

OPC XML-DA Client Driver

© 2026 PTC Inc. All Rights Reserved.

Table of Contents

OPC XML-DA Client Driver	1
Table of Contents	2
Welcome to the OPC XML-DA Client Driver Help Center	4
Overview	4
Project Architecture	5
Channel Properties – General	5
Tag Counts	6
Channel Properties – Write Optimizations	6
Channel Properties – Advanced	7
Channel Properties – OPC Server	7
Channel Properties – HTTP Proxy	8
Channel Properties – HTTP Authentication	8
Device Properties – General	9
Operating Mode	9
Tag Counts	10
Device Properties – Scan Mode	10
Device Properties – OPC Group	11
Device Properties – Exception Mode Options	12
Device Properties – Communications Parameters	13
Device Properties – Import	13
Data Types Description	15
Address Descriptions	16
Error Descriptions	17
Access is denied for item <address>	17
An invalid subscription handle was passed to the server on <channel/device name>	17
Invalid continuation point	18
Invalid filter	18
Invalid hold time for item <address>	18
Invalid item name for item <address>	18
Invalid path for item <address> with path <path>	18
Item <address> is Write Only	19
General SOAP fault on <channel/device name>	19
Server is busy on <channel/device>	19
Server is in an abnormal state on <channel/device name>	19
SOAP data queue overflow	19
SOAP HTTP communication error on <channel/device name>	20
SOAP SSL authentication error on <channel/device name>	20
SOAP TCP communication error on <channel/device name>	20
SOAP XML parse error on <channel/device name>	20
The data type cannot be accepted for item <address>	20

The item name <Channel>.<Device>.<Item Path>.<Item Name> is no longer available in the server address space	21
The item path <Channel>.<Device>.<Item Path> is no longer available in the server address space	21
The Property ID is invalid for the item <address>	21
The server reported an unspecified failure on <channel/device>	21
The server reported that the operation timed out on <channel/device name>	21
The server returned a 'Not Supported' error on item <address>	22
The server returned an "Out of memory" error on <channel/device name>	22
The specified write value for <address> is out of range	22
Unknown data type for tag <item> on device <device>. Using Default	22
Time-Sensitive Data Acquisition	23
Index	24

Welcome to the OPC XML-DA Client Driver Help Center

This help center is the user documentation for Kepware OPC XML-DA Client Driver. This help center is updated regularly to reflect the latest functionality and information.

Overview

What is the OPC XML-DA Client Driver?

Setup

How do I configure devices for use with this driver?

Data Types Descriptions

What data types does this driver support?

Address Descriptions

How do I address a data location on an OPC XML-DA Client device?

Error Descriptions

What error messages does the OPC XML-DA Client Driver produce?

Time-Sensitive Data Acquisition

What is the best way to configure time-sensitive subscriptions?

Version 1.040

© 2026 PTC Inc. All Rights Reserved.

Overview

The OPC XML-DA Client Driver provides a reliable way to connect OPC XML-DA enabled devices through the Internet or factory intranet. It provides the following:

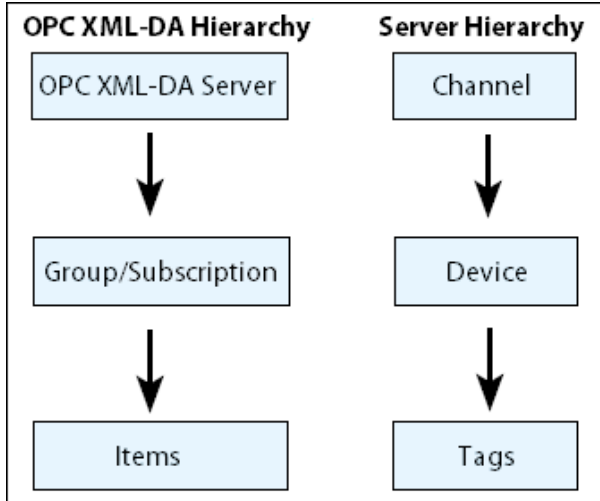
- A reliable connection point for accessing data from multiple remote or local OPC XML-DA enabled devices.
- Secure communications between devices on different networks with the use of HTTPS and SSL certificates.

Channel and Device Limits

The maximum number of channels supported by this driver is 128. The maximum number of devices supported by this driver is 256 per channel.

Project Architecture

An OPC XML-DA Client Driver channel represents a connection to an OPC XML-DA Server; an OPC XML-DA device represents an OPC XML-DA Group or Subscription while using Polled or Exception-based communications (respectively). The image below displays an example of the similarities between the OPC XML-DA and server functional hierarchies.



Note: The XML-DA client supports simultaneous connections to multiple servers.

Channel Properties – General

This server supports the use of multiple simultaneous communications drivers. Each protocol or driver used in a server project is called a channel. A server project may consist of many channels with the same communications driver or with unique communications drivers. A channel acts as the basic building block of an OPC link. This group is used to specify general channel properties, such as the identification attributes and operating mode.

<p>Property Groups</p> <p>General</p> <p>Write Optimizations</p> <p>Advanced</p>	<p><input type="checkbox"/> Identification</p> <p>Name</p> <p>Description</p> <p>Driver</p> <p><input type="checkbox"/> Diagnostics</p> <p>Diagnostics Capture Disable</p> <p><input type="checkbox"/> Tag Counts</p> <p>Static Tags 10</p>
---	--

Identification

Name: Specify the user-defined identity of this channel. In each server project, each channel name must be unique. Although names can be up to 256 characters, some client applications have a limited display window when browsing the OPC server's tag space. The channel name is part of the OPC browser information. The property is required for creating a channel.

