

Type	Description
ANSI type	ANSI SW_General
	SW_Isolated
IEC type	SW_Fused
	SW_Fused_Isolated
	SW_General
	SW_Knife

The bind properties available for switches are:

- TopBusColor
- IsSwitchOn
- BottomBusColor

The graphic representation of the default position and the bind properties is explained in the below table:

Bind Property	Description	Graphic
NA	Switch is in default position.	
IsSwitchOn	Switch is On.	
	If the switch is Off, the circuit representation looks as shown.	

Automatic Transfer Switch

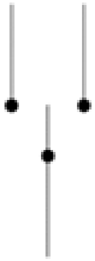
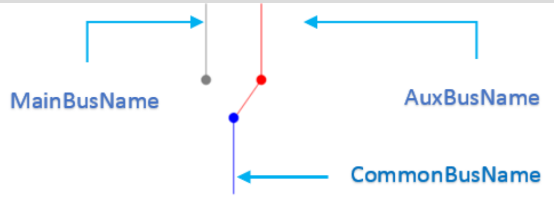
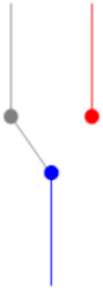
An Automatic Transfer Switch (ATS) is an electrical switch that automatically reconnects electric power source from its primary source to a standby source when it senses a failure or outage in the primary source.

The bind properties available for ATS are:

- MainBusColor
- AuxBusColor

- CommonBusColor
- ActiveConditionMain
- ActiveConditionAux

The graphic representation of the default position and the bind properties is explained in the below table:

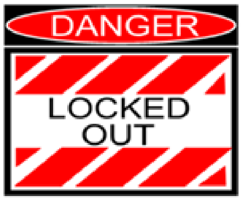
Bind Property	Description	Graphic
NA	ATS is in default position.	
ActiveConditionAux	ATS is connected to auxiliary power source.	
ActiveConditionMain	ATS is connected to main power source.	

Lockout/Tagout

Lockout/tagout refers to specific practices and procedures to safeguard employees from the unexpected energization or startup of machinery and equipment, or the release of hazardous energy during service or maintenance activities.

Lockout/Tagout has only one bind property; LockOut.

The graphic representation of the bind property is shown in the below table:

Bind Property	Graphic
LockOut	

Generator

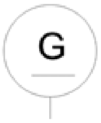
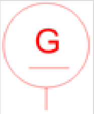
A generator is a device that converts mechanical energy into electrical energy for use in an external circuit.

The different types of generators that are supported in the Power SCADA Operation are:

- Gen_AC
- Gen_DC
- Gen_nd_1
- Gen_nd_2

Generator has only one bind property; BusColor.

The graphic representation of the default position and the bind property is shown in the below table:

Bind Property	Description	Graphic
NA	Generator is in default position.	
BusColor	Sets the bus color.	

Transformer

A transformer is a device that transfers electrical energy from one electrical circuit to another without any change of frequency through the process of electromagnetic induction.

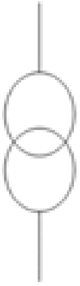
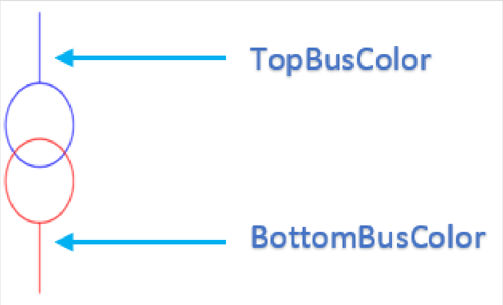
The different types of transformers that are supported in the Power SCADA Operation are:

- Trans_nd_1
- Trans_nd_2
- Trans_sd_1
- Trans_sd_2
- Trans_ss_1
- Trans_ss_2

The bind properties available for transformer are:

- TopBusColor
- BottomBusColor

The graphic representation of the default position and the bind properties is shown in the below table:

Bind Property	Description	Graphic
NA	Transformer is in default position.	
TopBusColor	Sets the bus color.	
BottomBusColor		

Utilities

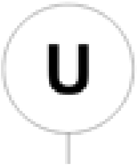
Utility is a commercial entity that owns and operates equipment and facilities for the generation, transmission, and distribution of electric energy.

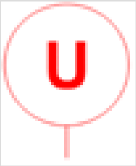
The different types of utilities that are supported in the Power SCADA Operation are:

- Util_nd_1
- Util_nd_2

Utility has only one bind property; BusColor.

The graphic representation of the default position and the bind property is shown in the below table:

Bind Property	Description	Graphic
NA	Utility is in default position.	

