OPC Quick Client

Table of Contents

OPC Quick Client

Help version 1.033

CONTENTS

Overview
What is the OPC Quick Client and how is it used?

The Main Window
What does the OPC Quick Client do?

Server Connection
How do I connect to a server and start receiving data?

Operations
What type of operations can I perform?

Test Mode
How can I perform a series of tests on an OPC server?

License Agreement
It is recommended that users review the license agreement.

Overview

The OPC Quick Client assists in the testing and development of the OPC Data Access 1.0 and 2.0 servers. It supports both local and remote OPC server connections. Remote connections are handled through the operating system's DCOM interface.

Note: The OPC Quick Client does not fully support OPC DA 3.0. This help file assumes the use of Windows XP or Server 2003. Service Pack 4 is recommended.

Client Preparation

For information on how to prepare the client, follow the instructions below.

1. To start, create a server connection (which will be used to connect to an OPC server).

2. Next, add a group to the connection that contains varying properties (such as update rate, dead band and time bias).

3. Add items to the individual groups. The groups contain properties such as an initial active state, data type, and access path.

Note: Active items will automatically receive data change notifications. The current value and quality will be displayed in the Item View.

See Also: Server, Group, and Items.

Main Window

The following image is a view of the application.
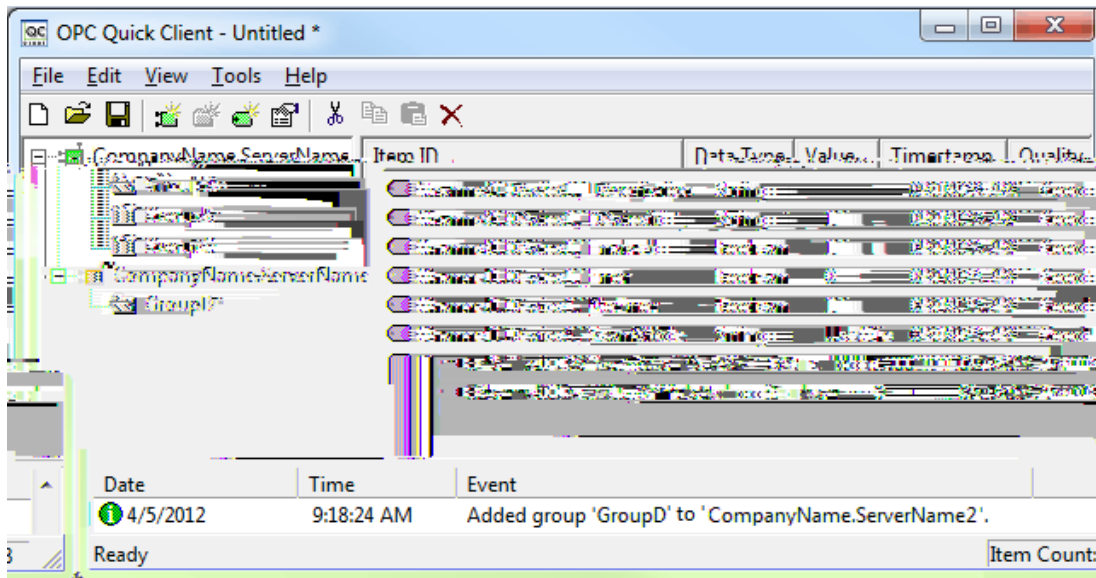
Title Bar
The Title Bar displays the name of the loaded project file. An asterisk (*) following the name indicates that it has been modified.

Toolbar
Descriptions of the icons are as follows:

Creates a new project.

Opens an existing project from disk.

Saves the current project to disk.

Creates a new server connection.

Creates a new group.

Creates a new item.

Edits the properties of the selected server, group, or item.

Cuts the current selection to the clipboard.

Copies the current selection to the clipboard.

Pastes the server, group, or items from the clipboard.

Deletes the current selection.

Group Window
The Group window manages a list of server connections along with their group objects. To display a menu of options applicable to the server/group, right-click on it. To display the property sheet for the server/group, double-click on it. For more information, refer to Server and Group.

Item Window
The Item window displays a list of item information for the selected group. To display a menu of options applicable to the item, right-click on it. To display the properties dialog for the item, double-click on it. For more information, refer to Items.

Note: Clicking on the Item ID column heading will cause the item list to be sorted based on the Item ID. To change column widths, click and drag the header for the desired column to the left or right end of the header button until the column is the desired size.

Message Log
The Message Log window logs status messages generated by the application. As with the Item window, the column headers may be resized. This window does not support sorting.

