

Technical Document

Niagara OpcUa Driver Guide

October 17, 2019

niagara

This PDF was created from documentation on docs.tridium.com. For the most current Tridium product documentation, go to docs.tridium.com.

OPC UA Driver Guide

Tridium, Inc.
3951 Westerre Parkway, Suite 350
Richmond, Virginia 23233
U.S.A.

Confidentiality

The information contained in this document is confidential information of Tridium, Inc., a Delaware corporation (“Tridium”). Such information and the software described herein, is furnished under a license agreement and may be used only in accordance with that agreement.

The information contained in this document is provided solely for use by Tridium employees, licensees, and system owners; and, except as permitted under the below copyright notice, is not to be released to, or reproduced for, anyone else.

While every effort has been made to assure the accuracy of this document, Tridium is not responsible for damages of any kind, including without limitation consequential damages, arising from the application of the information contained herein. Information and specifications published here are current as of the date of this publication and are subject to change without notice. The latest product specifications can be found by contacting our corporate headquarters, Richmond, Virginia.

Trademark notice

BACnet and ASHRAE are registered trademarks of American Society of Heating, Refrigerating and Air-Conditioning Engineers. Microsoft, Excel, Internet Explorer, Windows, Windows Vista, Windows Server, and SQL Server are registered trademarks of Microsoft Corporation. Oracle and Java are registered trademarks of Oracle and/or its affiliates. Mozilla and Firefox are trademarks of the Mozilla Foundation. Echelon, LON, LonMark, LonTalk, and LonWorks are registered trademarks of Echelon Corporation. Tridium, JACE, Niagara Framework, NiagaraAX Framework, and Sedona Framework are registered trademarks, and Workbench, WorkPlaceAX, and AXSupervisor, are trademarks of Tridium Inc. All other product names and services mentioned in this publication that are known to be trademarks, registered trademarks, or service marks are the property of their respective owners.

Copyright and patent notice

This document may be copied by parties who are authorized to distribute Tridium products in connection with distribution of those products, subject to the contracts that authorize such distribution. It may not otherwise, in whole or in part, be copied, photocopied, reproduced, translated, or reduced to any electronic medium or machine-readable form without prior written consent from Tridium, Inc.

Copyright 2018 Tridium, Inc. All rights reserved.

The product(s) described herein may be covered by one or more U.S. or foreign patents of Tridium.

About this Guide

>

This topic contains important information about the purpose, content, context, and intended audience for this document.

Product Documentation

This document is part of the Niagara technical documentation library. Released versions of Niagara software include a complete collection of technical information that is provided in both online help and PDF format. The information in this document is written primarily for Systems Integrators. In order to make the most of the information in this book, readers should have some training or previous experience with Niagara 4 or NiagaraAX software, as well as experience working with JACE network controllers.

Document Content

This document describes the procedures used to create an OPC UA Server in a station for the purpose of exposing control points, alarms, and histories to OPC UA clients. This document also describes the procedures used to create an OPC UA Client in the station which provides connectivity to OPC UA Servers for the purpose of integrating Opc UA data into the Niagara Framework for monitor and control operation.

Related Links

- [Document change log](#)
- [Related documentation](#)

Document change log

Changes to this document are listed in this topic.

- December 19, 2017: Initial publication

Related Links

- [About this Guide \(Parent Topic\)](#)

Related documentation

This topic lists documents that are related to this guide.

- Niagara Drivers Guide
- Niagara 4 Platform Guide

Related Links

- [About this Guide \(Parent Topic\)](#)

Overview

OPC is the interoperability standard for secure and reliable data exchange in the industrial automation space, as well as other industries. The OPC standard is a series of specifications that define the interface between Clients and Servers, as well as Servers and Servers, including access to real-time data, monitoring of alarms and events, access to historical data and other applications.

The OPC standard, initially restricted to the Windows operating system, derives its acronym from OLE (object linking and embedding) for Process Control. These specifications are now known as OPC Classic. The OPC Unified Architecture (UA) is a platform independent service-oriented architecture that integrates all the functionality of the individual OPC Classic specifications into an extensible framework.

The Opc UA Driver is the Niagara implementation of this service-oriented architecture.

Related Links

- [Driver component model](#)
- [Requirements](#)
- [Compatibilities](#)

Driver component model

The OPC UA Server driver and the OPC UA Client driver both follow the Niagara Driver Framework model.

OpcUaServer driver component model

The two main components are the OpcUaServer and OpcUaNamespace. The OpcUaServer component is used to model an Opc UA Server instance in the driver framework. There should be only one instance of the OpcUaServer in a station. The OpcUaNamespace component is used to model an Opc UA Namespace in the parent OpcUaServer. It is typically used to provide a logical grouping of OPC UA variables, histories, and events that can be accessed by an OPC UA client.

OpcUaClient driver component model

The primary OpcUaClient driver components are:

- OpcUaNetwork is the parent container for all of Opc UA Client devices being modeled.
- OpcUaDevice: is the Niagara representation of an Opc UA Client.
- OpcUaClientPointDeviceExt is the device extension that contains proxy control points used to proxy values to and from the Opc UA Server.
- OpcUaClientAlarmDeviceExt is the device extension provides the ability to subscribe to alarm events from the Opc UA Server.
- OpcUaClientHistoryDeviceExt is the device extension provides the ability to import histories from the Opc UA Server.
- OpcUaClientProxyExt is the control point proxy ext used to identify a specific Opc UA Server data variable to proxy through this control point.
- ImportHistoryExt is the history point extension that is used to import the history data associated with the specific Opc UA Server data variable defined in the proxy ext.

Related Links

- [Overview \(Parent Topic\)](#)

Requirements

This topic describes the licensing and software requirements for using the Niagara OPC UA Driver.

Software requirements

The Niagara OPC UA Server and Client Drivers are available for Niagara 4.3 or later.

OPC UA Server driver required modules

The following modules must be installed on the host in order to use the OPC UA Server driver.

- `opcUaCore-rt` — contains Java Class files and resources common to both OPC UA client and OPC UA server functionality.
- `opcUaServer-rt` — contains Java Class files and resources that support OPC UA Server run-time functionality.
- `opcUaServer-wb` — contains Java Class files and resources that support OPC UA Server Workbench functionality.

OPC UA Client driver required models

The following modules must be installed on the host in order to use the OPC UA Client driver.

- `opcUaCore-rt` — contains Java Class files and resources common to both OPC UA client and OPC UA server functionality.
- `opcUaClient-rt` — contains Java Class files and resources that support OPC UA Client run-time functionality.
- `opcUaClient-wb` — contains Java Class files and resources that support OPC UA Client Workbench functionality.

License Requirements

The `opcUaClient` module is a licensed feature. You must have a target controller host that is licensed with the “`opcUaClient`” feature. In addition, other OPC UA Client device, proxy point, or history limits may exist in your license.

The `opcUaServer` module is a licensed feature. If intending to serve data via OPC UA you must have a target controller host that is licensed with the “`opcUaServer`” feature. Additionally, other `opcUaServer` device limits or proxy point limits, or history limits may exist in your license.

You can check to see if your software installation is licensed for `opcUaClient` and/or `opcUaServer` by opening the license file from the License Manager view. The feature name is present only if your platform is licensed for the feature, as shown here:

```
< feature name="opcUaClient" expiration="2020-12-31" history.limit="" point.limit="" device.limit="" / >< feature name="opcUaServer" expiration="2020-12-31" history.limit="" device.limit="" / >
```

Related Links

- [Overview \(Parent Topic\)](#)

Compatibilities

OPC UA is a platform-independent, service-oriented architecture specification. It integrates all functionality from existing OPC Classic specifications, providing a more secure and scalable solution. The specification is backwards compatible with OPC Classic.

Operating systems

The OPC UA driver software functions on QNX-based controllers or Windows operating systems, starting with Windows 7 Service Pack 1 and later.

Niagara Platforms

The OPC UA driver software functions on any QNX-based platform running Niagara 4.3 or later (ex., JACE-8000), as well as Windows based platforms running Niagara 4.3 or later, such as a PC licensed for the OPC UA Client and/or OPC UA Server.

Related Links

- [Supported Opc UA profiles](#)
- [Supported OPC UA facets](#)
- [Overview \(Parent Topic\)](#)

Supported Opc UA profiles

- UA Generic Client Profile
- UA Data Access Client Profile
- UA History Data Access Client Profile
- UA Alarm

Related Links

- [Compatibilities \(Parent Topic\)](#)

Supported OPC UA facets

The following table lists the OPC UA Client facets supported by the Niagara OPC UA driver. Note that this is a subset of available OPC UA facets.

OPC UA Facet	Description
Base Client Behavior Facet	This Facet indicates that the Client supports behavior that Clients shall follow for best use by operators and administrators. They

OPC UA Facet	Description
	include allowing configuration of an endpoint for a server without using the discovery service set; Support for manual security setting configuration and behavior with regard to security issues; support for Automatic reconnection to a disconnected server.
Core Client Facet	This Facet defines the core functionality required for any Client. This Facet includes the core functions for Security and Session handling.
AddressSpace Lookup Client Facet	This Facet defines the ability to navigate through the AddressSpace and includes basic AddressSpace concepts, view and browse functionality and simple attribute read functionality.
Attribute Read Client Facet	This Facet defines the ability to read Attribute values of Nodes.
DataChange Subscriber Client Facet	This Facet defines the ability to monitor Attribute values for data change.
DataAccess Client Facet	This Facet defines the ability to utilize the DataAccess Information Model, i.e., industrial automation data like analog and discrete data items and their quality of service.
Alarm and Event Client Facet	Partially supported. This Facet defines the ability to subscribe for Event Notifications. This includes basic AddressSpace concept and the browsing of it, adding events and event filters as monitored items and adding subscriptions.
Method Client Facet	This Facet defines the ability to call arbitrary Methods.
Historical Access	This Facet defines the ability to read, process, and update historical data.

Related Links

- [Compatibilities \(Parent Topic\)](#)

Opc Ua Server tasks

The following topics describe how to set up an OPC UA Server, add server points with alarm and history extensions, and configure the server for OPC UA Client users.

Following are Opc Ua server tasks.

Related Links

- [Setting up the OPC UA Server](#)
- [Adding a server point with alarm and history extensions](#)
- [Configuring the server to support an Opc Ua Client user](#)
- [Verifying set up using client simulation software](#)

Setting up the OPC UA Server

This procedure describes the steps to set up the OPC UA Server on the station.

Prerequisites:

- opcUaCore-rt and opcUaServer (-rt,-wb) modules are installed
- Connected to the station
- opcUaServer palette is open
- Internet connectivity

NOTE: If you intend to install only the OPC UA Client driver then skip ahead to the chapter, “OPC UA Client tasks”.

The example client used in this procedure is the ProSys Simulation OPC UA Client. However, many other simulation OPC UA clients are available. Typically in an actual job, other 3rd party OPC UA clients will connect to the Niagara OPC UA Server installed on your station.

Perform the following steps:

1. Confirm that the JACE controller is licensed with the “opcUaServer” feature.
2. Add the OpcUaServer component from the palette to the Drivers folder in the station (**Station > Config > Drivers**).
The OpcUaServer is added to the station. By default, the Enabled field is True, and if this is the initial server setup then server Status will likely be {ok}.
3. Add an OpcUaNamespace component to the OpcUaServer (either drag from the palette or double-click on the OpcUaServer to open the Opc Ua Server Device Manager view and click **New > OK**).

A second New dialog appears containing several component configuration fields.

4. Configure the following fields as needed and click **Save**.
 - Name: [enter a name for the server’s Namespace or use the default name](#).

- Type: **Opc Ua Namespace** is the default type and the only available option.
- Namespace Url: [use the default value].
- Enabled: **true by default**.

The new namespace appears in the view.

5. In the Nav tree, right-click on the added OpcUaServer component to open a property sheet view and note the server's Opc Tcp Connection Address value, as shown.

▶ Opc Tcp Endpoint	Opc Tcp Endpoint
▶ User Authentication Methods	<input checked="" type="checkbox"/> Anonymous <input checked="" type="checkbox"/> Username/Password
▶ Max Session Count	500
▶ Max Session Timeout	+00000h 05m 00s
▶ Max Subscription Count	50
▶ Opc Tcp Connection Address	opc.tcp://IE3BLT1GKS6C2.global.ds.honeyw
▶ Server Info	Opc Ua Build Info
▼ Session Info	No active sessions

This connection address is required by any OPC UA Client attempting to connect to the server.

NOTE: The port specified in the Opc Tcp Connection Address may be blocked by your PC/network firewall. The firewall settings may need to be adjusted to allow data transfer on this port.

The OPC UA Server set up is complete. For any OPC UA Client connection to succeed, the client must be configured with the server's Opc Tcp Connection Address, as well as a Security Mode, Security Policy, and User Authentication method allowable by the server.

NOTE: Username and password values also must be defined in the station's UserService.

