EcoStruxure™

OPC UA Server Expert

User Guide

Original instructions

MFR53158.03
08/2025

www.se.com

Legal Information

The information provided in this document contains general descriptions, technical characteristics and/or recommendations related to products/solutions.

This document is not intended as a substitute for a detailed study or operational and site-specific development or schematic plan. It is not to be used for determining suitability or reliability of the products/solutions for specific user applications. It is the duty of any such user to perform or have any professional expert of its choice (integrator, specifier or the like) perform the appropriate and comprehensive risk analysis, evaluation and testing of the products/solutions with respect to the relevant specific application or use thereof.

The Schneider Electric brand and any trademarks of Schneider Electric SE and its subsidiaries referred to in this document are the property of Schneider Electric SE or its subsidiaries. All other brands may be trademarks of their respective owner.

Schneider Electric does not grant any right or license for commercial use of the document or its content, except for a non-exclusive and personal license to consult it on an "as is" basis.

Schneider Electric reserves the right to make changes or updates with respect to or in the content of this document or the format thereof, at any time without notice.

To the extent permitted by applicable law, no responsibility or liability is assumed by Schneider Electric and its subsidiaries for any errors or omissions in the informational content of this document, as well as any non-intended use or misuse of the content thereof.

Table of Contents

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 documentation or on the equipment to warn of potential hazards or to call attention to information that clarifies or simplifies a procedure.

The addition of this 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 and operation of electrical equipment and its installation, and has received safety training to recognize and avoid the hazards involved.

Before You Begin

Do not use this product on machinery lacking effective point-of-operation guarding. Lack of effective point-of-operation guarding on a machine can result in serious injury to the operator of that machine.

⚠WARNING

UNGUARDED EQUIPMENT

- Do not use this software and related automation equipment on equipment which does not have point-of-operation protection.
- Do not reach into machinery during operation.

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

This automation equipment and related software is used to control a variety of industrial processes. The type or model of automation equipment suitable for each application will vary depending on factors such as the control function required, degree of protection required, production methods, unusual conditions, government regulations, etc. In some applications, more than one processor may be required, as when backup redundancy is needed.

Only you, the user, machine builder or system integrator can be aware of all the conditions and factors present during setup, operation, and maintenance of the machine and, therefore, can determine the automation equipment and the related safeties and interlocks which can be properly used. When selecting automation and control equipment and related software for a particular application, you should refer to the applicable local and national standards and regulations. The National Safety Council's Accident Prevention Manual (nationally recognized in the United States of America) also provides much useful information.

In some applications, such as packaging machinery, additional operator protection such as point-of-operation guarding must be provided. This is necessary if the operator's hands and other parts of the body are free to enter the pinch points or other hazardous areas and serious injury can occur. Software products alone cannot protect an operator from injury. For this reason the software cannot be substituted for or take the place of point-of-operation protection.

Ensure that appropriate safeties and mechanical/electrical interlocks related to point-of-operation protection have been installed and are operational before placing the equipment into service. All interlocks and safeties related to point-of-operation protection must be coordinated with the related automation equipment and software programming.

NOTE: Coordination of safeties and mechanical/electrical interlocks for point-of-operation protection is outside the scope of the Function Block Library, System User Guide, or other implementation referenced in this documentation.

Start-up and Test

Before using electrical control and automation equipment for regular operation after installation, the system should be given a start-up test by qualified personnel to verify correct operation of the equipment. It is important that arrangements for such a check are made and that enough time is allowed to perform complete and satisfactory testing.

⚠WARNING

EQUIPMENT OPERATION HAZARD

- Verify that all installation and set up procedures have been completed.
- Before operational tests are performed, remove all blocks or other temporary holding means used for shipment from all component devices.
- Remove tools, meters, and debris from equipment.

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

Follow all start-up tests recommended in the equipment documentation. Store all equipment documentation for future references.

Software testing must be done in both simulated and real environments.

Verify that the completed system is free from all short circuits and temporary grounds that are not installed according to local regulations (according to the National Electrical Code in the U.S.A, for instance). If high-potential voltage testing is necessary, follow recommendations in equipment documentation to prevent accidental equipment damage.

Before energizing equipment:

- Remove tools, meters, and debris from equipment.
- Close the equipment enclosure door.
- Remove all temporary grounds from incoming power lines.
- Perform all start-up tests recommended by the manufacturer.

Operation and Adjustments

The following precautions are from the NEMA Standards Publication ICS 7.1-1995:

(In case of divergence or contradiction between any translation and the English original, the original text in the English language will prevail.)

- Regardless of the care exercised in the design and manufacture of equipment or in the selection and ratings of components, there are hazards that can be encountered if such equipment is improperly operated.
- It is sometimes possible to misadjust the equipment and thus produce unsatisfactory or unsafe operation. Always use the manufacturer’s instructions as a guide for functional adjustments. Personnel who have access to these adjustments should be familiar with the equipment manufacturer’s instructions and the machinery used with the electrical equipment.
- Only those operational adjustments required by the operator should be accessible to the operator. Access to other controls should be restricted to prevent unauthorized changes in operating characteristics.

About the Document

Document Scope

This manual describes the features and use of EcoStruxure OPC UA Server Expert software.

Validity Note

This document has been updated for the release of EcoStruxure OPC UA Server Expert 2.01 SP3.

Product Related Information

⚠WARNING

LOSS OF CONTROL

- Perform a Failure Mode and Effects Analysis (FMEA), or equivalent risk analysis, of your application, and apply preventive and detective controls before implementation.
- Provide a fallback state for undesired control events or sequences.
- Provide separate or redundant control paths wherever required.
- Supply appropriate parameters, particularly for limits.
- Review the implications of transmission delays and take actions to mitigate them.
- Review the implications of communication link interruptions and take actions to mitigate them.
- Provide independent paths for control functions (for example, emergency stop, over-limit conditions, and error conditions) according to your risk assessment, and applicable codes and regulations.
- Apply local accident prevention and safety regulations and guidelines.[1]
- Test each implementation of a system for proper operation before placing it into service.

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

[1] For additional information, refer to NEMA ICS 1.1 (latest edition), *Safety Guidelines for the Application, Installation, and Maintenance of Solid State Control* and to NEMA ICS 7.1 (latest edition), *Safety Standards for Construction and Guide for Selection, Installation and Operation of Adjustable-Speed Drive Systems* or their equivalent governing your particular location.

⚠WARNING

UNINTENDED EQUIPMENT OPERATION

- Only use software approved by Schneider Electric for use with this equipment.
- Update your application program every time you change the physical hardware configuration.

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

The examples in this document are given for information only.

⚠WARNING

UNINTENDED EQUIPMENT OPERATION

Adapt examples given herein to the specific functions and requirements of your industrial application before you implement them.

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

General Cybersecurity Information

In recent years, the growing number of networked machines and production plants has seen a corresponding increase in the potential for cyber threats, such as unauthorized access, data breaches, and operational disruptions. You must, therefore, consider all possible cybersecurity measures to help protect assets and systems against such threats.

To help keep your Schneider Electric products secure and protected, it is in your best interest to implement the cybersecurity best practices as described in the Cybersecurity Best Practices document.

Schneider Electric provides additional information and assistance:

- Subscribe to the Schneider Electric security newsletter.
- Visit the Cybersecurity Support Portal web page to:
  - Find Security Notifications.
  - Report vulnerabilities and incidents.
- Visit the Schneider Electric Cybersecurity and Data Protection Posture web page to:
  - Access the cybersecurity posture.
  - Learn more about cybersecurity in the cybersecurity academy.
  - Explore the cybersecurity services from Schneider Electric.

Product Related Cybersecurity Information

Refer to *Modicon Controller Platform Cybersecurity, User Guide* under *Related Documents*.

Related Documents

Title of documentation	Reference number
Modicon Controller Platform Cybersecurity, User Guide	EIO0000001999 (ENG) EIO0000002004 (CHS) EIO0000002001 (FRE) EIO0000002000 (GER) EIO0000002002 (ITA) EIO0000002003 (SPA)
Modicon M580, BMENUA0100 OPC UA Embedded Module, Installation and Configuration Guide	PHA83350 (ENG) PHA83351 (FRE) PHA83352 (GER) PHA83353 (ITA) PHA83354 (SPA) PHA83355 (CHS)

Title of documentation	Reference number
Modicon M580, Frequently Used Architectures, System Guide	HRB62666 (ENG) HRB65318 (FRE) HRB65319 (GER) HRB65320 (ITA) HRB65321 (SPA) HRB65322 (CHS)
Modicon M580, Complex Topologies, System Guide	NHA58892 (ENG) NHA58893 (FRE) NHA58894 (GER) NHA58895 (ITA) NHA58896 (SPA) NHA58897 (CHS)

To find documents online, visit the Schneider Electric download center (www.se.com/ww/en/download/).

Information on Non-Inclusive or Insensitive Terminology

As a responsible, inclusive company, Schneider Electric is constantly updating its communications and products that contain non-inclusive or insensitive terminology. However, despite these efforts, our content may still contain terms that are deemed inappropriate by some customers.

Terminology Derived from Standards

The technical terms, terminology, symbols and the corresponding descriptions in the information contained herein, or that appear in or on the products themselves, are generally derived from the terms or definitions of international standards.

In the area of functional safety systems, drives, and general automation, this may include, but is not limited to, terms such as *safety*, *safety function*, *safe state*, *fault*, *fault reset*, *malfunction*, *failure*, *error*, *error message*, *dangerous*, etc.

Among others, these standards include:

Standard	Description
IEC 61131-2:2007	Programmable controllers, part 2: Equipment requirements and tests.
ISO 13849-1:2023	Safety of machinery: Safety related parts of control systems. General principles for design.
EN 61496-1:2013	Safety of machinery: Electro-sensitive protective equipment. Part 1: General requirements and tests.
ISO 12100:2010	Safety of machinery - General principles for design - Risk assessment and risk reduction
EN 60204-1:2006	Safety of machinery - Electrical equipment of machines - Part 1: General requirements
ISO 14119:2013	Safety of machinery - Interlocking devices associated with guards - Principles for design and selection
ISO 13850:2015	Safety of machinery - Emergency stop - Principles for design
IEC 62061:2021	Safety of machinery - Functional safety of safety-related electrical, electronic, and electronic programmable control systems
IEC 61508-1:2010	Functional safety of electrical/electronic/programmable electronic safety-related systems: General requirements.
IEC 61508-2:2010	Functional safety of electrical/electronic/programmable electronic safety-related systems: Requirements for electrical/electronic/programmable electronic safety-related systems.
IEC 61508-3:2010	Functional safety of electrical/electronic/programmable electronic safety-related systems: Software requirements.
IEC 61784-3:2021	Industrial communication networks - Profiles - Part 3: Functional safety fieldbuses - General rules and profile definitions.
2006/42/EC	Machinery Directive
2014/30/EU	Electromagnetic Compatibility Directive
2014/35/EU	Low Voltage Directive

In addition, terms used in the information contained herein may tangentially be used as they are derived from other standards such as:

Standard	Description
IEC 60034 series	Rotating electrical machines
IEC 61800 series	Adjustable speed electrical power drive systems
IEC 61158 series	Digital data communications for measurement and control – Fieldbus for use in industrial control systems

Finally, the term *zone of operation* may be used in conjunction with the description of specific hazards, and is defined as it is for a hazard zone or danger zone in the Machinery Directive (2006/42/EC) and ISO 12100:2010.

NOTE: The aforementioned standards may or may not apply to the specific products cited in the present documentation. For more information concerning the individual standards applicable to the products described herein, see the characteristics tables for those product references.

Trademarks

OPC Foundation is either a registered trademark, trademark, or service mark of the OPC Foundation.

Microsoft, Windows, and Windows Server are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries/regions.

EcoStruxure OPC UA Server Expert Server

What's in This Part

Features

What's in This Chapter

Supported Profiles

Introduction

The primary purpose of the EcoStruxure OPC UA Server Expert server (herein, OPC UA server) is to provide an OPC UA communication channel between a Modicon controller or Modbus devices and OPC UA clients. The data of the Modicon controller or Modbus devices is mapped to variables in the OPC UA server and made available to OPC UA clients.

NOTE: The terms of each connection between an OPC UA client and the OPC UA server are determined by the client, which sets the attributes of the connection between the client and server.

Supported Profile

EcoStruxure OPC UA Server Expert supports the profile Standard 2017 UA Server Profile, with restrictions related to supported OPC UA services, page 15. For more information, visit the OPC Foundation website at http://opcfoundation.org/UA-Profile/Server/StandardUA2017.

Supported Facets

EcoStruxure OPC UA Server Expert supports the following Facets:

- **Server Category > Facets > Core Characteristics**:

  Core Server Facet (http://opcfoundation.org/UA-Profile/Server/CoreFacet) with restrictions related to supported OPC UA services, page 15.

- **Server Category > Facets > Data Access**:

  Data Access Server Facet (http://opcfoundation.org/UA-Profile/Server/DataAccess)

- **Security Category > Facets > Security Policy**:
  - None (http://opcfoundation.org/UA/SecurityPolicy#None)

    NOTE: You can enable the security policy **None** using the EcoStruxure OPC UA Server Expert Configuration Tool, page 47.
  - Basic256Sha256 (http://opcfoundation.org/UA/SecurityPolicy#Basic256Sha256)
  - Aes128_Sha256_RsaOaep (http://opcfoundation.org/UA/SecurityPolicy#Aes128_Sha256_RsaOaep)
  - Aes256_Sha256_RsaPss (http://opcfoundation.org/UA/SecurityPolicy#Aes256_Sha256_RsaPss)

- **Security Category > Facets > User Token > General**:

  Anonymous Facet
  (http://opcfoundation.org/UA-Profile/Security/UserToken/Anonymous)

  NOTE: You can enable the user token **Anonymous** using the EcoStruxure OPC UA Server Expert Configuration Tool, page 47.
- **Security Category > Facets > User Token > Server**:
  - User Name Password Server Facet
    (http://opcfoundation.org/UA-Profile/Security/UserToken/Server/UserNamePassword)
  - X509 Certificate Server Facet
    (http://opcfoundation.org/UA-Profile/Security/UserToken/Server/X509Certificate)

    NOTE: You can enable the user token **X509 Certificate** using the EcoStruxure OPC UA Server Expert Configuration Tool, page 47.
- **Transport Category > Facets > Client-Server**:

  UA-TCP UA-SC UA-Binary
  (http://opcfoundation.org/UA-Profile/Transport/uatcp-uasc-uabinary)

Server Stack Services

Supported OPC UA Services

The OPC UA server supports the following service sets and services.

Service set	Services
Attribute	• *Read* • *Write*
Method	• *Call*
MonitoredItem	• *CreateMonitoredItems* • *ModifyMonitoredItems* • *DeleteMonitoredItems* • *SetMonitoringMode*
SecureChannel	• *OpenSecureChannel* • *CloseSecurechannel*
Session	• *CreateSession* • *ActivateSession* • *CloseSession*
Subscription	• *CreateSubscription* • *ModifySubscription* • *DeleteSubscription* • *SetPublishingMode* • *Publish* • *Republish*
View	• *Browse* • *BrowseNext* • *TranslateBrowsePathToNodeIds*

NOTE: For a description of these service sets and services, refer to OPC Unified Architecture Specification Part 4: Services *(Release 1.05)*.

Operation

What's in This Chapter

Overview

This chapter describes the operating specifications, performance considerations, and guidelines for operating the EcoStruxure OPC UA Server Expert server.

Operating Specifications

Limits

- The maximum number of nodes that can be published in EcoStruxure OPC UA Server Expert data access address space is 400,000 if *Flat Mode* for the **DA Address Space** property is cleared, page 63. If *Flat Mode* is selected, you can exceed this limit, but hardware resources such as the CPU and memory may be limiting factors.

  NOTE: If *Flat Mode* is cleared and you exceed the limit, a message is displayed, but you can continue running EcoStruxure OPC UA Server Expert.

- The maximum memory size that can be allocated to EcoStruxure OPC UA Server Expert is 4 GB (3 GB on x86 platforms).

The following table describes the other limits, the context in which they occur, and their impact if exceeded.

Limit	Value	OPCUA service	Service parameter	Impact
Maximum session count	100	*CreateSession*	Not applicable	*Bad_TooManySessions* service result code
Minimum session timeout	10 s		Requested *SessionTimeout*	*revisedSession* Timeout
Maximum session timeout	3,600 s		Requested *SessionTimeout*	*revisedSession* Timeout
Maximum subscription count	300	*CreateSubscription*	Not applicable	*Bad_TooManySubscriptions* service result code
Minimum publishing interval	100 ms		Requested Publishing Interval	*revisedPublishingInterval*
Maximum publishing interval	3,600 s		Requested Publishing Interval	*revisedPublishingInterval*
Maximum subscription lifetime	3,600 s		Min(Requested Publishing Interval, 3,600,000) * Requested *LifetimeCount*	*revisedLifetimeCount*
Maximum notifications per publish	10,000		*maxNotificationsPerPublish*	Notifications maximum capacity is (1,000/*revisedPublishingInterval*) * 1,000 notifications per second
Minimum sampling interval	100 ms	*CreateMonitoredItems*	*MonitoringParameters SamplingInterval*	*revisedSampling* Interval
Maximum message queue size	100		*MonitoringParameters QueueSize*	*revisedQueueSize*
Maximum monitored items count	400,000(1)		Not applicable	*Bad_TooManyMonitoredItems* service result code
(1) If *Flat Mode* for the **DA Address Space** property is cleared, page 63. Otherwise, 600,000.				

Performance

Application Performance

If the execution of an EcoStruxure OPC UA Server Expert task cannot be completed within the configured time, the notification `Application overload` is displayed as follows:

- In the **Diagnostics** tab if the server is running as a standalone application, page 23.
- In the notification area if the server is running as a service, page 29.

Notifications can occur at the following frequency:

- Intermittently: It indicates a transient host system overload.
- Persistently: It indicates that the host system (memory, processor capacity, and/or other applications running on the host) lacks capacity to run EcoStruxure OPC UA Server Expert.

The following table describes the additional information that the notification `Application overload` can contain.

Notification	Description	Probable root cause
Poller task overrun (Delay: <delay> ms, Base: <base> ms)	The data source poller task is delayed by *<delay>* ms	There are too many communication requests polling data sources (controllers or Modbus devices).
Publishing timer overrun for session <SessionId> (Delay: <delay> ms, Base: <base> ms)	The per-session publishing timer is delayed by *<delay>* ms for session <SessionId>	Too many notifications are published for the session, or too many notifications are published for at least one subscription.
Publish overrun for subscription <SubscriptionID> (Delay: <delay> ms, Base: <publishing interval> ms)	The publish operation duration is *<delay>* ms higher than *<publishing interval>* for subscription *<SubscriptionID>*	Too many notifications are published for the subscription.
Receiver task overrun (Delay: <delay> ms, Base: <base> ms)	The network listening task is delayed by *<delay>* ms	Network latency or significant application overload.

Communication Performance

When the communication flow sent to a device exceeds the communication capacity of that device:

- The notification `Communication overload` is displayed in the **Diagnostics** tab (or in the notification area) with the device configured address.
- The configured **Monitored items sampling overrun policy** is applied, page 62.

Notifications can occur at the following frequency:

- Intermittently: It indicates a transient host system overload.
- Persistently: It indicates that the number of polling requests sent to the device to fulfill the *MonitoredItems* sampling rate exceeds the device capacity.

  NOTE: You can verify the number of requests sent to a device per timer in the *Timer* list in the Network tab, page 23.

For example, if the *Timer* list includes the following:

Timer number: 3

Timer 100 ms: 1 polling and 0 custom requests

Timer 1,000 ms: 87 polling and 0 custom requests

Timer 2,000 ms: 0 polling and 3 custom requests

The following formula yields the maximum number of requests that can be sent per second to the device communication adapter:

Max Flow = sum(1,000*Polling requests/Timer (in ms))

In this example:

Max Flow = (1,000*1/100) + (1,000*87/1,000) + 0 = 97 requests/s

Coordinating Server, Device, and Controller Communications

Each controller can process a different number of requests in each scan of the MAST task. The following table gives the number of requests per controller scan time according to the controller reference.