Server Connection

A server connection provides a link between an OPC server and this client. Groups are added through this connection. To create a new server connection, click **Edit | New Server Connection**. Alternatively, click the **New Server** toolbar button.
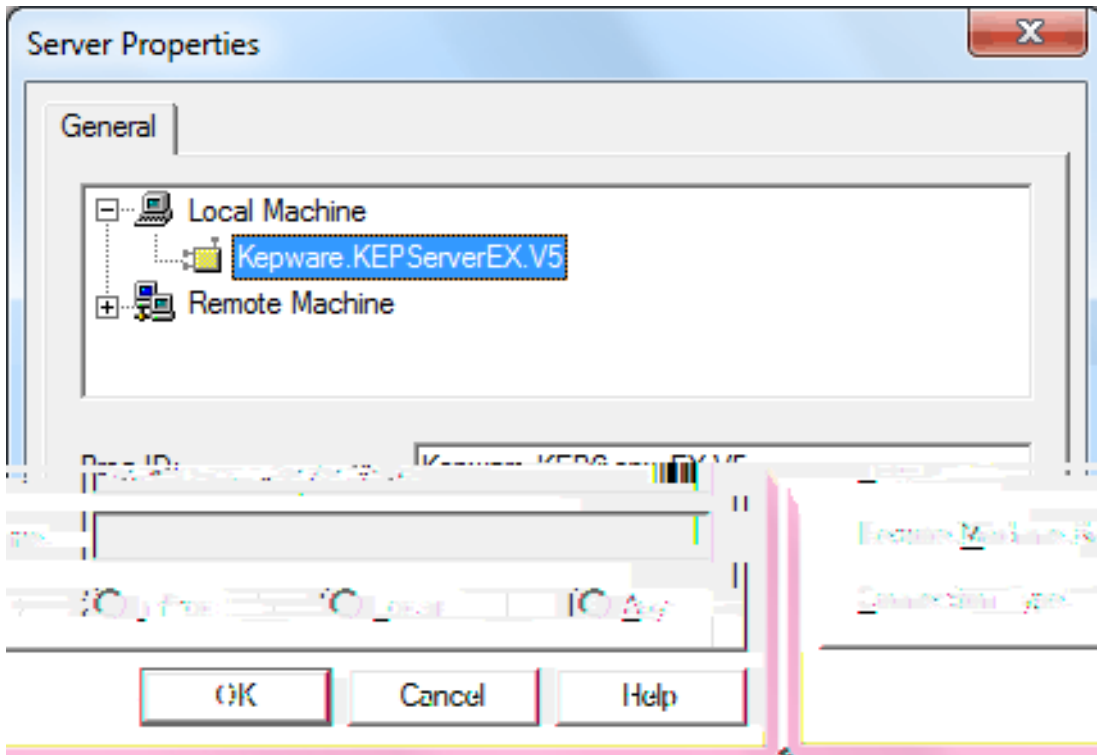
Descriptions of the parameters are as follows:

- **Prog ID**: This parameter specifies the Prog ID of the OPC Server to which the client will connect. Use the tree display to browse for registered servers. The Prog ID field can be automatically updated by double-clicking on any registered server.
- **Remote Machine Name**: This parameter specifies the name of the machine on which the OPC server resides. This field should be left blank if the server is located on the same machine as the client.
- **Connection Type**: This parameter specifies the type of connection the client should establish with the OPC server on the local machine. The options are **InProc**, **Local**, or **Any**. If the server is registered as InProc, then an InProc connection is attempted; otherwise, a Local connection will be attempted. This option is not available when connecting to a remote machine.

Server Properties

Once the client makes a successful connection to a server, the Server Properties dialog may be invoked by clicking **Edit | Properties**.

```
Server Properties                                                    X

  Status  Interfaces  Identification

      ProgID:              Kepware.KEPServerEX.V5
      Vendor:              Kepware
      Start Time:          11:08:16:301
      Current Time:        11:21:21:158
      Last Update Time:    11:21:21:050
      Server State:        Running
      Group Count:         3
      Major Version:       5
      Minor Version:       7
      Build Number:        55
      Connection Type:     Any (InProc/Local)




                              OK          Cancel        Help
```
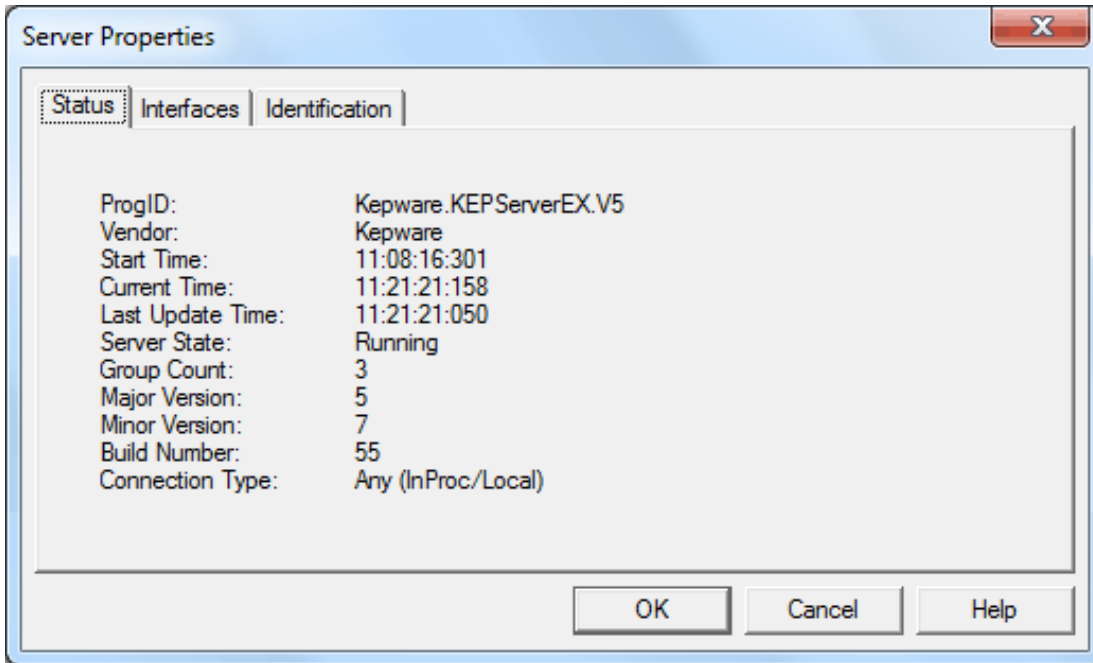
The client queries the server's status and obtains the following information:

| Status Item	Description
Prog ID	The Prog ID of the OPC Server to which the client will connect.
Vendor	A string that specifies vendor-specific information.
Start Time	Time at which the server was started.
Current Time	Current time as recognized by the server.
Last Update Time	Time that the server sent the last data update to the client on this connection.
Server State	The current state of the server. Valid states are Running, Failure, No Configuration, Suspended, and Test.
Major Version	Major version of the server software.
Minor Version	Minor version of the server software.
Build Number	Build number of the server software.
Connection Type	Type of connection with the server. This connection may be inproc, local, or any that the OS chooses.

Interfaces
The Interfaces tab lists the interfaces that are required at the server level. Interfaces that are not required are considered optional. The Supported column states which interfaces are supported.
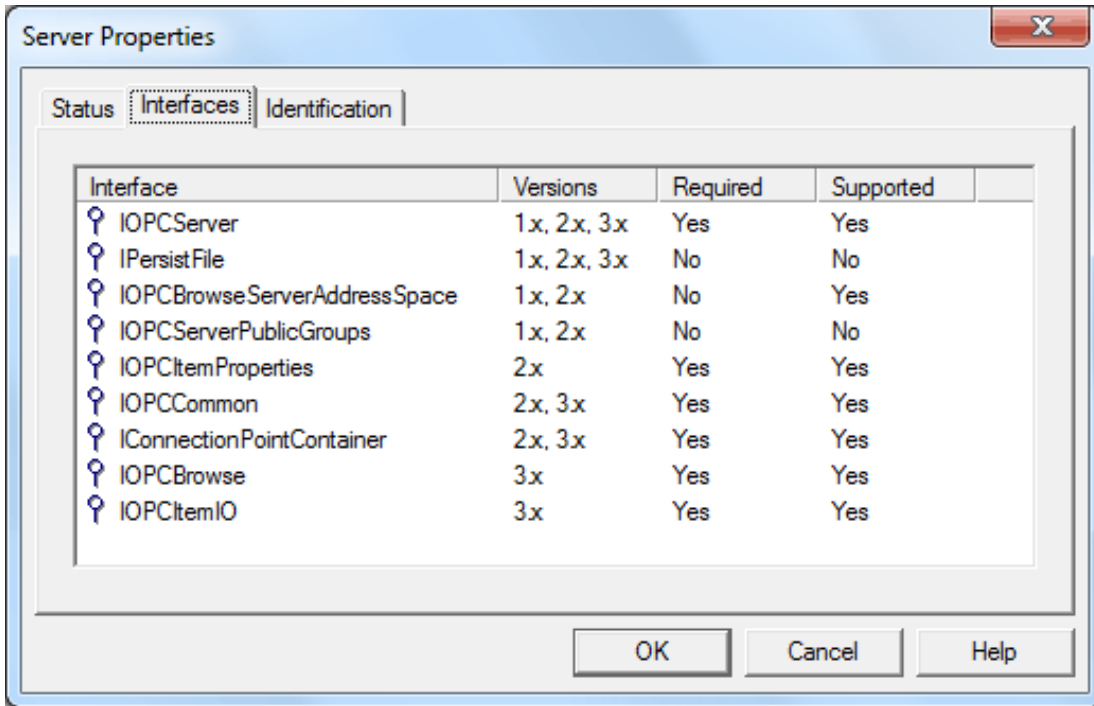
Identification
The Identification tab allows users to enter an optional name to identify the client to the server. The default client name is "Quick Client".
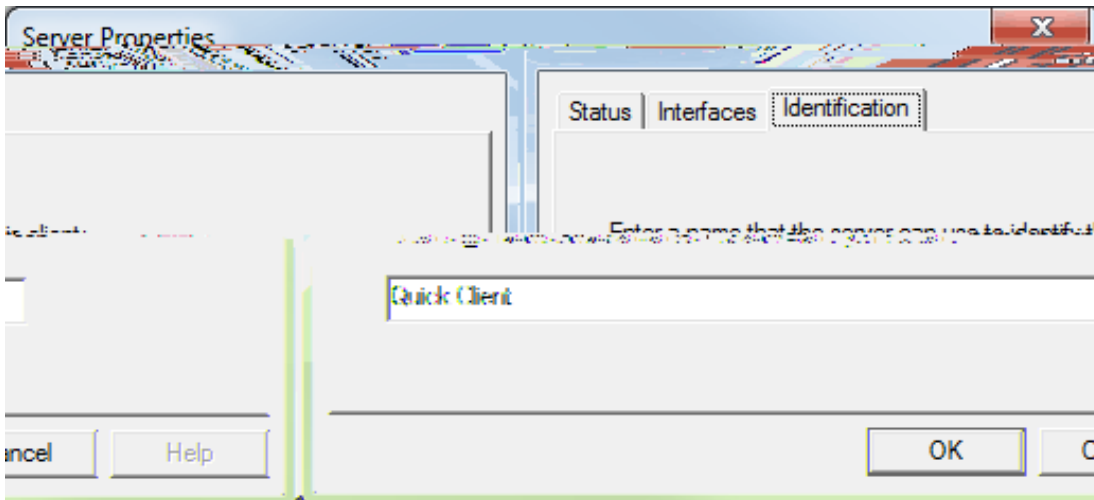
Visual/Operational Changes
When the OPC Quick Client adds, connects, or disconnects a server connection, there will be a change in its appearance. The image that appears to the left of each server connection will change to one of the following:

The client is connected to the server.

The client is disconnected from the server.

The client could not connect to the server (the server may not be properly registered).

Group

A group is used to organize a collection of items with a common set of properties. The group also specifies several properties, such as group name, update rate, time bias, percent dead band, Language ID, active state, and the data connection type (among others). To create a new group, click **Edit | New Group**. Alternatively, click **New Group** in the toolbar.

General

Group Properties

General

Name:

Update Rate (ms.): 1000

Time Bias (min.): 0

Percent Deadband: 0

Language ID: 1033

Update Notification: OPC 2.0/3.0   ☑ Active State

Keep Alive Rate (ms): 0

OK   Cancel   Help

Descriptions of the parameters are as follows:

- **Name:** This parameter specifies a name that will identify the group. If it is left empty, the OPC server will generate a unique name.
- **Update Rate:** This parameter specifies how often the OPC server will provide updates for changing data. The default setting is 1000 milliseconds.
- **Time Bias:** This parameter specifies the time difference between the client/server and the actual device. The information allows the client to convert the timestamp received by the server back to the local time of the device. The default setting is 0 minutes.
- **Percent Dead Band:** This parameter specifies the percent change in data required to notify the client of a data change. When scaling is enabled for the item, the required data change will be calculated using the scaled range. When scaling is not enabled for the item, the required data change will be calculated using the full range of the specified data type. The default setting is 0.
- **Language ID:** This parameter specifies the language that will be used by the server when returning values as text for operations on this group.
- **Update Notification:** This parameter specifies the method to be used when receiving data change notifications or initiating asynchronous requests. Valid selections are OPC 1.0 (No Timestamp), OPC 1.0 (With Timestamp) and OPC 2.0/3.0. Data Access 2.0/3.0 servers automatically supply timestamps with all data transactions.
- **Active State:** This parameter specifies the active state of the group. When the group is active, the client will receive data change updates for all active items at the specified update rate. The default setting is checked.
- **Keep Alive Rate:** This parameter specifies the amount of time between keep-alive messages sent from the server to the client. The default setting is 0 milliseconds.

Interfaces

Once a group has been successfully added to the project, click Edit | Properties and then select Interfaces. This will display the information on the interfaces that are required at the group level.
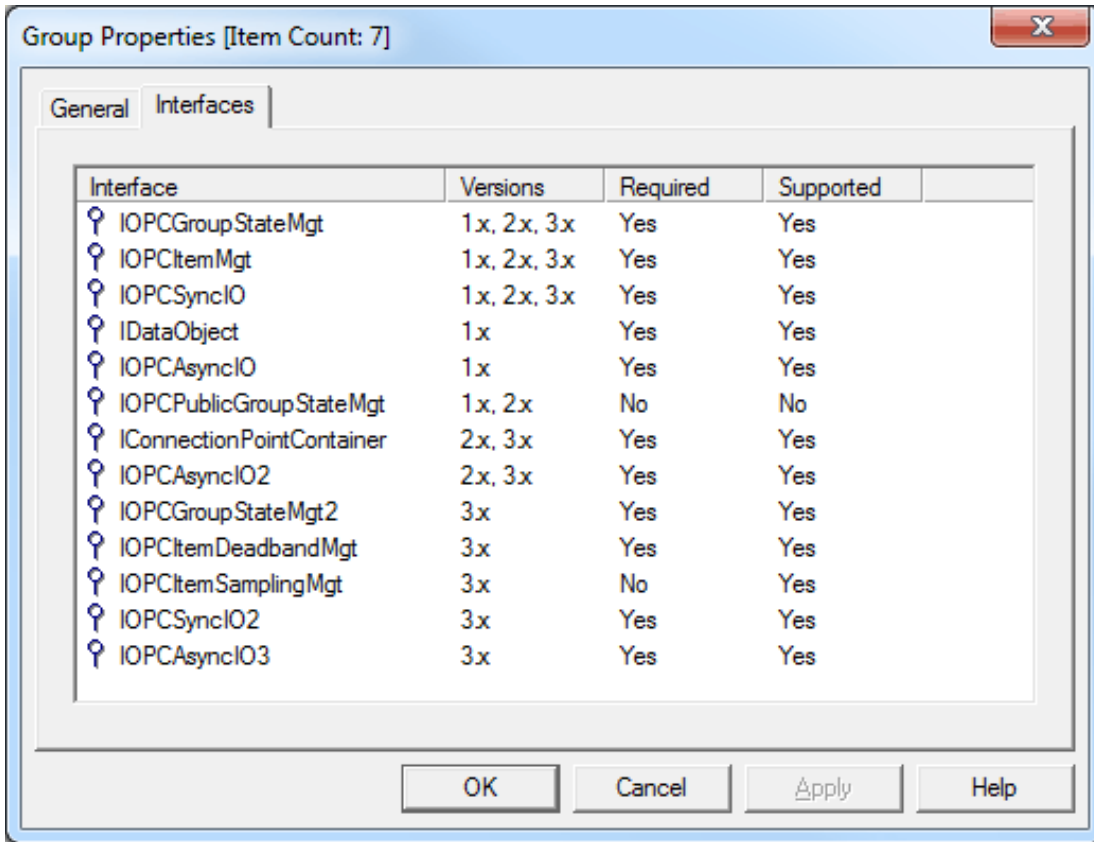
Note: Interfaces that are not required are considered optional. The Supported column states which group interfaces are supported by the server.