Related Links

- [Opc Ua Server tasks \(Parent Topic\)](#)

Related References

- [opcUaServer-OpcUaServer](#)
- [opcUaServer-OpcUaNamespace](#)

Adding a server point with alarm and history extensions

This procedure describes how to add server point with alarm and history extensions. Server points provide live and historical data as well as alarm events. Server points are visible to a connected OPC UA clients. Note that this procedure assumes that the station does not already have control points available.

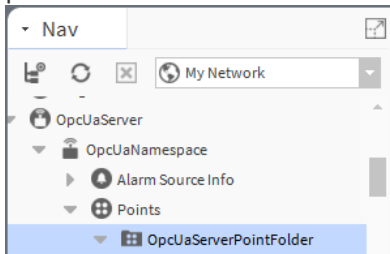
Prerequisites:

- Connected to a station with configured OpcUaServer driver
- opcUaServer palette

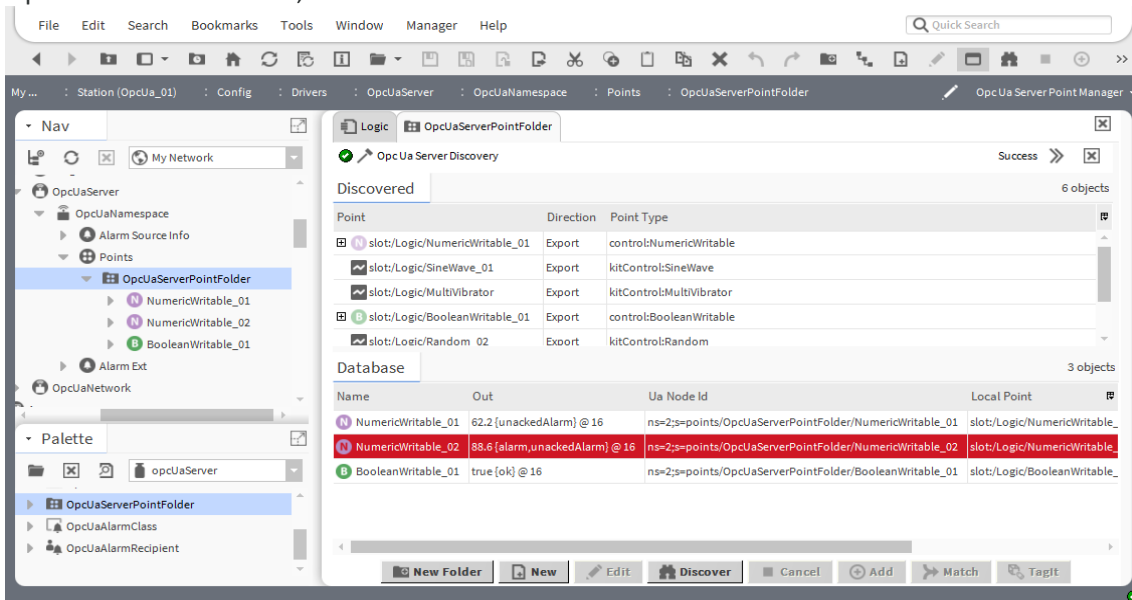
NOTE: Adding a history extension to a control point in the OpcUaServer's Namespace makes the associated history visible to a connected OPC UA client. Similarly, adding an alarm extension to a control point in the OpcUaServer's Namespace and setting the alarm extension's AlarmClass to OpcUaAlarmClass sends OPC UA events to a connected OPC UA client that subscribes for these events.

Perform the following steps:

1. In the Nav tree, click to expand the OpcUaServer and OpcUaNamespace components.
2. In the Points Folder under the OpcUaNamespace, add an OpcUaServerPointFolder from the palette.



3. Under the station Drivers folder, add a new NumericWritable point and configure the point with a NumericCov history extension and an OutOfRangeAlarmExt alarm extension.
4. Double-click the OpcUaServerPointFolder to open the Opc Ua Server Point Manager view, click **Discover**, and in the ORD selection select the folder to search and click **OK**. The Discovered pane presents a list of control points that exist in the station that can be mapped to the OpcUaServerPointFolder.
5. Select the NumericWritable in the Discovered pane and click **Add**. The point is added to the Database pane and becomes visible under the OpcUaServerPointFolder, as shown here.



By default, a discovered writable point has the “Export” direction, the point is exported from the Server as read-only. However, you can add a writable point that has the “Import” direction,

the point (which is writable by the Client) is imported to the Server which is able to read the data from the Client. To do this, expand the writable point in the Discovered pane and select the point with a Direction value of "Import" and click Add.

After adding the point in the server, you can add Niagara history or alarm extensions to enable histories or alarms for the point.

The server points in this OpcUaServerPointFolder are served up as consumable data, including live and historical data and alarm events, that is visible to a connected OPC UA client.

Related Links

- [Opc Ua Server tasks \(Parent Topic\)](#)

Related References

- [opcUaServer-OpcUaServerAlarmDeviceExt](#)

Configuring the server to support an Opc Ua Client user

This procedure defines a user for third party Opc Ua Clients when connecting to the Niagara OpcUaServer. The authentication scheme assures that the identity of a user can be verified and may use roles to limit user access only to certain areas. User authentication is achieved when the client passes user credentials to the server which is already configured for that user. Any new OPC UA client users should be authenticated via the Opc Ua Authentication Scheme.

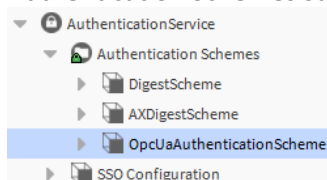
Prerequisites:

- Connected to a running station, already configured with the OpcUaServer
- The opcUaServer palette is open

NOTE: When making any server-side changes on the OPC UA Server, you must first disable and then re-enable the server.

Perform the following steps:

1. In the Nav tree, expand the station's Config and Drivers nodes, and open a Property Sheet view on the OpcUaServer component.
2. Under the Enabled property, click on the dropdown list and click **false**. The OpcUaServer is disabled.
3. In the Nav tree, expand the station's Services and AuthenticationService nodes.
4. Drag the OpcUaAuthenticationScheme component from the opcUaServer palette to the station Authentication Schemes subfolder under AuthenticationService, as shown here.



5. Double-click UserService node to open the User Manager view.

6. In the User Manager view, click **New** and in the New popup window click **OK** to add a single new user.
7. In the second **New** window, configure these properties:
 - a. Enter a user Name
 - b. Under `Authentication Scheme Name`, click the dropdown list and click on the **OpcUaAuthenticationScheme**.
 - c. In the **Password** and **Confirm** fields, enter a strong password.
A strong password requires at least 10 characters, plus at least one of each of the following characters: lowercase, uppercase, and a digit.

NOTE: If you use the same default password for all new users be sure to set the `Force Reset At Next Login` value to "true".

8. Click **Ok**.
9. Under the station Drivers node, re-open a Property Sheet view on the OpcUaServer component, and set `Enabled` value to **true** and click **Save**.

Related Links

- [Opc Ua Server tasks \(Parent Topic\)](#)

Related References

- [opcUaServer-OpcUaAuthenticationScheme](#)

Verifying set up using client simulation software

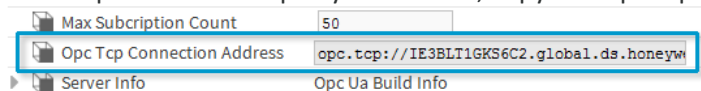
This procedure describes the steps to verify the OpcUaServer set up using the ProSys OPC UA Client software. Several other OPC UA Clients are available. The basic procedure would remain the same.

Prerequisites:

- Connected to the station
- Configured N4OpcUaServer
- OPC UA Client simulation software installed and running
- Internet connectivity

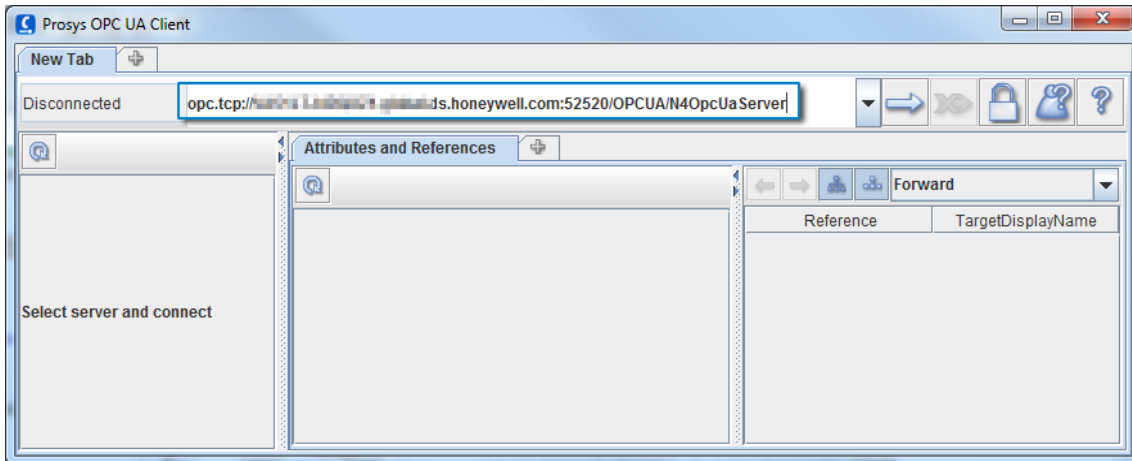
Perform the following steps:

1. In the OpcUaServer Property Sheet view, copy the Opc Tcp Connection Address, as shown.

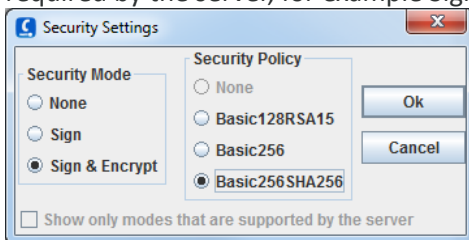


NOTE: The port specified in the Opc Tcp Connection Address may be blocked by the PC/ network firewall. The firewall settings may need to be adjusted to allow data transfer on this port.

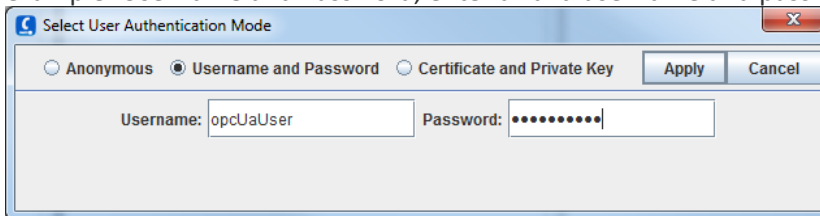
2. In the simulation client window, paste the Opc Tcp Connection Address (copied in step 1) into the available field, as shown.



- Click on Security Options (🔒) and select a security mode and security policy setting as required by the server, for example Sign & Encrypt and Basic256SHA256, and click **Ok**.



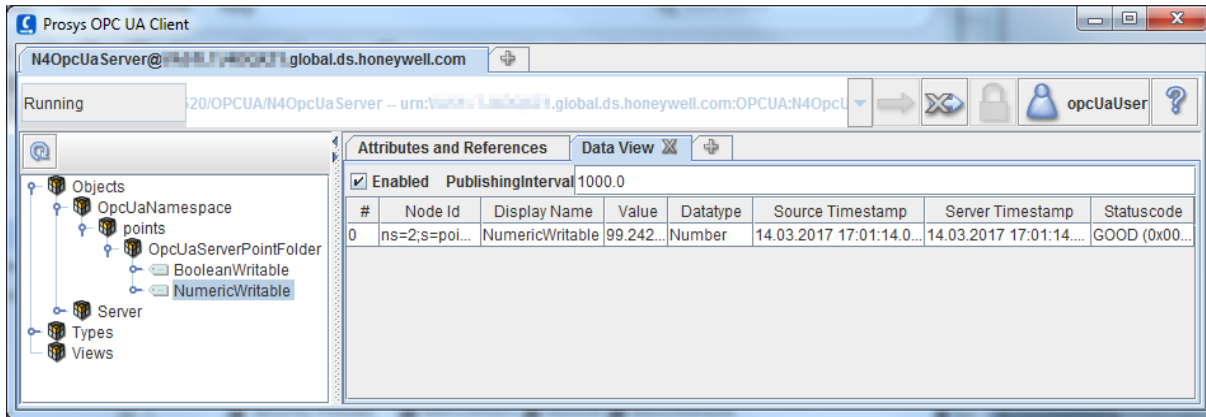
- Click on User ID (👤) and select a user authentication mode as required by the server, for example: Username and Password, enter a valid username and password and then click **Apply**.



NOTE: The Username and Password values must be for a valid username and password that is defined in the station's Users Service.

- Click Connect (➡) to connect to the server.

The client successfully connects to the OpcUaServer. In the example shown, expanded Objects reveal the OPC UA Namespace with Server points and the right-side Data View tab allows you to monitor data values for selected points.



Related Links

- [Opc Ua Server tasks \(Parent Topic\)](#)

Related References

- [Third-party simulation software](#)

Opc Ua Client tasks

The following topics describe how to set up the OPC UA Client, connect to an OPC UA Server, discover and add points with alarm and history extensions, and subscribe for OPC UA alarm events.

