OPC DA Client Driver Help

© 2012 Kepware Technologies

Table of Contents

OPC DA Client Driver Help

Help version 1.021

CONTENTS

Overview

The OPC DA Client Driver provides an easy and reliable way to connect OPC DA Client devices to OPC Client applications, including HMI, SCADA, Historian, MES, ERP and countless custom applications. It can be used to consolidate data from a number of OPC servers and is made available to server clients through all supported interfaces (such as OPC DA, OPC DX, DDE, SuiteLink, NIO/PDB, and so forth).

The OPC DA Client Driver has the ability to do the following:

- Provide a single, reliable connection point for accessing data from multiple OPC servers on both local and remote machines.
- Optimize OPC servers' performance through OPC groupings (which may be different than those that are required or allowed by clients).
- Configure connection monitoring and reconnect behavior for each OPC server.
- Provide connectivity to remote OPC servers for clients that do not support DCOM.
- Provide connectivity to multiple OPC servers from clients that do not support multiple connections (or handle them well).
- Allow connectivity to OPC servers that use different interfaces that are supported by clients (such as DDE, SuiteLink, NIO/PDB, and so forth).

Note: The maximum number of channels supported is 128. The maximum number of devices supported per channel is 256.

OPC Compliance

This driver supports OPC DA 1.0, 2.0, and 3.0 connections with underlying servers. It will use 2.0 calls if supported, and will automatically fall back to 1.0 if not supported. It does not support 3.0 advanced operations.

Note: By default, the driver will request data updates by exception (active groups and items). It can be configured to poll items (inactive groups and active items) using asynchronous reads. All writes will be asynchronous.

Project Architecture

An OPC DA Client Driver channel represents a connection to an OPC server; an OPC DA Client Driver device represents an OPC group. The OPC client can connect to OPC servers provided by one or more vendors. The following diagram shows the OPC client and server relationship.
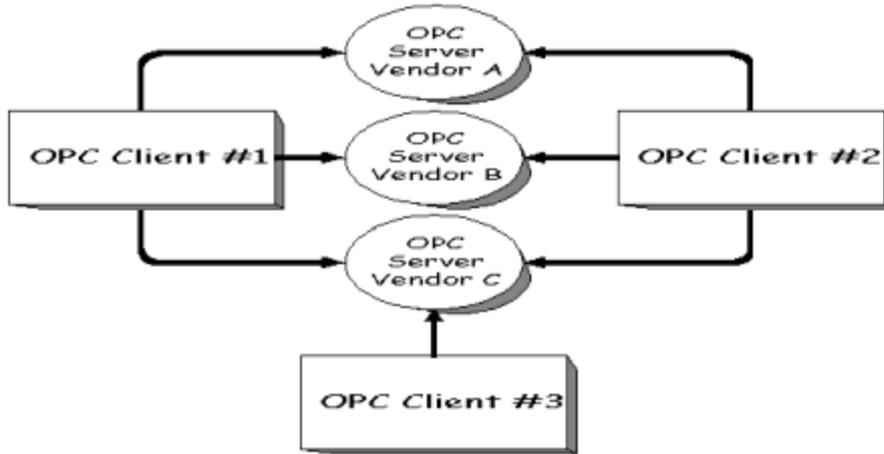
The OPC group object maintains information about itself and provides the mechanism for containing and logically organizing OPC items. The following diagram shows the group and item relationship.
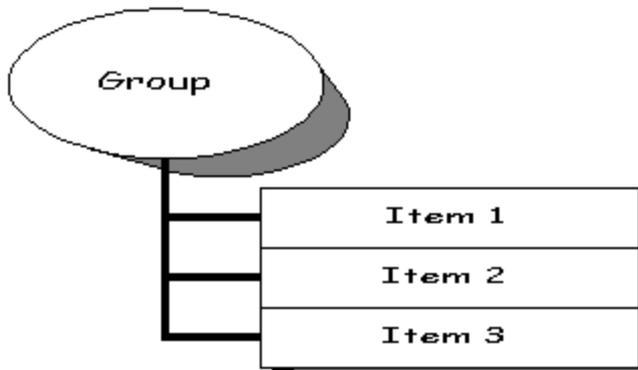
OPC Server