Note: For information on reserved characters, refer to "How To... Properly Name a Channel, Device, Tag, and Tag Group" in the server help.

Description: Specify user-defined information about this channel.

Many of these properties, including Description, have an associated system tag.

Driver: Specify the protocol / driver for this channel. Specify the device driver that was selected during channel creation. It is a disabled setting in the channel properties. The property is required for creating a channel.

Note: With the server's online full-time operation, these properties can be changed at any time. This includes changing the channel name to prevent clients from registering data with the server. If a client has already acquired an item from the server before the channel name is changed, the items are unaffected. If, after the channel name

has been changed, the client application releases the item and attempts to re-acquire using the old channel name, the item is not accepted. Changes to the properties should not be made once a large client application has been developed. Utilize proper user role and privilege management to prevent operators from changing properties or accessing server features.

Diagnostics

Diagnostics Capture: When enabled, this option makes the channel's diagnostic information available to OPC applications. Because the server's diagnostic features require a minimal amount of overhead processing, it is recommended that they be utilized when needed and disabled when not. The default is disabled.

● **Note:** This property is not available if the driver or operating system does not support diagnostics.

● *For more information, refer to Communication Diagnostics and Statistics Tags in server help.*

Tag Counts

Static Tags: Provides the total number of defined static tags at this level (device or channel). This information can be helpful in troubleshooting and load balancing.

Channel Properties – Write Optimizations

The server must ensure that the data written from the client application gets to the device on time. Given this goal, the server provides optimization properties to meet specific needs or improve application responsiveness.

Property Groups	Write Optimizations	
General	Optimization Method	Write Only Latest Value for All Tags
Write Optimizations	Duty Cycle	10

Write Optimizations

Optimization Method: Controls how write data is passed to the underlying communications driver. The options are:

- **Write All Values for All Tags:** This option forces the server to attempt to write every value to the controller. In this mode, the server continues to gather write requests and add them to the server's internal write queue. The server processes the write queue and attempts to empty it by writing data to the device as quickly as possible. This mode ensures that everything written from the client applications is sent to the target device. This mode should be selected if the write operation order or the write item's content must uniquely be seen at the target device.
- **Write Only Latest Value for Non-Boolean Tags:** Many consecutive writes to the same value can accumulate in the write queue due to the time required to actually send the data to the device. If the server updates a write value that has already been placed in the write queue, far fewer writes are needed to reach the same final output value. In this way, no extra writes accumulate in the server's queue. When the user stops moving the slide switch, the value in the device is at the correct value at virtually the same time. As the mode states, any value that is not a Boolean value is updated in the server's internal write queue and sent to the device at the next possible opportunity. This can greatly improve the application performance.
 - **Note:** This option does not attempt to optimize writes to Boolean values. It allows users to optimize the operation of HMI data without causing problems with Boolean operations, such as a momentary push button.
- **Write Only Latest Value for All Tags:** This option takes the theory behind the second optimization mode and applies it to all tags. It is especially useful if the application only needs to send the latest value to the device. This mode optimizes all writes by updating the tags currently in the write queue before they are sent. This is the default mode.

Duty Cycle: is used to control the ratio of write to read operations. The ratio is always based on one read for every one to ten writes. The duty cycle is set to ten by default, meaning that ten writes occur for each read operation. Although the application is performing a large number of continuous writes, it must be ensured that read data is still given time to process. A setting of one results in one read operation for every write operation. If there are no write operations to perform, reads are processed continuously. This allows optimization for applications with continuous writes versus a more balanced back and forth data flow.

● **Note:** It is recommended that the application be characterized for compatibility with the write optimization enhancements before being used in a production environment.

Channel Properties – Advanced

This group is used to specify advanced channel properties. Not all drivers support all properties; so the Advanced group does not appear for those devices.

Property Groups	<input type="checkbox"/> Non-Normalized Float Handling	
General	Floating-Point Values	Replace with Zero
Write Optimizations	<input type="checkbox"/> Inter-Device Delay	
Advanced	Inter-Device Delay (ms)	0

Non-Normalized Float Handling: A non-normalized value is defined as Infinity, Not-a-Number (NaN), or as a Denormalized Number. The default is Replace with Zero. Drivers that have native float handling may default to Unmodified. Non-normalized float handling allows users to specify how a driver handles non-normalized IEEE-754 floating point data. Descriptions of the options are as follows:

- **Replace with Zero:** This option allows a driver to replace non-normalized IEEE-754 floating point values with zero before being transferred to clients.
- **Unmodified:** This option allows a driver to transfer IEEE-754 denormalized, normalized, non-number, and infinity values to clients without any conversion or changes.

Note: This property is disabled if the driver does not support floating-point values or if it only supports the option that is displayed. According to the channel's float normalization setting, only real-time driver tags (such as values and arrays) are subject to float normalization. For example, EFM data is not affected by this setting.

For more information on the floating-point values, refer to "How To ... Work with Non-Normalized Floating-Point Values" in the server help.

Inter-Device Delay: Specify the amount of time the communications channel waits to send new requests to the next device after data is received from the current device on the same channel. Zero (0) disables the delay.

Note: This property is not available for all drivers, models, and dependent settings.