Visual/Operational Changes
When the OPC Quick Client adds, activates, or deactivates a group, the appearance of the folder icon will change to one of the following:

The group is active.

The group is inactive.

The client could not add the specified group to the server connection.

Items

Items represent data that may be accessed v   z   o   n

Descriptions of the parameters are as follows:

- **Access Path**: This parameter is required by some OPC servers to complete an item definition. To determine whether an access path is needed, refer to the OPC server's help documentation.
- **Item ID**: This parameter specifies the OPC server item that will be used to reference data. To determine valid item names for the location in question, refer to the OPC server's help documentation. If the server supports tag browsing, then an item can be selected by using the browse controls.
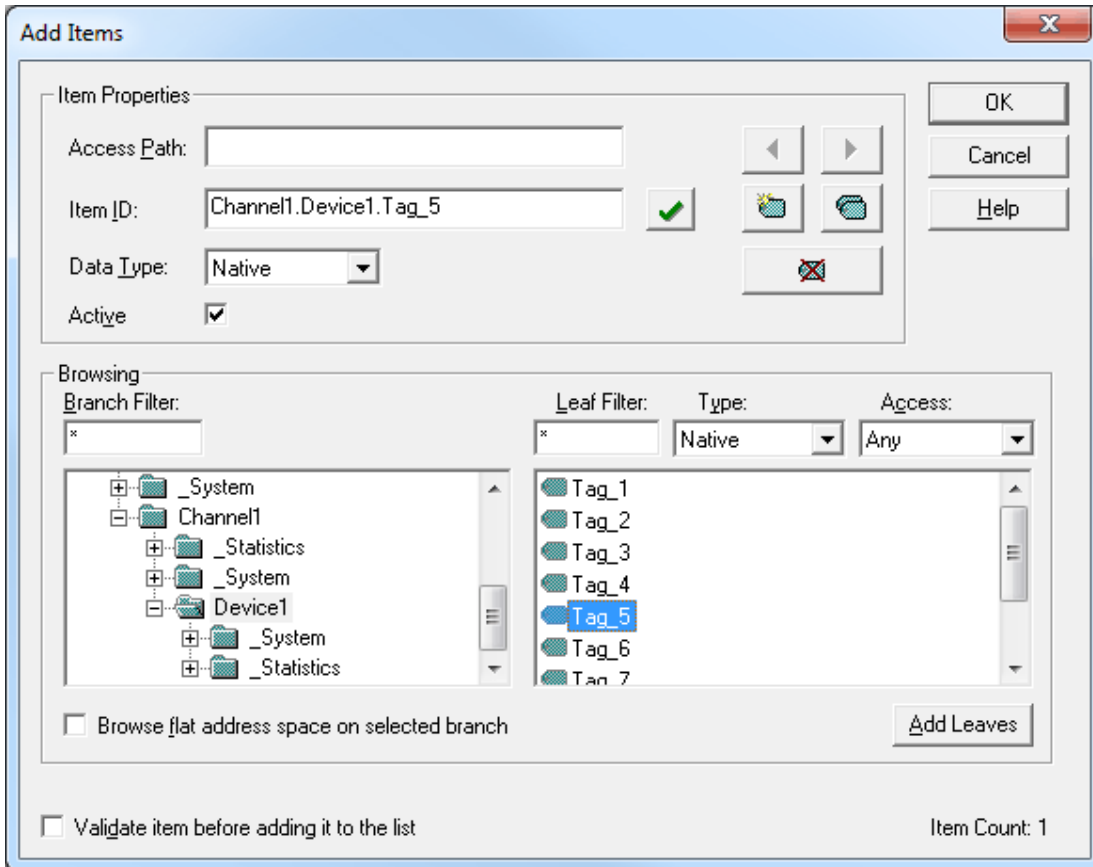- **Data Type**: This parameter specifies the requested data type that will be used when communicating between the OPC server and the OPC Quick Client. This should be specified to agree with the size and type of the register or memory location that is being addressed. For more information, refer to Data Types Description.
- **Active**: This parameter specifies the item's initial active state. It may be changed later. The default setting is checked.
- **Browse flat address space on selected branch**: When checked, this option modifies the item list by attaching the channel and device name. The default setting is unchecked.
- **Validate item before adding it to the list**: When checked, this option validates the item before it is added to the item list. The default setting is checked.
- **Add Leaves**: When clicked, this button adds the item that is currently selected in the item list to the Item ID field.