Controller reference	Communication adapter	Request per controller scan time	*Max channels/Max pending* possible configuration, page 56
BMXP581020	Embedded port	8	8/0, 8/8, 4/8
BMXP5820•0	Embedded port	12	12/0, 12/12, 4/12
BMXP5830•0	Embedded port	16	16/0, 16/16, 4/16
BMXP5840•0	Embedded port	16	16/0, 16/16, 4/16
BMXP585040	Embedded port	16	16/0, 16/16, 4/16
BMXP586040	Embedded port	16	16/0, 16/16, 4/16
BMXP3410••	Embedded port	8	8/0, 8/8, 4/8
BMXP3420••	Embedded port	8	8/0, 8/8, 4/8

A general rule for computing the communication capacity in requests per second is:

Max Capacity = 1,000*requests per controller scan time/controller scan time (in ms)

This rule presents the theoretical case. In practice, the controller scan time is replaced by *request time*, where:

Request time = transmission time request + (2 x controller scan time) + transmission time answer + (2*OPC UA Server Expert receiver scan), where:

OPC UA Server Expert receiver scan = 10 ms and transmission time is typically negligible compared to other times.

NOTE: If the controller communication is shared by EcoStruxure OPC UA Server Expert and other Modbus clients (and\or if multiple BMENOC•••• communication modules are operated) then, the formula above may not be predictive, especially in case of high communication flow. In this case, replace request time with the actual average access time displayed in the **Network** tab of the controller.

NOTE: If *Max Flow* is greater than *Max Capacity*, the monitored items sampling interval is overrun. The message `Communication overload` is displayed and the configured **Monitored items sampling overrun policy** is applied, page 62. To mitigate permanent communication overload, increase the monitored items sampling rate, or decrease the number of monitored items.

For example, with request per controller scan time = 16 and controller scan time = 50 ms:

Max Capacity = 1,000*16/50 = 320 requests/s

NOTE: With controller request length =1,024, each read request can contain up to 500 values of OPC variables with data type INT16 or UINT16.

Design Best Practices

When designing an EcoStruxure Control Expert application, take note of the number of OPC UA nodes your design will create.

For example:

- For a one-dimensional ARRAY instance with <*array elements count*> elements, <*array elements count*> nodes are created in the EcoStruxure OPC UA Server Expert data access address space.

- For a multi-dimensional ARRAY instance of <n> dimensions, with <*array elements count1*>, <*array elements count2*> … <*array elements countn*> elements per respective dimension, <*array elements count1*>*<*array elements count2*>*…*<*array elements countn*> nodes are created.
- For a DDT instance, one node for each DDT member is created.

In the above examples, if an ARRAY element or a DDT member is itself an ARRAY or a DDT, the same node computation rules apply on the element.

When configuring the OPC UA client session timeout and subscriptions lifetime settings (calculated as *LifetimeCount * Publishing rate*), set values that will reduce the likelihood that the server will unnecessarily prolong inactive sessions and subscriptions. Setting too high values can cause the server to significantly increase its workload, and reach the monitored items limitation.

Because EcoStruxure OPC UA Server Expert does not support the *TransferSubscription* OPCUA service, a subscriptions lifetime setting that is longer than the session timeout is overridden by the session timeout. This is because, as soon as the session is deleted on timeout, subscriptions are deleted without waiting for the expiration of the subscription lifetime.

Operating Modes

What's in This Chapter

Overview

Operating the OPC UA Server as a Windows Service or a Standalone Application

You can configure the OPC UA server to run on the host computer as either:

- A Windows service
- A standalone application

In both modes, you can use the OPC UA server to monitor, diagnose, and log the following types of data:

- Application diagnostic data: Includes detected errors related to the application execution. When the optional verbose mode is configured, it also logs OPC UA events.
- Communication diagnostic data: Includes detected errors related to communications between controllers and devices. When the optional verbose mode is configured, it also logs communication information.
- Snapshot file data: Contains runtime and statistical information displayed in the **Network** and **Sessions** tabs, page 22.

  NOTE: Use the following settings to enable error detection and information logging:

  - Detection of application errors is enabled in the **Settings > Diagnostic** window in the EcoStruxure OPC UA Server Expert Configuration Tool by selecting and configuring the **Application > Diagnostic** setting.
  - Detection of communication errors is enabled in the **Settings > Diagnostic** window in the EcoStruxure OPC UA Server Expert Configuration Tool by selecting and configuring the **Communication > Diagnostic** setting.
  - Logging of dynamically updated runtime information and statistics is enabled in the **Settings > Diagnostic** window by selecting and configuring the **Snapshot File** setting.

The configuration is performed using the EcoStruxure OPC UA Server Expert Configuration Tool, page 47.

Shutting Down and Restarting the Server

When you close the standalone application or stop the service (service mode), there is a short delay before the EcoStruxure OPC UA Server Expert shut down completes. For this reason, wait a short period before you restart it; otherwise, a detected error message is displayed.

Configuring the OPC UA Server to Run as a Windows Service

To configure the OPC UA server to run as a Windows service, proceed as follows.

Step	Action
1	Open the configuration tool.
2	In the navigation tree, select **Settings** > **Options**.
3	Select **Enable Service mode**.
4	Select **File** > **Save configuration**.

NOTE: When the OPC UA server operates as a service, the EcoStruxure OPC UA Server Expert Service Notifier icon is displayed in the system tray.

For information on how to start, stop, and use the service, refer to Operating EcoStruxure OPC UA Server Expert as a Windows Service, page 29.

Configuring the OPC UA Server to Run as a Standalone Application

To configure the OPC UA server to run as a standalone application, disable the Windows service. Proceed as follows:

Step	Action
1	Open the configuration tool.
2	In the navigation tree, select **Settings** > **Options**.
3	Clear **Enable Service mode**.
4	Select **File** > **Save configuration**.

NOTE: When the OPC UA server operates as a standalone application, the EcoStruxure OPC UA Server Expert Service Notifier icon is not displayed in the system tray.

For information on how to start, stop, and use the standalone application, refer to Operating EcoStruxure OPC UA Server Expert as a Standalone Application, page 22.

Operating EcoStruxure OPC UA Server Expert as a Standalone Application

Introduction

When EcoStruxure OPC UA Server Expert is configured to operate as a standalone application, page 22, you can open it either of the following ways:

- Navigating to the installation folder and double-clicking the executable file.
- Clicking the Windows Start button and selecting EcoStruxure OPC UA Server Expert.

Standalone Application Interface

The EcoStruxure OPC UA Server Expert user interface displays:

- Menu items:
  - **File**
  - **Help**
- An address area that identifies the OPC UA URIs supported by the server.
- A main screen that includes three sub-tabs:
  - **Diagnostics**
  - **Network**
  - **Sessions**
- A status bar that displays:
  - Server **Status**: **Running** or **Stopped**
  - **Start Date**: The date and time (mm-dd-yy hh:mm:ss) the server was started.
  - **Current Date**: The present date and time (mm-dd-yy hh:mm:ss).
  - **Version**: The EcoStruxure OPC UA Server Expert version.

File Menu

The **File** menu contains the following commands:

- **Save Snapshot**: Saves a snapshot in the form of a text file that contains the field controller configuration settings, active OPC UA client session, subscription information, and monitored service messages. The snapshot file is stored in the location specified in the **Settings > Diagnostic** window, page 57.
- **Save As Snapshot**: Opens a dialog box where you can create a snapshot file and specify the path and file name.

Diagnostics Tab

The **Diagnostics** tab displays informational and detected error messages. Detected error messages are related to:

- OPC UA server service detected errors.
- Controller and device communication detected errors.

Network Tab

The **Network** tab contains the following information:

- Network-related data describing interaction of the server with the network.
- Device-specific data for each monitored device.

The following table describes the fields of the **Network** tab.

Field	Description
Devices	The number of devices managed.
DA Nodes	The number of nodes under the Data Access (DA) folder.
Monitored DA Nodes	The number of monitored nodes under the DA folder.
Timers	The number of polling timers.
Poller task tick	*Keep alive* counter of the poller task.
Receiver task tick	*Keep alive* counter of the receiver task.
Answer task tick	*Keep alive* counter of the answer task.
Mast requests	The total number of requests sent on the network.
Mast answers	The total number of answers received on the network.
Mast errors	Total number of communication detected errors.

The following table describes the device-specific data displayed for each listed device.

Field	Description
<Device address>=<alias name> [Primary Address Standby Address]	The combination of: • Device address 1 • Configured alias name • Primary address and communication status, page 41. • Standby address and communication status **NOTE:** *Undefined* if not configured.
UNITY device or *MODBUS* Device	The device configured type.
Device Identity	The device identity, or *Unknown* if not configured.
Device Version	The device version, or *Unknown* if not configured.
Device Status, page 40	• *Good* • *Uncertain* • *Bad*
Frame TO	The configured **Frame timeout (ms)**.
Device TO	The configured **Device timeout (ms)**.
Max Channel	• *Used*: The number of channels used. • *Configured*: The **Max channels** value configured for the device alias in the EcoStruxure OPC UA Server Expert Configuration Tool (**Adjustment Information** category).
Max Pending Req	The actual *Max Pending* value used; not the **Max pending** configured setting.
Max Waiting Req	The device-specific requests waiting list size.
Topologic objects is	The setting of the **Topological object** property: • *Set* • *Not set*
Symbol table from PLC *Symbol table used: <filepath>* *No Symbol table*	The type of configured symbol link: • Data dictionary • Full path of the XVM or CSV file • None
Is using PLC DataDictionary or *NOT Using PLC DataDictionary*	The type of configured symbol link.
Application (Symbol table)	The application name taken from the XVM symbol link file.
Application (Device)	The application name taken from the controller.
Consistency Status	• *TestedUnknown*: For Modbus devices. • *Consistent*: If the controller application and controller symbol link are consistent. Does not apply to Modbus devices. For other cases, refer to symbol link consistency policy, page 72.
Device	The read and write properties of the device set in the **Option > Read Only** parameter, page 55: • *READ_ONLY* • *READ_WRITE*
Device Access	The data access, page 41 status: • *ENABLED* • *DISABLED*
Request Length	The length of controller communication requests.
Req Sent+Rcv	The sum of the number of sent and received requests.
Bytes Sent	The number of bytes sent to the controller.
Bytes Rcv	The number of bytes received from the controller.
Ref Count	The reference counter of the device.

Field	Description
Nb of Error	The number of communication detected errors.
Last Error Code	Allows diagnosing the last detected communication error.
State Cnt Good =	The number of devices with the respective device status, page 40.
State Cnt Uncertain =	
State Cnt Bad =	
Nb of Waiting Req (max reached)	The present number of waiting requests and the maximum number of waiting requests reached.
Best Access Time	The shortest time between communication request and communication access.
Worst Access Time	The longest time between communication request and communication access.
Last Access Time	The most recent time between communication request and communication access.
Average Access Time	The average time between communication request and communication access.
Nb Var Desc	The number of internal objects for *DataItem* management, page 76.
Nb Specific Var Desc	Number of internal objects for *Specific DataItems* management, page 80.
Timer Number	The number of active timers.
Timer list	The list of active timers with the number of polling and custom requests sent to the device.
Application Version (Symbol table)	The application signatures list.
Application Version (Device)	The device signatures list.
DataDictionnary option	The server uses the controller embedded data dictionary as a source for variable symbols: • Selected: Detection is enabled • Cleared: Detection is disabled For more information, refer to **Controller Embedded data > Using Data Dictionary** in the General Alias properties window, page 55.
No communication break option	On an application build change, the server is configured to switch to a pre-loaded new data dictionary to help avoid a communication break: • Selected: Enabled • Cleared: Disabled For more information, refer to **Controller Embedded data > No Communication Break** in the General Alias properties window, page 55.
New symbol detection option	The automatic detection and population of new symbols: • Selected: Enabled • Cleared: Disabled For more information, refer to **Dynamic consistency > New symbol detection** in the General Alias properties window, page 55.

The following table describes the values *Last Error Code* can have.

Value (Hex)	Description	Comment
AFE20006	Network disconnection	The connection with the controller is interrupted (permanently or temporarily).
EFE20007	Network overload	The communication channel is too busy to operate. It is generally reset by the server.
EFEF61DA	Communication overload	For more information, refer to Coordinating Server, Device, and Controller Communications, page 19.
80004005	Communication timeout	No answer was received from the controller since the configured **Frame timeout (ms)**.
EFEF6243	Communication task overload	The server communication task is overloaded.
EFEF61C6	Controller Modbus server overload	The controller Modbus server raised a Modbus exception, generally due to Modbus server overload.

Sessions Tab

The **Sessions** tab contains two panes:

- The **Sessions** pane (top) describes the OPC UA server session.
- The **Subscriptions** pane (bottom) describes a list of subscriptions requested by an OPC UA client and created by the OPC UA server.

The following table describes the **Sessions** pane data.

Field	Description
Session Name	A readable string that identifies the session, page 38, set in the *CreateSession()* service parameter.
Identity	The data access status: • *Anonymous*: For an anonymous session. • *UserName*: Provided by a *UserName* identity token, page 37. • *Certificate Subject Name*: provided by an X509 identity token, page 37.
Session ID	The *NodeId* assigned by the server to the session.
Connection	The local date and time when *CreateSession()* was invoked.
Last Contact	The local date and time when the last OPC UA service was invoked.
Subscriptions Count	The number of subscriptions in the session.
Monitored Items	The total number of *MonitoredItems* in the session.
Session Timeout (ms)	The present session timeout in ms. In brackets, the requested session timeout (set in the *CreateSession()* service parameter).
Publishing Time Overrun Count	The total cumulative number of times the per-instance subscription publishing timer was in overrun condition. A notification is displayed each time an overrun occurs. For more information, refer to Performance, page 17.

The following table describes the **Subscriptions** pane data.

Field	Description
SubscriptionId	The server-assigned identifier for the subscription.
PublishingEnabled	• TRUE: Publishing of Notification Messages is enabled for the subscription. • FALSE: Publishing of Notification Messages is disabled for the subscription.
Publishing Interval	The publishing interval that the server will use (in ms).
Keep Alive Count	The number of *keep alive* messages sent to the client during the present subscription. When the number of consecutive publishing cycles (in which there have been no notifications to report to the client) reaches this counter, a publish request is de-queued and used to return a *keep alive* message. This *keep alive* message informs the client that the subscription is still active.
Life Time Count	When the number of consecutive publishing cycles (in which there have been no publish requests available to send a publish response) reaches this counter, the subscription is closed.
Max Per Publish	The maximum number of notifications per publish response.
Items (Disabled) Count	The number of items attached to the subscription, including disabled items.
Notifications (Overflow \Discarded) Count	• *Notifications*: The total cumulative number of *Notifications* sent by the subscription. • *Overflow*: The total number of times a monitored item dropped notifications because of a queue overflow (not applicable if the queue size is ≤ 1). • *Discarded*: The total number of times a monitored item dropped notifications because of a cache value overwrite (not applicable if the queue size is > 1).
Publish (Late) Count	• *Publish*: The total number of de-queued publish requests. • *Late*: The total number of times the subscription entered the *LATE* state, that is to say, the number of times the publish timer expires and there are unsent notifications.
Seq No (Unack\Discarded Count)	• *Seq No*: The sequence number for the next Notification Message. • *Unack*: The total number of unacknowledged Notification Messages saved in the republish queue. • *Discarded*: The total number of Notification Messages that were discarded before they were acknowledged.
Publish Overrun Count	The total number of times the publish operation was in overrun condition. A notification is displayed each time an overrun occurs. For more information, refer to Performance, page 17.

Operating EcoStruxure OPC UA Server Expert as a Windows Service

Starting the Service

On first start, when the OPC UA server is configured as a service, page 22, the server is not running. There are two locations where you can start and stop the server:

- In the EcoStruxure OPC UA Server Expert Configuration Tool:
  - Select **Service > Start server as service** to start the service.
  - Select **Service > Stop server** to stop the service.

    NOTE: You can also use the **Start server as service** and **Stop server** toolbar buttons.
- In the **Services** tab of Windows **Task Manager**, right-click *OPCUAServerExpert* and select:
  - **Start** to start the service.
  - **Stop** to stop the service.

  NOTE: Only user accounts with administrator privileges can use the EcoStruxure OPC UA Server Expert Configuration Tool to configure the OPC UA server service mode, and start or stop the server as a service.

  NOTE: If restarting the OPC UA server service from the **Services** tab of the Windows **Task Manager** does not succeed, wait a short period of time and then, restart the service from EcoStruxure OPC UA Server Expert Configuration Tool or the **Services** tab of the Windows **Task Manager**. On system reboot, the EcoStruxure OPC UA Server Expert Service Notifier icon is immediately displayed, but the *OPCUAServerExpert* service start may be delayed for a period of time that is determined by your operating system. For example, a delay of 2 minutes can occur with Windows 10.

Monitoring Service Operations

When the OPC UA server operates as a service, you can use EcoStruxure OPC UA Server Expert Service Notifier icon to monitor operations and do the following:

- Diagnose the OPC UA server status: Place the pointer on the Service Notifier icon in the system tray to display the status, page 23.
- Monitor OPC UA server status changes: When the server status changes (from **Running** to **Stopped**, or from **Stopped** to **Running**) a message is displayed. A record of the change is logged in the OPC UA server service **Notifications** window.

  NOTE: To display the **Notifications** window, right-click the Service Notifier icon in the system tray and select **Show Notifications**.
- Log OPC UA server messages and detected errors: These are the same items as those logged in the **Diagnostics** tab when the OPC UA server operates as a standalone application. These items are also displayed in the **Notifications** window.

Certificate Management

What's in This Chapter

Application Instance Certificates

Types of Application Instance Certificates

EcoStruxure OPC UA Server Expert supports the use of two types of certificates:

- Self-signed certificates
- Certificates issued by a Certificate Authority (CA)

Self-Signed Certificates

A self-signed certificate is an identity certificate that is signed by the same entity whose identity it certifies. When you install EcoStruxure OPC UA Server Expert, you also install the following self-signed instance application certificate:

OPCUAServerExpert-Server.der

This self-signed certificate is installed on the host computer at the following location:

C:\Program Files (x86)\Schneider Electric\OPC UA Server Expert\PKI

The self-signed certificate is renewed either by running the *renewcertificate* tool, page 35 or by updating the certificate hostnames list, page 65.

Certificates Issued by a Certificate Authority

Instead of using self-signed certificates in your project, you can use a CA. The CA can be used to issue both OPC UA server and OPC UA client certificates.

When you install the OPC UA Server application, you also install the following Certificate Signing Request (CSR):

OPCUAServerExpertCsr.pem

The CSR can be used to issue the OPC UA server certificate. It is installed on the host computer at the following location:

C:\Program Files (x86)\Schneider Electric\OPC UA Server Expert\PKI

The CSR can be renewed either by running the *renewcertificate* tool, page 35 or by updating the certificate hostnames list, page 65

Certificate Deployment

When Using Self-Signed Certificates

To allow the OPC UA Server and an OPC UA client to communicate when using self-signed certificates, do the following:

- Add the OPC UA Server self-signed application instance certificate to the OPC UA client Certificate Trust List (CTL) (vendor-dependent). On server certificate renewal, repeat this operation.
- Add the OPC UA client self-signed application instance certificate to the OPC UA Server CTL, page 31. On client certificate renewal, repeat this operation.

  NOTE: The *OPC UA Server Expert Sample Client* self-signed certificate is automatically added to the OPC UA Server CTL on certificate renewal.

For more information, refer to Certificate Renewal, page 35.

When Using a Certificate Authority

To allow the OPC UA Server and an OPC UA client to communicate when using a CA, add the CA certificate or CA chain certificate to the OPC UA client CTL (Vendor-dependent) and to the OPC UA Server CTL, page 31.

Both the OPC UA Server and the OPC UA client must be enrolled.

Enrolling the OPC UA Server

To enroll the OPC UA Server, proceed as follows.