Channel Properties – OPC Server

Property Groups	<input type="checkbox"/> OPC XML-DA Server Settings	
General	Server Endpoint URL	http://localhost:8080/xmlbase...
Write Optimizations	Keep Alive (s)	0
Advanced	<input type="checkbox"/> HTTPS Security	
OPC Server	Trusted Certificates Path	
HTTP Proxy	<input type="checkbox"/> Item Path	
HTTP Authentication	Item Path Delimiter	\

OPC XML-DA Server Settings

- **Server Endpoint URL:** Specify the address to be used to connect to the remote XML-DA server. It is a required field.
- **Keep Alive:** Specify the rate at which a GetStatus call is sent to the server to check on the server's operational status. The valid range is 0 to 100 hours. The default setting is 0 seconds. When 0 is specified, no Keep Alive are sent.
 - **Note:** An error message is posted to the Event Log if a GetStatus call fails.

HTTP Security

This section is visible if the Server Endpoint URL starts with "https". It is recommended that HTTP Security be used when there are concerns about security.

- **Trusted Certificates Path:** Specify a path to a file that contains certificates that the client driver should trust. This property is required when using SSL.

Item Path

Item Path Delimiter: Specify the item path delimiter, which is used in the tag address with the format `<ItemPath><Delimiter><ItemName>`. Options include backslash, forward slash, exclamation point, vertical bar, period, underscore, and dash. The default setting is backslash (\).

● **Note:** This is optional. If the server does not use item paths, the item path and delimiter are not needed.

Channel Properties – HTTP Proxy

Property Groups	<input checked="" type="checkbox"/> HTTP Proxy	
General	Server Address	
OPC Server	Port	8080
HTTP Proxy	Username	
HTTP Authentication	Password	*****

Server Address: Specify the address that will be used as a proxy. Either an IP or a Domain name may be used.

Port: Specify the port that will be connected to on the remote server. The valid range is 0 to 65535. The default setting is 8080.

Username: Specify the user name (if one is required to connect to the server).

Password: Specify the password (if one is required to connect to the server).

● **Note:** The Username and Password properties are only required if the proxy requires authentication. When using a proxy without HTTPS, the username and password will be secured according to the type of proxy being used. This means that a clever and malevolent user could gain access to the Username and Password.

HTTPS/SSL

The client driver supports Simple SSL, which is the version used by most HTTPS-secured websites. In this configuration, the client driver must have the server certificate or root CA's certificate in the trusted list.

Channel Properties – HTTP Authentication

Property Groups	<input checked="" type="checkbox"/> HTTP Authentication	
OPC Server	Username	
HTTP Proxy	Password	*****
HTTP Authentication		

User Name: Specify the user name if required by the XML-DA server. The default setting is blank.

Password: Enter the password if required by the XML-DA server. The default setting is blank.

● **Caution:** HTTP Authentication is not secure. User names and passwords will be transmitted in plain text.

Device Properties – General

A device represents a single target on a communications channel. If the driver supports multiple controllers, users must enter a device ID for each controller.

<table border="1"> <tr> <th>Property Groups</th> </tr> <tr> <td>General</td> </tr> </table>	Property Groups	General	<table border="1"> <tr> <td colspan="2">[-] Identification</td> </tr> <tr> <td>Name</td> <td></td> </tr> <tr> <td>Description</td> <td></td> </tr> <tr> <td>Channel Assignment</td> <td></td> </tr> <tr> <td>Driver</td> <td></td> </tr> <tr> <td>Model</td> <td></td> </tr> <tr> <td>ID Format</td> <td>Decimal</td> </tr> <tr> <td>ID</td> <td>2</td> </tr> </table>	[-] Identification		Name		Description		Channel Assignment		Driver		Model		ID Format	Decimal	ID	2
Property Groups																			
General																			
[-] Identification																			
Name																			
Description																			
Channel Assignment																			
Driver																			
Model																			
ID Format	Decimal																		
ID	2																		

Identification

Name: Specify the name of the device. It is a logical user-defined name that can be up to 256 characters long and may be used on multiple channels.

- **Note:** Although descriptive names are generally a good idea, some OPC client applications may have a limited display window when browsing the OPC server's tag space. The device name and channel name become part of the browse tree information as well. Within an OPC client, the combination of channel name and device name would appear as "ChannelName.DeviceName".
- For more information, refer to "How To... Properly Name a Channel, Device, Tag, and Tag Group" in server help.

Description: Specify the user-defined information about this device.
 ● Many of these properties, including Description, have an associated system tag.

Channel Assignment: Specify the user-defined name of the channel to which this device currently belongs.

Driver: Selected protocol driver for this device.

Model: Specify the type of device that is associated with this ID. The contents of the drop-down menu depend on the type of communications driver being used. Models that are not supported by a driver are disabled. If the communications driver supports multiple device models, the model selection can only be changed when there are no client applications connected to the device.

- **Note:** If the communication driver supports multiple models, users should try to match the model selection to the physical device. If the device is not represented in the drop-down menu, select a model that conforms closest to the target device. Some drivers support a model selection called "Open," which allows users to communicate without knowing the specific details of the target device. For more information, refer to the driver documentation.

ID: Specify the device's driver-specific station or node. The type of ID entered depends on the communications driver being used. For many communication drivers, the ID is a numeric value. Drivers that support a Numeric ID provide users with the option to enter a numeric value whose format can be changed to suit the needs of the application or the characteristics of the selected communications driver. The format is set by the driver by default. Options include Decimal, Octal, and Hexadecimal.

- **Note:** If the driver is Ethernet-based or supports an unconventional station or node name, the device's TCP/IP address may be used as the device ID. TCP/IP addresses consist of four values that are separated by periods, with each value in the range of 0 to 255. Some device IDs are string based. There may be additional properties to configure within the ID field, depending on the driver.