This parameter specifies the OPC server to which the channel will connect.
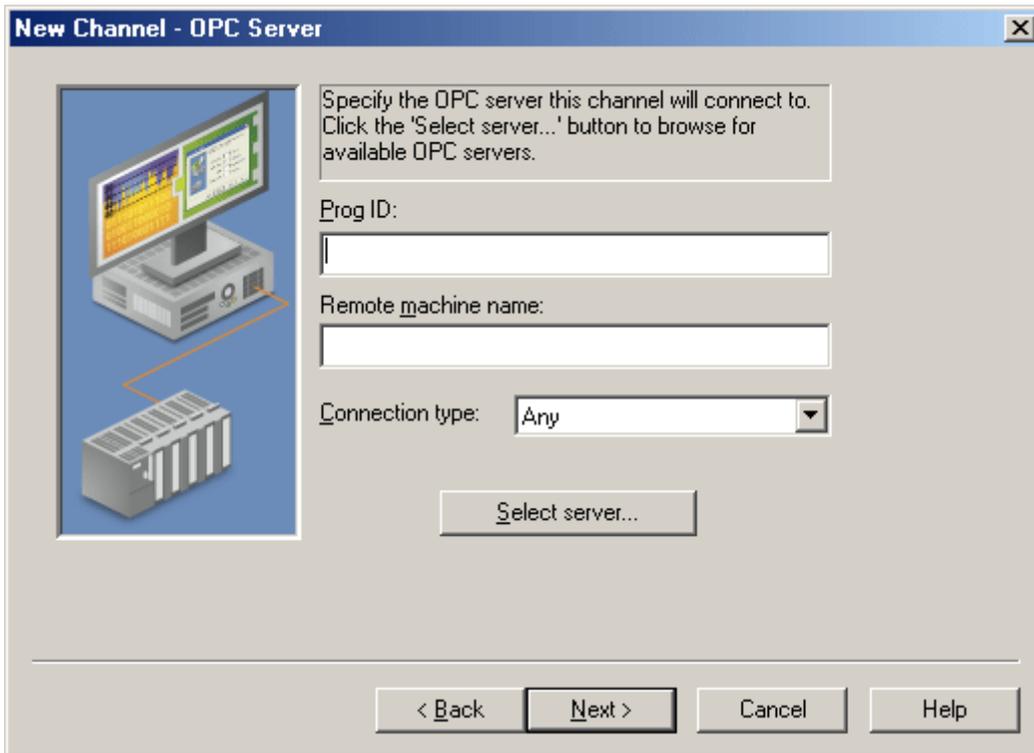
Descriptions of the parameters are as follows:

- **Prog ID:** This parameter specifies the Program ID of the OPC server to which the driver will connect.
- **Remote Machine Name:** This parameter specifies the name of the machine in which the OPC server resides, as specified by the Prog ID. This field should be left blank if the server is located on the same machine as the driver.
- **Connection Type:** This parameter specifies the type of connection that the driver should establish with the OPC server on the local machine. Options include InProc, Local, and Any. The default setting is Any.

    Note: When the server is registered as InProc, an InProc connection will be attempted. Otherwise, a Local connection will be attempted.
- **Select Server:** When pressed, this button invokes the Select OPC Server dialog. For more information, refer to the subtopic below.