Following are OPC UA client tasks.

Related Links

- [Adding the OPC UA Network](#)
- [Connecting to an OPC UA Server](#)
- [Discovering OPC UA Server Points](#)
- [Adding Points to the Station database](#)
- [Adding points containing OPC UA Histories](#)
- [Adding control point to invoke an OPC UA Method](#)
- [Adding Points for OPC UA Object Arrays](#)
- [Adding an OPC UA Folder with child OPC UA Variables](#)
- [Importing OPC UA Histories without using a Control Point](#)
- [Subscribing for OPC UA Alarm Events](#)

Adding the OPC UA Network

This procedure describes steps to add an OPC UA Network to the station.

Prerequisites:

- opcUaCore-rt and opcUaClient (-rt, -wb) modules are installed
- Controller is licensed with the opcUaClient feature
- Connection to the station
- opcUaClient palette
- Internet connectivity

Perform the following steps:

1. In the Workbench the Nav Tree, expand the **Station > Config** nodes and double click on **Drivers**.
2. In the Driver Manager view, click on the **New** button.
3. In the New dialog, select **OpcUaNetwork** from the dropdown list and click **OK**.

The OpcUaNetwork is added to the station, and appears in the Driver Manager view. By default, the Enabled field is True, and if this is the initial network setup then network Status will likely be {down} until an OpcUaDevice is added and server connection data is configured.

Related Links

- [Opc Ua Client tasks \(Parent Topic\)](#)

Related References

- [opcUaClient-OpcUaNetwork](#)

Connecting to an OPC UA Server

This procedure describes steps to add an OpcUaDevice to the network and configure the device to connect to an OPC UA Server.

Prerequisites:

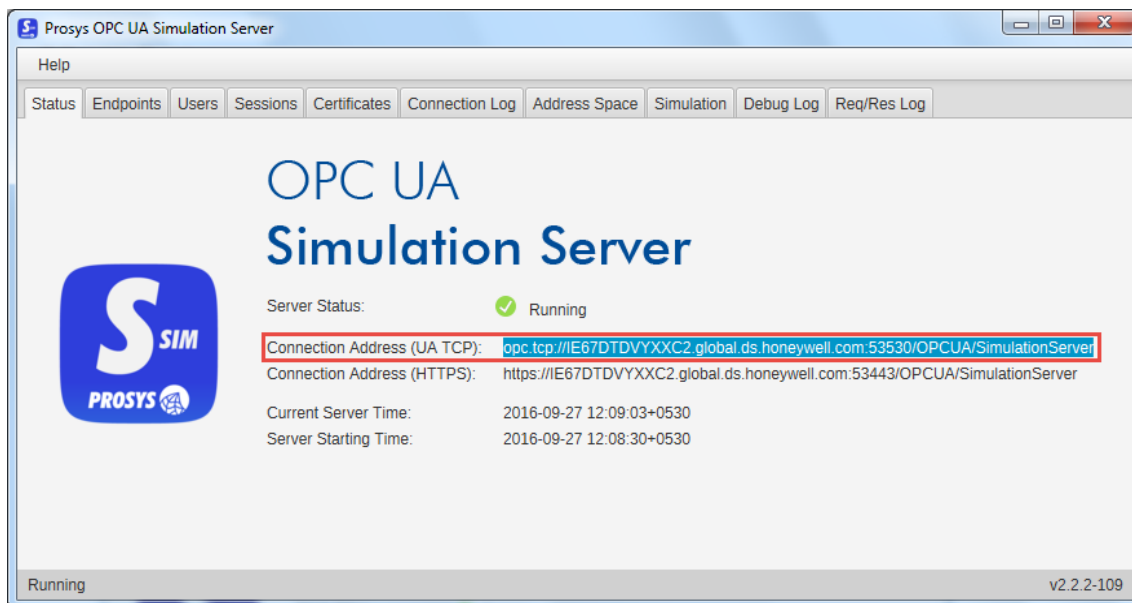
- opcUaCore-rt and opcUaClient (-rt, -wb) modules are installed
- Controller is licensed with the opcUaClient feature
- Connection to the station
- opcUaClient palette
- Internet connectivity

NOTE: The example server used for this procedure is the ProSys Simulation Server. However, it is more likely that you will open a connection to an OPC UA Server known to you that serves-up actual historical and live data.

Perform the following steps:

1. Identify and copy the server's Opc Tcp connection address (opc.tcp://*), as well as the required security mode and user authentication method.

For example, when using the ProSys Simulation Server copy the connection address from the Server Status screen, as shown here.



NOTE: Within Niagara the default configuration for the OPC UA Server and Client is security mode "Sign and SignEncrypt" and security policy "Basic256SHA256". These are the

recommended settings for high security. Other security policies and modes are supported within the Niagara OPC UA driver with a warning that alerts users to the security risk involved. The user or administrator must acknowledge this alert to proceed and this is logged in the system for audit purposes.

2. In the Workbench the Nav Tree, expand the **Station > Config > Drivers** nodes and double-click on the OpcUaNetwork.
3. In the Opc Ua Client Device Manager view and click on **New** to add a new device.
4. In the New dialog, select the **OpcUaDevice** from the dropdown list and click **OK**.

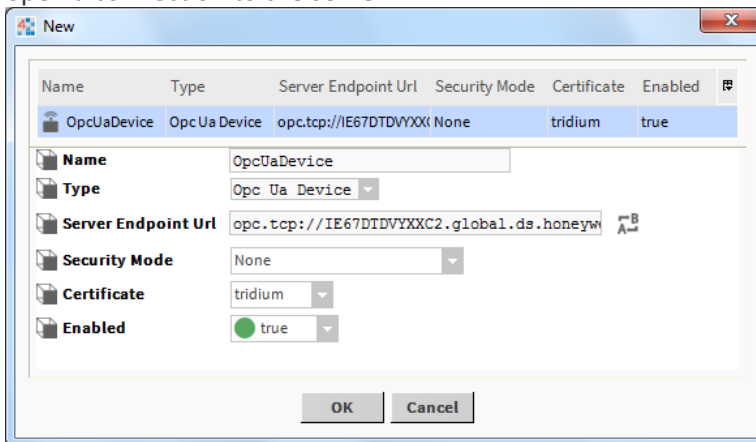
A second New dialog appears containing several device configuration fields.

5. Configure the following fields and click **OK**.
 - **Server Endpoint Url:** enter the OPC UA Server's Connection Address (identified in step 1)
 - **Security Mode:** by default this is set to "Sign Ecript Basic256 Sha256". The value must match the server's Security Mode configuration.

NOTE: The default Security Mode configuration for both OPC UA Server and Device is "Sign Ecript Basic256 Sha256" which means security modes Sign and Encrypt with security policy EncryptBasic256SHA256.

- **User Authentication Mode:** select an option available from the dropdown list, as required by the server. For example used here, the Prosys simulation server is configured for Anonymous user, as shown. Note that if you select the option "User Name And Password" then you must enter the credentials as they are configured on the server.
- **Certificate:** by default the value is "tridium". The value must match the server's Certificate configuration.

Once the server endpoint URL is entered and you click OK, the driver automatically attempts to open a connection to the server.



You can confirm that the connection to the server was successful by inspecting the property sheet of the OpcUaDevice. The Server State property should indicate "Running" and the Server Current Time, Server Start Time, and Server Info properties should be populated with the current values.

Related Links

- [Opc Ua Client tasks \(Parent Topic\)](#)

Related References

- [opcUaClient-OpcUaDevice](#)

Discovering OPC UA Server Points

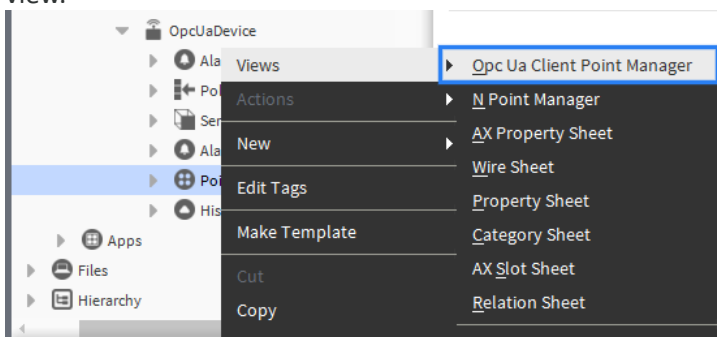
This procedure describes how to start a discover job that interrogates the connected OPC UA Server to discover the OPC UA objects being served up.

Prerequisites:

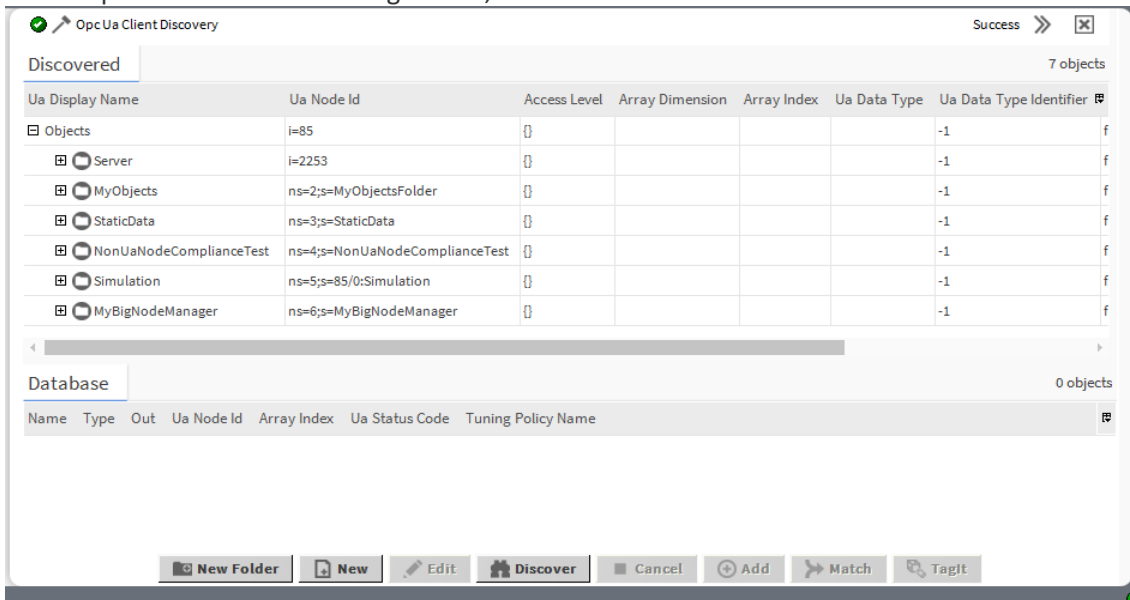
- Connected to an OPC UA Server

Perform the following steps:

1. In the Nav Tree, expand the **Station > Drivers > OpcUaNetwork > OpcUaDevice** node.
2. Right-click on the **Points** node and select **Views > Opc Ua Client Point Manager** to open the view.



3. In the Opc Ua Client Point Manager view, click on the **Discover** button.



This starts a discovery job that communicates with the OPC UA Server to discover the object structure in the server.

You can browse the discovered objects by expanding the items in the **Discovered** pane.

Related Links

- [Opc Ua Client tasks \(Parent Topic\)](#)

Related References

- [opcUaClient-OpcUaDevice](#)

Adding Points to the Station database

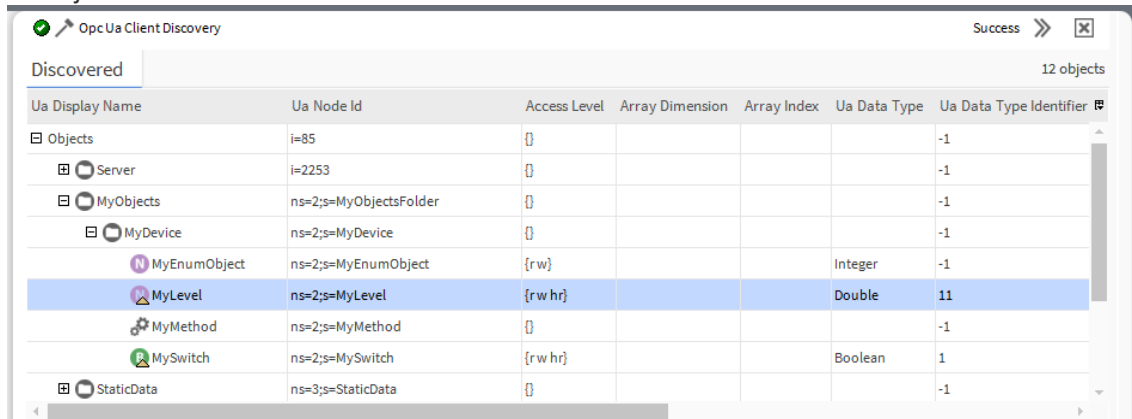
This procedure describes how to add control points to the station for selected discovered objects.