Operating Mode

<table border="1"> <tr> <th>Property Groups</th> </tr> <tr> <td>General</td> </tr> </table>	Property Groups	General	<table border="1"> <tr> <td colspan="2">[+] Identification</td> </tr> <tr> <td colspan="2">[-] Operating Mode</td> </tr> <tr> <td>Data Collection</td> <td>Enable</td> </tr> <tr> <td>Simulated</td> <td>No</td> </tr> </table>	[+] Identification		[-] Operating Mode		Data Collection	Enable	Simulated	No
Property Groups											
General											
[+] Identification											
[-] Operating Mode											
Data Collection	Enable										
Simulated	No										

Data Collection: This property controls the device's active state. Although device communications are enabled by default, this property can be used to disable a physical device. Communications are not attempted when a device is disabled. From a client standpoint, the data is marked as invalid and write operations are not accepted. This property can be changed at any time through this property or the device system tags.

Simulated: Place the device into or out of Simulation Mode. In this mode, the driver does not attempt to communicate with the physical device, but the server continues to return valid OPC data. Simulated stops physical communications with the device, but allows OPC data to be returned to the OPC client as valid data. While in Simulation Mode, the server treats all device data as reflective: whatever is written to the simulated device is read back and each OPC item is treated individually. The data is not saved if the server removes the item (such as when the server is reinitialized). The default is No.

● **Notes:**

1. Updates are not applied until clients disconnect and reconnect.
2. The System tag (_Simulated) is read only and cannot be written to for runtime protection. The System tag allows this property to be monitored from the client.
3. In Simulation mode, the item's memory map is based on client update rate(s) (Group Update Rate for OPC clients or Scan Rate for native and DDE interfaces). This means that two clients that reference the same item with different update rates return different data.
4. When a device is simulated, updates may not appear faster than one (1) second in the client.

● Simulation Mode is for test and simulation purposes only. It should never be used in a production environment.

Tag Counts

Property Groups	<input type="checkbox"/> Identification <input type="checkbox"/> Operating Mode <input type="checkbox"/> Tag Counts	
General	Static Tags	130

Static Tags: Provides the total number of defined static tags at this level (device or channel). This information can be helpful in troubleshooting and load balancing.

Device Properties – Scan Mode

The Scan Mode specifies the subscribed-client requested scan rate for tags that require device communications. This setting works differently in this driver than in other drivers. Instead of determining the rate of device communications, it determines the rate that newly cached values are available to subscribed clients. It does not affect the rate of reads sent to the target server. Read frequency is controlled entirely by static settings configured in [Device Properties – OPC Group](#). Generally, Scan Mode should be left in its default state. Synchronous and asynchronous device reads and writes are processed as soon as possible; unaffected by the Scan Mode properties.

● **Note:** In this driver, tag reads occur at the rate defined in [Device Properties – OPC Group](#) regardless of the Scan Mode setting.

Property Groups	<input type="checkbox"/> Scan Mode	
General	Scan Mode	Respect Client-Specified Scan Rate ▼
Scan Mode	Initial Updates from Cache	Disable

Scan Mode: Specify how tags in the device are scanned for updates sent to subscribing clients. Descriptions of the options are:

- **Respect Client-Specified Scan Rate:** This mode uses the scan rate requested by the client.
- **Request Data No Faster than Scan Rate:** This mode specifies the value set as the maximum scan rate. The valid range is 10 to 99999990 milliseconds. The default is 1000 milliseconds.

- **Note:** When the server has an active client and items for the device and the scan rate value is increased, the changes take effect immediately. When the scan rate value is decreased, the changes do not take effect until all client applications have been disconnected.
- **Request All Data at Scan Rate:** This mode forces tags to be scanned at the specified rate for subscribed clients. The valid range is 10 to 99999990 milliseconds. The default is 1000 milliseconds.
- **Do Not Scan, Demand Poll Only:** This mode does not periodically poll tags that belong to the device nor perform a read to get an item's initial value once it becomes active. It is the OPC client's responsibility to poll for updates, either by writing to the `_DemandPoll` tag or by issuing explicit device reads for individual items. *For more information, refer to "Device Demand Poll" in server help.*
- **Respect Tag-Specified Scan Rate:** This mode forces static tags to be scanned at the rate specified in their static configuration tag properties. Dynamic tags are scanned at the client-specified scan rate.

Initial Updates from Cache: When enabled, this option allows the server to provide the first updates for newly activated tag references from stored (cached) data. Cache updates can only be provided when the new item reference shares the same address, scan rate, data type, client access, and scaling properties. A device read is used for the initial update for the first client reference only. The default is disabled; any time a client activates a tag reference the server attempts to read the initial value from the device.