Select OPC Server
This dialog is used to browse the network for an OPC server.
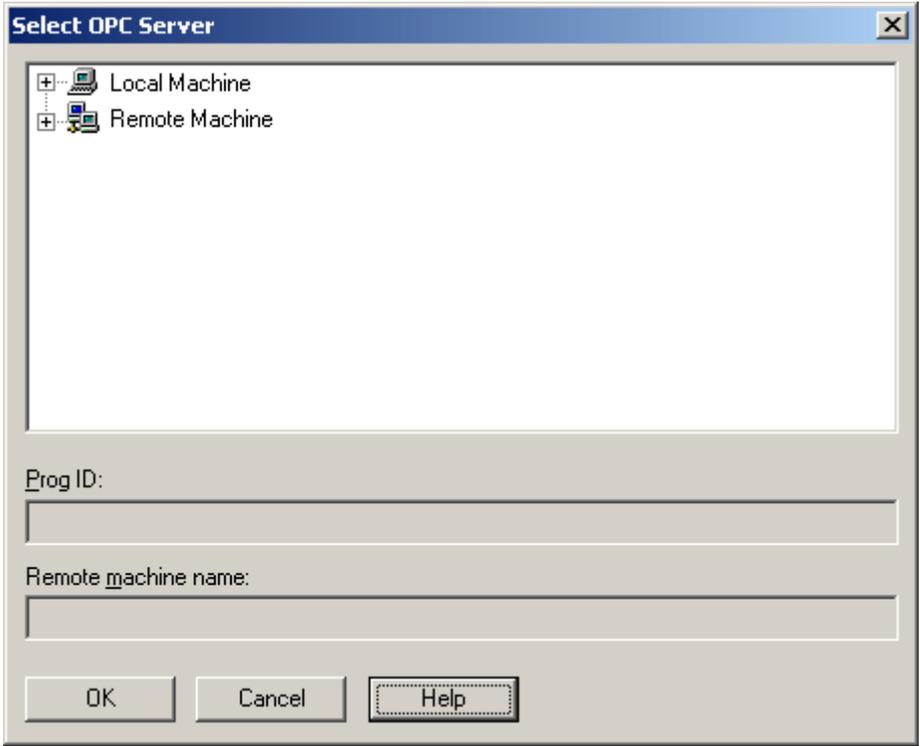
Connection

This parameter specifies the time interval between connection retry attempts and integrity polls. Additional integrity checks may be configured at the group level. For more information, refer to **Watchdog**.
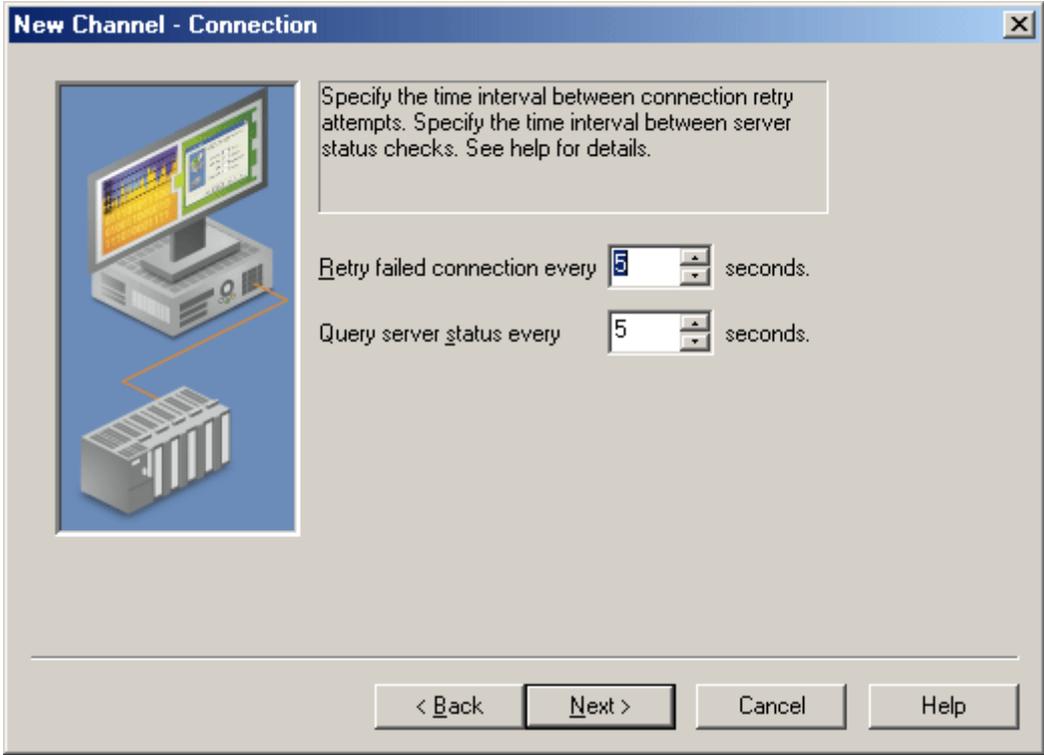
Descriptions of the parameters are as follows:

- **Retry Failed Connection Every ___ Seconds:** This parameter specifies the time between connection attempts. The valid range is 5 to 600 seconds. The default setting is 5 seconds.