Bind Property	Description	Graphic
BusColor	Sets the bus color.	

Busbar

A busbar is a metallic strip or bar (typically copper, brass or aluminium) that conducts electricity within a switchboard, distribution board, substation, battery bank, or other electrical apparatus to make a common connection between several circuits in a system.

Busbar has only one bind property; BusColor.

The graphic representation of the default position and the bind property is shown in the below table:

Bind Property	Description	Graphic
NA	Busbar is in default position.	
BusColor	Sets the bus color.	

Trends references

This section contains reference information related to using Trends.

Use the following links to find the content you are looking for:

Trends UI
Trends options

Trends UI

The Trends user interface consists of a trends display pane and a Trend Library pane.

Trends display pane

The Trends display pane shows the trends selected in the **Trend Library**. When you create a trend, it automatically opens in the display pane and the trend name is selected in the **Trend Library**. You can select multiple trends to be shown simultaneously in the display pane. Scroll the display pane to view all of the trends that you selected in the **Trend Library**. For information on the options and controls available in the trend view, see Trends options.

If you log out of the application, your selections are retained and are loaded in the Trends display pane the next time you log in.

Trend Library

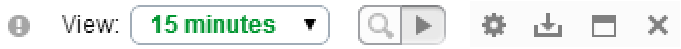
The **Trend Library** contains all the trends that are configured in the system. Trends can be listed individually or they can be organized within folders. You use the Trend Library to select the trends you want to view.

> **TIP**: To hide the library, click the Hide Library icon (|← or →|) in the top right corner of the library. To show the library, click the Show Library icon (→▯ or ▯←) at the top of the library ribbon, or click anywhere in the minimized library ribbon.

For information on how to configure Trends, see Trends configuration.

Trends options

The following options are available in the upper right area of the trend in the display pane.



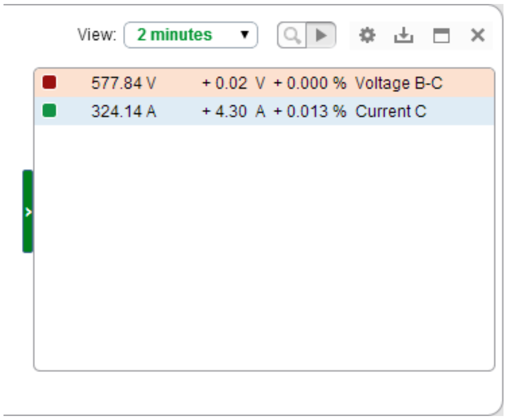
These options are summarized in the following table.

View: 15 minutes ▾	View list	The setting for the time range on the X-axis. Select a time range from the dropdown list. The view window reflects the time in minutes or hours from the last data point read from the source. For example, if you are viewing a 15 minute window and the last data point occurred 20 minutes ago, then the trend time range spans the previous 35 to 20 minutes.
🔍 ▶	Inspect	Acts as toggle to enable and disable the inspection mode for the trend. When you enable inspection mode, inspect icons appear on the trend when you place your pointer anywhere on the diagram. A slider also opens below the X-axis. Use the slider to adjust the time range for the trend. Data values are not updated in the trend but they continue to be updated in the legend. When you disable inspection mode, all data that was captured is shown.
⚙	Edit	Opens the Trend Setup dialog. You can modify any of the settings for the trend.
⤓	Download trend data as CSV	Saves the trend data that is displayed in the diagram in a CSV file on your system. When events occur, you can download the data to a CSV file for further analysis.

🗖	Maximize	Displays the trend in a full browser page. Click the Restore icon 🗗 to return to the default size in the trend display area.
✕	Close	Closes the trend. This also clears the checkbox for the trend in the **Trend Library**.

Trend legend

The legend opens on the right of the trend by default. You can select **Left** or **Off** on the **Chart** tab in the Add Trend or Trend Setup dialogs to change the location of the legend or to remove it from the trend display.



View: 2 minutes

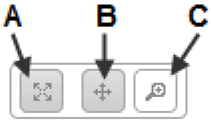
```
🟥  577.84 V   + 0.02  V  + 0.000 %  Voltage B-C
🟩  324.14 A   + 4.30  A  + 0.013 %  Current C
```

The legend provides the following capabilities:

- You can close and open the legend by clicking the arrow on the left side of the legend.
- If you have enabled multiple axes in your trend, when you place your mouse pointer over a measurement series in the legend, it indicates which axis the series is drawn on.
- You can temporarily disable a measurement series by clicking the color swatch for the series.
- The background color of a measurement series entry changes to match the threshold colors when the series passes into the upper or lower threshold. You set the threshold colors on the **Axes** tab of the Add Trend or the Trend Setup dialog.