Device Properties – OPC Group

Property Groups	<input checked="" type="checkbox"/> OPC Group	
General	Update Mode	Exception
Scan Mode	Update/Poll Rate (ms)	5000
OPC Group	Language ID	1033

Update Mode: Specify the OPC Group's update mode. Options include Exception and Poll. The default setting is Exception. Descriptions of the options are as follows:

- **Exception:** In this mode, a subscription is created for a set of tags and the server is instructed to update the client when the data changes. The client driver makes `SubscriptionPolledRefresh` requests at the Update/Poll rate. If the server does not receive a response from the client referencing a particular subscription, it terminates the subscription after the maximum allowed time (65.5 seconds) has elapsed.
 - **Note:** The Exception Mode for the OPC XML-DA Client Driver differs from other drivers. All function requests made through the OPC XML-DA Client Driver are synchronous and blocking, meaning that the response from the server immediately follow the client request. If a request is made, the client waits until a response from the server is received. The OPC XML-DA Client Driver is designed to divide computational resources among devices/groups within each channel. The channel operates independently from other channels, whereas the devices within each channel will take turns making requests. This can cause problems when using Exception-based communications where multiple devices share a channel. For example, if two devices share a channel and one has a long hold time and wait time, the second driver has to wait for the first driver to finish its request/response transaction before the second driver can begin its request. In this case, it may be helpful to delegate devices with long hold times to their own channel.
- **Poll:** In this mode, no subscription is created. Tags will be read from the server at the interval specified in the Update Rate property .

Update/Poll Rate: When in Exception Mode, this property specifies how often the underlying OPC server will provide updates for changing data. When in Poll Mode, this property specifies how often the driver reads the items attached to the group. The valid range is 0 milliseconds to 1 hour. The default setting is 5000 milliseconds.

Language ID: Specify the language to be used by the underlying server when returning values as text for operations. The default setting is 1033 (English).

Device Properties – Exception Mode Options

Property Groups	<input checked="" type="checkbox"/> Exception Mode Options	
General	Hold Time (ms)	0
OPC Group	Wait Time (ms)	0
Exception Mode Options	Percent Deadband	0

Hold Time: Specify the amount of time that the server will wait before either continuing to wait or returning updates. The valid range is 0 to 360000000 milliseconds (0 to 100 hours). The default setting is 0 milliseconds.

● **Note:** Because the client specifies the amount of time that the server is instructed to wait (the client's present time plus the specified duration in milliseconds), synchronization between the client and server is critical to eliminate the differences in time propagating themselves through to the hold time.

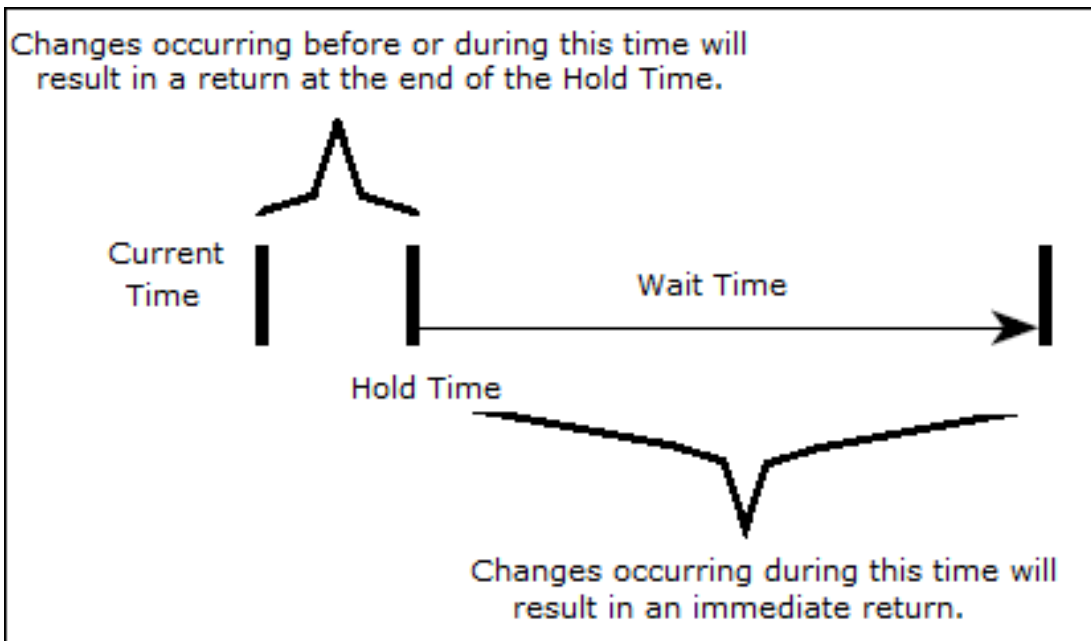
Wait Time: Specify the amount of time that the server will wait before returning updates from a SubscriptionPolledRefresh request if no data change has taken place. Otherwise, the server will return a value any time during the Wait Time period on a data change. The valid range is 0 to 360000000 milliseconds. The default setting is 0 milliseconds.

Percent Deadband: Specify the minimum percent change needed in a tag's value to return the value to the client. This will be based on range values that are determined by the server. The valid range is 0 to 100%. The default setting is 0.

● **Note:** If the Update Rate is less than the Hold Time, the blocking nature of the API calls will cause the next refresh request to be sent when the previous request completes. For more information, refer to "Differences in Exception Mode" below.

Differences in Exception Modes

Because HTTP is an inherently synchronous protocol, the classic model from OPC DA of opening ports and listening for messages from a server is impossible. To simulate Exception Mode, the OPC XML-DA Client Driver Specification explains that when in Exception Mode, the client sends a Subscribe request to the server and then follows it with a Refresh request at the Update Rate. The client has the option of specifying a Hold Time and a Wait Time at each refresh request.



Descriptions of the terms are as follows:

- **Hold Time:** This is the minimum data resolution that is expected in a subscription-based setup. When an XML-DA server receives a refresh request with a Hold Time, the server delays the refresh response for the

- given Hold Time.
- **Wait Time:** This is the maximum data resolution that is expected in a subscription-based setup. When an XML-DA server receives a refresh request with a Wait Time, the server sends the refresh response when the data has changed since the last refresh or until the end of the wait time.