Step	Action
1	Locate the server CSR, page 30.
2	Send the CSR to the intended CA that processes the CSR and issues a new certificate (*<IssuedCertificate>*).
3	A user with administrator privileges must replace the self-signed application certificate with the CA issued certificate by running from a Command Line Interface: `"C:\Program Files (x86)\Schneider Electric\OPC UA Server Expert\PKI\setcertificate" <IssuedCertificateFullPath>` **NOTE:** The *setcertificate* tool is installed as part of the EcoStruxure OPC UA Server Expert installation. Only use certificates that have the *.der* file name extension.
4	A user with administrator privileges must add the CA certificate or CA chain to the CTL, page 31.

NOTE: On OPC UA Server certificate renewal, repeat the OPC UA Server enrollment. The *OPC UA Server Expert Sample Client* does not support enrollment. For more information, refer to Certificate Renewal, page 35.

Managing the OPC UA Server Certificate Trust List

Introducing the Certificate Trust List

The OPC UA server Certificate Trust List (CTL) contains the application certificates of OPC UA clients with which the OPC UA server can communicate. The CTL is stored on the local computer Windows Certificate Store (WCS), in the *OFSUA_CTL* store

A user with administrator privileges must trust a self-signed OPC UA client certificate (or a CA chain root certificate) by adding the certificate to the CTL.

NOTE: Only a user with administrator privileges can untrust an OPC UA client certificate by removing it from the CTL.

A user with administrator privileges can add an application certificate to the CTL in either of two ways:

- By running the *trustentity* tool from a Command Line Interface. Using this tool, the user with administrator privileges adds a certificate to the CTL, without requiring any knowledge of the WCS from the user.

  NOTE: The *trustentity* tool is installed on the host computer as part of the EcoStruxure OPC UA Server Expert installation.

- By running the WCS Explorer CTL management application. Using this tool, a user with administrator privileges can access a graphical user interface and execute commands for adding, removing, and editing certificates.

Adding a Certificate by Running the *trustentity* Tool

To use the *trustentity* tool to add a certificate to the CTL from a Command Line Interface, proceed as follows.

Step	Action
1	On the host computer, open a Command Line Interface.

2	Execute the following command:
	`"C:\Program Files (x86)\Schneider Electric\OPC UA Server Expert \PKI \trustentity" <CertificateToTrustFullPath>`
	Where:
	`<CertificateToTrustFullPath>` represents the full path to the certificate residing on the local computer. For example: `C:\certs\mycertificate.der`

Managing Certificates Using the WCS Explorer

To use the WCS Explorer to manage certificates in the CTL, proceed as follows.

Step	Action
1	Open the WCS Explorer in either of the following ways: • Select the **EcoStruxure OPC UA Server Expert Windows Certificate Store** shortcut icon that was created on the host computer during the OPC UA server installation. • Run the following command from a Command Line Interface: *C:\Program Files (x86)\Schneider Electric\OPC UA Server Expert\PKI\OPCUA ServerExpert.msc*
2	In the WCS Explorer, select **OFSUA_CTL**.
3	You can perform the following actions: • Add a certificate: Select **Action > All Tasks > Import** and follow the instructions in the **Certificate Import Wizard**. • Remove a certificate: Select the certificate to be removed and then, select **Action > Delete**. • Edit a certificate: Double-click a certificate and then perform the required edits.

Certificate Revocation List Management

A Certificate Revocation List (CRL) is a list of the certificates that have been issued and subsequently revoked by a CA. Because the certificates in a CRL were created by a remote CA, only online CRLs are supported.

NOTE: An offline or local CRL may present a security breach caused by the potential inconsistency between a local CRL and the CRL updated by the CA. For example, a certificate recently revoked by the CA may continue to be considered valid by the local OPC UA server.

An online CRL verification assumes that one of the following is defined in the certificate, making it possible to verify the continuing validity of the certificate:

- An Online Certificate Status Protocol (OCSP) endpoint
- A CRL Distribution Point (DP) in the form of a URL that can be accessed online or offline from the HTTP cache.

An online CRL verification can result in a long delay while the certificate authority is contacted, especially if no Internet connection is available. For this reason, the default setting of the **Certificate Revocation** setting in the **Settings > Security** window, page 63 of the EcoStruxure OPC UA Server Expert Configuration Tool disables this online verification feature.

Certificate Validation Policy

Strict Validation Policy

The OPC UA server applies a strict validation policy to each certificate path when it starts or when a client initiates an OPC UA connection. Rejected certificates are added to the **Rejected Certificate Store** if this selection is enabled in the **Settings > Security** window, page 63 of the EcoStruxure OPC UA Server Expert Configuration Tool.

Potential Validation Process Detected Errors

The validation policy applied by the OPC UA server can detect the following validation errors. The following table describes each potential detected error, including if the detected error can be ignored, and whether it is ignored by the applied validation policy.

Validation step	Detected error	Cause	Severity	
			Can be ignored	Ignored by policy
Host Name	*Bad_CertificateHostNameInvalid*	The *HostName* in the URI that is used to connect to the server does not match a *HostName* specified in the server certificate.	Yes	No
Certificate Structure	*Bad_CertificateInvalid*	The certificate structure is invalid or one of the issuer certificate structures is not valid.	No	No
Signature	*Bad_SecurityChecksFailed*	The certificate signature is invalid or one of the CA certificates in the chain has an invalid signature or is not found. An issuer certificate may be not found even if it exists, but is not a CA.	No	No
Trust List Verification	*Bad_SecurityChecksFailed*	The certificate is not trusted and at least one of the issuers in the chain is not trusted.	No	No
Validity Period	*Bad_CertificateTimeInvalid* *Bad_CertificateIssuerTimeInvalid*	The present time does not fit the certificate validity period or one of the issuer certificate validity periods.	Yes	No
Certificate Usage	*Bad_CertificateUseNotAllowed* *Bad_CertificateIssuerUseNotAllowed*	The set used for the certificate does not match use requested for the certificate (for example, Application, Software or CA)(1).	Yes	No
Find Revocation List	*Bad_CertificateRevocationUnknown* *Bad_CertificateIssuerRevocationUnknown*	A detected error is returned during the CA certificate revocation list verification (for example, a network interruption prevented the application from accessing the list). No detected error is returned if **Certificate Revocation** is not selected in the **Settings > Security** window, page 63 of the configuration tool.	Yes	No
Revocation Verification	*Bad_SecurityChecksFailed*	The certificate was revoked and cannot be used.	No	No

(1) The application certificate must include: *digitalSignature*, *nonRepudiation*, *keyEncipherment*, and *dataEncipherment*.

Causes of Certificate Type Validation Detected Errors

The certificate type validation policy applies to certificates of the following types:

- OPC UA server application instance certificates
- OPC UA client application instance certificates provided by the *OpenSecureChannel()* and *CreateSession()* services, provided that the client establishes a security policy other than *None*
- X509 user token certificate

The following table describes the effect and probable cause of a certificate validation detected error.

Certificate type	Use case	Effect	Probable cause
OPC UA server application instance certificate	Starting the OPC UA server	The application start is aborted.	Validity period or revocation verification
OPC UA client application instance certificate	*OpenSecureChannel* or *CreateSession*	Service detected error. Session activation detected error.	Trust List verification or validity period, or revocation verification
	At least one session is created	The secure channel is closed by the OPC UA server at secure channel renewal (each hour).	Validity period or revocation verification
X509 certificate user token	*ActivateSession*	Service detected error. Session activation detected error.	Trust List verification or validity period or revocation verification

Causes of User Token Validation Detected Errors

The following table describes the probable cause and effect of a user token validation detected error, for user token types other than *Anonymous*.

Type	Use case	Effect	Probable cause
UserNamePassword	*ActivateSession*	Service detected error. Session activation detected error.	The token does not match a Windows account. The Windows account does not belong to *OFSUA_ACL* group.
X509Certificate	For the certificate type *X509 certificate user token*, refer to Causes of Certificate Type Validation Detected Errors, page 34.		

Certificate Renewal

You can renew the OPC UA Server self-signed certificate, the CSR, and the EcoStruxure OPC UA Server Expert Sample Client self-signed certificate by executing the following command in a Command Line Interface. Administrator privileges are required.

C:\Program Files (x86)\Schneider Electric\OPC UA Server Expert\PKI \renewcertificate" [-lm lifetimeInMonths]

The self-signed certificate has a default 60-month (5-year) validity period (*lifetime*) that takes effect as of the date the renewal is performed.

You can override the default value by setting the optional *[-lm lifetimeInMonths]* parameter in the command line. Possible values range from 12 to 600. An out-of-range value is overridden by the value 60.

Hostnames specified in the self-signed server certificate and CSR must match the connection configuration. For more information, refer to **Hostname** Section, page 65.

NOTE: If renewed, the OPC UA self-signed server certificate must be re-deployed, page 31. If the OPC UA server certificate is issued by a CA, it must be re-enrolled, page 31.

Security Management

What's in This Chapter

Overview

This chapter describes default security settings, and user authentication and authorization methods.

Default Security Policies

Security Policy Settings at Installation

When you install EcoStruxure OPC UA Server Expert, the OPC UA server is installed and default settings are applied.

Default Security Configuration Settings

The OPC UA server is installed with the following default security settings, page 64:

- The **Security Policy > Allow None** option is cleared. Thus, when attempting to connect with the OPC UA server in its default configuration, an OPC UA client must use the **Basic256Sha256** security policy.
- The **Anonymous user token > Allow** option is cleared. Thus, when attempting to activate a session with the OPC UA server in its default configuration, an OPC UA client must use one of the following user token options:
  - *UserName* identity token
  - X509 identity token
- The **User authentication > Activate** option is selected. Thus, if the user authentication or authorization verifications do not succeed, the OPC UA client cannot establish a connection with the OPC UA server.

For more information on these settings and how to use the EcoStruxure OPC UA Server Expert Configuration Tool to edit them, refer to **Security** Settings, page 63.

User Authentication and Authorization

Authenticating and Authorizing User Tokens

When user authorization is activated and anonymous user tokens are not allowed for the OPC UA server (for example, the default configuration settings, page 63), an OPC UA client must provide a user identity token when executing the *ActivateSession()* service. The user identity token can be a *UserName* identity token or an X509 identity token, if X509 user tokens are allowed, page 63. Only an authenticated and authorized user can connect to the OPC UA server.

The following descriptions of identity tokens assume user authorization has been activated and X509 user tokens are allowed.

UserName Identity Token

When the OPC UA client provides a *UserName* identity token, the OPC UA server performs authentication and authorization verifications in the following ways:

- Authentication verification is based on Windows user accounts. A *UserName* identity token is authenticated if its content matches a Windows user account (local or domain) with the same username and password combination.
- Authorization verification is based on an Access Control List (ACL). A Windows user group (*OFSUA_ACL*) is created by the setup and initially includes a user with administrator privileges with permission to add authorized Windows user accounts to the ACL.

Only an authenticated user, with an associated Windows user account that is included in an *OFSUA_ACL* group, can connect to the OPC UA server.

X509 Identity Token

When the OPC UA client provides an X509 identity token, the OPC UA server performs user authentication and authorization verifications in the following ways:

- Authentication relies on the use of public and private keys, as follows:
  - The certificate signature set in the *userTokenSignature* parameter of the *ActivateSession()* service (generated with the private key associated with the X509 certificate) is decrypted using the public key provided in the certificate.
  - The decrypted certificate signature is compared to a certificate signature that is computed using the signature algorithm provided in the *userTokenSignature* parameter. If both signatures are equal, the user token is authenticated.
- Authorization is based on the OPC UA server Certificate Trust List (CTL), page 31. If, for the user, the CA and all its intermediate CAs (if applicable) is displayed as trusted in the CTL, the user token is authorized.

Only an authenticated user with a certificate issuer that is trusted in the CTL can connect to the OPC UA server.

Communication Links

What's in This Chapter

Overview

This chapter describes how an OPC UA client can establish a connection with EcoStruxure OPC UA Server Expert, how controller data can be accessed, and how to establish redundant connections between the OPC UA server and the controller.

Connecting an OPC UA Client to the OPC UA Server

OPC UA Client Connecting to the OPC UA Server

You can programmatically obtain the EcoStruxure OPC UA Server Expert supported endpoint URLs using either the *GetEndPoints()* or the *CreateSession()* service. In addition, the URLs supported by the server can be viewed in the address area of the OPC UA server when it is operating as a standalone application, page 22.

NOTE: The OPC UA server does not support registration with a Local Discovery Server nor a Global Discovery Server.

For information on the supported endpoint URLs, refer to **Connection Settings** Section, page 65.

OPC UA Client Monitoring OPC UA Server Status

An OPC UA client can query the OPC UA server *ServiceLevel* variable (as formally defined in the OPC Unified Architecture part 5) to monitor the ability of the OPC UA server to provide data. The following features are combined in *ServiceLevel* computation:

- The server license status: *Missing*, *DemoMode*, or *Activated*.

  For more information, refer to *Licensing Instructions* in the *Modicon PAC System Software and Hardware Release Notes*.
- The data access address space state: *Empty*, *Built*, *Partially Built*, or *Updating*.

The *ServiceLevel* variable can have the following values.

***ServiceLevel* value**	**License status**	**Address space state**
1	Missing and DEMO mode expired	*Any*
3	Activated or DEMO mode	*Empty*
4	Activated or DEMO mode	*Partially Built*
199	Activated or DEMO mode	*Updating*
255	Activated or DEMO mode	*Built*

For example, *ServiceLevel* value 199 indicates that services in the *View Service Set* may generate a detected error. In this case, it is a best practice that an OPC

UA client waits until the *ServiceLevel* value returns to value 255 before attempting to perform any of the *View Service Set* services.

Communicating with Controllers and Devices

Supported Communication Protocols and Devices

EcoStruxure OPC UA Server Expert can operate with the following platforms and communication networks.

Platform	Communication Protocol			
	Modbus TCP/IP	**USB**	**Modbus serial**	**XIP**
M340	Built-in channel BMXNOE0100 BMXNOE0110 BMXNOC0401	Built-in channel	Built-in channel	Not applicable
M580	Built-in channel BMENOC03•• BMXNOC0402	Built-in channel	BMXNOM02••	
MC80	Built-in channel	Built-in channel	Built-in channel	
Modbus devices	Platform-dependent	Not applicable	Platform-dependent	
Momentum	Built-in channel	Built-in channel	Built-in channel	
Quantum	140NOE771••	Built-in channel	Built-in channel	
Premium	Not applicable	Built-in channel	Not applicable	Built-in channel TSXETY••••

NOTE: Quantum and Premium platforms not managed with Unity Pro or EcoStruxure Control Expert are not supported by EcoStruxure OPC UA Server Expert 2.0 and any subsequent versions.

⚠WARNING

UNINTENDED EQUIPMENT OPERATION

Ensure that you operate a controller platform supported by EcoStruxure OPC UA Server Expert.

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

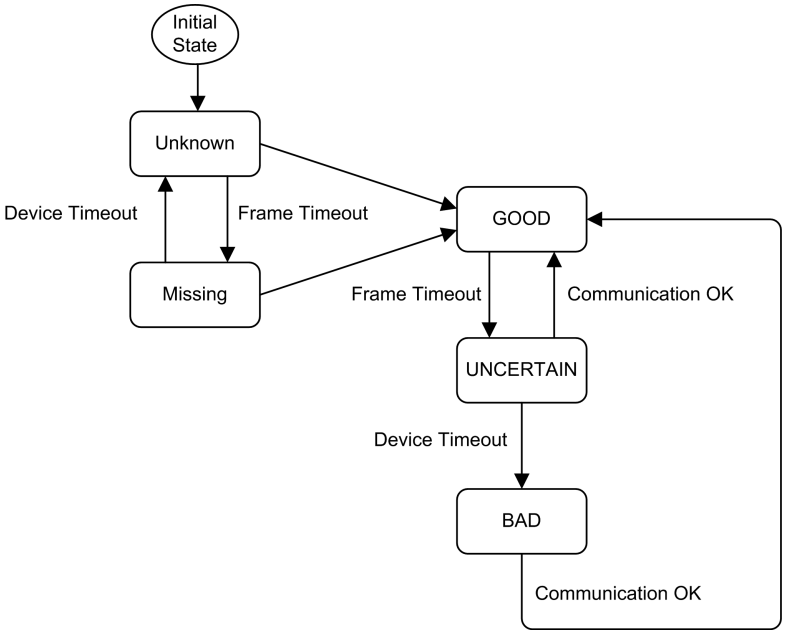
Requirements for Access to Controller or Device Variables Using OPC UA *DataItems*

The ability of the OPC UA server to read or write controller or device variables using *DataItems* depends on these preconditions:

- A communications path is established between the OPC UA server and the controller or device.
- Communication status, page 41 is *ONLINE*.
- Data access status, page 41 is *ENABLED*.

Device Status

A controller or device can transition through several states as depicted in the following state diagram.



NOTE: Device timeout (ms) and **Frame timeout (ms)** can be configured using the EcoStruxure OPC UA Server Expert Configuration Tool in the **Devices > Device overview** window, page 56.

NOTE: Device Status is displayed in the EcoStruxure OPC UA Server Expert **Network** tab, when the server is running as a standalone application, page 23.

Communication Status

The following table describes the relation between the communication status of a device and the device status.

Communication status	Description
ONLINE	Device status is *Good* or *Uncertain*.
OFFLINE	The cause for an *OFFLINE* communication status can be: • Device Status is *Bad*. • Server communication is stopped because the *DEMO* mode expired and licensing information is missing. • Device communication is disabled because controller link redundancy, page 43 is configured, and the standby communication path did not pass the communication path consistency verification. • Device communication is disabled because of the configuration of the disconnection timeout property, page 56.

NOTE: The communication status is included in both the primary address and the standby address as displayed in the EcoStruxure OPC UA Server Expert **Network** tab when the server is running as a standalone application, page 23.

Data Access Status

The following table describes the relation between the data access status of a device and the device status.

Data access status	Description
ENABLED	Data access status is *ENABLED* if the following conditions are true: • Communication status is *ONLINE*. • The symbol link (the collection of symbols derived from controller variables) is consistent with the source controller variables.
DISABLED	Data access status is *DISABLED* if either of the following conditions is true: • Communication status is *OFFLINE*. • The symbol link is not consistent with the source XVM symbol link file.

NOTE: The data access status is displayed as **Device access** in the EcoStruxure OPC UA Server Expert **Network** tab, when the server is running as a standalone application, page 23.

Controller Link Redundancy

Introduction

You can configure the OPC UA server to establish redundant Unified Messaging Application Services (UMAS) links to an EcoStruxure Control Expert controller. Communication redundancy relies on the use of:

- Two separate physical communication paths: A primary path and a standby path.
- Two different IP addresses attributed to the same controller with a unique alias.

The two physical communication paths are achieved using:

- Two separate Ethernet ports in the controller local main rack.

  For example, the two separate connections could be made to either of the following:

  - Two ports, each on a different Ethernet communication module.
  - A controller module port and a port on an Ethernet communication module.
- Two Ethernet ports, each on a separate Network Interface Card (NIC) on the host computer.

The switchover from primary communication path to standby communication path can be automatically performed if the primary path is interrupted or in response to a command. In each case, the switchover does not impact the OPC UA client application, for example a SCADA system.

Using the controller link redundancy feature, you can design a redundant network architecture that provides for a communication path switchover that helps maintain communications if the connection to part of the network is interrupted.

NOTE: The controller link redundancy feature only applies to standalone controller architectures supporting Modbus TCP/IP communications. It does not apply to redundant controller architectures that provide their own redundancy features.

Establishing Primary and Standby Communication Paths

Controller link redundancy is enabled using the EcoStruxure OPC UA Server Expert Configuration Tool to enter two different IP addresses for the controller in the **Device address 1** and **Device address 2** fields.

Neither communication path, with its associated IP address, is configured as the primary path. Instead, the first communication path to establish a connection and successfully transmit application information is designated the primary communication path. At this moment, the primary communication path status is *ONLINE* and communication is supported by the OPC UA server.