Prerequisites:

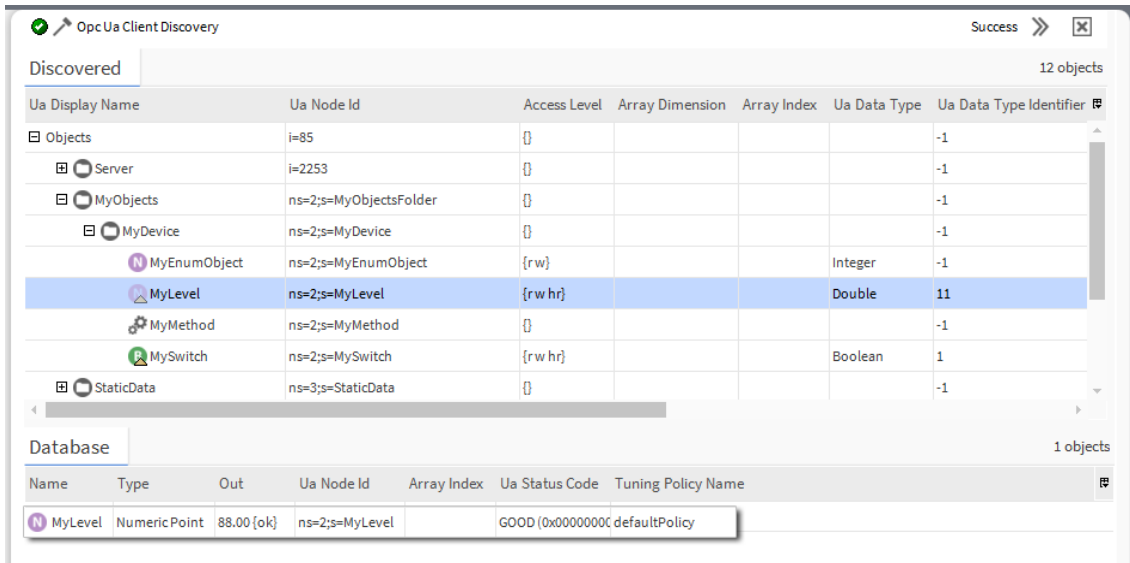
- Discovered points on the OPC UA server are visible in the Opc Ua Client Point Manager view.

Perform the following steps:

1. In the Opc Ua Client Point Manager view, expand the items in the **Discovered** pane and select an object to add.

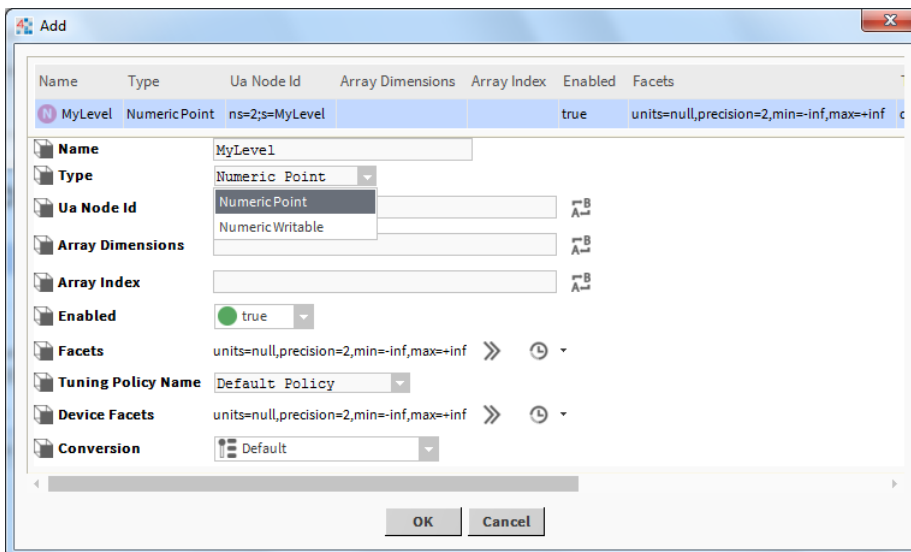


2. Click the **Add** button (or double-click on the selected object).
3. In the Add dialog, click **OK** to add the point to the Database.
The added point is entered into a subscribed state and it begins updating with real-time values, as shown here.



NOTE: If the OPC UA object has engineering units, precision, and/or range information configured, it will attempt to configure the point's facets to match. This may not always be possible.

Also, if the type of the selected object is writable, then you can change the point **Type** to be writable type, as shown below.



Related Links

- [Opc Ua Client tasks \(Parent Topic\)](#)

Related References

- [opcUaClient-OpcUaDevice](#)

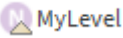
Adding points containing OPC UA Histories

If the OPC UA object being added has an OPC UA history, the point added to the station will automatically include a history extension (ImportHistoryExt).

Prerequisites:

- Discovered objects with histories on the OPC UA server visible in the Opc Ua Client Point Manager view.



The OPC UA objects that have histories can be identified in a couple of ways:

- A triangular history badge will be placed over the icon of the OPC UA object in the Discovered pane:  MyLevel
- In the Discovered pane, the **Historizing** column value will be true.

Perform the following steps:

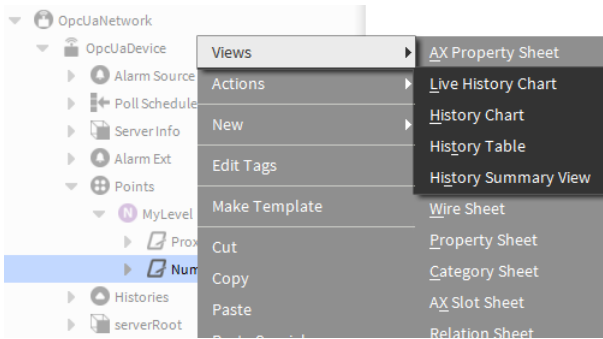
1. In the Discovered pane, locate and select one or more objects with histories, as shown here.

Opc Ua Client Discovery

Ua Display Name	Ua Node Id	Access L
Objects	i=85	{}
Server	i=2253	{}
MyObjects	ns=2;s=MyObjectsFolder	{}
MyDevice	ns=2;s=MyDevice	{}
MyEnumObject	ns=2;s=MyEnumObject	{rw}
 MyLevel	ns=2;s=MyLevel	{rw hr}
MyMethod	ns=2;s=MyMethod	{}
 MySwitch	ns=2;s=MySwitch	{rw hr}

The default history extension period for uploading history data is once in 10 minutes. This can be modified via the Execution Time property, by changing the Interval value, if required.

From the History extension you can also view the history data, as shown here.




Related Links

- [Opc Ua Client tasks \(Parent Topic\)](#)

Related References

- [opcUaClient-OpcUaClientHistoryDeviceExt](#)

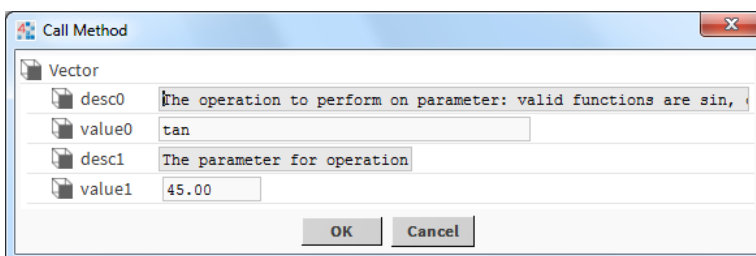
Adding control point to invoke an OPC UA Method

The driver provides a means of invoking OPC UA Method objects defined in a OPC UA server. A discovered OPC UA Method can be identified by clicking  `MyMethod`.

Prerequisites: The discovered objects on the OPC UA server are visible in the Opc Ua Point Manager view.

Perform the following steps:

1. Select an Opc Ua Method and click **Add**.
It adds an OPC UA Method in the Discovered pane. The added control point will have a `callMethod` action.
2. Right-click on the **OpC UA Method** and click **Actions > Call Method**.



If the OPC UA Method has defined input arguments for the method, a **Call Method** dialog will be displayed for the arguments. It will have a descriptor that describes each of the arguments. You may need to scroll the descriptor to see the complete text.

3. In the Call Method dialog, enter the desired argument values and click OK.

Database						
Name	Type	Out	Ua Node Id	Array Index	Ua Status Code	Tuning Policy Name
MyMethod	Opc Ua Method	1.00 [ok]	ns=2;s=MyMethod			defaultPolicy

NOTE: The OpcUaMethod control point is a subclass of BStringPoint. The out property will be a StatusString type. The data type of the method return value may not be a String. In addition to setting the out property by converting the method results to a string, it will also add a “result” property of the appropriate BStatusValue type.


This invokes the method in the OPC UA server with the provided arguments.

On completion, the added value will be reflected and displayed in the Out property for the control point.

Related Links

- [Opc Ua Client tasks \(Parent Topic\)](#)

Adding Points for OPC UA Object Arrays

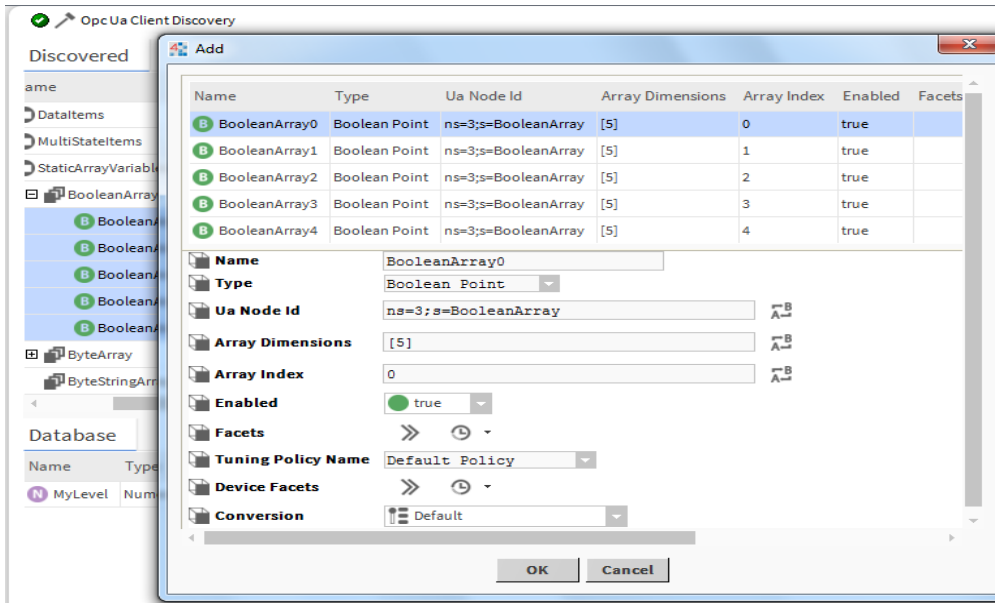
A discovered OPC UA object that is an array can be identified by:  BooleanArray . You can select and add one or more items from the discovered OPC UA object array.

Prerequisites: Discovered Boolean array objects on the OPC UA server visible in the Opc Ua Client Point Manager view.

Perform the following steps:

1. Click to expand a BooleanArray object to display the list of items.
2. Select one or more items and click on the **Add** button.

A single control point is used for an individual array item. The “Array Dimension” column indicates the size of the array. The Array Index is automatically set based on the selection.



The selected points are added to the Database.

Database							5 objects
Name	Type	Out	Ua Node Id	Array Index	Ua Status Code	Tuning Policy Name	
BooleanArray0	Boolean Point	true {ok}	ns=3;s=BooleanArray	0	GOOD (0x00000000) ""	defaultPolicy	
BooleanArray1	Boolean Point	false {ok}	ns=3;s=BooleanArray	1	GOOD (0x00000000) ""	defaultPolicy	
BooleanArray2	Boolean Point	true {ok}	ns=3;s=BooleanArray	2	GOOD (0x00000000) ""	defaultPolicy	
BooleanArray3	Boolean Point	false {ok}	ns=3;s=BooleanArray	3	GOOD (0x00000000) ""	defaultPolicy	
BooleanArray4	Boolean Point	false {ok}	ns=3;s=BooleanArray	4	GOOD (0x00000000) ""	defaultPolicy	

Related Links

- [Opc Ua Client tasks \(Parent Topic\)](#)

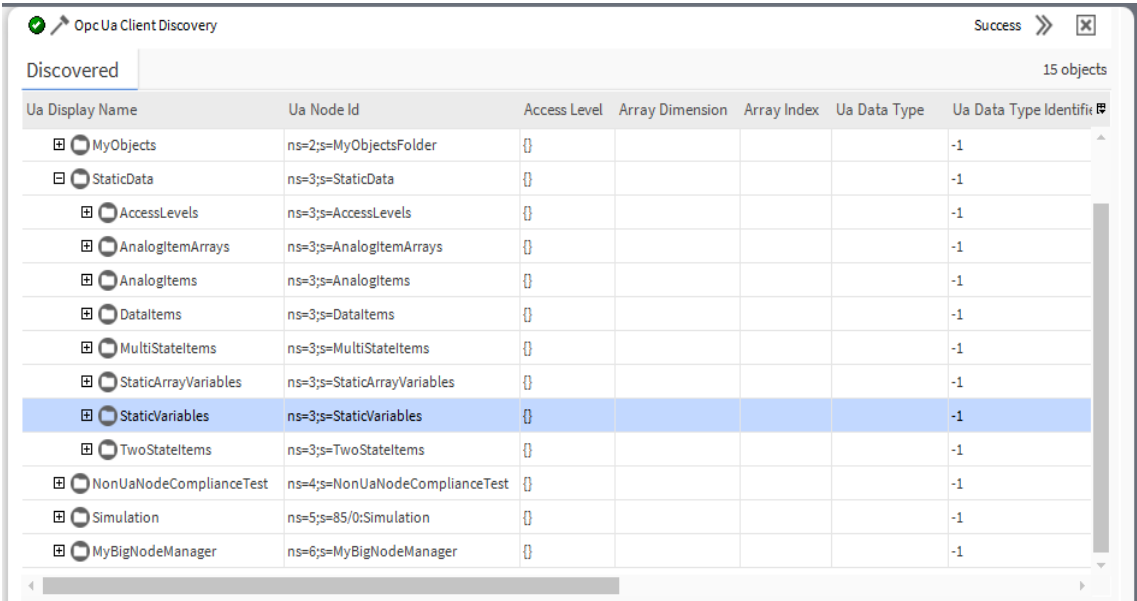
Adding an OPC UA Folder with child OPC UA Variables

It is possible to add multiple control points at once by adding a discovered OPC UA Object Folder to the station. Note that the collection of control points must be in a discoverable container.