Device Properties – Communications Parameters

Property Groups	[-] Communications Parameters	
General	Maximum Request Size (bytes)	256
Scan Mode		
Communications Parameters		

Request Size

Max. Items per Read: Specify the maximum number of items that can be included in a single Read request. The valid range is 1 to 512. The default is 512.

Max. Items per Write: Specify the maximum number of items that can be included in a single Write request. The valid range is 1 to 512. The default is 512.

Request Timeout

Read Timeout: Specify the amount of time that the driver will wait for a Read Complete notification to be returned from the server before any other Read or Write requests will be sent. If the expected notification is not received, the items that were included in the request will be set to Bad quality, and will remain Bad until the next successful Read. The valid range is 100 to 3600000 milliseconds. The default is 5000 milliseconds.

Write Timeout: Specify the amount of time that the driver will wait for a Write Complete notification to be returned from the server before sending any other Write or Read request. If the expected notification is not received, the driver will log a Write failed message on timeout. The valid range is 100 to 3600000 milliseconds. The default is 5000 milliseconds.

Read after Write: Choose Enable to force an explicit read after a write command (to confirm the new value). Choose Disable to update after the next publish or poll response. The default is Enable.

Device Properties – Import

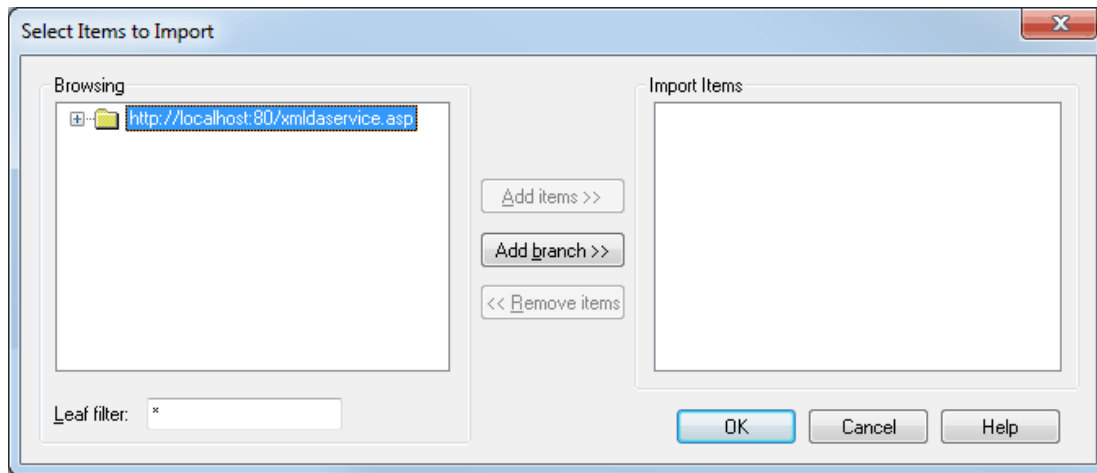
Property Groups	[-] Import	
Import	Import Items	Select Import Items...

Select import items: When clicked, this button invokes the Select Items to Import property group for browsing the server for tags. For more information, refer to "Selecting Items to Import" below.

Selecting Items to Import

The driver displays all items in the tree. To add items for import, select them from the tree and then click **Add Items >>**. The Import Items list will then display the items for import. To remove items, select them and then click **<< Remove Items**. Once finished, click **OK**.

● **Note:** The **Leaf Filter** specifies a wild card string that conforms to the Visual Basic LIKE operator (which is used to filter leaf names).



● **Note:** The Import Items list box only displays the list of tags that the user would like to import. The driver will validate the tags after the device has completed configuration. Any tags found invalid will not be created, and a message will be posted on the Event Log.

Data Types Description

Data Type	Description
Boolean	Single bit
Byte	Unsigned 8-bit value bit 0 is the low bit bit 7 is the high bit
Char	Signed 8-bit value bit 0 is the low bit bit 6 is the high bit bit 7 is the sign bit
Word	Unsigned 16-bit value bit 0 is the low bit bit 15 is the high bit
Short	Signed 16-bit value bit 0 is the low bit bit 14 is the high bit bit 15 is the sign bit
DWord	Unsigned 32-bit value bit 0 is the low bit bit 31 is the high bit
Long	Signed 32-bit value bit 0 is the low bit bit 30 is the high bit bit 31 is the sign bit
QWord	Unsigned 64-bit value bit 0 is the low bit bit 63 is the high bit
LLong	Signed 64-bit value bit 0 is the low bit bit 62 is the high bit bit 63 is the sign bit
Float	32-bit floating point value. The driver interprets two consecutive 16-bit registers as a floating point value by making the second register the high word and the first register the low word.
Double	64-bit floating point value
String	Null-terminated character array
Date	Date YYYY-MM-DDTHH:MM:SS.MMM

Address Descriptions

The OPC XML-DA Client Driver address descriptions refer to fully-qualified Item IDs defined on an OPC XML-DA server. Arrays are supported for all data types.

The screenshot shows the OPC XML-DA Client Driver software interface. The main window displays a tree view on the left and a table of tag descriptions on the right. The tree view shows a hierarchy starting with 'Channel1', which contains 'Device1', which in turn contains 'Channel_1', '0Statistics', '0System', 'Device_1', and 'Device_2'. The table on the right lists the following tags:

Tag Name	Address	Data Type	Scan Rate	Scaling
Bool_1	Channel_1.Device_1.Bool_1	Boolean	100	None
Tag_1	Channel_1.Device_1.Tag_1	Short	100	None
Tag_2	Channel_1.Device_1.Tag_2	Short	100	None
Tag_3	Channel_1.Device_1.Tag_3	Short	100	None

Below the table is a log window with columns for Date, Time, Source, and Event. The status bar at the bottom indicates 'Ready', 'Default User', 'Clients: 0', and 'Active tags: 0 of 0'.