After the primary communication path is established, the remaining path becomes the standby communication path. A connection verification and a consistency verification are performed on the standby communication path every 10 seconds. The results of these communication path verifications can be:

- Communication verification: **OK** or **Not OK**.
- Consistency verification: **OK** if the following conditions are true; otherwise **Not OK**:
  - The controller firmware versions (in the primary and standby paths) are the same.
  - The application signatures (in the primary and standby paths) are compatible.

    NOTE: The application GUID and the variable layout must be the same.
  - The tested network frame lengths are the same.

Configuring Controller Link Redundancy to Operate with Two Controllers

If the application GUID is the same for both the primary communication path and the standby communication path, it is possible to operate the controller link redundancy feature by connecting each path to a different controller.

When building the controller application for a two-controller design, so that EcoStruxure Control Expert applies the same application signature to the applications in the primary controller and the standby controller, proceed as follows.

Step	Action
1	Build the application in EcoStruxure Control Expert for the primary controller. Use the communication settings, including IP address, that apply to the primary communication path.
2	Download the application from EcoStruxure Control Expert to the primary controller.
3	Use the **Build Changes** command in EcoStruxure Control Expert to re-build the application for the standby controller. Use the communication settings, including IP address, that apply to the standby communication path.
4	Download the application from EcoStruxure Control Expert to the standby controller.

Impact of Communication Path Status on Controller Link Redundancy

After the primary communication path is established and is operating *ONLINE*, the status of the primary and standby communication paths is determined by the communication verification and consistency verification of the standby communication path, as follows:

- If the controller link redundancy feature is configured, but the standby communication path is not identified (for example, not physically connected), then:
  - The standby communication path is *OFFLINE* and no switchover can be performed.
  - The primary communication path remains *ONLINE*.
- If the standby communication path is identified, the consistency verification is performed.
  - If the standby communication path consistency verification is successful:

    - The standby communication path is *ONLINE*.

    - A switchover can be performed from primary to standby communication paths.
  - If the standby communication path consistency verification is unsuccessful:

    - The system is not operational. In this case, both the primary and standby communication paths are *OFFLINE*.

    - No switchover can be performed.

⚠CAUTION

INCONSISTENT CONFIGURATION NOT DETECTED

Keep the primary communication path and standby communication path connected to the same controller.

Failure to follow these instructions can result in injury or equipment damage.

Triggering a Switchover

A switchover from the primary communication path to the standby communication path can be triggered in two ways:

- Automatically, in response to an interruption of the primary communication path (for example, by a send or receive detected error).

  NOTE: A send or receive detected error can arise from a request timeout due to a device communication latency, which cannot be differentiated from a communication interruption. To help avoid data loss, the last request is resent through the new primary communication path after a switchover. As a result, switching from primary to standby communication paths has no effect on client application variables.

- By writing to the *#SwitchPrimaryAddress Specific DataItem*, page 85.

  Writing a value operates a switchover from the primary to the standby communication path. A *Bad_Unexpected* service result code is generated in response to the write operation if the standby communication path is not configured or if it is *OFFLINE*..

Managing a Switchover

The switchover from one communication path to the other is managed as follows:

- If the standby communication path was *ONLINE* at the last connection verification, the switchover can be performed. The standby communication path becomes the new primary communication path.
- If the standby communication path was *OFFLINE* due to disconnection at the last verification, a switchover can be attempted. A switchover will be performed when access to the standby communication path is achieved, unless the original primary communication path is re-established and *ONLINE*.
- If the standby communication path was OFFLINE as a result of not passing the most recent consistency verification, no switchover can be performed.

OPC UA Server Redundancy

Overview

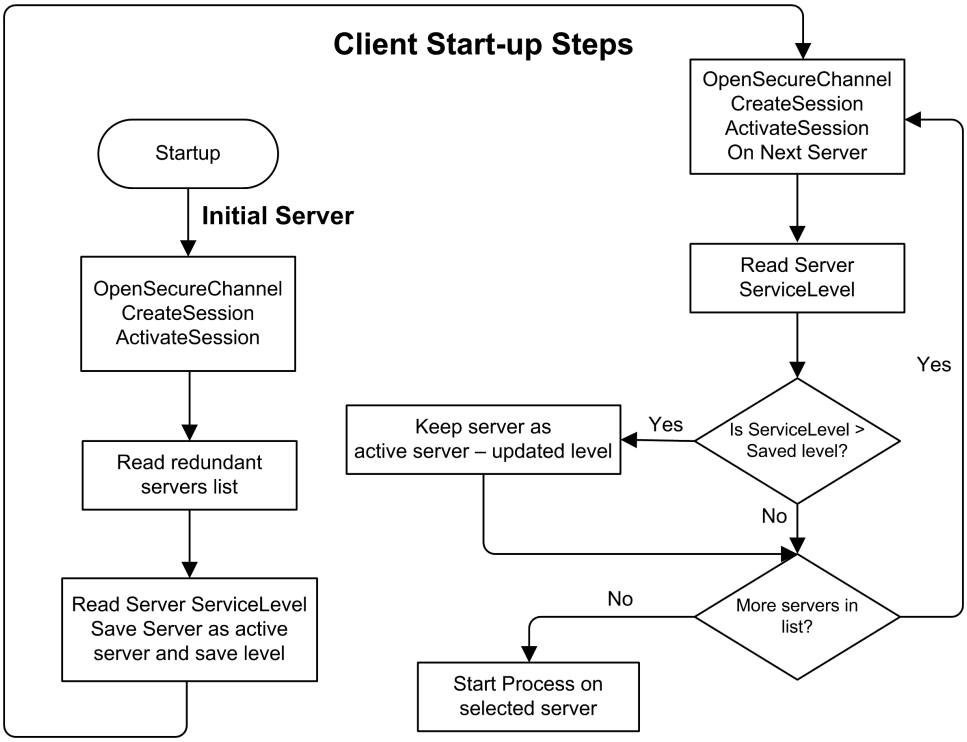
EcoStruxure OPC UA Server Expert supports non-transparent server redundancy with warm failover mode as defined in the *OPC Unified Architecture Specification Part 4, Services, Non-transparent Redundancy, Release 1.05*.

In warm failover mode, the backup or alternate server (or servers) can be active, but cannot connect to data points. Typically, this occurs in systems where the underlying devices are limited to a single connection. For example, the underlying devices such as controllers may have limited resources that permit only a single server connection. In this case, only one server at a time can consume data.

Initial Client Connection to a Redundant Server Set

The following flowchart gives an overview of the steps an OPC UA client typically performs when first connecting to a redundant server set.

NOTE: This flowchart does not cover all the possible detected error use cases.



NOTE: The redundant servers list must be configured in the OPC UA client in the same way it is implemented in EcoStruxure OPC UA Server Expert.

Supported Client Actions

The following tables indicate which client actions are supported in non-transparent redundancy with warm failover mode.

During initial connection, in addition to actions of active server:

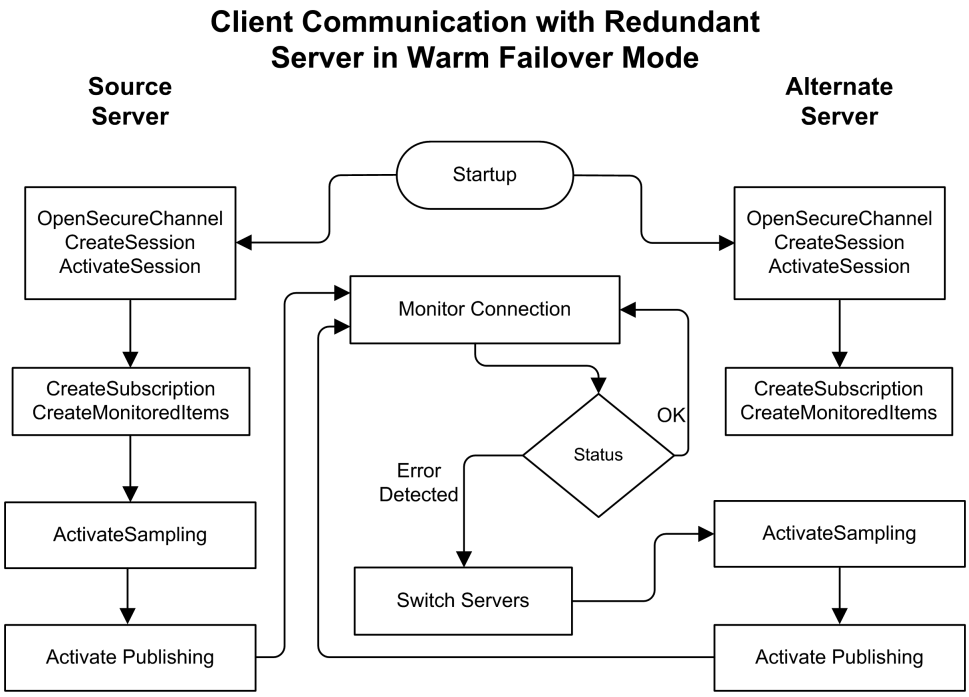
Client actions in warm failover mode	Supported
Connect to more than one OPC UA server	No
Create subscriptions and add monitored items	No
Activate sampling on the subscriptions	Yes
Activate publishing	Yes

At failover:

Client actions in warm failover mode	Supported
OpenSecureChannel to backup OPC UA server	No
CreateSession on backup OPC UA server	No
ActivateSession on backup OPC UA server	No
Create subscriptions and monitored items	No
Activate sampling on the subscriptions	Yes
Activate publishing	Yes

Server Communications

The following flowchart describes the steps a client performs when communicating with a server using warm failover mode.



EcoStruxure OPC UA Server Expert Configuration Tool

What's in This Part

Description

Using the EcoStruxure OPC UA Server Expert Configuration Tool

A user with administrator privileges can use the configuration tool to:

- Create an alias that represents a controller or another OPC UA client device.
- Configure properties for an alias.
- Configure global properties for the OPC UA server.

  NOTE: Configurations for an alias or for the OPC UA server are made statically and only take effect after the OPC UA server is stopped and restarted. Dynamically making changes while the OPC UA server is running is not supported.

Graphical User Interface

The configuration tool graphical user interface consists of the following elements:

- A menu bar with a toolbar beneath it.
- A navigation tree on the left-hand side with:
  - A **Devices** branch with child elements to create, page 50 and configure, page 53 device alias items.
  - A **Settings** branch with child elements to configure the OPC UA server, page 57.
- A display area on the right-hand side to configure settings associated with the item selected in the navigation tree.

Menu Commands

The configuration tool menu bar contains the following commands (which are also displayed on the toolbar):

Menu	Command	Description
File	**New Device Alias**	Creates a controller or other device alias.
	Open Archive	Restores the configuration from a saved backup file.
	Save Archive as	Saves the configuration to a file for storage.
	Save configuration	Saves changes, and applies them to the configuration when the OPC UA server restarts.
	Print Preview	Displays a preview of selected printing content.
	Print	Prints, or saves to a file, the OPC UA server settings.
Edit	**Copy**	Copies the selected device alias.
	Paste	Pastes the selected device alias.
	Delete	Deletes the selected device alias.
	Rename	Lets you rename the selected device alias.
View	—	Lets yous display the list of device alias items in different viewing modes.
Service(1)	**Start server as service**	Starts the *OPCUAServerExpert* service.
	Stop server	Stops the *OPCUAServerExpert* service.
Settings	—	Lets you select the available display language.
Help	—	Contains commands to open and search the EcoStruxure OPC UA Server Expert HTML help, view the license file, and display information about the EcoStruxure OPC UA Server Expert Configuration Tool.

(1) The menu is enabled when the OPC UA server is configured to operate as a Windows service, page 21.

Creating and Configuring a Device Alias

What's in This Chapter

Overview

This chapter describes how to use the EcoStruxure OPC UA Server Expert Configuration Tool to create and configure a device alias.

Managing Device Aliases

Creating a Device Alias

To create and configure a device alias using the EcoStruxure OPC UA Server Expert Configuration Tool, proceed as follows.

Step	Action
1	In the configuration tool menu bar, click **File > New Device Alias**. **Result**: A device alias is created with default settings.
2	In the **Devices > Device overview** window, assign the device alias a unique **Device name**.
3	Click the ellipsis button (...) for **Device address 1**. **Result**: The **Device address wizard** opens.
4	Configure the settings for **Device address 1** including the following, and click **OK**: • Network driver • Network type • Device address, page 51
5	In the **General** property section, click the ellipsis button (...) for **Symbol table file**. **Result**: A file selection dialog box opens.
6	Select an .XVM, page 72 or .CSV, page 86 symbol link file, and click **OK**.
7	Configure the remaining **General** and **Configuration** properties.
8	Click **File > Save Configuration** and review the messages that may be displayed.

Device Address Parameters

The EcoStruxure OPC UA Server Expert Configuration Tool enables you to configure the address of the following devices:

- **DIRECT** devices:
  - **ModbusSerial**
  - **TCP IP**
  - **USB**
  - **ModbusTCPGateway**
- **XWAY** devices:
  - **TCP IP**
  - **XWAY Gateway**

The following sections and parameters are available.

Section/parameter	Description
PLCs	Identifies the type of controller or device: • **UNITY / Control Expert**: For devices configured using EcoStruxure Control Expert (or Unity Pro). The *./U* suffix is added to the device alias address. • **MODBUS Device**: For other devices supporting the Modbus protocol. ◦ **Minimum commnucation profile**: When the check box is selected, it enables communication with frame length = 59 registers or 944 coils. The *./NQ* suffix is added to the device alias address. Otherwise, it increases the Modbus communication bandwidth, enabling communication with frame length = 125 registers or 2,000 coils. The *./Q* suffix is added to the device alias address. ◦ **Swap 32-bit Values**: Select the check box if the device encodes REAL or DWORD values in big-endian order. The *./NQW* suffix is added to the device alias address (or *./QW* if **Minimum commnucation profile** is cleared).
Adapter number	Applies to Modbus drivers and defines the instance of the driver to be used.
Instance number	There is one instance per driver installed. Generally, equal to 1. Each driver corresponds to a communication card in the computer.
Port number	This setting is available for the TCP/IP drivers. You can use it to define a TCP port other than the standard port number (502) to manage complex networking.
ALIAS ACCESS PATH	Displays the read-only alias string resulting from the selections made in this window.

The following additional parameters are available for **DIRECT** devices.

Parameter	Description
Data master/ Program master	Not supported.
PLC Node	Applies to Modbus Serial drivers and defines the server number of the device on the Modbus network.
Gateway	The **MODBUS index** that is available for **TCP IP** and **ModbusTCPGateway** drivers, page 52. It defines the server number of the device on the Modbus network. The **Gateway** check box applies to the **ModbusTCPGateway** driver. When it is selected, it defines the gateway device. You can set the **Max channels** and **Max pending** values. These values are automatically set on the devices configured with the same IP address or DNS.

You can configure the following parameters for an **XWAY** device address using the **TCP IP** driver.

Parameter	Description
Addressing levels	Addressing levels for XWAY addresses.
Network **Station** **Gate** **Index**	XWAY address. Depending on the selection made for **Addressing levels**, additional text boxes (without label) can be enabled. **Index** is the XWAY gateway number. To obtain an index number, create a virtual alias without associated symbol table file by using the **XWAY Gateway** driver. For more information, refer to **XWAY Gateway** Driver, page 53.
Through ETY (IP)	Use this setting with TSXETY110 and TSXETY120 communication modules. When the check box is selected, enter an IP address.

ModbusTCPGateway Driver

The **ModbusTCPGateway** driver enables you to configure a physical device functioning as a Modbus gateway, generally accessed with Modbus TCP/IP.

The gateway can be a communication bottleneck if server Modbus devices are independently configured. Create a virtual alias on behalf of the gateway that allows you to configure a shared communication channel.

Proceed as follows:

Step	Action
1	In the EcoStruxure OPC UA Server Expert Configuration Tool, create a device alias and click the ellipsis button (...).
2	In the **Device address wizard**, for the **ModbusTCPGateway** driver, select the **Gateway** check box.
3	Set the IP address of the physical gateway.
4	Configure each Modbus server alias address that is to be addressed through the gateway and set the same IP address as you set it for the **ModbusTCPGateway** driver.
5	Configure the **Max channels** and **Max pending** properties of the gateway alias with values that allow accessing each device. **NOTE:** The **Max channels** and **Max pending** values of each device alias addressed through the gateway are automatically set from the **Max channels** and **Max pending** properties of the gateway alias and are read-only.

XWAY Gateway Driver

The **XWAY Gateway** driver enables you to configure a set of XWAY devices configured with an inter-station XWAY address and accessed through the same XWAY device, functioning as a gateway.

The gateway can be a communication bottleneck if inter-station devices are independently configured. Create a virtual alias on behalf of the gateway that allows you to configure a shared communication channel.

Proceed as follows:

Step	Action
1	In the EcoStruxure OPC UA Server Expert Configuration Tool, create a device alias and click the ellipsis button (...).
2	In the **Device address wizard**, for the **XWAY Gateway** driver, set the **XWAY Gateway Index** check box to identify the gateway alias.
3	Configure each XWAY alias address that is to be addressed through the gateway and set the same XWAY gateway index as you set it for the **XWAY Gateway** driver.
4	Configure the **Max pending** property of the gateway alias with a value that allows accessing each device. **NOTE:** The **Max pending** value of each device alias addressed through the gateway is automatically set from the **Max pending** property of the gateway alias and is read-only.

Device Alias Properties

Device Alias Default Properties

The default values applied to a device alias are set in the **Devices > Default devices** navigation tree branch. If you edit these device alias settings in the **Default devices** branch, they will be applied to subsequently created device aliases.

Alternatively, you can edit device settings for a specific device, without changing default values, in the **Devices > Device overview** window.

Editing Properties for a Selected Device Alias

Select **Devices > Device overview** to display the collection of existing device aliases. Select a device alias in the top right-hand pane to display its configured properties in the lower pane.

The following properties/categories can be configured for a selected device alias:

- **Device name**, **Device address 1**, and **Device address 2**
- **General**
- **Communication information**:
  - **Adjustment Information**
  - **Enhanced adjustment information**

Name and Address Properties

The following table describes the device alias name and address properties. They must be unique for each device alias and have no default settings.

Property	Description
Device name	Name of the device.
Device address 1	The required main address, page 51 of the device, and the only address if **Device address 2** is not configured. This value is input using the **Device address wizard**.
Device address 2	Optional. An alternate address, page 51 of the device. It is used to create a second communication path between EcoStruxure OPC UA Server Expert and the OPC UA client device (typically a controller) for controller link redundancy, page 41. This value is input using the **Device address wizard**.

General Properties

On creation of a device alias, these non-unique, general properties are set to the default values set in the **Devices > Default devices** window. You can configure the following properties for a device alias.

Property	Description
Topological objects	Optional. Select the **Populate** check box to enable the population of State RAM topological objects for a device configured in EcoStruxure Control Expert. Default: Cleared
Symbol table file	Optional. If required, click the ellipsis (...) button, navigate to, and select a symbol file to be associated with the device alias. Two file types are supported: • .XVM, page 72: Exported from EcoStruxure Control Expert, this file provides access to the symbols for the variables of this device. • .CSV, page 86: If the device is configured as a Modbus device.
PLC Embedded Data	• **Using Data Dictionary**: Optional. If selected, it embeds the EcoStruxure Control Expert data dictionary that provides access to the symbols for the variables of this device. It is effective only if the **Data dictionary** option in **Tools > Project Settings... > General > PLC embedded data** of EcoStruxure Control Expert is selected. Default: Selected. • **No Communication Break**: Optional. If selected, in the event of an application build change, it helps supports uninterrupted operations by permitting the OPC UA server to preload and then switch to a new data dictionary. It is effective only if the **Data dictionary > Preload on build changes** option in **Tools > Project Settings... > General > PLC embedded data** of EcoStruxure Control Expert is selected. Default: Cleared
Preload settings	Select one of the following symbol loading and device connection options that are performed when the OPC UA server starts: • **No Preload**: No device connection nor symbol access is performed. • **Symbol table**: Symbol loading is performed; device connection is performed only if **PLC Embedded Data > Using Data Dictionary** is selected. • **Device** (default): Symbol loading and device connection are performed.
Dynamic consistency	**New Symbol Detection**: Optional. If selected, the OPC UA server detects and populates new symbols created in the EcoStruxure Control Expert application. It is effective only if the **PLC Embedded Data > Using Data Dictionary** check box is selected. Default: Cleared
Option	• **Simulated**: Optional. If selected, no physical connection is made to the device. The variables values are simulated by the server, according to the server simulation settings, page 58. Default: Cleared • **Read Only**: Optional. Defines the read/write access to the property: ◦ If selected, variables relating to the device have read-only access rights. ◦ If cleared, variables relating to the device have read/write access rights. Default: Cleared
Comment	A free-form text field where you can enter a comment.