Prerequisites: One or more discovered OPC UA Object Folders on the OPC UA server visible in the Opc Ua Client Point Manager view.

Perform the following steps:

1. Select a Discovered Opc Ua Object Folder and click on **Add** by double-clicking on the discovered row or by selecting and clicking the **Add** button.



Opc Ua Client Discovery Success

Discovered 15 objects

Ua Display Name	Ua Node Id	Access Level	Array Dimension	Array Index	Ua Data Type	Ua Data Type Identifier
MyObjects	ns=2;s=MyObjectsFolder	{}				-1
StaticData	ns=3;s=StaticData	{}				-1
AccessLevels	ns=3;s=AccessLevels	{}				-1
AnalogItemArrays	ns=3;s=AnalogItemArrays	{}				-1
AnalogItems	ns=3;s=AnalogItems	{}				-1
DataItems	ns=3;s=DataItems	{}				-1
MultiStateItems	ns=3;s=MultiStateItems	{}				-1
StaticArrayVariables	ns=3;s=StaticArrayVariables	{}				-1
StaticVariables	ns=3;s=StaticVariables	{}				-1
TwoStateItems	ns=3;s=TwoStateItems	{}				-1
NonUaNodeComplianceTest	ns=4;s=NonUaNodeComplianceTest	{}				-1
Simulation	ns=5;s=85/0:Simulation	{}				-1
MyBigNodeManager	ns=6;s=MyBigNodeManager	{}				-1

2. In the resulting Confirmation dialog, click **Yes**.

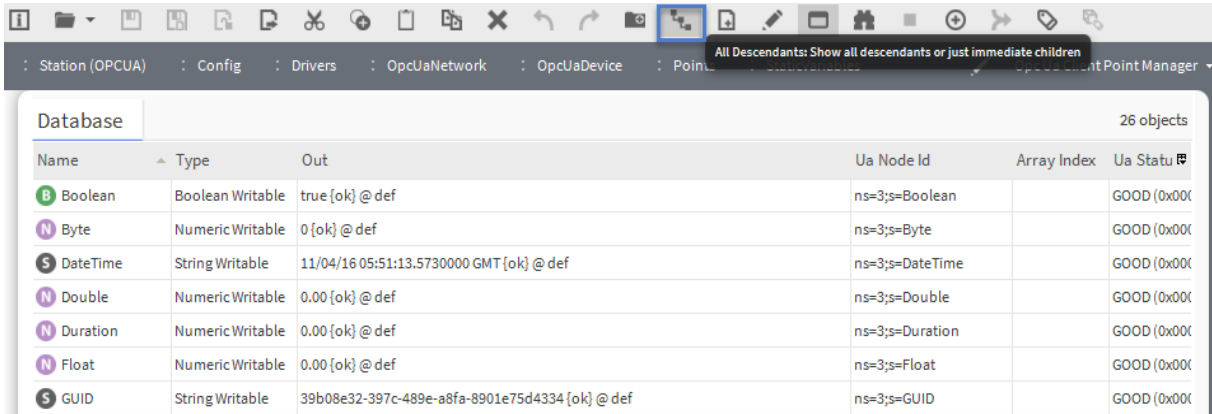


This adds a point folder and control points for all of the child OPC UA variables.

NOTE: Clicking **No** in the dialog adds only the point folder.

The added point folder in the database pane is configured to match the variables in the discovered OPC UA folder.

NOTE: The show **All Descendants** icon on the tool bar must be selected to show the child points, as shown here.



Station (OPCUA) : Config : Drivers : OpcUaNetwork : OpcUaDevice : Point Manager

All Descendants: Show all descendants or just immediate children

Database 26 objects

Name	Type	Out	Ua Node Id	Array Index	Ua Status
Boolean	Boolean Writable	true {ok} @ def	ns=3;s=Boolean		GOOD (0x0000)
Byte	Numeric Writable	0 {ok} @ def	ns=3;s=Byte		GOOD (0x0000)
DateTime	String Writable	11/04/16 05:51:13.5730000 GMT {ok} @ def	ns=3;s=DateTime		GOOD (0x0000)
Double	Numeric Writable	0.00 {ok} @ def	ns=3;s=Double		GOOD (0x0000)
Duration	Numeric Writable	0.00 {ok} @ def	ns=3;s=Duration		GOOD (0x0000)
Float	Numeric Writable	0.00 {ok} @ def	ns=3;s=Float		GOOD (0x0000)
GUID	String Writable	39b08e32-397c-489e-a8fa-8901e75d4334 {ok} @ def	ns=3;s=GUID		GOOD (0x0000)

Related Links

- [Opc Ua Client tasks \(Parent Topic\)](#)

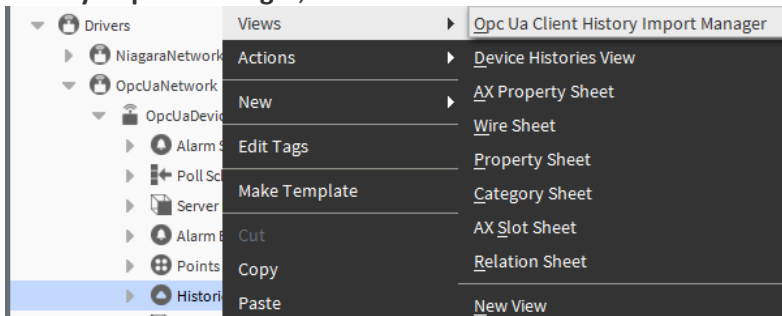
Importing OPC UA Histories without using a Control Point

The OpcUaDevice component's "Histories" OpcUaClientHistoryDeviceExt component provides the ability to import OPC UA histories without creating a control point.

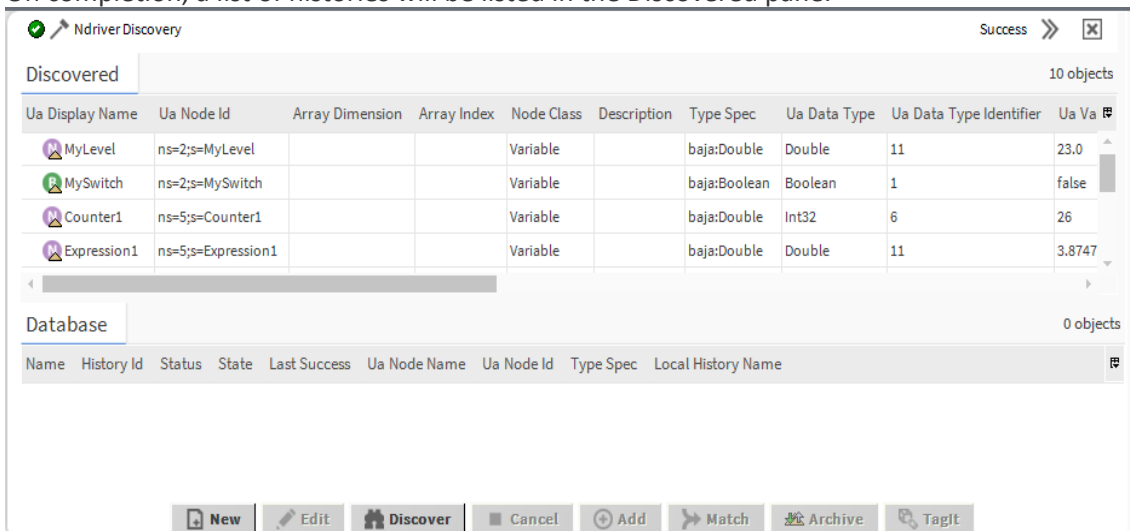
WARNING: Be aware that it is possible to duplicate an OPC UA History import by unintentionally using both the OpcUaClientHistoryImport and a Control Point with an ImportHistoryExt for the same discovered object. See "Adding Points that also have OPC UA Histories".

Perform the following steps:

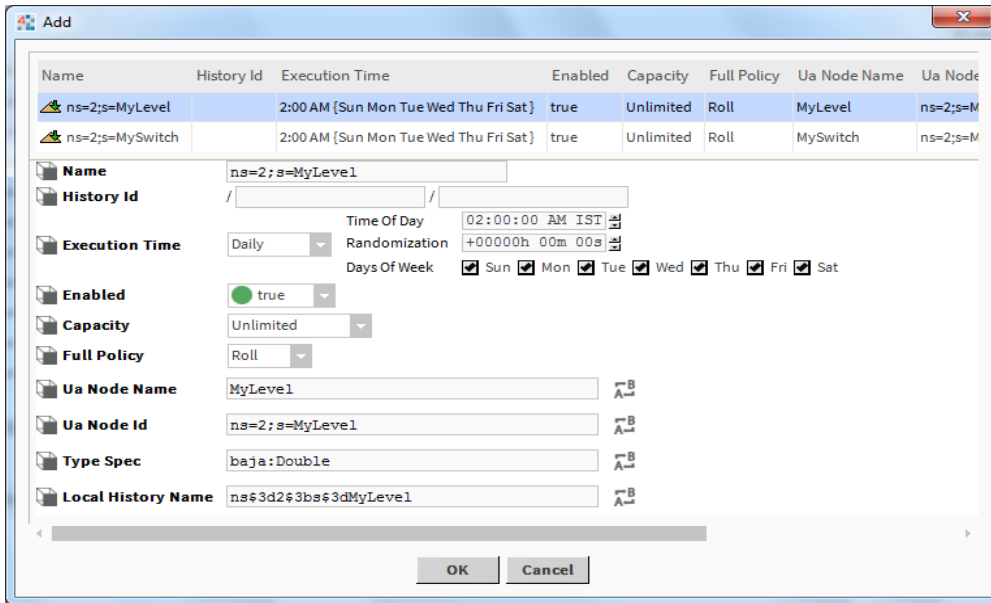
1. In the Nav tree, right-click on **Histories** under the OpcUaDevice and click **Views > Opc Ua Client History Import Manager**, as shown.



2. In the view, click **Discover** and wait a few moments for the job to run. On completion, a list of histories will be listed in the Discovered pane.



3. In the Discovered pane, select one or more histories to be imported and click **Add**.
4. In the **Add** dialog, modify properties to setup the history import as desired and click **OK**.



The histories are imported to the station's history database at the time specified in the Execution Time property shown above.