Error Descriptions

The following error / warning messages may be generated. Click on the link for a description of the message.

[Access is denied for item <address>](#)

[An invalid subscription handle was passed to the server on <channel/device name>](#)

[General SOAP fault on <channel/device name>](#)

[Invalid continuation point](#)

[Invalid filter](#)

[Invalid hold time for item <address>](#)

[Invalid item name for item <address>](#)

[Invalid path for item <address> with path <path>](#)

[Item <address> is Write Only](#)

[Server is busy on <channel/device>](#)

[Server is in an abnormal state on <channel/device name>](#)

[SOAP data queue overflow](#)

[SOAP HTTP communication error on <channel/device name>](#)

[SOAP SSL authentication error on <channel/device name>](#)

[SOAP TCP communication error on <channel/device name>](#)

[SOAP XML parse error on <channel/device name>](#)

[The data type cannot be accepted for item <address>](#)

[The item name <Channel>.<Device>.<Item Path>.<Item Name> is no longer available in the server address space](#)

[The item path <Channel>.<Device>.<Item Path> is no longer available in the server address space](#)

[The Property ID is invalid for the item <address>](#)

[The server reported an unspecified failure on <channel/device>](#)

[The server reported that the operation timed out on <channel/device name>](#)

[The server returned a "Not Supported" error on item <address>](#)

[The server returned an "Out of Memory" error on <channel/device name>](#)

[The specified write value for <address> is out of range](#)

[Unknown data type for tag <item> on device <device>. Using Default](#)

 For information on errors that are not listed above, contact the OPC XML-DA server manufacturer.

Access is denied for item <address>

Error Type:

Error

Possible Cause:

The server denied Read and/or Write access to the specified item. This is usually caused by Web Service security (such as, globally-disabled write capabilities).

Solution:

This problem cannot be rectified by the client: users may need to modify the OPC XML-DA server permissions or contact the OPC XML-DA server provider.

An invalid subscription handle was passed to the server on <channel/device name>

Error Type:

Error

Possible Cause:

The OPC XML-DA server referenced by the channel and device terminated the subscription abnormally.

Solution:

If this error persists, contact the OPC XML-DA server provider.

Invalid continuation point

Error Type:

Error

Possible Cause:

The server timed out.

Solution:

Retry browsing.

Invalid filter

Error Type:

Error

Possible Cause:

The filter string is invalid.

Solution:

Enter a valid filter string.

Note:

For information on acceptable filter strings, refer to the OPC XML-DA server documentation.

Invalid hold time for item <address>

Error Type:

Error

Possible Cause:

The Hold Time is too long.

Solution:

Change the Hold Time to a smaller value.

Note:

The Hold Time's valid range is determined by the OPC XML-DA server.

See Also:

[Exception Mode Options](#)

Invalid item name for item <address>

Error Type:

Error

Possible Cause:

The item name does not conform to the server's syntax.

Solution:

Change the item name to match one that is specified in the OPC XML-DA server.

Invalid path for item <address> with path <path>

Error Type:

Error

Possible Cause:

The Item Path does not conform to the server's syntax.

Solution:

If this error persists, contact the OPC XML-DA server provider.

Item <address> is Write Only

Error Type:

Warning

Possible Cause:

An attempt was made to read a value that is Write Only. This value may not be read from or returned as part of a write response.

Solution:

Do not attempt to read the item.

General SOAP fault on <channel/device name>

Error Type:

Error

Possible Cause:

The OPC XML-DA server referenced by the channel and device is not responding.

Solution:

Check the SOAP endpoint URL, proxy settings, security settings, and network connection.

Server is busy on <channel/device>

Error Type:

Error

Possible Cause:

The OPC XML-DA server referenced by the channel and device is processing another polled refresh for one or more of the subscriptions.

Solution:

If the action is a Write, resend the request. Otherwise, the error will be handled by the driver and the request will be resubmitted automatically.

Server is in an abnormal state on <channel/device name>

Error Type:

Error

Possible Cause:

The OPC XML-DA server referenced by the channel and device is in an abnormal state. The requested operation could not be completed.

Solution:

If this error persists, contact the OPC XML-DA server provider.

SOAP data queue overflow

Error Type:

Error

Possible Cause:

The OPC XML-DA server referenced by the channel and device is not responding.

Solution:

Check the SOAP endpoint URL, proxy settings, security settings, and network connection.

SOAP HTTP communication error on <channel/device name>

Error Type:

Error

Possible Cause:

An HTTP error occurred while communicating with the OPC XML-DA server referenced by the channel and device.

Solution:

If this error persists, contact the OPC XML-DA server provider.

SOAP SSL authentication error on <channel/device name>

Warning

Possible Cause:

An error occurred authenticating the OPC XML-DA server referenced by the channel.

Solution:

Check the SOAP endpoint URL, proxy settings, security settings, and network connection.

SOAP TCP communication error on <channel/device name>

Error Type:

Error

Possible Cause:

The OPC XML-DA server referenced by the channel and device is not responding.

Solution:

Check the SOAP endpoint URL, proxy settings, security settings, and network connection.