Adjustment information Properties

You can configure the following properties for a device alias.

Property	Description
Max channels	The number of channels allocated to the device. Default values: • 1 for USB devices • 4 for non-USB devices (minimum value)
Max pending	The maximum number of requests that can be sent in parallel. The default is 0, which sets the value to the **Max channels** value. **NOTE:** For the Modbus TCP/IP protocol, up to 4 times **Max channels** is supported.
Device timeout (ms)	A delay used to manage the status transition of the device. Range: 3,000...32,767 ms. Default: 5,000 ms. **NOTE:** It is a good practice to use a value of at least 3 times **Frame timeout (ms)**. The value can be monitored using the *#PLCQualStatus Specific DataItem*, page 84.
Frame timeout (ms)	The permissible delay (in milliseconds) between request and answer. The maximum length of time the OPC UA server will wait for an answer from a device after sending a request. Range 1,000...10,900 ms.to a maximum of one-third the configured **Device timeout (ms)**. Default: 1,000 ms.

Enhanced Adjustment Information Properties

If the OPC UA server detects a communication error or it is disconnected from the device, you can configure the following optional properties.

Property	Description	Range	Default value
Reconnection retry number	The number of times the OPC UA server attempts to resend an unsuccessful request to the device. For more information, refer to **Network** Tab, page 23.	0...3	0
Disconnection timeout (mn)	The duration (in minutes) for which the network connection is disabled if the device is disconnected when the OPC UA server starts or during operation. **NOTE:** If the device is still disconnected after the timeout expired, the timeout is renewed.	0...1,440	0

NOTE: These parameters do not apply to Modbus TCP gateway and XWAY gateway devices, but they apply to devices connected behind a Modbus TCP or XWAY gateway.

NOTICE
DEVICE COMMUNICATION IS MAINTAINED DISABLED Ensure that the **Disconnection timeout (mn)** value is adapted to your system. **Failure to follow these instructions can result in equipment damage.**

Configuring Global OPC UA Server Settings

What's in This Chapter

Overview

This chapter describes non-unique configuration settings for device aliases.

Diagnostic Settings

Configuring Log and Snapshot Files

Use the **Settings > Diagnostic** window to configure diagnostic log and snapshot features. Both log and snapshot files contain the information displayed in these locations:

- The **Diagnostics** tab when the OPC UA server is running as a standalone application, page 23.
- The **Notifications** window when the server is running as a Windows service, page 21.

Log Trace Section

You can configure the following properties.

Property	Description
Overwrite logs	Overwrites the log files you enabled each time the OPC UA server starts. Default: Cleared
Application diagnostic	Select the check box to enable the *Application diagnostic* log file, then configure the following: • Log file name and path: Click the ellipsis button (...) to select a log file (.txt)(1). • **Max Size (MB):** Set the maximum log file size. Range: 500...4,000 MB Default: 1,000 MB When the log file exceeds the set size, it is discarded. Default: Cleared
Communication diagnostic	Select the check box to enable the *Communication diagnostic* log file, then configure the following: • Log file name and path: Click the ellipsis button (...) to select a log file (.txt)(1). • **Max Size (MB):** Set the maximum log file size. Range: 500...4,000 MB Default: 1,000 MB When the log file exceeds the set size, its content is overwritten. Default: Cleared
Snapshot	Select the check box to enable the *Snapshot* log file, then configure the following: • Log file name and path: Click the ellipsis button (...) to select a log file (.txt)(1). • **Max Size (MB):** Set the maximum log file size. Range: 500...4,000 MB Default: 1,000 MB When the log file exceeds the set size, it is discarded. Default: Cleared **NOTE:** If the OPC UA server is running in Standalone mode, the configured path is used by the **Menu > Save snapshot** command.
Snapshot period (s)	Rate (in seconds) at which the snapshot file is updated. The property is enabled if you enable the *Snapshot* log trace file and set to 60 seconds. Range: 10...120 in 1 second increments Default: Disabled
Verbose mode	Select the check box to enable the retrieval of information in the *Application diagnostic* and *Communication diagnostic* log files. Default: Cleared
(1) The default path is *C:\Users\<username>\AppData\Local\Temp*. When EcoStruxure OPC UA Server Expert is configured to run as a Windows service, it cannot write to files in this folder. Select a log file that is not under the Windows *C:\Users* folder.	

Simulation Settings

Overview

Use the **Settings > Simulation** window to enable simulation mode for OPC UA clients connected to the UPC UA server, and to tune the values generated by the server during simulated operations.

Simulation Settings Section

You can configure the following properties.

Property	Description
Simulator mode	Defines how simulated operations are performed: • Selected: The OPC UA server s generates simulated operational values for connected OPC UA clients, according to the settings made in this window. • Cleared: The OPC UA server only generates simulated operational values for device alias items with the **Option > Simulated** property selected. For more information, refer to the **General** properties section of the **Device overview** window, page 55. Default: Cleared
Initial values	Defines how simulated values are initialized: • **Zero**: Initialized to zero. • **Random** (default): Initialized to randomly generated values.
Max value	Select the maximum value for a simulated numeric variable. Range 0...32,767. Default: 100 **NOTE:** A variable value is increased as follows: • Boolean variables are inverted. • Floating point variable values are increased by 0.3. • A numeric value returns to 0 when the maximum value is reached.
Notification adjust	Indicates the probability of a variable value being adjusted at each cycle, according to the following formula: Probability = 1/N, where the value of N is in the range 1...10. **NOTE:** The update rate of simulated variables is determined by the value of N as follows: • N=1: Simulated variables are updated at the same monitored item sampling interval a read attribute operation is performed. • 1<N≤10: At each period there is one chance in N that the simulated variable is updated. There is no correlation between different declared variables; their values change individually. The decrease in the probability value (increase of N) reduces the number of notifications, and thus reduces the load on the CPU.

PLC Software Settings

Overview

Use the **Settings > PLC Software** window to set the period for the cyclic consistency verifications of the symbol file, and the data type mapping of the data types used by the EcoStruxure Control Expert software.

Dynamic Consistency Section

You can configure the following property.

Property	Description
Cyclic consistency check rate (s)	The period (in seconds) for the symbol database (only an XVM symbol link file or the data dictionary) consistency verification. An inconsistency triggers a database reload and an update of all variables definition. A reload is also triggered by a communication detected error on inconsistent items. Default: 10 s

Project Files Options Section

You can configure the following properties.

<table>
<tr><th>Property</th><th>Description</th></tr>
<tr><td rowspan="15">Unity / Control Expert symbols</td><td colspan="3">If the Use native types check box is selected, variable instances linked to EcoStruxure Control Expert String, DATE, TOD, DT and TIME data types are converted to the OPC UA built-in STRING data type, in accordance with the IEC1131-3 representation.

Default: Selected</td></tr>
<tr><td colspan="3">If cleared, the following data type conversion occurs:</td></tr>
<tr><td>EcoStruxure Control Expert Type</td><td colspan="2">OPC UA data type</td></tr>
<tr><td>STRING</td><td colspan="2">BYTE ARRAY</td></tr>
<tr><td>DATE</td><td colspan="2">UINT32</td></tr>
<tr><td>TOD</td><td colspan="2">UINT32</td></tr>
<tr><td>DT</td><td colspan="2">DOUBLE</td></tr>
<tr><td>TIME</td><td colspan="2">UINT32</td></tr>
<tr><td colspan="3">Example 1:

If Use native types is selected, the following conversion examples apply.</td></tr>
<tr><td>EcoStruxure Control Expert Type</td><td>Value sample</td><td>OPC UA built-in type</td></tr>
<tr><td>DATE</td><td>D#2017-05-17</td><td>STRING</td></tr>
<tr><td>TOD</td><td>TOD#07:44:01</td><td>STRING</td></tr>
<tr><td>DT</td><td>DT#2017-05-17-07:44:01</td><td>STRING</td></tr>
<tr><td>TIME</td><td>T#07h44m01s100ms</td><td>STRING</td></tr>
<tr><td colspan="3">Example 2:

If Use native types is cleared, the following conversion examples apply.</td></tr>
<tr><td>EcoStruxure Control Expert Type</td><td>Value sample</td><td>OPC UA data type</td></tr>
<tr><td>DATE</td><td>0x20170517</td><td>UINT32</td></tr>
<tr><td>TOD</td><td>0x07440100</td><td>UINT32</td></tr>
<tr><td>DT</td><td>4.2922532071416873e-154</td><td>DOUBLE</td></tr>
<tr><td>TIME</td><td>27841100</td><td>UINT32</td></tr>
<tr><td>Unity / Control Expert DATE, DOT and DT instances</td><td colspan="2">If Use native types is selected, selecting the Use regional settings check box enables formatting the STRING datatype attribute of variable instances linked to EcoStruxure Control Expert DATE, TOD and DT datatypes according to configured Windows regional settings.

Default: Cleared

Example:

DATE can be formatted:
• M/d/yyyy
• MM/dd/yyyy
• yyyy-MM-dd
Where:
• MM = Month with leading 0
• M = Month without leading 0
• dd = Day with leading zero
• d = Day without leading zero
• yyyy = 4-digit year
TIME can be formatted:
• H:mm:ss
• HH:mm:ss
Where:
• HH = 24-hour display with leading 0
• H = 24-hour display without leading 0
• mm = Minute with leading zero
• m = Minute without leading zero</td></tr>
</table>

Property	Description
Unity / Control Expert TIME instances	If **Use native types** is selected, selecting the **Display underscore** check box enables formatting EcoStruxure Control Expert variable instances of the STRING data type by displaying the underscore. Default: Cleared
Byte array management	When the **Manage as ByteString**, page 78 check box is selected, the *DataItem* data type *Byte array* is converted to the *ByteString* data type. Default: Cleared

Communication Settings

Overview

Use the **Settings > Communication** window to configure communication settings for EcoStruxure OPC UA Server Expert.

Communication Settings Section

You can configure the following properties.

Property	Description
Monitored item sampling overrun policy	Defines the policy in case of communication overload, page 19. • • if **Bad statusCode** is selected (default), it sets the *MonitoredItem DataValue statusCode* to *Bad_NoCommunication* = 8031 0000 hex. • if **Sampling interval adapt** is selected, it updates the *MonitoredItem DataValue* at the present sampling interval. For more information on *DataValue*, refer to *DataValue* Type, page 70. **NOTE:** If the communication overload persists, tune the device configuration.
Modbus serial driver	If a device is in timeout on a Serial Modbus network due to a missing device, or if **Frame timeout (ms)** is greater than the configured frame timeout value, the OPC UA Server sends commands to the serial Modbus driver to adapt the baud rate that generates the unexpected workload. If the **Disable driver configuration on timeout** check box is selected, a re-configuration is not performed. Default: Cleared
Alias with data dictionary set	**Allow disconnection at server startup**, page 85. Default: Cleared
#DeviceServiceLevel computation	Defines whether extended rules are taken into account for the calculation of *#DeviceServiceLevel*. • if **Extended rules** is selected, it includes standalone and redundant controller states Run, Stop, and Wait. if, in addition, **Always apply Standalone CPU rules** is selected, it uses standalone controller states for redundant controllers. • if **Extended Rules** is cleared, it uses the default rules without controller state. For more information on the calculation of *#DeviceServiceLevel*, refer to Specific *DataItems*, page 80. Default: Cleared

Options Settings

Overview

Use the settings in the **Settings > Options** window to configure optional settings for EcoStruxure OPC UA Server Expert.

Features Section

You can configure the following properties.

Property	Description
DNS Scanning TCP/IP	When the check box is selected, the OPC UA server uses the DNS to resolve device names for OPC UA clients (for example, *Controller_Pump* instead of 192.168.2.10). Default: Selected
Enable service mode	Defines whether the OPC UA server runs as a Windows service, page 29. Default: Cleared
Processor affinity	Defines a 32-bit vector where each bit set to 1 represents a logical processor on which the tasks of the OPC UA server process can run. Bit 0 represents processor 0, bit 1 represents processor 1, and so on. The bit vector is logically combined using the AND operator with the system bit vector ((2^n) -1, where *n* is the number of logical processors). A null bit vector (value = 0) is set to system bit vector. The value is displayed in hexadecimal. Default: 0
DA Address Space	When the **Flat mode** check box is selected, node references are skipped so that the Data Access (DA) address space *Built* state can be achieved faster. The maximum number of nodes that can be published is increased, page 16. Browse services (*Browse*, *BrowseNext* and *TranslateBrowsePathsToNodeIds*) invoked on the DA address space are not supported. Default: Cleared
	When the **Case sensitive** check box is selected, DA address space *NodeIds* are managed with case sensitivity. Default: Selected

Security Settings

Overview

Use the **Settings > Security** window to configure OPC UA security settings for EcoStruxure OPC UA Server Expert.

Settings

You can configure the following properties.

Property	Description
SecurityPolicy	When the **Allow "None"** check box is selected, an OPC UA client can create a session or open a channel using an OPC UA client application instance certificate that is not validated, page 34. Default: Cleared
Anonymous user token	When the **Allow** check box is selected, an OPC UA client can create an anonymous session (i.e. a session where the user identity token cannot be validated, page 35). Default: Cleared
X509 user token	When the **Allow** check box is selected, an OPC UA client can create a session using an X509 identity token. Default: Selected
User authentication	When the **Activate** check box is selected, the OPC UA server applies user authentication and authorization, page 36. Default: Selected
Certificate revocation check	When the **Enable** check box is selected, the OPC UA server maintains a Certificate Revocation List (CRL) and performs an online CRL verification of OPC UA client certificates, page 33. Default: Cleared **Timeout(s)**: If the CRL verification is enabled, it defines the time period (in seconds) for verification completion. Range 15...120 Default: 60
Rejected certificate store	If selected, a read/write file is created that stores a list of certificates rejected by the validation policy, page 33. You can select the file storage folder. Default: Cleared

Connection Settings

Use the **Settings > Connection** window to configure connection settings for EcoStruxure OPC UA Server Expert.

Connection Settings Section

Settings

You can configure the following settings.

Property	Description
Endpoint URLs	Defines whether legacy paths for the server endpoint URLs are supported. • When the **Use legacy path** check box is selected, the following endpoint URLs are supported: ◦ *opc.tcp://<server_computer_name>:<listening_port>/UA/OFS-UA* ◦ *opc.tcp://<server_IPAddress>:<listening_port>//UA/OFS-UA* • When the **Use legacy path** check box is cleared, the following endpoint URLs are supported: ◦ *opc.tcp://<server_computer_name>:<listening_port>/OPCUAServerExpert* ◦ *opc.tcp://<server_IPAddress>:<listening_port>/OPCUAServerExpert* Where: • <*server_computer_name*> is the name of the computer running OPC UA server. • <*server_IPAddress*> is the IPv4 or IPv6 address of the computer running the OPC UA server (excluding IPv4 local host and IPv6 link local address). • <*listening_port*> is set to 49152 by default, but can be configured in the range 49152...65535. The value 4840 is also supported. Default: Cleared
TCP listening port	Configures the TCP listening port of the OPC UA server in the range 49152...65535 or with the alternate value 4840. The listening port is part of the OPC UA server URL. Default: 49152 **NOTE:** You can click **Check Availability** to verify whether the port you configured conflicts with an already used port. If a conflict is detected, the next available port is identified. The availability verification only indicates whether a port is available. It does not identify the application that is using a port. Close the OPC UA server before performing the verification.

Hostname Section

Overview

In certain applications, it may be required to restrict the hostname list published through server endpoints. Without restriction, in case of a hostname modification (addition or removal of an IP address) on OPC UA Server certificate renewal, the updated server endpoint list may include unwanted endpoints.

In the **Hostname** section of the **Connection** window, you can restrict the hostname list to a fixed list of known endpoints.

A hostname can refer to either the computer hostname or the computer IP address.

NOTE: Hostnames starting with fe80: or 127, and IP address 90.0.0.2 are not taken into account.

NOTE: The local machine hostname is automatically included in the hostname list.

Settings

You can configure the following settings.

Property	Description
Dynamic mode	When the **Allow** check box is selected, the dynamic mode is enabled; otherwise, EcoStruxure OPC UA Server Expert operates in static mode. Default: Selected Buttons: • **Refresh**: Updates the hostname list and display the present system hostnames (removing discarded hostnames and adding new hostnames). • **Configure**: Opens the **Configure IP Addresses** dialog box. **NOTE:** The **Configure** button is disabled in dynamic mode.

Dynamic Mode

Use dynamic mode to maintain the consistency between server endpoints URLs and system hostnames.

In dynamic mode:

- The **Hostname** section initially displays the hostnames embedded in the OPC UA Server certificate. It is consistent with the self-signed server certificate installed during setup or after a certificate renewal, page 35.
- Using the *renewcertificate* tool updates the self-signed OPC UA Server certificate and CSR with the present system hostnames. For more information, refer to Certificate Renewal, page 35.

  NOTE: If the collection of system hostnames has changed, discarded hostnames are displayed in red.

Static Mode

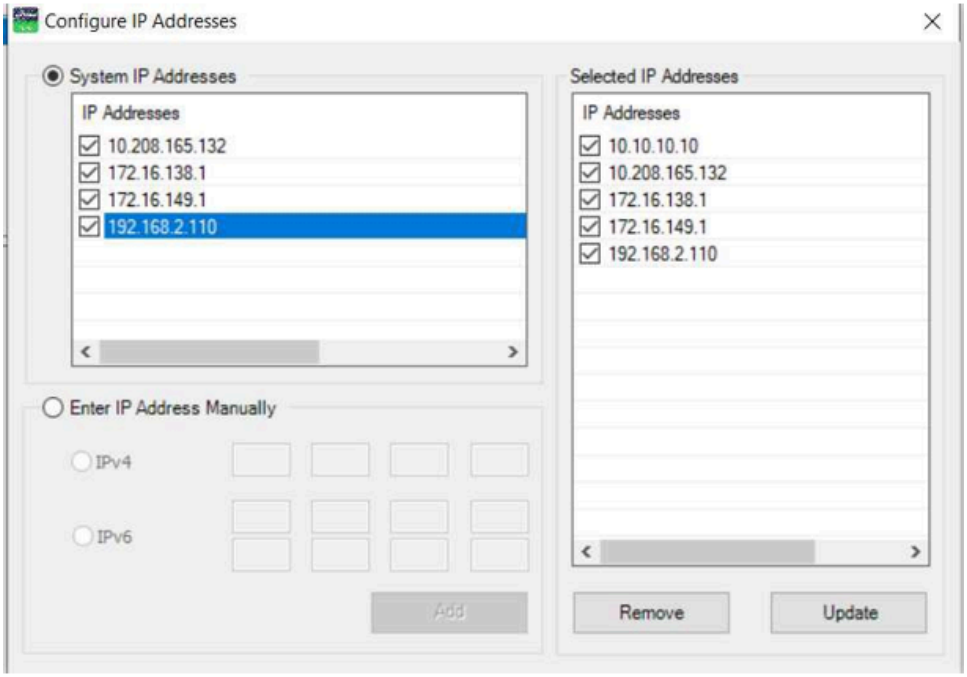
Use static mode to manage a fixed configured list of server endpoint URLs ignoring system hostname modifications, and to configure a hostname.