Related Links

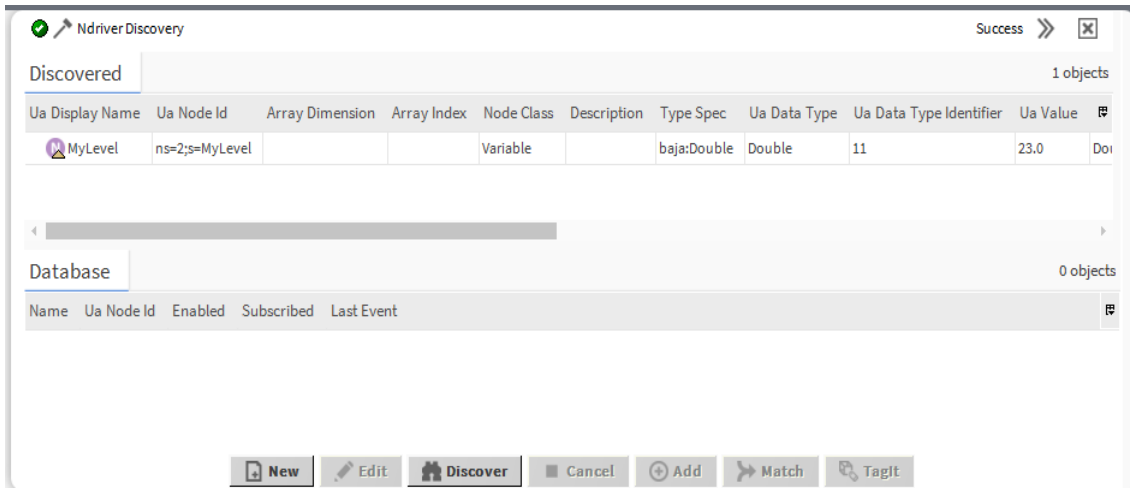
- [Opc Ua Client tasks \(Parent Topic\)](#)

Subscribing for OPC UA Alarm Events

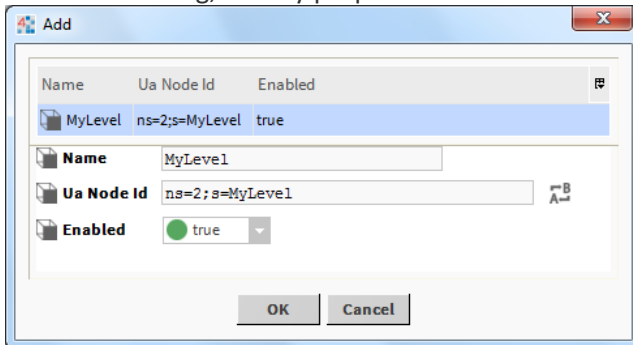
The OpcUaDevice component's "Alarm Ext" OpcUaClientAlarmDeviceExt component provides the ability to subscribe to OPC UA alarm events.

Perform the following steps:

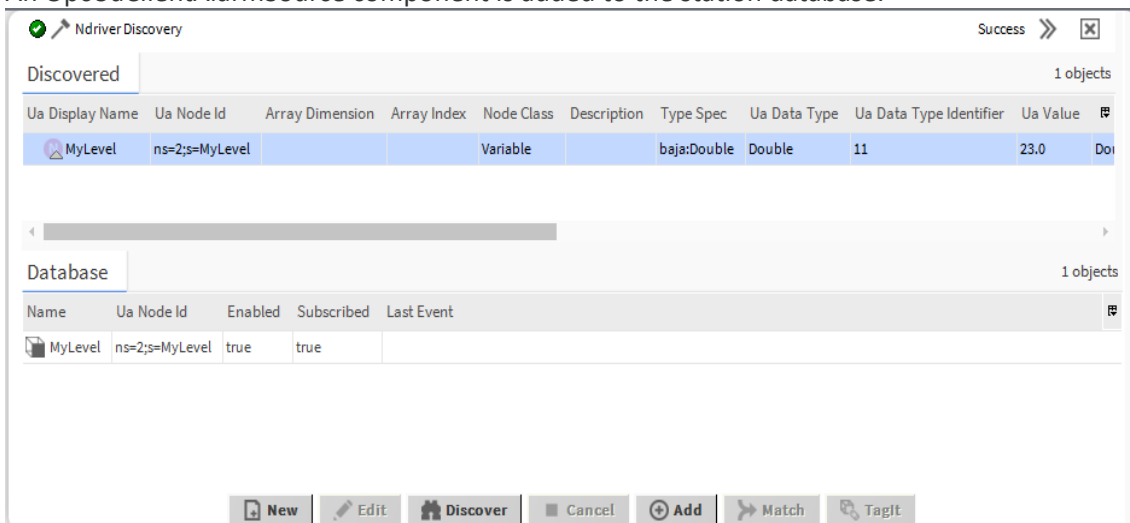
1. In the Nav tree, right-click on **Alarm Ext** under the OpcUaDevice and click **Views > Opc Ua Client Alarm Manager**, as shown.
2. In the view, click **Discover**. It will scan the Opc UA server looking for OPC UA Variables that are "alarmable".



3. In the Discovered pane, select the variable to be imported and click **Add**.
4. In the **Add** dialog, modify properties as desired and click **OK**.



An OpcUaClientAlarmSource component is added to the station database.



NOTE: The Last Event column displays a summary of the last alarm event received from the OPC UA Server for the specified OPC UA node.

5. The added OpcUaClientAlarmSource contains an AlarmSourceInfo component. The AlarmSourceInfo can be configured to route the received Opc Ua Alarm Events to Alarm Recipients.

Property Sheet

MyLevel (Opc Ua Client Alarm Source)

Status	{ok}
Enabled	<input checked="" type="checkbox"/> true
Ua Node Id	ns=2;s=MyLevel
Subscribed	<input checked="" type="checkbox"/> true
Last Event	07-Nov-16 3:29 PM IST ns=2;s=MyLevel.Ala.
Alarm Source Info	Alarm Source Info
Alarm Class	Default Alarm Class
Source Name	\$parent.displayName\$?
To Fault Text	? ?

The Enabled property is used to subscribe or unsubscribe to receive Opc Ua Alarm Events on the given OPC UA Node.

When a new OPC UA alarm event is received it will be routed through the specified AlarmClass in the station's Alarm Service. It is then treated as any other alarm.

Related Links

- [Opc Ua Client tasks \(Parent Topic\)](#)

Related References

- [opcUaClient-OpcUaClientAlarmDeviceExt](#)

Opc Ua Reference

The following topics provide conceptual information on the OPC UA security architecture, including descriptions of the security policies, security mode, security profiles and user authentication.

Also covered is information on point type and facet mapping, server alarm processing and alarm acknowledgment processing, links for third-party simulation software, as well as descriptions of the components and views present in the opcUaServer and opcUaClient modules.

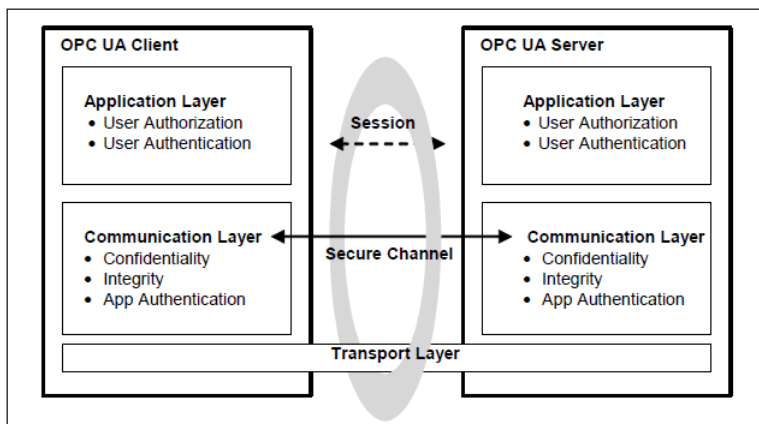
Related Links

- [About the security architecture](#)
- [Point type mapping](#)
- [Point facet mapping](#)
- [OpcUaServer alarm processing](#)
- [OpcUaServer alarm acknowledgment processing](#)
- [Third-party simulation software](#)
- [Components and views](#)

About the security architecture

The OPC UA security architecture is structured in an application layer and a communication layer on top of the transport layer, as shown here.

Figure 1. OPC UA Security Architecture



The routine work of a Client application and a Server application to transmit information, settings, and commands is done in a session in the application layer. The application layer also manages the security objectives user authentication and user authorization. The application layer communicates over a secure channel that is created in the communication layer and relies upon it for secure communication. The secure channel provides encryption to maintain confidentiality, message

signatures to maintain integrity and digital certificates to provide application authentication for data that comes from the application layer and passes the “secured” data to the transport layer.

Related Links

- [Security policies](#)
- [Security mode](#)
- [Security profiles](#)
- [User authentication](#)

- [Opc Ua Reference \(Parent Topic\)](#)

Security policies

A security policy specifies which security mechanisms are to be used in OPC UA. Security policies are used by the Server to announce which mechanisms it supports and by the Client to select one to use with the secure channel it wishes to open.

Security policies include the following information:

- algorithms for signing and encryption
- algorithm for key derivation

The choice of security policy is normally made by the administrator typically when the Client and Server products are installed. The available security policies are listed below

- None — This defines a policy for configurations with the lowest security needs. This policy will result in NO channel security during communication. The specifications recommend that, by default this Security Policy should be disabled if any other Security Policies are available.
- Basic128RSA15 — This defines a policy for configurations with medium security. It uses RSA15 as Key-Wrap-algorithm and 128-Bit for encryption algorithms.
- Basic256 — This defines a policy for configurations with medium to high security needs. A suite of algorithms that are for 256-Bit encryption.
- Basic256SHA256 — The default configuration for OPC UA server and client in Niagara, this defines a policy for configurations high security needs.

Related Links

- [About the security architecture \(Parent Topic\)](#)

Security mode

The Security Mode is an enumeration that specifies what security should be applied to message exchanges during a session. Following are the security modes supported by OPC UA.

- None — This security mode means No security is applied.
- Sign — This security mode means all messages are signed but not encrypted. In this mode the messages are transported as plain text, but the signature or signing the message allows detection if it has been manipulated by any third party.
- Sign and Encrypt — The default configuration for the OPC UA server and client, this security mode means all messages are signed and encrypted. In this mode the messages are encrypted

and not transported as plain text. The signature or signing the message allows detection if it has been manipulated by any third party.

Related Links

- [About the security architecture \(Parent Topic\)](#)

Security profiles

In the Niagara OPC UA implementation the combination for Security mode and Security Policy form a Security Profile, different Profiles specify different details such as which encryption algorithms are required for which OPC UA functions. Some of the Profiles specify security functions and others specify other functionality that is not related to security. The following security profiles are available in Niagara Opc Ua.

- None
 - No security mode or no security profile, NO channel security during commutation.
- Sign with Security Policy
 - SignBasic128RSA15 — Medium channel security during commutation with message transported as plain text with a digital signature.
 - SignBasic256 — Medium to high channel security during commutation with message transported as plain text with a digital signature.
 - SignBasic256SHA256 — high channel security during commutation with message transported as plain text with a digital signature.
- Sign and Encrypt with Security Policy
 - SignEncryptBasic128RSA15 — Medium channel security during commutation with message transported with a digital signature and encrypted.
 - SignEncryptBasic256 — Medium to high channel security during commutation with message transported with a digital signature and encrypted.
 - SignEncryptBasic256SHA256 — High channel security during commutation with message transported with a digital signature and encrypted.

NOTE: In Niagara the default configuration for OPC UA security mode is Sign and Encrypt with Security Policy: SignEncryptBasic256SHA256 The other security policies and modes will be supported within for OPC UA server and client with a warning message on the security risk involved, the User or Administrator must agree to this and it is logged in the system for audit purposes.

Related Links

- [About the security architecture \(Parent Topic\)](#)

User authentication

Authentication is a security objective that assures that the identity of an entity such as a Client, Server, or User can be verified. User Authentication is achieved when the client passes user credentials to the server.

Following is the current set of supported user authentication mechanisms for the Niagara OPC UA driver:

- Anonymous — Indicates that the client has no user credentials.

NOTE: Choosing Anonymous will invoke a warning message about the security risk involved. The User or Administrator must accept the risk and that acceptance is logged in the system for audit purposes.

- Username/password — The default User Authentication mode, the client passes simple username and password credentials to the server.

Related Links

- [About the security architecture \(Parent Topic\)](#)

Point type mapping

The following table maps the OPC UA built-in data types to Niagara point types.

OPC UA Data Type	Boolean	Enum	Numeric	String
Boolean	X			
SByte			X	
Byte			X	
Int16			X	
UInt16			X	
Int32			X	
UInt32			X	
Int64			X	
UInt64			X	
Float			X	
Double			X	
String				X
DateTime				X
Guid				X
ByteString				
XmlElement				X
NodeId				X
ExpandedNodeId				

OPC UA Data Type	Boolean	Enum	Numeric	String
StatusCode				
LocalizedText				X
ExtensionObject				
Variant (array of)	X	X	X	X
DiagnosticInfo				
Enumeration		X		

Related Links

- [Opc Ua Reference \(Parent Topic\)](#)

Point facet mapping

The OpcUaClient driver will attempt to initialize facets for a point based on the setup of the OPC UA Variable being proxied.

Control Point	Description
BooleanPoint	Will set the TrueText and FalseText if the OPC UA variable has "TrueState" and "FalseState" properties.
EnumPoint	Will set the EnumRange if the OPC UA variable has an "EnumStrings" property.
NumericPoint	Will set the Units if the OPC UA variable has an EUIInformation property and the Unit maps to a system Unit. Will set the Precision, Max, and Min values based on the OPC UA variable data type or a "Range" property if it exist for the variable.