SOAP XML parse error on <channel/device name>

Error Type:

Warning

Possible Cause:

The XML response from the OPC XML-DA server referenced by the channel and device could not be parsed.

Solution:

If this error persists, contact the OPC XML-DA server provider.

The data type cannot be accepted for item <address>

Error Type:

Error

Possible Cause:

The tag's data type does not match the data type of the item in the OPC XML-DA server.

Solution:

Ensure that the tag's data type matches the data type of the item in the OPC XML-DA server.

The item name <Channel>.<Device>.<Item Path>.<Item Name> is no longer available in the server address space

Error Type:

Error

Possible Cause:

The item has been removed by the OPC XML-DA server.

Solution:

Ensure that the requested item is available in the OPC XML-DA server.

The item path <Channel>.<Device>.<Item Path> is no longer available in the server address space

Error Type:

Error

Possible Cause:

The item path is not recognized by the OPC XML-DA server.

Solution:

If this error persists, contact the OPC XML-DA server provider.

The Property ID is invalid for the item <address>

Error Type:

Error

Possible Cause:

The Property ID is not valid for the item.

Solution:

If this error persists, contact the OPC XML-DA server provider.

The server reported an unspecified failure on <channel/device>

Error Type:

Error

Possible Cause:

The execution of a request failed due to unknown reasons, although the server is in a state that should support that request.

Solution:

If this error persists, contact the OPC XML-DA server provider.

The server reported that the operation timed out on <channel/device name>

Error Type:

Error

Possible Cause:

The operation took too long to complete on the OPC XML-DA server referenced by the channel and device.

Solution:

The length of time allowed for the request to complete is determined by the server manufacturer. If this error persists, contact the OPC XML-DA server provider.

The server returned a 'Not Supported' error on item <address>

Error Type:

Error

Possible Cause:

An attempt was made to write to the quality and/or timestamp (which is not supported by the server).

Solution:

N/A.

The server returned an "Out of memory" error on <channel/device name>

Error Type:

Error

Possible Cause:

The server does not have enough resources to fulfill the request.

Solution:

If the error persists, contact the OPC XML-DA server provider.

The specified write value for <address> is out of range

Error Type:

Error

Possible Cause:

An attempt was made to write a value that is not supported by the data type.

Solution:

Write a value within the supported range.

See Also:

[Data Types Description](#)

Unknown data type for tag <item> on device <device>. Using Default

Error Type:

Warning

Possible Cause:

1. The target server did not return a data type.
2. The data type that was entered is not supported by the driver.

Solution:

1. The server will determine the data type once a client is connected.
2. Enter a data type that is supported by the driver.

Time-Sensitive Data Acquisition

This driver operates very differently than other COM / DCOM-based drivers. XML is transmitted over the network in plain text with considerable serialization and deserialization overhead. Furthermore, all API calls are synchronous due to limitations in the SOAP serialization library and the HTTP protocol. This means that subscriptions on the same channel will compete for time.

To resolve this, users can put time-sensitive subscriptions onto their own channel where they will get a dedicated thread. This minimizes the interactions between subscriptions.

Index

A

Access is denied for item <address> 17

Address Descriptions 16

An invalid subscription handle was passed to the server on <channel/device name> 17

C

Channel Assignment 9

Channel Properties – Advanced 7

Channel Properties – General 5

Channel Properties – Write Optimizations 6

Communications Parameters 13

D

Data Collection 10

Data Types Description 15

Device Properties – General 9

Diagnostics 6

Do Not Scan, Demand Poll Only 11

Driver 9

Duty Cycle 6

E

Error Descriptions 17

Exception Mode Options 12

G

General 9

General SOAP fault on <channel/device name> 19

H

HTTP Authentication 8

HTTP Proxy 8

I

ID 9
Identification 5, 9
Import 13
Initial Updates from Cache 11
Inter-Device Delay 7
Invalid continuation point 18
Invalid filter 18
Invalid hold time for item <address> 18
Invalid item name for item <address> 18
Item <address> is Write Only 19

M

Model 9

N

Name 9
Non-Normalized Float Handling 7

O

OPC Group 11
OPC Server 7
Operating Mode 9
Optimization Method 6
Overview 4

P

Project Architecture 5

R

Replace with Zero 7
Respect Tag-Specified Scan Rate 11

S

Scan Mode 10
Server is busy on <channel/device> 19
Server is in an abnormal state on <channel/device name> 19
Simulated 10
SOAP data queue overflow 19
SOAP HTTP communication error on <channel/device name> 20
SOAP SSL authentication error on <channel/device name> 20
SOAP TCP communication error on <channel/device name> 20
SOAP XML parse error on <channel/device name> 20

T

Tag Counts 6, 10
The data type cannot be accepted for item <address> 20
The item name <Channel>.<Device>.<Item Path>.<Item Name> is no longer available in the server address space 21
The item path <Channel>.<Device>.<Item Path> is no longer available in the server address space 21
The Property ID is invalid for the item <address> 21
The server reported an unspecified failure on <channel/device> 21
The server reported that the operation timed out on <channel/device name> 21
The server returned a 'Not Supported' error on item <address> 22
The server returned an 'Out of memory' error on <channel/device name> 22
The specified write value for <address> is out of range 22
Time-Sensitive Data Acquisition 23

U

Unknown data type for tag <item> on device <device>. Using Default 22
Unmodified 7

W

Write All Values for All Tags 6
Write Only Latest Value for All Tags 6
Write Only Latest Value for Non-Boolean Tags 6