In static mode:

- The **Hostname** section displays the configured hostname list. This list is saved in the present configuration and in archives. When you switch between dynamic and static mode, your configuration is retained.
- Using the *renewcertificate* tool updates the self-signed OPC UA Server certificate and CSR with the hostnames you configured. For more information, refer to Certificate Renewal, page 35.

  NOTE: Hostnames that are not part of the present system hostnames (discarded hostnames or manually configured hostnames) are displayed in red.

The following figures shows an example of the **Configure IP Addresses** dialog box.



Section	Description
System IP Addresses	Displays the system hostnames.
Enter IP Addresses Manually	Lets you manually add an IPv4 or IPv6 IP address the hostname list.
Selected IP Addresses	Displays the hostnames that are used if you click **Update**, page 68. It initially displays the hostnames that are displayed in the **Hostname** section.

Adding IP Addresses

You can add IP addresses to the hostname list in the following ways:

- By selecting an IP address from the **System IP Addresses** list. The address is automatically added to **Selected IP Addresses**.
- By manually adding an IP address. Use this method to configure an IP address that is not part of the collection of system hostnames (for example, a Network Address Translation (NAT) address).

To manually add an IP address, proceed as follows.

Step	Action
1	Select **Enter IP Address Manually**.
2	Select **IPv4** or **IPv6**.
3	Enter the IP address. **NOTE:** For IPv6 addresses, the upper 4 text boxes are the first four 2-byte groups.
4	Click **Add**. **Result**: The address is added to **Selected IP Addresses**.

NOTE: You cannot add IP address 90.0.0.2 or addresses starting with fe80: or 127.

Removing an IP Address

You can remove an IP address from the **Selected IP Addresses** list in two ways:

- Clearing its check box in **System IP Addresses**.
- Clearing its check box in **Selected IP Addresses** and clicking **Remove**.

Updating the Hostname List

To update the hostname list after adding or removing IP addresses, click **Update**.

Result: The **Configure IP Addresses** dialog box closes and the IP addresses displayed in the **Hostname** section are updated.

NOTE: If you update the hostname list and save your configuration changes, it automatically runs the *renewcertificate* tool and you must re-deploy the renewed OPC UA Server self-signed certificate or CSR. For more information, For more information, refer to Certificate Deployment, page 31.

NOTE: You cannot update the hostname list if **Selected IP Addresses** contains an IP address with its check box cleared.

Cancelling Modifications

To cancel your modifications, click the Close button of the **Configure IP Addresses** dialog box.

Result: Your modifications are discarded.

NOTE: You cannot cancel modifications you applied by clicking the **Update** button.

OPC UA Information Model

What's in This Part

Overview

This part describes the OPC UA information model, with emphasis on the OPC UA server address space.

OPC UA Data Model

What's in This Chapter

Information Model

DataValue Type

As defined in *OPC UA Part 4: Services* the *DataValue* type is used to transport a *Variable* value and a timestamp in both the *Read* service result and the *DataChangeNotification* data that is used for *MonitoredItems*. In EcoStruxure OPC UA Server Expert, the *sourceTimestamp* reflects the timestamp that is applied when a value is received from the controller or device (real-time communication channel).

Namespace Index References

The following Namespace URIs (published in the *NamespaceArray* variable as defined in the *OPC Unified Architecture Specification Part 5: Information Model, Release 1.04 (November 22, 2017)*, for the associated *NamespaceIndex* values as defined in *OPC Unified Architecture Specification Part 3: Address Space Model, Release 1.04 (November 22, 2017)* are supported by EcoStruxure OPC UA Server Expert.

NamespaceIndex	**URI**	**Usage**
0	http://opcfoundation.org/UA/	OPC UA Namespace
1	urn:localhost:Schneider Electric: OPCUAServerExpert	Local server
2	https://schneider-electric.com	OPC UA server data access
3	http://opcfoundation.org/UA/Diagnostics	OPC UA diagnostics generated by EcoStruxure OPC UA Server Expert

OPC UA Server Data Access Address Space

What's in This Chapter

OPC UA Server Modeling Elements

DataItem, DA Root, and Alias Models

OPC UA Node Modeling

A *DataItem* designates an instance of an OPC UA variable node, with *hasTypeDefinition ReferenceType* targeting *DataitemType* as defined in the *OPC UA Specifications part 8: Data Access*.

A *Folder* designates an instance of an OPC UA Object node, with *hasTypeDefinition* targeting *FolderType* as defined in the *OPC UA Specifications part 5: Information Model*.

The following notation is used to qualify an OPC UA node:

<template>=<UAType>,<NodeId>,<DisplayName>,<TargetReference>

Where:

- *<template>* designates a template name for the node.
- *<UAType>* is the target of a *hasTypeDefinition ReferenceType*. Possible values include:
  - *"Folder"*
  - *"DataItem"*
- *<TargetReference>* designates a node template which is the (0..*) target of *Organizes ReferenceType*, where *<template>* is the *SourceNode*.

Other patterns used to set the OPC UA node attributes include:

Node attribute	Value
Node Type	String
NodeId Identifier	0:*<NodeId>*
BrowseName	2:*<DisplayName>*
DisplayName	*<DisplayName>*

DA Root and Alias Modeling

A root *Folder* node is published as *<ROOT_DA>*="*Folder*", "","*DA*",*<Alias>*

An alias configured in the EcoStruxure OPC UA Server Expert Configuration Tool with the name *<AliasName>* is structured as follows:

<Alias>="*Folder*",*<AliasName>*,*<AliasName>*,*<EntityName>*

Examples of *<EntityName>* are described in the following topics and include *DataItem*, *Mapping*, and *Specific*.

EcoStruxure Control Expert Address Space

Overview

This section describes the EcoStruxure Control Expert address space published in the OPC UA server.

Linking the OPC UA Server to EcoStruxure Control Expert Symbols

Accessing EcoStruxure Control Expert Address Space

The EcoStruxure Control Expert address space is published by translating the collection of variables (or symbols) associated with each controller and associating them with an OPC UA server alias created in EcoStruxure OPC UA Server Expert Configuration Tool. You can associate controller variable data with the OPC UA server in two ways:

- Embedding the controller data dictionary in the OPC UA server.
- Linking an XVM symbol link file containing controller variables to the OPC UA server.

  NOTE: The *NamespaceIndex* for all *NodeIds* published in the EcoStruxure OPC UA Server Expert address space, page 70 and related to data access is *2*.

For more information on creating and configuring an alias, refer to Managing Device Aliases, page 50.

Verification of the Symbol Link Consistency

On symbol link loading and modification, the OPC UA server verifies the consistency between linked symbols and variables in the controller application. On symbol link modification, the address space is verified for consistency, as follows:

- For modifications that do not change the controller application symbols layout (modification of the application, or new symbol creation), the symbol link is not reloaded.

  NOTE: New symbol creation is verifed only if the **Dynamic consistency > New Symbol Detection** option is set as a General Property of the device alias, page 55, and applies only to an embedded data dictionary symbol link.
- For modifications that modify the controller application symbols layout (for example, symbol removal, symbol property edits, or symbol internal memory allocation modification), the symbol link is reloaded and the address space is updated.

Until the address space update is completed:

- The *OPC UA View Service Set* is not available.
- Controller communication required to fulfill *OPC UA Attribute Service Set* and *OPC UA MonitoredItem Service Set* is interrupted.

  NOTE: You can take steps to help avoid a communication interruption where the symbol link is based on an embedded data dictionary, and preloading of the edited data dictionary is configured. For more information, refer to Configuring No OPC UA Server Communication Break During Data Dictionary Modification, page 74.

For a symbol link inconsistency based on an imported XVM symbol link file, communication (the data access status) with the controller is disabled, page 41.

The following table describes the possible inconsistency status conditions, displayed as *Consistency Status* (in the **Network tab** of the OPC UA server) when the server operates as a standalone application.

Consistency Status	**Description**
Critical	The XVM file cannot be loaded.
NotName	The application names in the XVM file and the controller application are different.
NotStamp	The controller application symbols layout in the XVM file and the controller are different.

Linking OPC UA Server Symbols to an Exported XVM File

To produce an XVM file that can be associated with a device alias by the OPC UA server, do the following:

- Configure EcoStruxure Control Expert to save the XVM file in the same location as specified for the **Symbol table file** property in the device alias **General** category, page 55.
- Export the XVM file in EcoStruxure Control Expert.

To configure EcoStruxure Control Expert for XVM file generation, proceed as follows.

Step	**Action**
1	In EcoStruxure Control Expert, click **Tools > Project Settings...**. **Result**: The **Project Settings** dialog box opens.
2	Click **Project Settings > General > Project autosaving on download**, click **save XVM**, and then click **OK**.
3	Click **Tools > Options...**. **Result**: The **Options Management** dialog box opens.
4	Click **Options > General > Paths**, and in the **XVM path** field, enter the same path as it is configured for the **Symbol table file** property of the device alias associated with this controller in the EcoStruxure OPC UA Server Expert Configuration Tool.
5	Click **OK**.

To export the XVM file, proceed as follows.

Step	**Action**
1	In EcoStruxure Control Expert, open the application.
2	In the **Project Browser**, right-click **Variables & FB Instances**, then select **Export...**. **Result**: The **Export** dialog box opens.
3	In the **Save as type** field, click **Data mapping (*.XVM)** and input in a **File name**.
4	Click **Export**.

Embedding the Controller Data Dictionary in the OPC UA Server

To embed the controller data dictionary variables in the OPC UA server, enable the data dictionary in EcoStruxure Control Expert. Optionally, you can limit the collection of embedded variables to only those variables configured for HMI display.

To embed the controller data dictionary in the OPC UA server, proceed as follows.

Step	Action
1	In EcoStruxure Control Expert, click **Tools > Project Settings...**. **Result**: The **Project Settings** dialog box opens.
2	Click **Project Settings > General > PLC embedded data**, then click **Data dictionary**.
3	Optional: In the **Project Settings > General > PLC embedded data** screen, with **Data dictionary** selected, select **Only HMI variables.**
4	Click **OK**.

Configuring No OPC UA Server Communication Break During Data Dictionary Modification

A change in the data dictionary symbols layout (for example, symbol removal, symbol property edits, or symbol internal memory allocation modification) can cause a communication interruption between the OPC UA server and OPC UA client while the server undergoes a symbol link resynchronization during which the variable definitions are updated.

You can help avoid this communication interruption by allowing the OPC UA server to preload the edited data dictionary before it is applied to the controller application. To configure a data dictionary preload, proceed as follows.

Step	Action
1	In EcoStruxure Control Expert, click **Tools > Project Settings...**. **Result**: The **Project Settings** dialog box opens.
2	Click **Project Settings > General > PLC embedded data**, then with **Data dictionary** previously selected, click **Preload on build changes**.
3	In the **Effective Build changes time-out (sec)** field, enter a value that exceeds the time required by the OPC UA server to update the variables definition.
4	Click **OK** to close the dialog box.

Supported Data Types

Supported EcoStruxure Control Expert Variable Types

EcoStruxure OPC UA Server Expert provides access to symbolized EcoStruxure Control Expert variable instances of the following data type categories:

- Elementary Data Type (EDT)
- Derived Data Type (DDT)
- Arrays (single or multiple dimension)
- Device Derived Data Type (DDDT)
- Elementary Function Block (EFB) and Derived Function Block (DFB) inputs, outputs, inputs/outputs, and public parameters of any of the preceding types.

  NOTE: The *DataValue statusCode* of a forced EBOOL instance successfully accessed is *Good_LocalOverride* = 00960000 hex instead of *Good* = 00000000 hex. For more information on *DataValue*, refer to *DataValue* Type, page 70.

  NOTE: Generic data types (ANY_ARRAY x) are not supported.

Discoverable Variable Instances

The OPC UA server supports instances of variables declared in the following EcoStruxure Control Expert locations:

- Global namespace
- Process namespace
- Safety-related namespace (read-only)
- Program Units, including public and external variables, inputs and outputs EDT parameters

Discoverable Objects

The following objects are supported:

- State RAM topological objects, page 79 (%MW, %M, %S, %SW, %KW, %I, and %IW). They can be populated by alias configuration.
- Local I/O objects configured with topological (as opposed to DDDT) address: Only symbolized I/O objects are populated.
- Local I/O objects configured with IODDT instances: Only EcoStruxure Control Expert referencing IODDT fields are populated.
- Instances of DFBs and Program Units private members, including nested DFBs.

  NOTE: You must configure the application data dictionary in EcoStruxure Control Expert by clicking **Tools > Project Settings > PLC embedded data** and then, clicking **Include DFB/PU private variables**.

Variable Data Type Conversion

The OPC UA server can discover and convert to OPC UA data types the following EDTs supported by the EcoStruxure Control Expert data logic model.

EcoStruxure Control Expert EDT	OPC UA built-in data type
BOOL	Boolean
EBOOL	Boolean
INT	INT16
DINT	INT32
UINT	UINT16
UDINT	UINT32
REAL	FLOAT
BYTE	BYTE
WORD	UINT16
DWORD	UINT32
DATE	The display of time and string data types is determined by the configuration of **Use native types** and **Use regional settings** in the **PLC Software** window for the device alias. For more information, refer to **PLC Software** Settings, page 59.
TIME	
TOD	
DT	
STRING	

EcoStruxure Control Expert Variable Modeling

General Variable Structure

EcoStruxure Control Expert variable instances, except Elementary Function Blocks (EFBs), Derived Function Blocks (DFBs), and program units are structured as follows:

<EntityName>=*"DataItem"*,<*VariableNameBrowsePath*>,<FinalName>, <VariableChild>

Where:

- <VariableChild >=*"DataItem"*,<*VariableNameBrowsePath*>,<FinalName>, <VariableChild>
- <VariableNameBrowsePath>=<AliasName>!<FullVariableBrowsePath>
- <FullVariableBrowsePath> describes an EDT instance with name <FinalName> or the full browse path used to access a final element of a DDT or ARRAY instance for which the final name is <FinalName>.
  - The browse path to access a DDT element <DDTElement> of a DDT instance <DDTInstance> is formed with <DDTInstance>.<DDTElement> syntax.
  - The browse path to access an ARRAY element at index <Index> of an ARRAY instance <ArrayInstance> is formed with <ArrayInstance>[<Index>]

For more information on OPC UA modeling elements, refer to DataItem, DA Root, and Alias Models, page 71.

EFB, DFB and Program Unit Variable Structure

EcoStruxure Control Expert EFBs, DFBs, and program units with name <ContainerName> are structured as follows:

<EntityName>=*"Folder"*,<*ContainerNameBrowsePath*>,<ContainerName>, <VariableChild>

Where <*ContainerNameBrowsePath*>=<AliasName>!< ContainerName>.

Example

Consider the following example of a variable declaration made in EcoStruxure Control Expert:

```
DDT1 as Struct

aBool BOOL

anArray ARRAY[0...1, 0...1] of BOOL
```

For the variable *aDDT1* that is an instance of DDT1 and is accessed with the alias *MyAlias*, the complete list of populated nodes and the target node of the *Organizes ReferenceType* are presented in the following table.

Node	Target Node
NodeAlias	Node1
Node1	Node11
Node1	Node12
Node12	Node121
Node12	Node122
Node121	Node1211
Node121	Node1212
Node122	Node1221
Node122	Node1222

The associated node attributes are described in the following table.

Node	*NodeId*	*BrowseName*	*DisplayName*
NodeAlias	ns=2;s=0:MyAlias	MyAlias	MyAlias
Node1	ns=2;s=0:MyAlias!aDDT1	2:aDDT1	aDDT1
Node11	ns=2;s=0:MyAlias!aDDT1.aBool	2:aBool	aBool
Node12	ns=2;s=0:MyAlias!aDDT1.anArray	2:anArray	anArray
Node121	ns=2;s=0:MyAlias!aDDT1.anArray[0]	2:anArray[0]	anArray[0]
Node12 2	ns=2;s=0:MyAlias!aDDT1.anArray[1]	2:anArray[1]	anArray[1]
Node1211	ns=2;s=0:MyAlias!aDDT1.anArray[0][0]	2:anArray[0][0]	anArray[0][0]
Node1212	ns=2;s=0:MyAlias! aDDT1.anArray[0][1]	2:anArray[0][1]	anArray[0][1]
Node1221	ns=2;s=0:MyAlias! aDDT1.anArray[1][0]	2:anArray[1][0]	anArray[1][0]
Node1222	ns=2;s=0:MyAlias! aDDT1.anArray[1][1]	2:anArray[1][1]	anArray[1][1]

DataItem Node Attributes

Description Node Attribute

The *Description* node attribute for a *DataItem* is set to the *Comment* property of the matching variable configured in EcoStruxure Control Expert.

AccessLevel Node Attribute

The *AccessLevel* node attribute of a *DataItem* includes the following bits:

- Bit 0 (*CurrentRead*) is set to 1.
- Bit 1 (*CurrentWrite*) is set to 0 if either of the following is true:
  - The matching EcoStruxure Control Expert variable *Constant* property is set to 1.
  - The **Read Only** check box (**Option** property of the device alias **General** category, page 55), to which the variable belongs, is selected.

  Otherwise, Bit 1 is set to 1.

Effect of the Byte array management Property On *Datatype*, *ValueRank* and *Value* Attributes

The settings described in the following tables apply to the *Datatype*, *ValueRank*, and *Value* attributes for the following EcoStruxure Control Expert categories and are determined by the state of the **Byte array management > Manage as ByteString** check box in the **PLC Software** settings of the EcoStruxure OPC UA Server Expert Configuration Tool. For more information, refer to **Project Files Options** Section, page 60.

When the check box is cleared:

Category	***Datatype***	***ValueRank*(1)**	***Value***
EDT	For more information, refer to Variable Data Type Conversion, page 75	*Scalar*	Scalar
DDT	BYTE	*OneDimension*	Array
ARRAY of EDT	For more information, refer to Variable Data Type Conversion, page 75	*OneDimension*	Array
Other ARRAY	BYTE	*OneDimension*	Array
(1) *Scalar* values can be negative, zero, or positive. *OneDimension* values can be 0 or positive.			

When the check box is selected:

Category	***Datatype***	***ValueRank*(1)**	***Value***
EDT	For more information, refer to Variable Data Type Conversion, page 75	*Scalar*	Scalar
DDT	ByteString	*Scalar*	Scalar
Array of EDT	For more information, refer to Variable Data Type Conversion, page 75	*OneDimension*	Array
Other ARRAY	ByteString	*Scalar*	Scalar
(1) *Scalar* values can be negative, zero, or positive. *OneDimension* values can be 0 or positive.			

DataItem Properties

DataItem Property Model

Every EcoStruxure Control Expert variable modeled as a *DataItem* has (0..*) *hasProperty ReferenceType* targeting a *Variable* node instance, which is defined as a property (that is to say, the instantiated variable has a *hasTypeDefinition ReferenceType* targeting a *PropertyType* as *VariableType*).

The following notation is used to qualify a *DataItem* property of a node, for which *NodeId* String Identifier is 0:<*VariableNameBrowsePath*>

<*PropertyId*>=<Description>,<*DataType*>

DataItem Property Patterns

The following patterns are used to set the OPC UA node attributes.

Node attribute	Value
NodeId Type	String
NodeId Identifier	2:<*VariableNameBrowsePath*>?<PropertyId>
BrowseName	2:<*Description*>
DisplayName	<*Description*>
DataType	<*DataType*>

DataItem Property List

DataItem properties can include the following.

<Proper-tyId>	<Description>	<Data-Type>
5000	The initial value of the variable.	INT32
5001	The type of variable. It may be elementary, structured, function-block, section, and so forth.	STRING
5005	The size, useful for non-elementary data types.	INT32
5007	The type name, as known by the programming tool.	STRING
5010	The free-form string of the variable.	STRING
5011	The topological address of the variable.	STRING
5012	Time stamp event support.	BOOL

State RAM Topological Objects

Populating State RAM Topological Objects

You can populate EcoStruxure Control Expert device State RAM topological objects in the address space. To do this, select the **Topological objects > Populate** check box in the **General** category , page 55 of the EcoStruxure OPC UA Server Expert Configuration Tool for an EcoStruxure Control Expert device alias. For more information, refer to Device Alias Properties, page 53.