- **Query Server Status Every ___ Seconds:** This parameter specifies how often the driver will test the connection to the underlying server by sending a GetStatus message. If a response is not received, the driver will automatically reconnect to the server. The valid range is 5 to 30 seconds. The default setting is 5 seconds.

OPC Group

This parameter specifies the properties of the OPC group associated with the device.

Descriptions of the parameters are as follows:

- **Name:** This parameter specifies an optional name for identifying the group. If nothing is entered, the underlying OPC server will generate a unique name.
- **Update Mode:** This parameter specifies the update mode. Options include Exception and Poll. Descriptions of the options are as follows:

    - **Exception:** This option will be added active, and the server will notify the driver of data value and quality changes. The driver will first try to establish an OPC 2.0 callback for the group. If that is not available, the driver will attempt to establish an OPC 1.0 callback.
    - **Poll:** This option will be added inactive, and the driver will read all items attached to the group/device at the configured update/poll rate (or as close as possible). Asynchronous reads will be used. OPC DA 2.0 asynchronous reads will be used if supported by the OPC server. If that is not supported, OCP DA 1.0 asynchronous reads will be used.

        Note: In Poll mode, all tags are added as inactive. The server will briefly set the items active during each poll cycle; as such, users may notice the active tag count in the server's status bar fluctuate. This is normal.
- **Update/Poll Rate (ms):** When Exception mode is used, this parameter specifies how often the underlying OPC server should provide updates for changing data. When Poll mode is used, this parameter specifies how often the driver should read the items attached to this group. The valid range is 0 to 3600000 milliseconds. The default setting is 1000 milliseconds.
- **Percent Deadband:** This parameter specifies the percent change in data required to notify the client of a data change. This setting is used for Exception mode only.
- **Language ID:** This parameter specifies the language that will be used by the underlying server when returning values as text for operations on the group. The default setting is 1033 (English).

Communications Parameters

The Communications Parameters specifies the maximum number of items that can be included in each Read and Write request, as well as the asynchronous Read and Write timeouts.
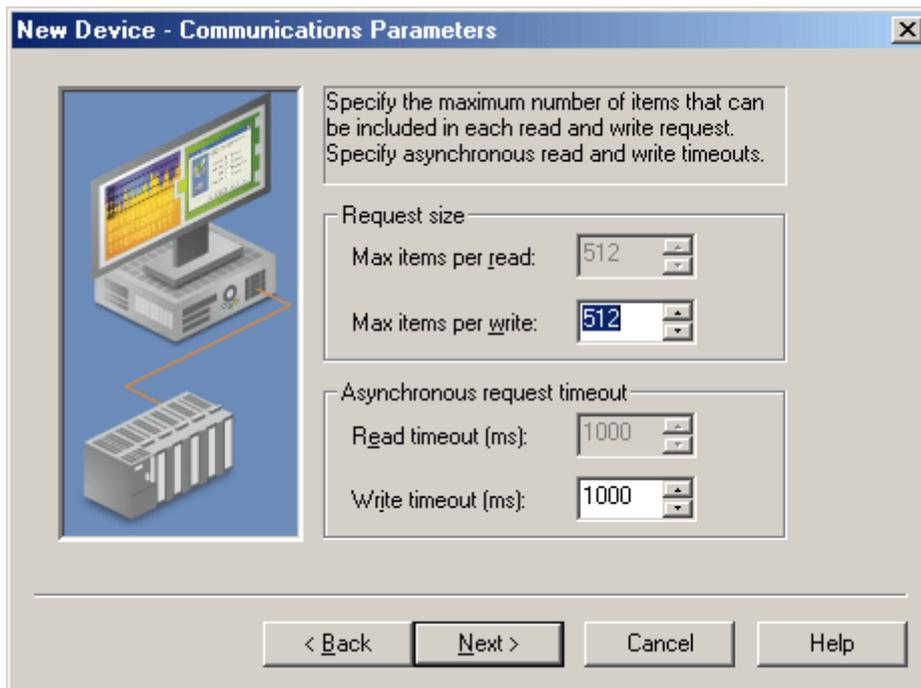