Inspection mode

The following icons appear when you enable the inspection mode and you place your pointer on the trend.



A	Reset Zoom (100%) - resets the trend to its default size.

B	Pan the chart - after you zoom in to an area of the diagram, click **Pan the chart**, then click and hold the left mouse button on the diagram and drag it left or right.
C	Zoom in to selection area - zooms in when you drag the mouse over an area of the chart. The zoom action occurs when you release the left mouse button.

When the trend is in inspection mode, the trend remains static until you toggle inspection mode off to return the trend to its update mode. Note that the data in the legend continues to update in real time with the latest values even though the trend remains static for analysis purposes. When you toggle inspection mode off, the trend refreshes and includes all of the data that was captured while you were in inspection mode.

You can drag the slider below the X-axis to the right to decrease the time range for the trend. For example, if the time range is set to 15 minutes and you drag the slider to the right, the range values decrease, and if you continue to drag the slider to the right, the values decrease further to show minutes and seconds on the scale.

Decommission references

This section contains detailed instructions for decommissioning your system. For an overview, see "Decommission" on page 832.

Choose **Destroy** or **Overwrite** to decommission your system.

You must decommission Power SCADA on all architecture components.

NOTE: You can choose **Destroy** or **Overwrite** for each computer in your Power SCADA system, depending on your requirements. For example, you can choose to destroy your Power SCADA and Power SCADA Anywhere server(s) and overwrite your Power SCADA clients and web clients.

Decommissioning does not completely restore your computers to the state they were in before Power SCADA was installed. Decommissioning does not remove third-party software used by Power SCADA (for instance, McAfee Application Control and the .NET framework), even if this software was installed using the Power SCADA installer.

NOTE: Decommissioning will not remove Power SCADA data that has been exported or Power SCADA information in third-party software. This includes, but is not limited to:

- Data exported to Power Monitoring Expert. To decommission Power Monitoring Expert, see the Power Monitoring Expert System Guide.
- Data exported to other systems using EcoStruxure™ Web Services (EWS), OFS, DDE, ODBC, CtAPI, FTP, CSV, SQL, or any other data export method.
- Registration information shared with Schneider Electric.
- Diagnostics and Usage data sent to Schneider Electric.
- System information sent to Schneider Electric for licensing.
- Schneider Electric License Manager and Floating License Manager.
- Power SCADA information configured in third-party whitelisting software (McAfee Application Control, Sentinel System Driver).
- Files or data copied, backed-up, exported, or otherwise saved to a file location other than the Power SCADA folder.

NOTE: Decommissioning is permanent. You cannot recover, reinstall, or otherwise retrieve any part of Power SCADA after Decommissioning. If you need to keep a copy of your Power SCADA data on another computer, see Backup your current project and relevant files.

Destroy

⚠ WARNING

HAZARD OF PHYSICAL INJURY

- Do not destroy hard drives without the proper safety training.
- Never burn a hard drive, put a hard drive in a microwave, or pour acid on a hard drive.

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

NOTE: If you do not have the proper safety training, consult your IT department to select an asset disposal company.

To destroy hard drives:

1. Identify all computers where Power SCADA is installed. In a distributed architecture, this includes all Power SCADA servers, Power SCADA Anywhere servers, Power SCADA clients, Advanced Reporting and Dashboards Module computers.
2. Remove all hard drives from the computers identified in the previous step.
3. Destroy each hard drive:
   a. Puncture, shatter, or sand the hard drive plates. Follow local regulations for proper disposal of the hard drive.
   b. or, provide the hard drive to an asset disposal company.
4. Identify all Power SCADA web client computers and follow the steps below to Decommission Power SCADA Web Clients.

Overwrite

NOTICE

UNINTENDED DATA LOSS OR LOSS OF SOFTWARE FUNCTION

- Only overwrite files and folders you are certain are from Power SCADA Operation with Advanced Reporting and Dashboards.
- Back-up important files from other software before overwriting Power SCADA Operation with Advanced Reporting and Dashboards.

Failure to follow these instructions can result in irreversible damage to software and databases.

To overwrite Power SCADA Operation 2020 with Advanced Reporting and Dashboards, follow these steps:

1. Select a Data Destruction Tool
2. Decommission Power SCADA Servers
3. Decommission Power SCADA Anywhere Servers
4. Decommission Power SCADA Clients

5. Decommission Power SCADA Web Clients
6. Decommission Advanced Reporting and Dashboards Module (if installed)
7. Decommission Event Notification Module 8.x (if installed)

See the following sections for details.