Related Links

- [Opc Ua Reference \(Parent Topic\)](#)

OpcUaServer alarm processing

When started, the OpcUaServer automatically adds to the station's AlarmService an OpcUaServerAlarmClass and an OpcUaServerAlarmRecipient (if they don't already exist). The OpcUaServerAlarmRecipient is responsible for triggering the OPC UA event from the Niagara alarms that are routed to it via the linked OpcUaAlarmClass. Alarms routed to the OpcUaServerAlarmRecipient must come from a control point that also contains an OpcUaServerProxyExt so that it can be related to a OPC UA NodeId. The OPC UA NodeId is a property of the OpcUaServerProxyExt. The OpcUaServerAlarmRecipient also has an OpcUaSeverity property. This property is used to map the alarm's AlarmState to OPC UA alarm severity.

The following image shows the default OpcUaSeverity mapping.

▼ ● Opc Ua Severity	700, 900, 500, 600	
● To Offnormal	<input type="text" value="700"/>	[1 - 1000]
● To Fault	<input type="text" value="900"/>	[1 - 1000]
● To Normal	<input type="text" value="500"/>	[1 - 1000]
● To Alert	<input type="text" value="600"/>	[1 - 1000]

Related Links

- [Opc Ua Reference \(Parent Topic\)](#)

OpcUaServer alarm acknowledgment processing

An OPC UA Client can subscribe and receive OPC UA Events from an OPC UA server. The OPC UA Client Alarm Manager may allow the user to Acknowledge a received event back to the OPC UA server. It may also allow the user to command the OPC UA server to Enable or Disable the source of the received event.

The Acknowledge, Enable, and Disable commands are processed by the OpcUaServer's OpcUaServerAlarmDeviceExt that is a child of each Namespace component. Each OpcUaServer Namespace has a frozen slot named AlarmExt that is of type OpcUaServerAlarmDeviceExt. It will keep track of all of the alarmable points under this Namespace. When an Acknowledge is received for the OPC UA Client it attempts to locate the alarm record being acknowledged and acknowledge the alarm. When a Disable command is received it will disable the alarm from generating any more OPC UA Events from the associated control point until an Enable command is received.

Related Links

- [Opc Ua Reference \(Parent Topic\)](#)

Third-party simulation software

Third-party OPC UA simulation servers and client devices are available for purposes of testing and evaluation. This section provides a few links.

If installing only the client driver, you will need to identify an Opc UA Server for the client driver to communicate with. Following are a couple that are available.

- Prosys OPC UA Simulation Server download:
<https://prosysopc.com/products/opc-ua-simulation-server/>
- OPC Foundation Sample Applications download:
<https://opcfoundation.org/developer-tools/developer-kits-unified-architecture/sample-applications/>

If installing only the server driver, you will need to identify an OPC UA Client device for the server driver to communicate with. Following are a couple that are available.

- Prosys OPC UA Client download:
<https://www.prosysopc.com/products/opc-ua-client/evaluate/>

- Unified Automation's UA Expert OPC UA Client download:
<https://www.unified-automation.com/downloads/opc-ua-clients.html>

Related Links

- [Opc Ua Reference \(Parent Topic\)](#)

Related Tasks

- [Verifying set up using client simulation software](#)

Components and views

Components include services, folders and other model building blocks associated with a module. You drag them to a property or wire sheet from a palette. Views are plugins that can be accessed by double-clicking a component in the Nav tree or right-clicking a component and selecting its view from the **Views** menu.

The component and view topics that follow appear as context-sensitive help topics when accessed by:

- Right-clicking on the object and selecting **Views > Guide Help**
- Clicking **Help > Guide On Target**

Related Links

- [opcUaServer-OpcUaServer](#)
- [opcUaServer-OpcUaNamespace](#)
- [opcUaServer-OpcUaServerPointDeviceExt](#)
- [opcUaServer-OpcUaServerAlarmDeviceExt](#)
- [opcUaServer-OpcUaServerDeviceFolder](#)
- [opcUaServer-OpcUaServerPointFolder](#)
- [opcUaServer-OpcUaAlarmClass](#)
- [opcUaServer-OpcUaAlarmRecipient](#)
- [opcUaServer-OpcUaAuthenticationScheme](#)
- [opcUaClient-OpcUaNetwork](#)
- [opcUaClient-OpcUaDevice](#)
- [opcUaClient-OpcUaBuildInfo](#)
- [opcUaClient-OpcUaClientAlarmDeviceExt](#)
- [opcUaClient-OpcUaClientPointDeviceExt](#)
- [opcUaClient-OpcUaDeviceFolder](#)
- [opcUaClient-OpcUaClientHistoryDeviceExt](#)
- [opcUaClient-OpcUaClientPointFolder](#)
- [opcUaServer-OpcUaServerDeviceManager](#)
- [Opc Ua Server Alarm Manager](#)
- [Opc Ua Server Point Manager](#)
- [Opc Ua Client Device Manager](#)
- [Opc Ua Client Alarm Manager](#)
- [Opc Ua Client Point Manager](#)
- [Opc Ua Client History Import Manager](#)
- [Opc Ua Reference \(Parent Topic\)](#)

opcUaServer-OpcUaServer

OpcUaServer is a network-level component. It contains the device-level OpcUaNamespace component and configuration parameters necessary for communications with OpcUa client devices. Additionally, the OpcUaServer point device extension (OpcUaServerPointDeviceExt) of Opc Ua Namespace is used to add points in the Niagara Opc Ua Server. This component is found in the OpcUaServer palette.

The OpcUaServer has the standard network component properties such as health and other status properties (see “About network components” in the Drivers Guide for general information). The default view for an OpcUaServer is the Opc Ua Server Device Manager.

Properties used to configure the OpcUaServer are listed here:

Name	Value	Description
Opc Ua Server Name	text string	Name of the server- edit this as needed, or use the default name: N4OpcUaServer.
Opc Tcp Endpoint		Container for configuration subproperties
Enabled	true (default), false	
Port	52520 (default)	Port number for Opc Tcp connections. NOTE: The port specified in the Opc Tcp Connection Address may be blocked by PC/network firewall. The firewall settings may need to be adjusted to allow data transfer on this port.
Security Mode	none, sign (default), signEncrypt (default)	By default, “sign” and “signEncrypt” are selected.
Security Policies	Basic128Rsa15, Basic256, Basic256Sha256	By default, all are selected.
User Authentication Methods	Anonymous User Name and Password (default) Certificate (default) Issued Token (default)	By default, all except Anonymous are selected. Opc Ua Device settings must match the server’s settings for a successful connection. Note that username and password values must be defined in the station’s User Service.
Max Session Count	500 (default)	
Max Session Timeout	00001h 00m 00s (default)	
Max Subscription Count	50 (default)	

Name	Value	Description
Opc Tcp Connection Address	opc.tcp://URL for example (opc.tcp:// ABCD1234.global.ds.honeywell.com: 52520/OPCUA/N4OpcUaServer)	Connection address for the Opc Ua Server. The station must be running for this field to be populated. This address includes the specified port number for the Opc Tcp Endpoint.

Actions

- Ping - Sends a ping monitor request to verify device health.

Related Links

- [Components and views \(Parent Topic\)](#)

Related Tasks

- [Setting up the OPC UA Server](#)

opcUaServer-OpcUaNamespace

OpcUaNamespace is a device-level component that must be added to the OpcUaServer node. The Namespace provides a standard method for an Opc Ua Server to represent objects to Opc Ua Clients. It defines objects in terms of variables and applicable attributes. It is typically used to provide a logical grouping of OPC UA variables, histories, and events that can be accessed by an OPC UA client.

If you intend to add server points to the Opc Ua Server then you must first add a NameSpace component. Within this Namespace you can discover points on a station and add those points to the Namespace. Once that is done, an OPC UA Client can connect to the OPC UA Server and subscribe to all these points for monitoring.

This component is found in the OpcUaServer palette.

The OpcUaNamespace has the standard component properties such as health and other status properties.

Properties used to configure the OpcUaNamespace component are listed here:

Name	Value	Description
Name	text string	Use the default name or enter some other meaningful name such as HVAC, etc.
Type	text string	Opc Ua Namespace is the default type and the only available option.
Namespace Url	text string	Use the default url value.
Enabled	true (default), false	Use to enable/disable this component

Related Links

- [Components and views \(Parent Topic\)](#)

Related Tasks

- [Setting up the OPC UA Server](#)

opcUaServer-OpcUaServerPointDeviceExt

OpcUaServerPointDeviceExt (Points) is the OPC UA implementation of PointDeviceExt, a frozen device extension under every OpcUaServerNamespace. Its primary view is the Opc Ua Server Point Manager.

The Opc Ua Server Point Device Ext (Points) is also available in the opcUaServer palette.

Related Links

- [Components and views \(Parent Topic\)](#)

opcUaServer-OpcUaServerAlarmDeviceExt

This component is a child of each Namespace component. Each OpcUaServer Namespace has a frozen slot named AlarmExt that is of type OpcUaServerAlarmDeviceExt. It keeps track of all of the alarmable points under this Namespace.

When an Acknowledge is received for the OPC UA client it attempts to locate the alarm record being acknowledged and acknowledge the alarm. When a Disable command is received it disables the alarm from generating any more OPC UA Events from the associated control point until an Enable command is received.

This component is found in the OpcUaServer palette.

Related Links

- [Components and views \(Parent Topic\)](#)

Related Tasks

- [Adding a server point with alarm and history extensions](#)

opcUaServer-OpcUaServerDeviceFolder

Opc Ua Server Device Folder is the Opc Ua Server implementation of a folder under the OpcUaServer. Typically, you add such folders using the New Folder button in the Opc Ua Server Device Manager view of the OpcUaServer. The component is also available in the opcUaServer palette.

Related Links

- [Components and views \(Parent Topic\)](#)

opcUaServer-OpcUaServerPointFolder

Opc Ua Server Point Folder is the Opc Ua Server implementation of a folder under an OpcUaNamespace Points extension. You add such folders using the New Folder button in the Opc Ua Server Point Manager view of the Opc Ua Server Point Device extension. The Opc Ua Server Point Folder is also available in the opcUaServer palette.

Related Links

- [Components and views \(Parent Topic\)](#)

opcUaServer-OpcUaAlarmClass

OpcUaAlarmClass is used to group the alarms that have the same routing. It collects the alarms of OPC UA Server points if the alarm extension is configured. This component is found in the OpcUaServer palette.

Related Links

- [Components and views \(Parent Topic\)](#)

opcUaServer-OpcUaAlarmRecipient

OpcUaAlarmRecipient is used to route alarms to OPC UA Clients. This also maps alarm conditions to OPC UA Severity.

This component is found in the OpcUaServer.

Related Links

- [Components and views \(Parent Topic\)](#)

opcUaServer-OpcUaAuthenticationScheme

The OpcUaAuthenticationScheme authenticates the user on an OPC UA Server, and ensures the password requirements are adhered to. Any new OPC UA user should be associated with this scheme.

The OpcUaAuthenticationScheme component must be added to the station's Authentication Service. Then, when creating a new OPC UA specific user, simply configure the user's authentication scheme as: "OpcUaAuthenticationScheme".

NOTE: When making any changes on OPC UA server side, you must disable and then re-enable the OPC UA Server.

Property	Value	Description
Global Password Configuration	container	Container component for the subproperties
Password Strength	container	Container component for the subproperties configured for a strong password which requires a minimum of 10 characters; and at least 1 of each of the following characters: lowercase, uppercase, digit.
Expiration Interval	365d 00h 00m 00s (default)	The password is valid for this period of time.
Warning Period	030d 00h 00m 00s	The period of time prior to password expiration that a warning is issued.
Password History Length	1–10	Configures the number of previously used passwords to be retained. This setting restricts the user from reusing a set number of previously used passwords. The range is 1–10, with the default being 2.

Related Links

- [Components and views \(Parent Topic\)](#)

Related Tasks

- [Configuring the server to support an Opc Ua Client user](#)

opcUaClient-OpcUaNetwork

The OpcUaNetwork is a top-level container component for an OpcUaNetwork in a station. It represents a network of manageable Opc Ua Client devices. Within a network you can create multiple Opc Ua Client devices. This network provides a common “Monitor” that is used to monitor the status of connected OPC UA Servers. This component is available in the opcUaClient palette.

Actions

- Ping - A Ping action attempts communication with device and request to verify device health. If successful, the status is set to ok. If it fails, the status is set to down.
- Learn – Discovers server objects.
- Reset Comm –

Related Links

- [Components and views \(Parent Topic\)](#)

Related Tasks

- [Adding the OPC UA Network](#)

opcUaClient-OpcUaDevice

OpcUaDevice is the device-level component in an OpcUaNetwork, and represents a client connection to a specific OpcUaServer. It contains configuration parameters necessary for the driver to communicate with that server. The OpcUaDevice has a Points device extension (OpcUaClientPointDeviceExt) that contains all subscribed proxy points.

The OpcUaDevice has the standard device component properties such as status and enabled (see “Common device components” in the Drivers Guide for general information). The default view for an OpcUaDevice is the Opc Ua Client Manager.