Descriptions of the parameters are as follows:

- **Max Items Per Read:** This parameter specifies a limit on the number of items that can be included in a single Read request. The valid range is 1 to 512. The default setting is 512.
- **Max Items Per Write:** This parameter specifies a limit on the number of items that can be included in a single Write request. The valid range is 1 to 512. The default setting is 512.
- **Read Timeout (ms):** This parameter specifies how long the driver will wait for a Read complete notification to be returned from the server before any other Read or Write requests are sent. If the expected notification is not received, the items included in the request will be set to Bad quality (and will remain Bad until the next successful Read).

  Note: Read settings will be disabled when the Update Mode is Exception. The driver will not issue Read requests to the server.
- **Write Timeout (ms):** This parameter specifies how long 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.

Watchdog

When enabled, this parameter monitors callback integrity when using the Exception Update mode. A watchdog item is required to force callbacks at the requested update rate. Users can select the item that will be the watchdog; if one is not chosen, the driver will select an active item on its own.
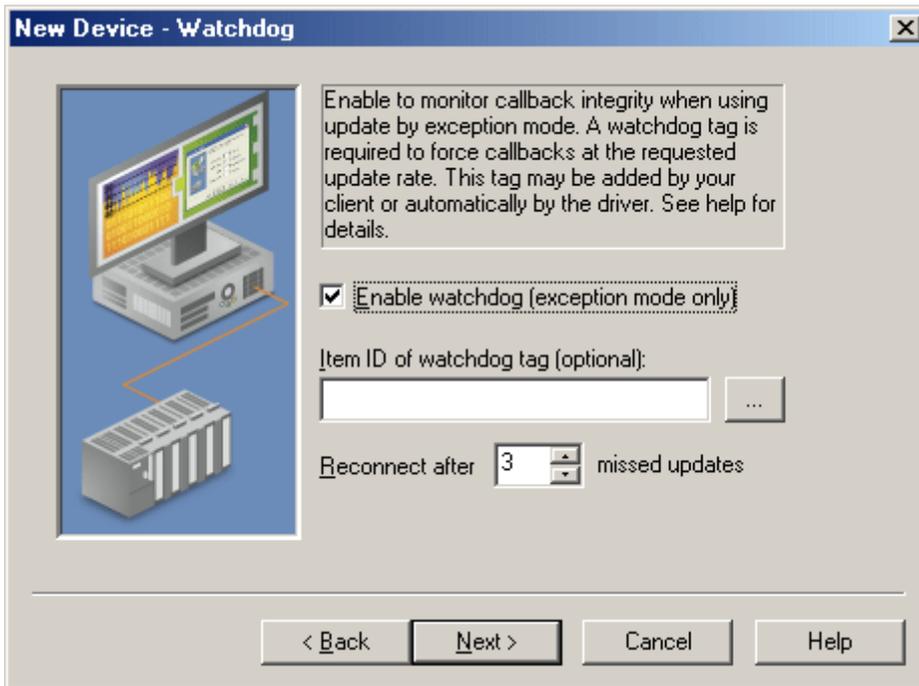
Descriptions of the parameters are as follows:

- **Enable watchdog:** When checked, this parameter will enable the watchdog callback monitoring feature. For more information, refer to **Watchdog Callback Monitoring**.
- **Item ID of watchdog tag:** This optional parameter specifies the Item ID of the tag that will be automatically added by the driver upon connection. This field may be left blank if one or more suitable watchdog tags will be active for the connection's duration.

  Note: To locate a watchdog Item ID on the network, click the browse button.
- **Reconnect after ___ missed updates:** This parameter specifies the number of update periods that may pass without a data change notification from the underlying OPC server before the driver assumes there is a problem and reconnects. The valid range is 2 to 10. The default setting is 3.