State RAM Topological Object Model Structure

The following structure is applied to State RAM topological objects:

<EntityName>= <Mapping>

Where:

- <Mapping>=“Folder”,<MappingBrowsePath>,”#Mapping”,<TopoObject>
- <MappingBrowsePath>=<AliasName>!#Mapping
- <TopoObject>=“DataItem”,<TopoObjectBrowsePath>,<TopoAddress>
- <TopoObjectBrowsePath>=<AliasName>!<TopoAddress>
- <TopoAddress> is the topological address of the variable (for example: %MW10)

For more information on OPC UA modeling elements, refer to DataItem, DA Root, and Alias Models, page 71.

Specific *DataItems*

Diagnostic Supporting *DataItems*

Specific *DataItems* are data items not related to controller variables, but supporting diagnostics operations. Specific *DataItems* are only populated for a device alias that represents a EcoStruxure Control Expert controller.

Modeling Structure

Specific *DataItems* are structured as follows:

<EntityName>=<Specific>

Where:

- <Specific>=“Folder”,<SpecificBrowsePath>,”#Specific”,<SpecificItem>
- <SpecificBrowsePath>=<AliasName>!#Specific
- <SpecificItem>=“DataItem”,<SpecificItemBrowsePath>,<SpecificItemName>
- <SpecificItemBrowsePath>=<AliasName>!<SpecificItemName>
- <SpecificItemName> is the name of the Specific *DataItem*

For more information on OPC UA modeling elements, refer to DataItem, DA Root, and Alias Models, page 71.

Specific *DataItems* Collection

The following table describes the collection of specific *DataItems* and their relationship to the *DataType* attribute and to the *CurrentWrite* value of the *AccessLevel* node attribute.

<SpecificItemName>	*DataType*	*CurrentWrite*	Description
#AddressSpaceState	BYTE	0	Application space state at alias level, page 81
#AppliID	INT32	0	Application signature set, page 82
#AppliName	STRING	0	Application name of the controller application
#AppliOMC	INT32	0	Application modification counter set, page 82
#AppliVersion	STRING	0	Application version of the controller application
#DeviceIdentity	STRING	0	Controller product reference
#DeviceServiceLevel	BYTE	0	Service level at device level, page 82
#PLCQualStatus	INT16	0	monitoring of the communication status with a controller device, page 84
#SwitchPrimaryAddress	UINT16	1	Monitoring the controller link redundancy function status, page 85

#AddressSpaceState

You can address *#AddressSpaceState* to monitor the alias address space state, page 88. Also refer to Alias and Data Access Address Space State Diagram, page 90.

The following table describes the possible values.

Value	Alias address space state	Description
0	*Empty*	Initial state
1	*Built*	The configured options are successfully populated
2	*Updating*	The address space is updating
3	*Partially Built*	At least one option is not successfully populated (if configured)

This item is populated in the *#Specific* folder. Nevertheless, it is created at server start and put in a cache. You can address it even if the alias address space is in the *Empty* state.

NOTE: While the alias address space state is not *Built*, any variable node access (including *MonitoredItems* creation) returns the *BadNodeIdUnknown* exception.

#AppliID

The values of *#AppliID* are a copy of the application signatures (IDs) read in the controller. You can address this item to monitor internal changes made by EcoStruxure Control Expert when it builds the application. Some of these signatures change each time the application is built while other signatures are only generated when the application is created.

DataType	*CurrentWrite*	Value Read
Int32 OneDimension array	0	8 values are published. Only the following values are pertinent (given by array index): • Index 0: Creation ID (CID) • Index 1: Overall modification ID (MID) • Index 2: Automatic modification ID (AID) • Index 3: Memory Layout ID (LID) • Index 4: Data block Deletion ID (DID)

#AppliOMC

The values of *#AppliOMC* are a copy of the Object Modification Counters (OMC) read in the controller. You can address this item to monitor the consistency of internal changes made by EcoStruxure Control Expert when it builds the application.

Coupled with *#AppliID*, it enables you to determine the type of modification made to the application.

DataType	*CurrentWrite*	Value Read
Int32 OneDimension array	0	16 values are published. Only the following values are pertinent (given by array index): • Index 0: Overall application • Index 1: Program • Index 2: Configuration • Index 3: Type definition • Index 4: Variables • Index 5: Animation tables • Index 6: Communication • Index 7: Functional modules • Index 9: Variable creation (variables and FB instances created in the application or inside a type)

#DeviceServiceLevel

Similarly to the global *ServiceLevel*, *#DeviceServiceLevel* supports monitoring of a level of service local to a device. The following are used for the calculation of *#DeviceServiceLevel*:

- Alias address space state, page 88
- Device communication status, page 41
- **#DeviceServiceLevel computation** (if used), page 62

This item is populated in the *#Specific* folder. Nevertheless, it is created at server start and put in a cache. You can address it even if the alias address space is in the *Empty* state.

The following table indicates the possible values of *#DeviceServiceLevel* when **#DeviceServiceLevel computation** is not used.

Value	Device communication status	Address space state
1	OFFLINE	*Any*
3	ONLINE	*Empty*
4		*Partially Built*
199		*Updating*
255		*Built*

NOTE: The *#DeviceServiceLevel DataValue statusCode* is *Good*. For more information on *DataValue*, refer to *DataValue* Type, page 70.

The following table indicates the possible values of *#DeviceServiceLevel* if **#DeviceServiceLevel computation** is used to take into account standalone controller rules.

Value	Device communication status	Address space state	Controller state
1	OFFLINE	*Any*	Any
3	ONLINE	*Empty*	
4		*Partially Built*	
120		*Updating* or *Built*	*Stop*
199		*Updating*	*Run*
255		*Built*	

NOTE: The *#DeviceServiceLevel DataValue statusCode* may be *Bad* in case of a communication error, page 39. For more information on *DataValue*, refer to *DataValue* Type, page 70.

The following table indicates the possible values of *#DeviceServiceLevel* if **#DeviceServiceLevel computation** is used to take into account redundant controller rules.

Value	Device communication status	Address space state	Primary controller state	Standby controller state
1	OFFLINE	*Any*	Any	Any
3	ONLINE	*Empty*		
4		*Partially Built*		
120		*Updating* or *Built*	*Stop*	*Stop*
150		*Updating* or *Built*	*Stop*	*Wait*
199		*Updating*	*Run*	*Run*
N.A.(1)		*Updating* or *Built*	*Stop*	*Run*
203		*Updating* or *Built*	*Run*	*Stop*
204		*Updating* or *Built*	*Run*	*Wait*
255		*Built*	*Run*	*Run*
(1) The value is not managed because the primary controller *Stop* state is a switchover condition.				

NOTE: The *#DeviceServiceLevel DataValue statusCode* may be *Bad* in case of a communication error, page 39.

NOTE: The *#DeviceServiceLevel DataValue statusCode* may be *Bad_ ConfigurationError* if either of the following is true:

- Neither the XVM symbol link file nor the data dictionary are configured. In this case, it is not possible to monitor *ECPU_HSBY_1*.
- The data dictionary is configured but *ECPU_HSBY_1* is not embedded as HMI variable.
- The standby controller is not connected or the alias is configured on the standby IP address (primary IP address+1)

For more information on *DataValue*, refer to *DataValue* Type, page 70.

#PLCQualStatus

#PLCQualStatus supports monitoring the communication status with a controller device, page 40. The following table describes the possible values.

Value (hex)	Description
0x00C0	Communication with the device is functional (Device state is *Good*).
0x0040	No communication with the device since less than the configured **Device timeout (ms)** (Device state is *Uncertain*).
0x000C	No communication with the device since the configured **Device timeout (ms)** (Device state is *Bad*).
0x0	The device is not identified (the device state is *Unknown* or *Missing*).
0x0018	The device is inconsistent, page 72.

#SwitchPrimaryAddress

#SwitchPrimaryAddress supports monitoring the status of the controller link redundancy function, page 41. You can use it to trigger a switchover from the primary communication path to the standby communication path, provided the standby communication path is both configured and ONLINE. The following table describes the possible values.

Value	Description
0	The primary communication path is **Device address 1**.
1	The primary communication path is **Device address 2**.
2	The standby communication path is not configured.

NOTE: When the value of *#SwitchPrimaryAddress* is:

- 0: You can change the primary communication path to **Device address 2** by using a write command to change the value to 1.
- 1: You can change the primary communication path to **Device address 1** by using a write command to change the value to 0.

#SwitchPrimaryAddress and *DataValue*

The following table describes the possible *DataValue statusCode* values used by EcoStruxure OPC UA Server Expert in relation to communication with devices.

***statusCode* value**	**Description**
Good	Communication with the device is correct and the primary communication path is ONLINE.
Bad	Communication with the device is interrupted, the primary and standby communication paths are OFFLINE.
Uncertain	Communication with the device is correct, but the standby communication path is OFFLINE.

For more information on *DataValue*, refer to *DataValue* Type, page 70.

Address Space Use Cases

Controller Disconnected at OPC UA Server Start

An alias configured in the EcoStruxure OPC UA Server Expert Configuration Tool with the **Topological objects > Populate** or **PLC Embedded Data > Using Data Dictionary** check box selected and for which the physical associated device is disconnected at OPC UA server start generates an *Empty* then, *Partially Built* address space state, page 89.

In some cases, a controller that is disconnected at OPC UA server start does not impact the system (for example, if an incremental commissioning is performed). The *ServiceLevel* is not affected and you can select the **Allow disconnection at server startup** global property to manage this use case. For more information, refer to **Communication** Settings, page 62.

As a result, if a controller with <Alias> is disconnected at OPC UA server start, the values you must take into account are determined as follows:

- **Allow disconnection at server startup** is selected.

Node	Value
ServiceLevel	255
<Alias> ! #DeviceServiceLevel	1
<Alias> ! #AddressSpaceState	3

- **Allow disconnection at server startup** is not selected.

Node	Value
ServiceLevel	4
<Alias> ! #DeviceServiceLevel	1
<Alias> ! #AddressSpaceState	3

Controller Disconnected During Operation

If a controller is disconnected after the alias address space is in the *Built* state, there is no impact on the address space state because the address space remains in the cache. As a result, there is no impact on the service level. Nevertheless, *#DeviceServiceLevel* = 1, page 85.

Modbus Device Address Space

Overview

This section describes the address space of a Modbus device.

Modbus Device Address Space Publication

CSV Symbol Table File Consistency Verification

Modbus device address space is published through translation of a Comma Separated Value (CSV) symbol table file associated with a device alias.

The CSV file does not provide for consistency verification of the application name and version, between the CSV file and the controller application. If the underlying CSV file is modified, the address space is not updated. In this case, the modified CSV file can only be loaded into EcoStruxure OPC UA Server Expert after a server restart.

CSV File Structure

The structure of each line in a CSV file is as follows:

<Address><Separator><Symbol><Separator><Comment>

Where:

- <Address> is a valid address for the device associated with that symbol file, page 87. The maximum length is 50 characters.
- <Separator> can be a comma, a space or a tab character.
- <Symbol> can be any string of characters without a comma, space, ,tab, or special character. The maximum length is 33 characters.
- <Comment> is a text field with a maximum length of 510 characters.

CSV File Address Structure

<Address> presents the following structure:

<ObjectAdress>[:<ArrayLength>] [;<Postfix>][;R]

Where <ObjectAdress> is a function of <ObjectType>, as indicated in the following table.

<ObjectType>	**<ObjectAddress>**
Coils(1)	000000...099999
Input status	100000...199999
Input register	300000...399999
Holding register(2)	400000...499999
(1) You can use the alternate syntax *%Mi* or *%Mi:<n>* where *<n>* is the array length. (2) You can use either of the following alternate syntax: • Access to one registers *%MWi* or *%MWi:<n>* • Access to two registers: *%MDi* or *%MDi:<n>* • Access to two registers with conversion to Float: *%MFi* or *%MFi:<n>* Where *<n>* is the array length.	

- <ArrayLength> supports the declaration of one-dimension array of <ArrayLength> objects starting at <ObjectAddress>
- <Postfix> applies only to holding registers. It allows a type conversion. The values are:
  - F: To convert 2 consecutive holding registers to 32 bit float.
  - D: To convert 2 consecutive holding registers to 32 bit signed integer.
  - R: To set the variable access right to *ReadOnly*.

For example, `400001:10;F,a_table_of_10_float` is a valid line in CSV file.

CSV File Addressing Iterations

The following table presents the collection of possible <ObjectType>, <ArrayLength>,<Postfix> and ;R addressing combinations, with their relationship to *DataItem* attributes.

<ObjectType>	**<Array Length>**	**<Postfix>**	**;R**	**DataType**	**CurrentWrite AccessLevel**	**ValueRank**	**Value array length**
Coils	NS	NS	NS	Boolean	1	Scalar	NA
Input Status	NS	NS	NS	Boolean	0	Scalar	NA
Input Register	NS	NS	NS	Int16	0	Scalar	NA

<ObjectType>	<Array Length>	<Postfix>	;R	DataType	CurrentWrite AccessLevel	ValueRank	Value array length
Holding Register	NS	NS	NS	Int16	1	Scalar	NA
Coils	NS	NS	S	Boolean	0	Scalar	NA
Input Status	NS	NS	S	Boolean	0	Scalar	NA
Input Register	NS	NS	S	Int16	0	Scalar	NA
Holding Register	NS	NS	S	Int16	0	Scalar	NA
Holding Register	NS	F	NS	Float	1	Scalar	NA
Holding Register	NS	D	NS	Int32	1	Scalar	NA
Holding Register	NS	F	S	Float	0	Scalar	NA
Holding Register	NS	D	S	Int32	0	Scalar	NA
Coils	S	NS	NS	Boolean	1	OneDimension	<ArrayLength>
Input Status	S	NS	NS	Boolean	0	OneDimension	<ArrayLength>
Input Register	S	NS	NS	Int16	0	OneDimension	<ArrayLength>
Holding Register	S	NS	NS	Int16	1	OneDimension	<ArrayLength>
Coils	S	NS	S	Boolean	0	OneDimension	<ArrayLength>
Input Status	S	NS	S	Boolean	0	OneDimension	<ArrayLength>
Input Register	S	NS	S	Int16	0	OneDimension	<ArrayLength>
Holding Register	S	NS	S	Int16	0	OneDimension	<ArrayLength>
Holding Register	S	F	NS	Float	1	OneDimension	<ArrayLength>
Holding Register	S	D	NS	Int32	1	OneDimension	<ArrayLength>
Holding Register	S	F	S	Float	0	OneDimension	<ArrayLength>
Holding Register	S	D	S	Int32	0	OneDimension	<ArrayLength>

S: The option is specified

NS: The option is not specified

NA: Not applicable

Symbol Data Access Structure

The following modeling structure is used to access a Modbus symbol:

<EntityName>=“DataItem”,<VariableNameBrowsePath>,<Symbol>

where <VariableNameBrowsePath>=<AliasName>!<Symbol>

For more information on OPC UA modeling elements, refer to DataItem, DA Root, and Alias Models, page 71.

OPC UA Server Address Spaces

This section describes the alias and data access address spaces published in the OPC UA Server.

Alias and Data Access Address Spaces

The EcoStruxure OPC UA Server Expert 2.0 *ServiceLevel* does not take into account errors detected during the loading of symbols. However, the

disconnection of a controller at server start while the data dictionary is set is taken into account.

Consequently, an OPC UA client cannot rely on *ServiceLevel* to detect whether it is possible to address a variable node in the address space. If a variable node is not populated when it is addressed, the OPC UA result code *Bad_NodeIdUnknown* is returned to the OPC UA client that must operate in recovery mode.

Similarly, it is not possible to detect a symbol loading error at startup or a symbol reloading error during operation in a warm redundant system, page 44.

To help improve this, starting from EcoStruxure OPC UA Server Expert 2.01, the new *ServiceLevel* value *4* and two new specific *DataItems* are provided to manage the following:

- The address space state of an alias: *#AddressSpaceState*, page 81
- The *ServiceLevel* local to the device: *#DeviceServiceLevel*, page 82

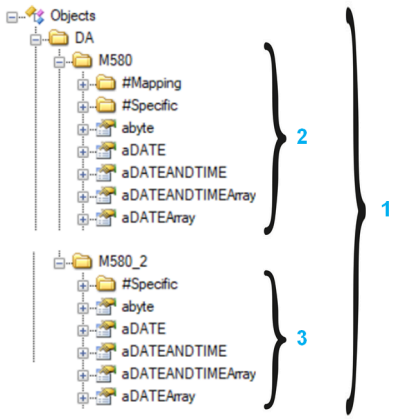
Alias Address Space

The alias address space is a composition of at least one *#Specific* folder and optional parts depending on the property that is selected or configured in the EcoStruxure OPC UA Server Expert Configuration Tool.

Property	Alias Address Space contents
Topological objects > Populate	*#Mapping* folder
Symbol table file	Controller symbols
PLC Embedded Data > Using Data Dictionary	Controller symbols

NOTE: Symbol table file and **PLC Embedded Data > Using Data Dictionary** exclude each other, while **Topological objects > Populate** can be combined with any of them.

The following figure shows the alias and data access address space organization, including the possible configuration options.



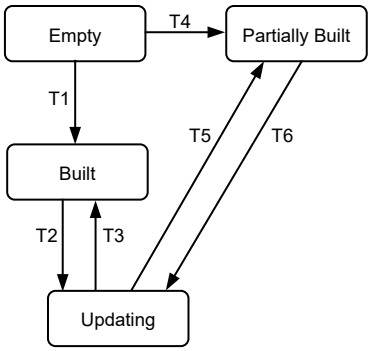
Item	Description
1	Data access address space
2	Alias address space
3	Controller symbols

The following table lists the errors that can be detected depending on the property that is selected or configured.

Property	Possible error causes
Topological objects > Populate	At server start, the controller is not connected or a network interruption has a duration greater than three times **Device timeout (ms)**.
Symbol table	The XVM or CSV file is not readable or incompatible.
PLC Embedded Data > Using Data Dictionary	• At server start, the controller is not connected or a network interruption has a duration greater than three times **Device timeout (ms)**. • The data dictionary is not embedded in the controller. • The data dictionary is not correctly loaded.

Alias and Data Access Address Space State Diagram

The alias and data access address spaces can have the states indicated in the following state diagram.



The following table describes the states and transitions for the alias and data access address spaces.

State \transition	Alias address space state description	Data access address space state description
Empty	Initial state at OPC UA server start.	Initial state at OPC UA server start.
T1	The alias address space populating is performed.	The alias address space populating is performed for all aliases.
Built	The alias address space has the *Built* state.	All alias address spaces have the *Built* state.
T2	A symbol link modification is detected.	A symbol link modification is detected on at least one alias.
Updating	The alias address space is updating.	The alias address spaces involved in T2 or T6 are updating.
T3	The symbol link reloading is successfully completed. Coming from T6, the alias address space takes the *Built* state.	The symbol link reloading of aliases involved in T2 completed successfully. Coming from T6, the alias address space of aliases involved in T4 or T5 take the *Built* state.
T4	The alias address space populating did not succeed, page 89.	At least one of the alias address space populating did not succeed, page 89.
Partially Built	The alias address space has the *Partially built* state.	At least one alias address space has the *Partially built* state.
T5	The symbol link reloading did not succeed, page 89.	The symbol link reloading of an alias involved in T2 did not succeed., page 89.
T6	The error that was detected in T4 or T5 is resolved.	The error that was detected in T4 or T5 is resolved.

Availability of OPC UA Service Sets

The data access address space states are considered for the calculation of the OPC UA server *ServiceLevel* variable value, page 38.

The following table indicates the availability of OPC UA service sets according to the data access address space state.