Properties used to configure the OpcUaDevice are listed here:

Name	Value	Description
Name	text string	Name of the device - edit this as needed.
Type		Type of device
Server Endpoint Url	opc.tcp://URL for example (opc.tcp:// IE67DTDVYXX.honeywell.com: 52520/OPCUA/N4OpcUaServer)	Connection address URL for the Opc Ua Server
Security Mode	Sign Ecript Basic256 Sha256 (default) Sign Ecript Basic256 Sign Basic256 Sha256 Sign Basic256 Sign Ecript Basic128Rsa15 Sign Basic128 Rsa15 None	Specifies what security should be applied to message exchanges during a session. <ul style="list-style-type: none"> • Sign Ecript Basic256 Sha256 — All messages are signed and encrypted. The signature or signing the message allows detection if it has been manipulated by any third party. For more details see, Security policies in the section “About the OPC UA security architecture”.
Certificate	tridium other	
Security Mode	Anonymous User Name and Password (default) Certificate (default) Issued Token (default)	User authentication mode is Anonymous by default. For successful connection, these settings must match OPC UA Server’s User Authentication settings.
Enabled	true (default), false	
User Name	text string	Required when the OPC UA Server’s user authentication mode is set to User Name and Password. The username and password must be defined in the station’s UserService.
Password	text string	Required when the OPC UA Server’s user authentication mode is set to User Name and Password. The username and password

Name	Value	Description
		must be defined in the station's UserService.

Actions

- Ping - Sends a ping monitor request to verify device health.
- Learn – Discovers server objects.
- Reset Comm – Enables/disables the device (closes and reopens server connection)

Related Links

- [Components and views \(Parent Topic\)](#)

Related Tasks

- [Connecting to an OPC UA Server](#)
- [Discovering OPC UA Server Points](#)
- [Adding Points to the Station database](#)

opcUaClient-OpcUaBuildInfo

This component is a child of each OpcUaDevice component. Each OpcUaDevice has a frozen slot named Server Info (OpcUaBuildInfo) on the OpcUaDevice component. This component displays read-only data on the OpcUaServer configured for the device, as shown.

Server Info (Opc Ua Build Info)	
Product Name	N4_OpcUaServer
Product Uri	urn:tridium.com:OPCUA:N4OpcUaServer
Manufacturer	Tridium
Software Version	4.3.58.16
Build Number	16
Build Date	05/10/17 18:46:26.2290000 GMT

Related Links

- [Components and views \(Parent Topic\)](#)

opcUaClient-OpcUaClientAlarmDeviceExt

This component is a child of each OpcUaDevice component. Each OpcUaDevice has a frozen slot named AlarmExt that is of type OpcUaClientAlarmDeviceExt. The default view is the Opc Ua Client Alarm Manager.

The OpcUaClientAlarmDeviceExt component provides the ability to subscribe to OPC UA alarm events. The view allows you to scan the connected OpcUaServer for OPC UA variables that are alarmable”

Related Links

- [Components and views \(Parent Topic\)](#)

Related Tasks

- [Subscribing for OPC UA Alarm Events](#)

opcUaClient-OpcUaClientPointDeviceExt

OpcUaClientPointDeviceExt (Points) is the OPC UA implementation of PointDeviceExt, a frozen device extension under every OpcUaClient device. Its primary view is the Opc Ua Client Point Manager.

The OpcUaClientPointDeviceExt (Points) is also available in the opcUaClient palette.

Related Links

- [Components and views \(Parent Topic\)](#)

opcUaClient-OpcUaDeviceFolder

Opc Ua Device Folder is the Opc Ua Client implementation of a folder under an OpcUaNetwork. Typically, you add such folders using the New Folder button in the Opc Ua Client Device Manager view of the OpcUaNetwork. The component is also available in the opcUaClient palette.

Related Links

- [Components and views \(Parent Topic\)](#)

opcUaClient-OpcUaClientHistoryDeviceExt

The OpcUaClientHistoryDeviceExt (default name Histories) is a frozen device extension of the OpcUaClient component. It allows the import of historical data from the OpcUaServer into the history space. Use its default Opc Ua Client History Import Manager view to add OpcUaClientHistoryImport descriptors.

For general information, see “About the Histories extension” and “History Import Manager” in the Niagara Drivers Guide.

Related Links

- [Components and views \(Parent Topic\)](#)

Related Tasks

- [Adding points containing OPC UA Histories](#)

opcUaClient-OpcUaClientPointFolder

Opc Ua Client Point Folder is the Opc Ua Client implementation of a folder under an OpcUaDevice Points extension. You add such folders using the New Folder button in the Opc Ua Client Point Manager view of the Opc Ua Client Point Device extension. The Opc Ua Client Point Folder is also available in the opcUaClient palette.

Related Links

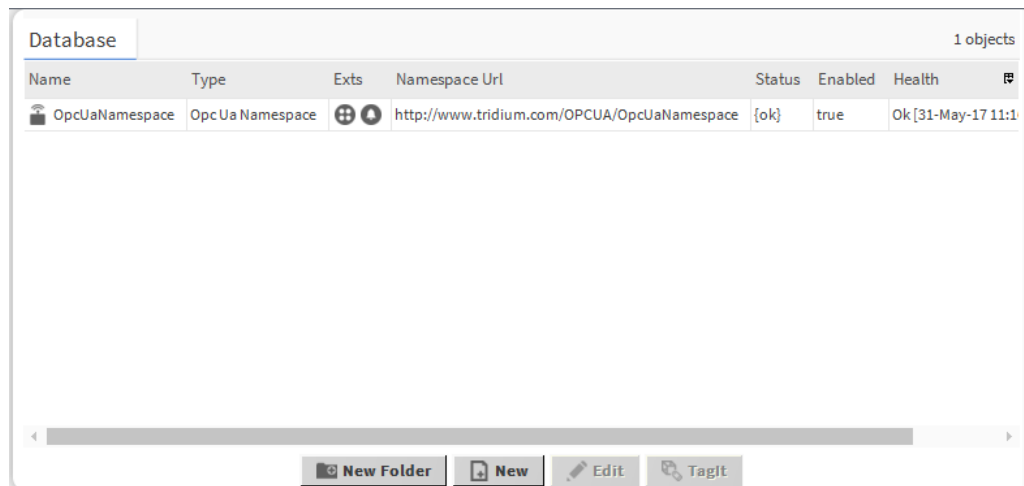
- [Components and views \(Parent Topic\)](#)

opcUaServer-OpcUaServerDeviceManager

The Opc Ua Server Device Manager view is the primary view for the OpcUaServer component. The view allows you to manage OpcUaNamespace components in the station.

To view, either double-click the OpcUaServer or right-click the OpcUaServer and select **Views > Opc Ua Server Device Manager**.

Figure 2. Opc Ua Server Device Manager with added OpcUaNamespace



Related Links

- [Components and views \(Parent Topic\)](#)

Opc Ua Server Alarm Manager

The Opc Ua Server Alarm Manager is the default view of the Alarms device extension (OpcUaServerAlarmDeviceExt) under these devices. To view, double-click the Alarms extension, or right-click and select **Views > Opc Ua Server Alarm Manager**.

Related Links

- [Components and views \(Parent Topic\)](#)

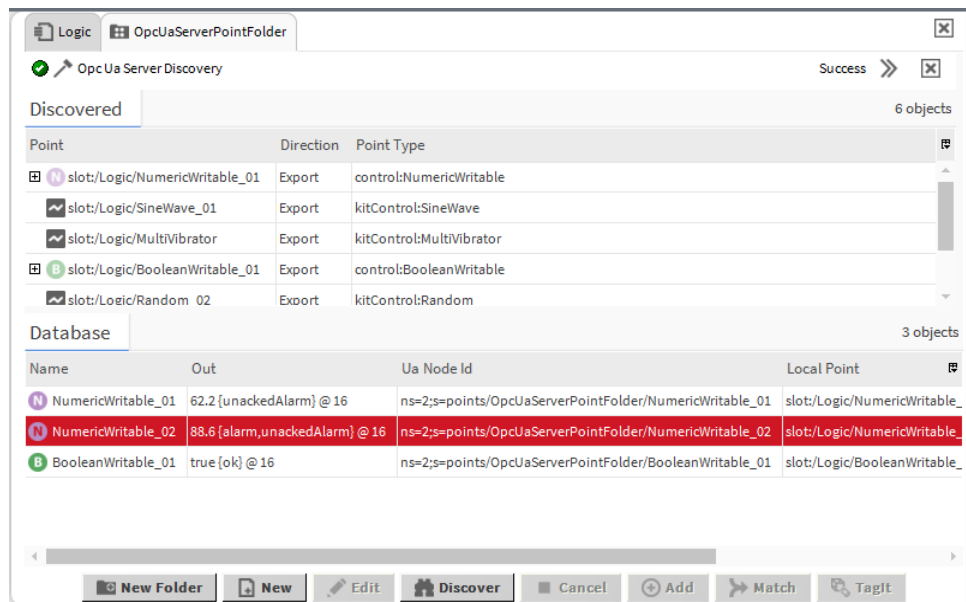
Opc Ua Server Point Manager

Opc Ua Server Point Manager is the default view for the OpcUaServerPointDeviceExt (Points container) under an Opc Ua Namespace. This is also the default view for any OpcUaServerPointFolder under the Points container of an Opc Ua Namespace.

Use this view to discover server points that exist in the station which can be mapped to the OpcUaServerPointFolder. Such server points are visible to a connected OPC UA client. Also, server points provide live and historical data as well as alarm events.

To view, right-click a OpcUaServerPointDeviceExt or OpcUaServerPointFolder and select **Views > Opc Ua Server Point Manager**

Figure 3. Opc Ua Server Point Manager view with discovered points added to station database



Related Links

- [Components and views \(Parent Topic\)](#)

Opc Ua Client Device Manager

The Opc Ua Device Manager is the default view of a OpcUaNetwork which helps to access OPC UA device components. To view, right-click a OpcUaNetwork and select **Views > Opc Ua Client Device Manager**.

Related Links

- [Components and views \(Parent Topic\)](#)

Opc Ua Client Alarm Manager

The Opc Ua Client Alarm Manager is the default view of the OpcUaClient AlarmDeviceExt. This view allows you to scan the OPC UA server to discover and subscribe to OPC UA alarm events.

Related Links

- [Components and views \(Parent Topic\)](#)

Opc Ua Client Point Manager

Opc Ua Client Point Manager is the default view for the OpcUaClientPointDeviceExt (Points container) under an Opc Ua Device. This is also the default view for any OpcUaClientPointFolder under the Points container of an OPC UA Device. Use this view to discover available points in order to add them to your station database.

To view, right-click a OpcUaClientPointDeviceExt or OpcUaClientPointFolder and select **Views > Opc Ua Client Point Manager**.

Figure 4. Opc Ua Client Point Manager view with selected discovered point added to station database

The screenshot shows the 'Opc Ua Client Discovery' window with a 'Success' status. It is divided into two main sections: 'Discovered' and 'Database'.

Discovered (12 objects):

Ua Display Name	Ua Node Id	Access Level	Array Dimension	Array Index	Ua Data Type	Ua Data Type Identifier
Objects	i=85	{}				-1
Server	i=2253	{}				-1
MyObjects	ns=2;s=MyObjectsFolder	{}				-1
MyDevice	ns=2;s=MyDevice	{}				-1
MyEnumObject	ns=2;s=MyEnumObject	{rw}			Integer	-1
MyLevel	ns=2;s=MyLevel	{rwhr}			Double	11
MyMethod	ns=2;s=MyMethod	{}				-1
MySwitch	ns=2;s=MySwitch	{rwhr}			Boolean	1
StaticData	ns=3;s=StaticData	{}				-1

Database (1 objects):

Name	Type	Out	Ua Node Id	Array Index	Ua Status Code	Tuning Policy Name
MyLevel	Numeric Point	88.00 [ok]	ns=2;s=MyLevel		GOOD (0x00000000)	defaultPolicy

Related Links

- [Components and views \(Parent Topic\)](#)

Opc Ua Client History Import Manager

Opc Ua Client History Import Manager is the default view for the OpcUaClientHistoryDeviceExt component which provides the ability to discover and import OPC UA histories without the creation of a control point.

To view, right-click a OpcUaClientHistoryDeviceExt and select **Views > Opc Ua Client History Import Manager**.

Figure 5. Opc Ua Client History Import Manager lists discovered points with histories

Ndriver Discovery Success >> [X]

Discovered 10 objects

Ua Display Name	Ua Node Id	Array Dimension	Array Index	Node Class	Description	Type Spec	Ua Data Type	Ua Data Type Identifier	Ua Va
MyLevel	ns=2;s=MyLevel			Variable		baja:Double	Double	11	23.0
MySwitch	ns=2;s=MySwitch			Variable		baja:Boolean	Boolean	1	false
Counter1	ns=5;s=Counter1			Variable		baja:Double	Int32	6	26
Expression1	ns=5;s=Expression1			Variable		baja:Double	Double	11	3.8747

Database 0 objects

Name	History Id	Status	State	Last Success	Ua Node Name	Ua Node Id	Type Spec	Local History Name
------	------------	--------	-------	--------------	--------------	------------	-----------	--------------------

Related Links

- [Components and views \(Parent Topic\)](#)