Button Definitions

Descriptions of the icons are as follows:

◀ Applies changes to the current tag and fills the dialog with properties of the previous tag in the list.

▶ Applies changes to the current tag and fills the dialog with properties of the next tag in the list. If no next tag exists, the client will attempt to automatically increment the current Item ID.

Creates a new item definition.

Duplicate (decimal, hexadecimal, or octal format) current item definition up to 32,767 items (automatically increments).

Deletes the current item definition.

✔ Validates the current item definition.

Browsing

If the server supports browsing, the browsing controls will be available for use in searching for selected items. Filters can be applied as needed in order to simplify the search.

Visual/Operational Changes

When the OPC Quick Client adds, activates, or deactivates an item, the appearance of the tag icon will change to one of the following:

The item is active.
The item is inactive.
The client could not add the specified item to the group.

Data Types Description

The OPC Quick Client supports the data types listed in the table below. Their default description is determined by the OPC server.

Native	Description
Boolean	Single bit
Boolean Array	Single bit array
Char	Signed 8-bit value
Char Array	Signed 8-bit value array
Byte	Unsigned 8-bit value
Byte Array	Unsigned 8-bit value array
Short	Signed 16-bit value
Short Array	Signed 16-bit value array
Word	Unsigned 16-bit value.
Word Array	Unsigned 16-bit value array
Long	Signed 32-bit value
Long Array	Signed 32-bit value array
DWord	Unsigned 32-bit value
DWord Array	Unsigned 32-bit value array
LLong	Signed 64-bit value
LLong Array	Signed 64-bit value array
QWord	Unsigned 64-bit value
QWord Array	Unsigned 64-bit value array
Float	Single precision floating-point value (32 bits)*
Float Array	Single precision floating-point value (32 bits) array*
Double	Double precision floating-point value (64 bits)*
Double Array	Double precision floating-point value (64 bits) array*
String	Zero-terminated character array
String Array	Zero-terminated string array
Date	YYYY-MM-DDThh:mm:ss.OOO

*When a floating-point value is Infinite, Indeterminate, or NaN, the following is displayed in the Value column of its Item Properties:

- 1.#INF
- -1.#INF
- -1.#IND
- 1.#QNAN
- -1.#QNAN

See Also: Options and Items.

Operations

The OPC Quick Client performs operations on an OPC server at the Server, Group, and Item level. This allows users to test the functionality of an OPC server.

Server Operations

Connect

To establish a connection to the server, click Tools | Server and then select Connect. All groups and items will be added.

Disconnect

To disconnect from the server, click Tools | Server and then select Disconnect.

Reconnect

To re-establish a connection to the server, click Tools | Server and then select Reconnect.

Get Error String

To obtain an error description for any valid error codes, click Tools | Server and then select Get Error String.
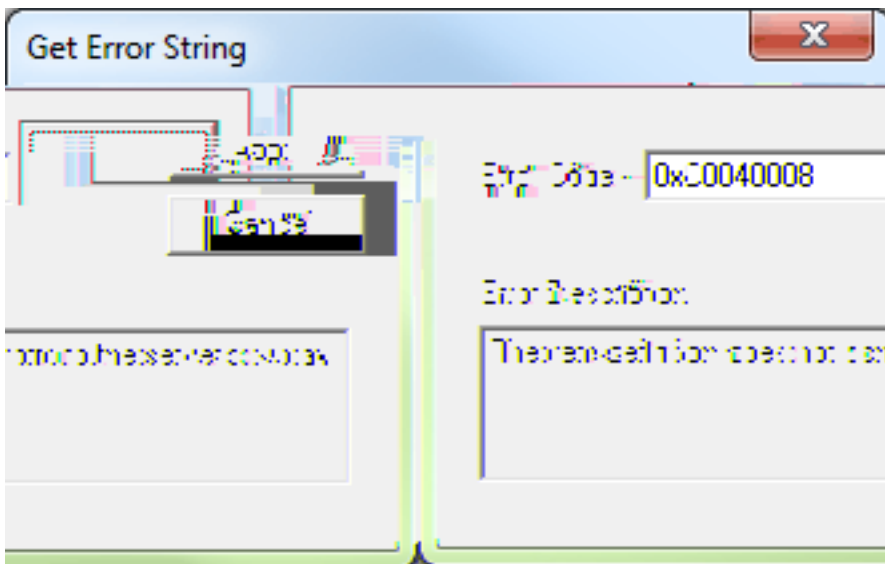
Note: To request an error description from the server, enter the error code and then press Apply. If the server can translate the error code, the description will appear. An OPC server should be able to translate any OPC error code, in addition to Win32 errors.