State	*Discovery/ GetEndoints*	*SecureChannel*	*Session*	*View*	*Attribute*	*Subscription*	*MonitoredItem*
Empty	Yes	Yes	Yes	No	No	Yes	No
Built	Yes	Yes	Yes	Yes	Yes	Yes	Yes
Updating	Yes	Yes	Yes	No for the alias involved. Yes for the others.	Yes	Yes	Yes
Partially Built	Yes	Yes	Yes	Yes	Yes for the populated nodes; otherwise, no	Yes	Yes for the populated nodes; otherwise, no

EcoStruxure OPC UA Server Expert Sample Client

What’s in This Part

Using the Sample Client

What's in This Chapter

Overview

This chapter describes the EcoStruxure OPC UA Server Expert Sample Client that is included with EcoStruxure OPC UA Server Expert and how to execute five tasks typically performed by an OPC UA client:

- Connect to an OPC UA server and create a session, page 95.
- Browse an OPC UA node, using different options, page 95.
- View the attributes of an OPC UA node, page 97.
- Read\Write the attributes of an OPC UA variable node, page 99.
- Subscribe to a variable node *MonitoredItem*, page 101.

Sample Client Description

Supported OPC UA Service Sets

The EcoStruxure OPC UA Server Expert Sample Client (hereafter, the OPC UA sample client) is an OPC UA client that supports the following OPC UA service sets:

- *Attribute*
- *MonitoredItem*
- *SecureChannel*
- *Session*
- *Subscription*
- *View*

These service sets are also included in the collection of OPC UA service sets supported by EcoStruxure OPC UA Server Expert. For more information, refer to Server Stack Services, page 15.

Supported Identity Tokens

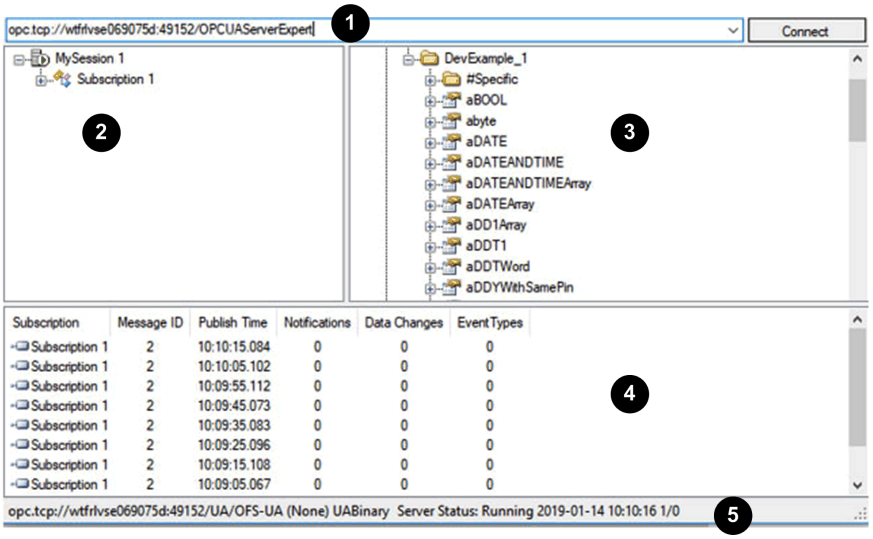
Only the following identity tokens are supported by the OPC UA sample client:

- *Anonymous*
- *UserName*

NOTE: The X509 identity token is not supported by the OPC UA sample client.

OPC UA Sample Client Window

The OPC UA sample client window presents five distinct areas.



1 Server connection area

2 Session area

3 Address space area

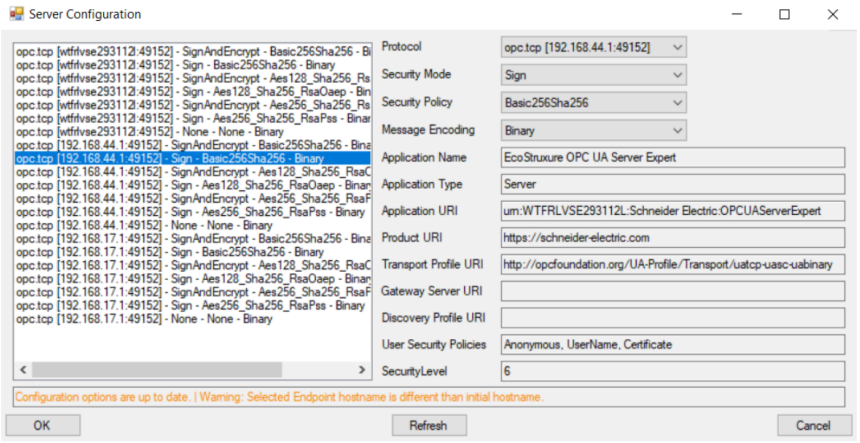
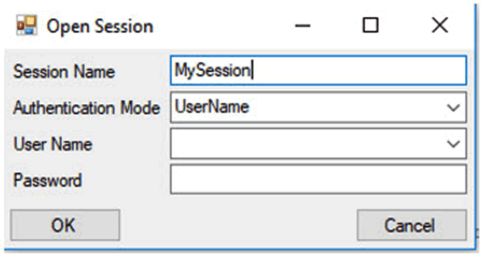
4 Global subscription notification area

5 Status bar

Connecting to an OPC UA Server and Creating a Session

Making a Connection and Configuring a Session

To communicate with an OPC UA server, open a connection, and begin a session, proceed as follows.

Step	Action
1	In the OPC UA sample client window, page 94, enter the target OPC UA server URI in Server connection area, then click **Connect**. **Result**: The **Server Configuration** dialog box opens. 
2	In the **Server Configuration** dialog box, select the settings to be used to create a session and click **OK**. **Result**: The **Open Session** dialog box opens.  **NOTE:** The selections you make in the **Server Configuration** dialog box become part of the endpoint returned by the OPC UA server after a connection is established using the *GetEndpoints* service.
3	In the **Open Session** dialog box, enter the appropriate information in the following fields and click **OK**: • **Session Name**: A name for the session. • **Authentication Mode**: A user identity token supported by the server that is determined by the menu selection. ◦ **UserName**: Enter a username and password. ◦ **Certificate**: Browse for and select a certificate and Private Key files. **NOTE:** When the session is created, the session area and address space area in the OPC UA sample client window , page 94 are populated.

Browsing OPC UA Nodes and Viewing Node Attributes

Browsing OPC UA Nodes and Setting Browse Options

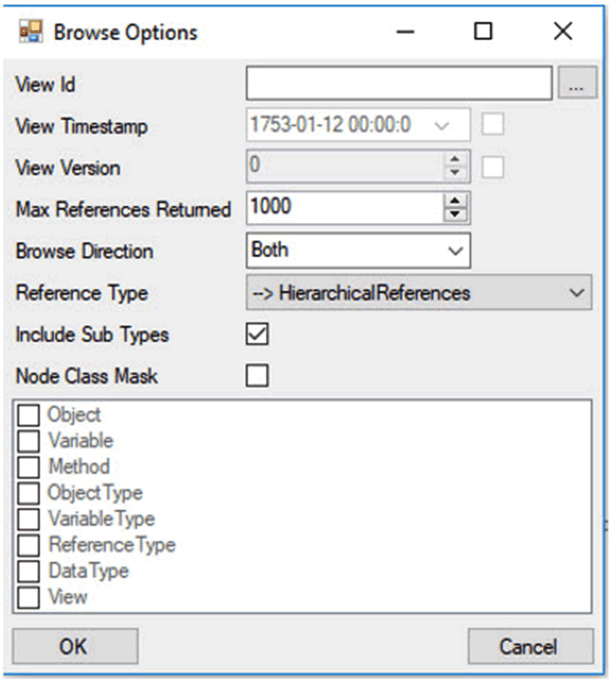
The OPC UA address space area in the OPC UA sample client window, page 94 displays a tree structure where each child element represents an OPC UA node

and each child thereof represents a related OPC UA node that is linked to its parent with an OPC UA *ReferenceType*.

By default, the **Browse Direction** of an OPC UA node is *FORWARD_0*. As a result, only OPC UA child nodes, linked to their parent with a *Forward* OPC UA *ReferenceType*, are displayed.

For references that are not symmetric, you can set the browse direction to *INVERSE_1* to manage inverse OPC UA *ReferenceType* nodes, or *BOTH_2* to manage both forward and inverse OPC UA *ReferenceType* nodes.

To edit browse options, proceed as follows.

Step	Action
1	In the address space area of the OPC UA client window, right-click the OPC UA parent node and select **Browse Options...** **Result**: The **Browse Options** dialog box opens. 
2	In the **Browse Options** dialog box, you can set the following options: • **Browse Options**: Select *Forward*, *Inverse*, or *Both*. • **Max References Returned**: Limit the tree view population by setting the maximum number of references to display.

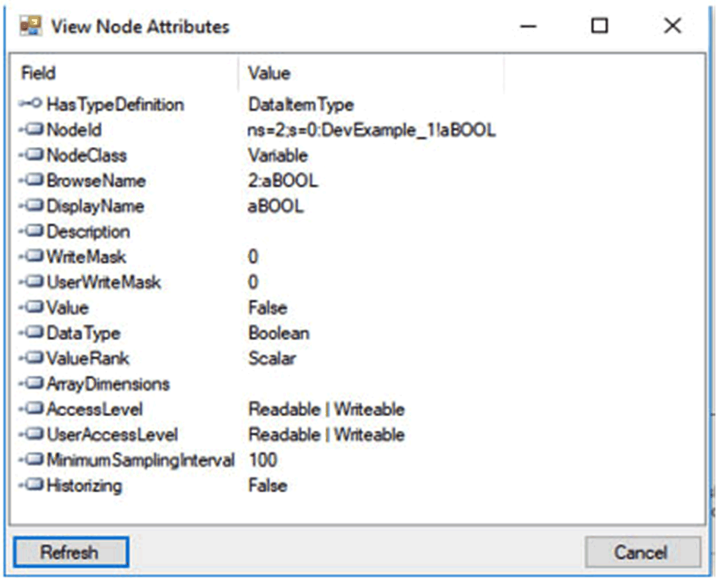
NOTE: By default, reference types are not displayed. To display reference types, in the address space area of the OPC UA client window, right-click the OPC UA parent node and select **Show References**.

Viewing OPC UA Node Attributes

OPC UA nodes are associated with attributes.

To view attributes, in the address space area of the OPC UA client window, right-click the OPC UA parent node and select **View Attributes...**

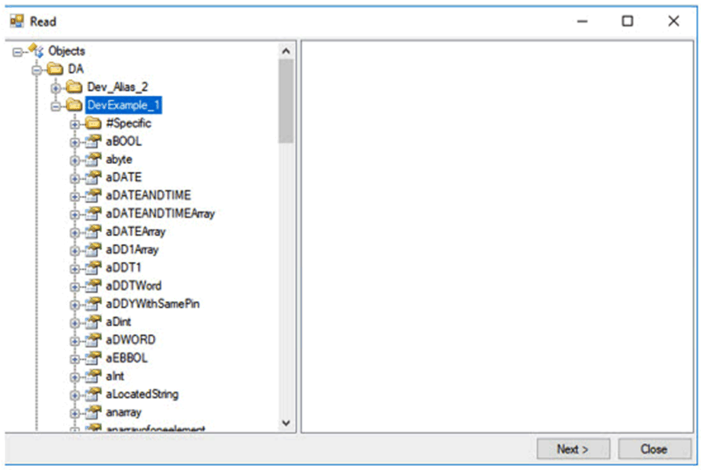
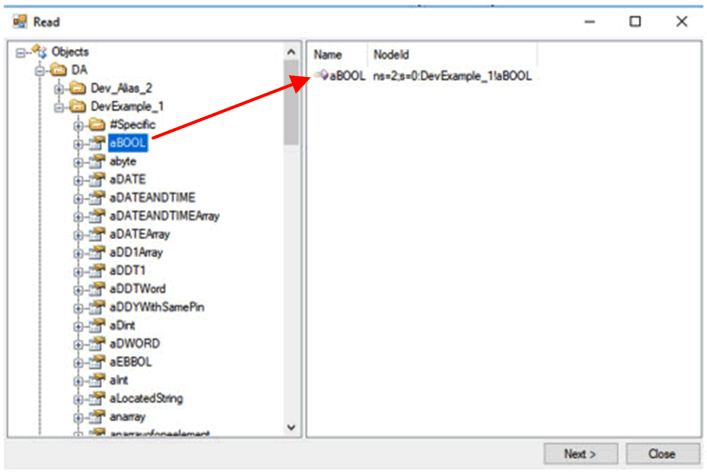
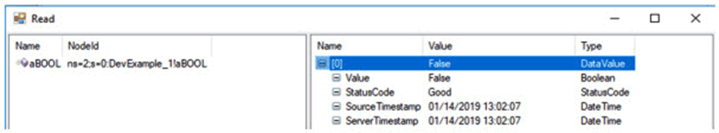
Result: The **View Node Attributes** dialog box opens.


View Node Attributes
Field
Value
HasTypeDefinition
DataItemType
NodeId
ns=2;s=0:DevExample_1!aBOOL
NodeClass
Variable
BrowseName
2:aBOOL
DisplayName
aBOOL
Description
WriteMask
0
UserWriteMask
0
Value
False
DataType
Boolean
ValueRank
Scalar
ArrayDimensions
AccessLevel
Readable | Writeable
UserAccessLevel
Readable | Writeable
MinimumSamplingInterval
100
Historizing
False
Refresh
Cancel

Reading\Writing the Attributes of a Variable Node

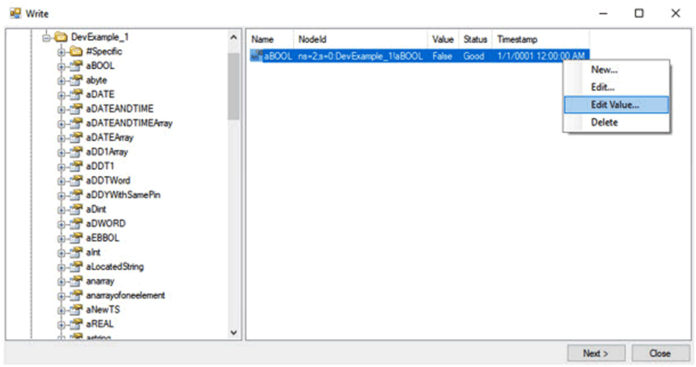
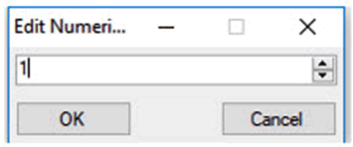
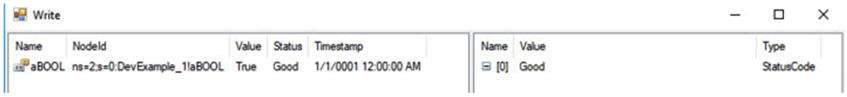
Reading the Attributes of a Variable Node

To read the attributes of a variable node, proceed as follows.

Step	Action
1	In the OPC UA session area in the OPC UA sample client window, page 94, right-click the session and select **Read...**. **Result**: The **Read** wizard opens. 
2	In the left-hand pane, select a variable node to read from and drop it in the right-hand pane. **Result**: The variable node is added. 
3	Click **Next** to view the attributes of the selected variable node. 

Writing the Attributes of a Variable Node

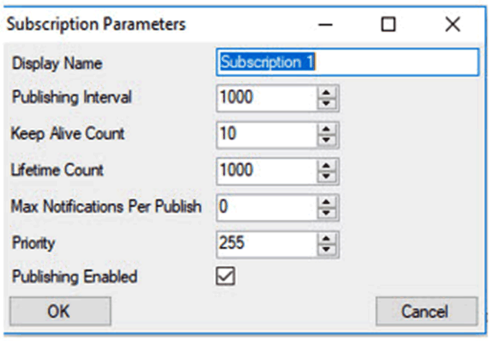
To write the attributes of a variable node, proceed as follows.

Step	Action
1	In the OPC UA session area in the OPC UA sample client window, page 94, right-click the session and select **Write...** to open the **Write** wizard.
2	In the left-hand pane, select a variable node to write to and drop it in the right-hand pane. **Result**: The variable node is added.
3	Right-click the added variable node and select **Edit Value...**:  **Result**: The**Edit Numeric** dialog box opens.
4	Input the value to be written and click **OK**. 
5	Click **Next** to view the attribute value and verify that the write operation was successful. 

Subscribing to a Variable Node *MonitoredItem*

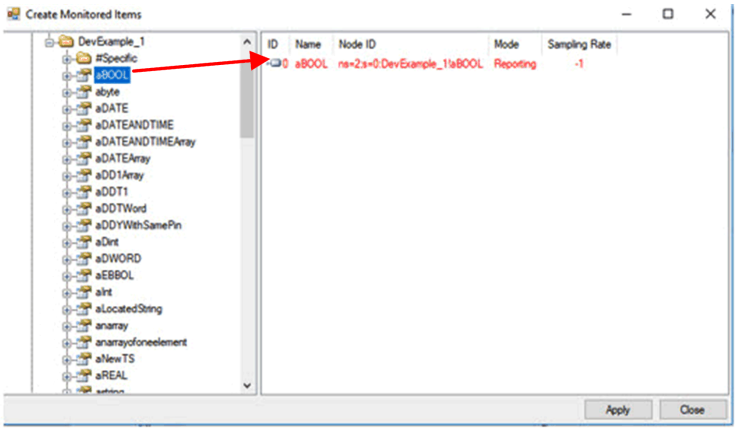
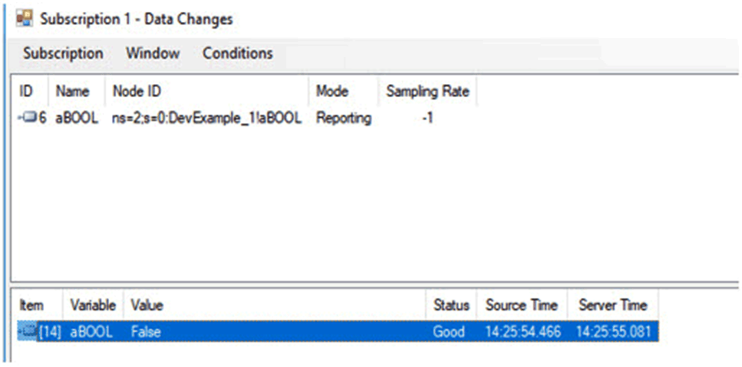
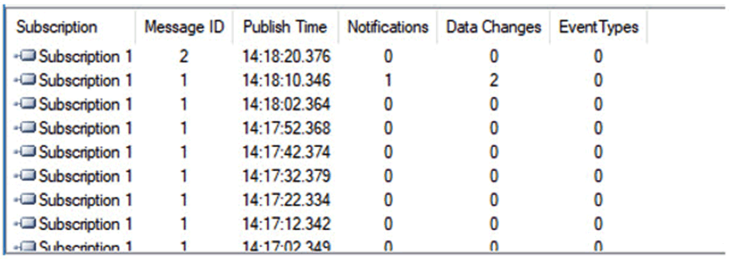
Creating a Subscription

To create a subscription, proceed as follows.

Step	Action
1	In the OPC UA session area in the OPC UA sample client window, page 94, right-click the session and select **Subscription > New...**. **Result**: The **Subscription Parameters** dialog box opens. 
2	Configure property values for the subscription and click **OK**. **Result**: In the session area, the subscription is displayed as a child of the session.  **NOTE:** When you select the subscription, the **Monitor Subscription** window is displayed in the right-hand pane.

Creating a *MonitoredItem*

To create a *MonitoredItem*, proceed as follows.

Step	Action
1	In the **Monitor Subscription** pane, select the menu item **Subscription > Create Monitored Items...**. **Result**: The **Create Monitored Items** wizard opens.
2	In the left-hand pane, select a variable node to be monitored as a *MonitoredItem* and drop it in the right-hand pane. **Result**: The variable node is added. 
3	Click **Apply** and then, **Close**. **Result**: The **Data Changes** window is created for the selected subscription, and updated information about data changes is display its bottom area.  Also, the global subscription notification area in the OPC UA sample client window, page 94 is updated. 

Index

R

S

T

V

W

X

Schneider Electric
35 rue Joseph Monier
92500 Rueil Malmaison
France

+ 33 (0) 1 41 29 70 00

www.se.com

MFR53158.03