1. Select a Data Destruction Tool

Select a data destruction tool. There are many commercial and open-source data destruction tools available. Consult your IT department if you are unsure about which tool to choose.

2. Decommission Power SCADA Servers

On your primary and secondary Power SCADA server computers:

Uninstall Power SCADA Software

1. Open the Windows Control Panel and select Programs and Features.
2. Uninstall Power SCADA Operation 2020.
3. Uninstall ArchestrA Data Store.
4. Uninstall Power SCADA Operation WebServer (if installed).
5. Uninstall Citect SCADA 2018 R2 Web Client (if installed).
6. Uninstall Power SCADA Operation Project DBF AddIn (if installed).
7. Uninstall Sentinel System Driver Installer (if installed).
8. Uninstall OPC Factory Server (if installed).
9. Uninstall Schneider Electric Floating License Manager (if necessary).

NOTE: The Schneider Electric Floating License Manager (FLM) may be used by other Schneider Electric software. Confirm Power SCADA Operation is the only software using the FLM before uninstalling it.

Delete ArchestrA and Citect Windows Security Groups

1. Delete the following Windows security groups, if they exist: **Asb.Deployment.***
2. Delete the following Window security groups, if they exist: **Citect.Driver.Users**

Overwrite Power SCADA Files

1. Install your data destruction tool.
2. Locate the **Power SCADA data folder** under Program Files. The default location of this folder is `..\Program Files (x86)\Schneider Electric\Power SCADA Operation\v2020 R2`.
3. Follow instructions provided with your data destruction tool to overwrite the entire **Power SCADA data folder** located in the previous step.

4. Locate the **ArchestrADataStore user folder**. The default location of this folder is `C:\Users\ArchestrADataStore`.
5. Follow instructions provided with your data destruction tool to overwrite the entire **ArchestrADataStore user folder** located in the previous step.

Overwrite Citect Data

1. Locate the **Citect user authentication file** *MachineName*.auth in your Windows user folder, where *MachineName* is the name of your computer. For example, the default location for the Windows user *Administrator* on a computer called *Standalone* is `C:\Users\Administrator\Documents\standalone.auth`.
2. Follow instructions provided with your data destruction tool to overwrite the **Citect user authentication file** located in the previous step.
3. Locate the **Citect web deployment folder**. The default location of this folder is `C:\inetput\wwwroot\Citect`.
4. Follow instructions provided with your data destruction tool to overwrite the **Citect web deployment folder** located in the previous step.

3. Decommission Power SCADA Anywhere Servers

Uninstall Power SCADA Software

On each of your Power SCADA Anywhere server computer(s):

1. Open the Windows Control Panel and select Programs and Features.
2. Uninstall Power SCADA Operation 2020 control client.
3. Uninstall Power SCADA Anywhere.

Delete Power SCADA Windows Security Groups

1. Delete the following Windows security groups, if they exist: **VJCAView** and **VJCAControl**.

4. Decommission Power SCADA Clients

On each of your Power SCADA client computer(s):

1. Open the Windows Control Panel and select Programs and Features.
2. Uninstall Power SCADA Anywhere.

5. Decommission Power SCADA Web Clients

Uninstall Vijeo Citect 2015 Web Client

On each of your web client computers:

1. Open the Windows Control Panel and select Programs and Features.
2. Uninstall Vijeo Citect 2015 Web Client.

Overwrite Citect Temporary Internet Files

On each of your web client computers:

1. Install your data destruction tool.
2. Locate the **Citect temporary Internet files folder**:
   a. Use the Windows "Run" command and enter %temp% to browse to the temporary Internet files folder.
   b. Locate the "Citect" folder in the temporary Internet files folder.
3. Follow instructions provided with your data destruction tool to overwrite the entire **Citect temporary Internet files folder** folder located in the previous step.

6. Decommission Advanced Reporting and Dashboards Module (if installed)

On each of your Advanced Reporting and Dashboards Module server computer(s):

Uninstall Advanced Reporting and Dashboards Module and ETL(PowerSCADA)

1. Open the Windows Control Panel and select Programs and Features.
2. Uninstall Advanced Reporting Module.
3. Uninstall ETL (PowerSCADA).