Import

This dialog is used to browse the server for tags.
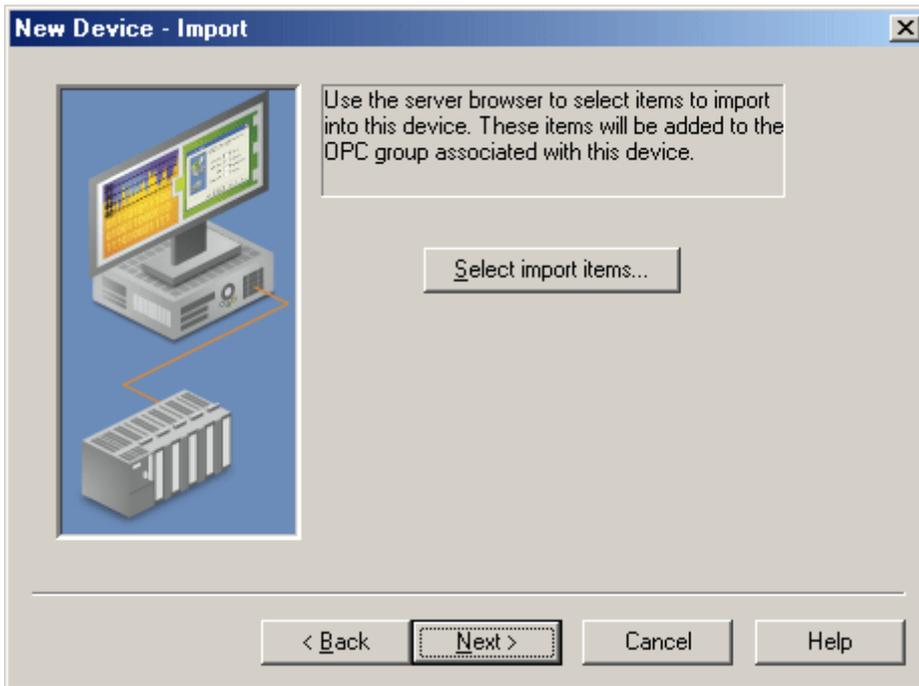
Description of the parameter is as follows:

- **Select Import Items:** When clicked, this button invokes the Select Items to Import dialog.

Select Items to Import

The driver adds all items to the tree and then validates them in the **Import Items** list. To add items to import, select them from the tree and click **Add Items >**. The Import Items list will show the items that will be imported. To remove items from import, simply select and click **< Remove Items**. Then, select **OK**.

The driver browses the server's address space and displays the results in the tree control on the left. Items selected for import will appear in the list control on the right. To populate the Import Items list, select one or more items in the browse tree and then click **Add Items >**. To add items under a specific branch, first select that branch and then click **Add branch >**. To remove items from the Import Items list, simply select one of more items in the browse tree and then click **< Remove Items**. To add the selected items to the current tag database, click **OK**.