This operation tests the following server functionality: IOPCServer::GetErrorString ().

Enumerate Groups

To enumerate groups available to the server connection, click Tools | Server and then select Enumerate Groups.
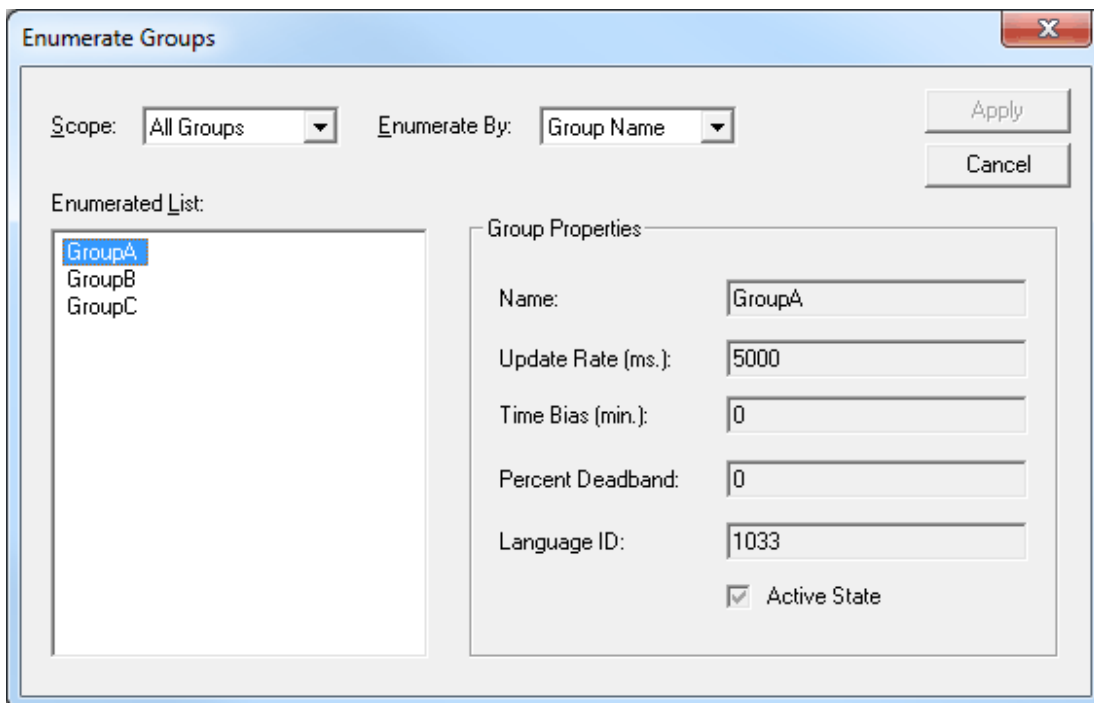
Note: The server can be requested to enumerate groups by name or by object. When enumerating by name, the server passes back an enumeration of strings each containing the name of a group. When enumerating by object, the server passes back an enumeration of group interface pointers (IUnknown), which the client can make use of by querying for other interfaces. Either way, the client uses both mechanisms to obtain the same results: a list of group names along with the properties of each group. The client can request that only private groups (groups this client added to the server connection), public groups (groups that may have been added by another client, but are made available) or all groups be enumerated.

This operation tests the following server functionality: IOPCServer::CreateGroupEnumerator ()., IOPCServer::GetGroupByName ( ), IOPCGroupStateMgt::GetState ( ).