Overwrite Advanced Reporting and Dashboards Module Data

1. Install your data destruction tool if it is not already installed.
2. Detach ION database archives:
   a. Open **SQL Server Management Studio**, enter your password if required and click **Connect** to access your SQL Server.
   b. In the **Object Explorer** pane on the left, expand **Databases**, right-click the database archive you want to detach and click **Tasks > Detach**… to open the **Detach Database** dialog.
   c. In the **Detach Database** dialog, click **OK**.
   d. Repeat the above steps for all ION database archives.
3. Locate your Advanced Reporting Module folder under Program Files. The Advanced Reporting Module folder contains the following subfolders:
   - `\applications`
   - `\config`
   - `\Database`
   - `\Floating License Manager`
   - `\License Manager`
   - `\Setup Logs`

- `\system`
- `\web`

4. Follow instructions provided with your data destruction tool to overwrite the entire Advanced Reporting Module folder located in the previous step.
5. Locate any custom Advanced Reporting Module files in folders outside of the Advanced Reporting Module folder. This may include, but is not limited to, following file types:
   - Vista and Designer files: `.cfg, .dgm, .wsn, .wsg`
   - ION databases and archives: `.LDF, .MDF`
   - ION database backups: `.bak`
   - Custom report packs: `.rdlc`
   - Advanced Reporting Module (PME) System Key: `.key`
6. Follow instructions provided with your data destruction tool to overwrite the files located in the previous step.

Overwrite ETL (PowerSCADA) Data

1. Locate the ETL (PowerSCADA) data folder under Program Files. The default location of this folder is `..\Program Files\Schneider Electric\ETL (PowerSCADA).`
2. Follow instructions provided with your data destruction tool to overwrite the entire ETL (PowerSCADA) data folder located in the previous step.

7. Decommission Event Notification Module 8.x (if installed)

If you migrated notifications from Event Notification Module (ENM) 8.x, you must follow these steps to decommission it. For more information, see "Migrate notifications" on page 339.

NOTE: ENM 9.0 does not require these decommissioning steps.

On each of your Power SCADA server computer(s):

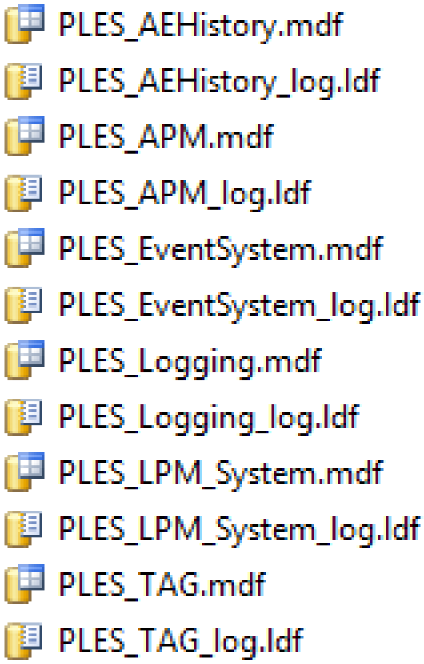
Uninstall Event Notification Module

1. Open the Windows Control Panel and select Programs and Features.
2. Uninstall Event Notification Module.

Overwrite Event Notification Module Data

1. Install your data destruction tool if it is not already installed.

2. Locate the ENM databases. By default, these are in the folder `..\Program Files\Microsoft SQL Server\MSSQL11.ION\MSSQL\DATA`. It contains these files:



PLES_AEHistory.mdf
PLES_AEHistory_log.ldf
PLES_APM.mdf
PLES_APM_log.ldf
PLES_EventSystem.mdf
PLES_EventSystem_log.ldf
PLES_Logging.mdf
PLES_Logging_log.ldf
PLES_LPM_System.mdf
PLES_LPM_System_log.ldf
PLES_TAG.mdf
PLES_TAG_log.ldf

3. Follow instructions provided with your data destruction tool to overwrite all the database files located in the previous step.

4. Locate the **ENM database backup files**. By default, these are in the folder `..\Program Files\Microsoft SQL Server\MSSQL11.ION\MSSQL\Backup`. It contains files with names similar to `MM-DD-YYYY MM-DD PLES_AEHistory.bak`.

5. Follow instructions provided with your data destruction tool to overwrite all the **ENM database backup files** located in the previous step.

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.

components

Standardized, predefined graphics for defined use.

All drawn objects are either graphics, that is, free-form drawings, or components. A component contains one or several graphic figures. It can also have predefined functionality. Components typically represent a feature or a component in a live system.

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 249 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.

snippets

Standardized, predefined functions for defined use in graphics.

Snippets typically represent a feature in a live system. Snippets are located in dedicated libraries and are displayed in the Snippets pane. Blink, which starts and stops a blink animation, is an example of a snippet.

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 587 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.se.com

As standards, specifications, and designs change from time to time, please ask for confirmation of the information given in this publication.

7EN02-0440-03 04/2022