Note: The Import Item list box is only a list of tags that the user would like to import. The driver will validate the tags after the device has completed configuration. Any tag found to be invalid will not be created. A message will be posted on the event log.
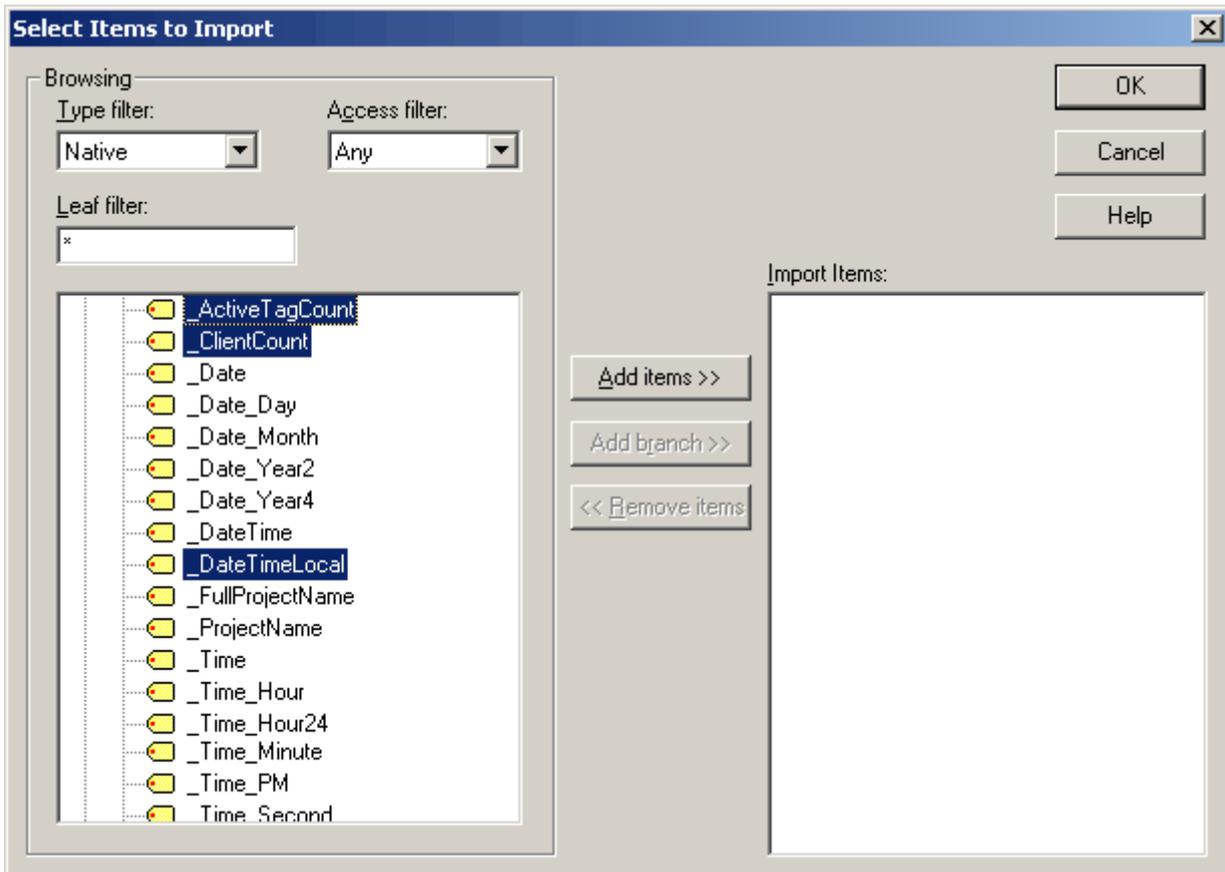
Watchdog Callback Monitoring

Applying the Exception selection for Update Mode is generally more efficient than polling. For example, users may not know whether a long time between update notifications is because none of the values associated with the group have remained the same or whether the underlying OPC server (or COM callback connection) has failed.

Because of this uncertainty, it is recommended that users employ a watchdog. This is an item configured in the underlying OPC server, whose value will change at least once per update period. As long as this item is referenced by the driver, it will force an update notification for every update period. When the watchdog monitoring feature is enabled, the driver will track the time since the last update. When this time exceeds a predefined limit, the driver will assume there has been a failure and will attempt to reconnect to the server.

**Note:** For more information, refer to **Watchdog**.

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
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	Zero terminated character array
Date	Date YYYY-MM-DDTHH:MM:SS.MMM

Address Descriptions

The OPC DA Client Driver's address descriptions refer to fully-qualified Item IDs of the server item.

Note: Arrays are supported for all data types.

Error Descriptions

The following error/warning messages may be generated. They are listed here in alphabetical order. Click on the link for a description of the message.

Add item failed for '<item>' on device '<device>'. Function error: <error code>

Error Type:
Serious

Possible Cause:
The server rejected an add items request.

Result:
The driver will not be able to read or write to the associated tag(s).

Solution:
The solution depends on the specific error code. There is most likely a problem in an underlying server (such as resource limits).

Add item failed for '<item>' on device '<device>s' Item error: <error code>

Error Type:
Serious

Possible Cause:
The server failed to add the specified item.

Result:
The driver will not be able to read or write to the associated tag(s).

Solution:
Verify that the Item ID set for the tag is valid.

Add watchdog item failed for ' <item>' on device '<device>'. Item error: '<error code>'. ' <error description>'

Error Type:
Warning

Possible Cause:

The driver failed to add the specified watchdog item. The particular reason can be determined from the error code. The driver will include an error description in this message when available from the server. The specified Item ID is not usually valid in the current server configuration.

Solution:
Verify the watchdog's Item ID and change it as needed.

Async Write 1.0 failed for '<item>' on device'<device>'. Callback item error: <error code>

Error Type:
Warning

Possible Cause:
The server processed the write request, but failed to successfully write to the underlying device.

Solution:
Verify that communications to device are good and that the point is writable.

Async Write 1.0 failed for '<item>' on device '<device>'. Function error: <error code>

Error Type:
Warning

Possible Cause:
The server rejected a write request that included the indicated item.

Solution:
The solution depends on the specific error code. There is most likely a problem in the underlying server (such as resource limits).

Async Write 1.0 failed for '<item>' on device '<device>'. Item error: <error code>

Error Type:
Warning

Possible Cause:
This message will be logged if the item was included in a write request that was rejected because of one or more invalid items. The error code here will indicate if this was one of the invalid items.

Solution:
Verify that Item ID is valid and that item is valid.

Async Write 2.0 failed for '<item>' on device '<device>'. Callback item error: <error code>

Error Type:
Warning

Possible Cause:
The server processed the write request, but failed to successfully write to underlying device.

Solution:
Verify that communications to device are good and that point is writable.

Async Write 2.0 failed for '<item>' on device '<device>'. Function error: <error code>

Error Type:
Warning

Possible Cause:

The server rejected a write request that included the indicated item.

Solution:
The solution depends on the specific error code.There is most likely a problem with an underlying server (such as resource limits).

Async Write 2.0 failed for '<item>' on device '<device>'. Item error: <error code>

Error Type:
Warning

Possible Cause:
The server was unable to process a write request for this item. Other items in request may have been accepted.

Solution:
Verify that the Item ID is valid and that the item is valid.

Failed to acquire item management interface for device '<device>'

Error Type:
Serious

Possible Cause:
The driver was not able to obtain the item management COM interface from the server.

Result:
The driver will not be able to read or write to tags associated with device/group.

Solution:
This is a required interface. Contact Technical Support.

Failed to add group for device '<device>'. Reason: <error code>

Error Type:
Serious

Possible Cause:
Server rejected request to add specified group.

Result:
The driver will not be able to read or write to tags associated with device/group.

Solution:
The solution depends on the specific error code. Because the error may have been caused by a duplicate group name, try a different name in the OPC Group device properties page. It may also be due to a problem in an underlying server (such as resource limits).

Failed to connect to server on channel '<channel>'

Error Type:
Serious

Possible Cause:
Could not connect to server.

Solution:
Check the Program ID, machine name and network connection.

Failed to establish callback for device '<device >'

Error Type:
Serious

Possible Cause:

The driver was not able to establish a COM callback for the device/group.

Result:
The driver will not be able to read, write or receive data change notification for tags associated with the device or group.

Solution:
1. Verify that OPC DA 1.0 or 2.0 is supported.
2. Verify the DCOM settings (if from a remote server).

Reconnecting server on channel '<channel>' due to GetStatus failure

Error Type:
Warning

Possible Cause:
The driver did not receive a response to the GetStatus request. A server, network or COM failure is assumed.

Solution:
The driver will automatically reconnect.

Reconnecting server on channel '<channel>' due to shutdown notification

Error Type:
Warning

Possible Cause:
The server has sent a notification that it is shutting down.

Solution:
The driver will automatically reconnect.

Reconnecting server on channel '<channel>' due to watchdog failure on device '<device>'

Error Type:
Warning

Possible Cause:
The time that elapsed since the last data change notification was longer than the configured limit. A server, network or COM failure is assumed.

Solution:
The driver will automatically reconnect.

Tag '<item>' on device '<device>' was not imported. Data type is not supported by this driver

Error Type:
Warning

Possible Cause:
The data type that was entered is not supported by the driver.

Solution:
Enter a data type that is supported by the driver.

Tag ' <item>' on device '<device>' was not imported. Item failed OPC server validation

Error Type:
Warning

Possible Cause:

The item selected for import did not have a valid Item ID. Some servers provide Item IDs that are intended to give the user information, but are not themselves valid Item IDs. For example, this server provides driver address syntax hints to the browser.

Solution:

This error can be avoided by not selecting special purpose Item IDs in the import item browser; however, no harm is done by selecting them.

Index

A

C

D

E

F

H

I

O

P

R

T

W