Get Group By Name
To obtain group properties by name, click Tools | Server and then select Get Group By Name.
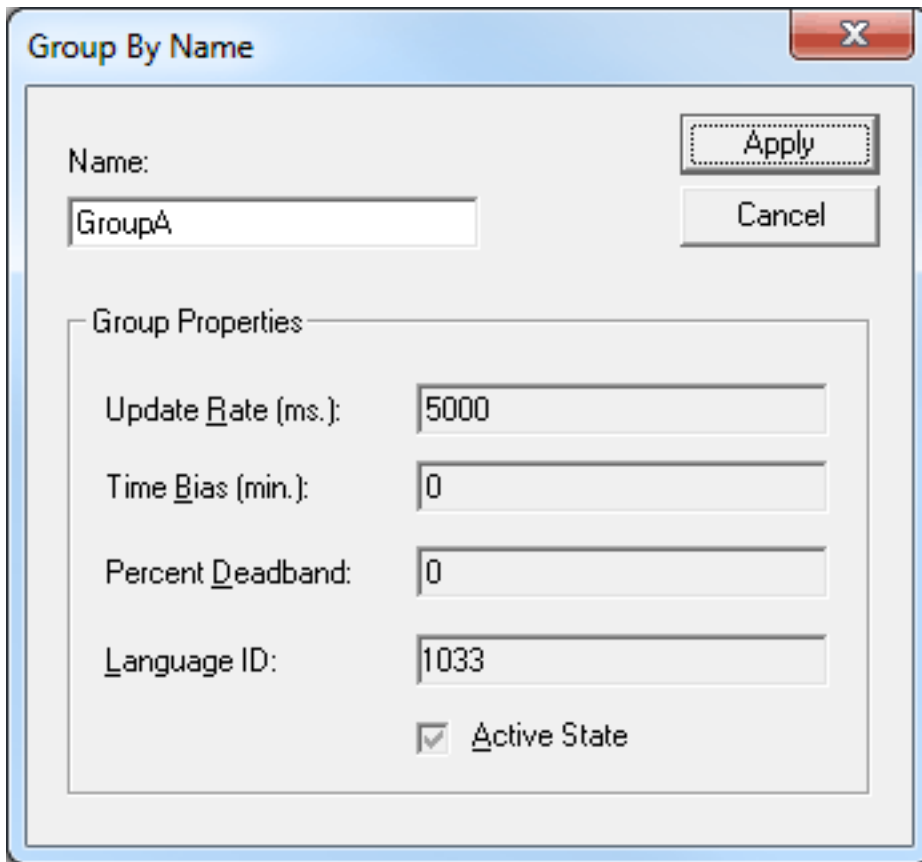
Note: To request group properties by name, enter the name of the group and then press Apply. If the server can resolve the group name, the client will proceed to obtain the properties for that group.

This operation tests the following server functionality: IOPCServer::GetGroupByName ( ), IOPCGroupStateMgt::GetState ( ).

Auto Create Tags
To automatically create tags from the connected OPC server, click Tools | Server and then select Auto Create Tags. For more information, refer to Options.

Group Operations
Clone Group
To clone the selected group, click Tools | Group and then select Clone Group. The OPC server should create a new group with a unique name and with identical properties to the group that is being cloned. If the cloned group contains items, those items should also be cloned. The state of the cloned group and items should be set to inactive.

This operation tests the following server functionality: IOPCGroupStateMgt::CloneGroup, SetState ( ), IOPCItemMgt::CreateEnumerator, SetClientHandles ( ), IEnumOPCItemAttributes::Reset, Next ( ).

Asynchronous Refresh (Cache/Device)
To perform an asynchronous refresh from cache/device on the selected items, click Tools | Group and then select Asynchronous 1.0/2.0 Cache Refresh or Asynchronous 1.0/2.0 Device Refresh. Refreshes done using 1.0 use IAdviseSink to return refresh notifications. Refreshes done using 2.0 use connection points to exchange data between client and server.

Item Operations
Set Active/Inactive
To activate/inactivate the selected item(s), click Tools | Item and then select Set Active | Inactive. Only active items should receive on data change updates.

Note: The group must be set to active.

Synchronous Read (Cache/Device)

To perform a synchronous read from the cache/device on the selected items, click Tools | Item and then select Synchronous Cache Read or Synchronous Device Read.

Asynchronous Read (Cache/Device)
To perform an asynchronous read from the cache/device on the selected items, click Tools | Item and then select Asynchronous 1.0/2.0 Cache Read or Asynchronous 1.0/2.0 Device Read. Reads done using 1.0 use IAdviseSink to return read complete notifications. Reads done using 2.0 use connection points to exchange data between the client and server.

Synchronous/Asynchronous Write
To perform a synchronous/asynchronous write on the selected items, click Tools | Item and then select Synchronous Write or Asynchronous 1.0/2.0 Write.
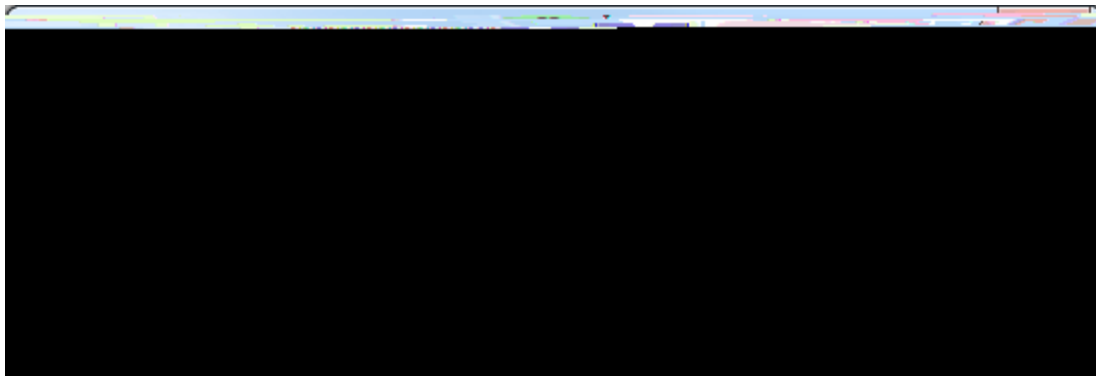
Note: Both types of writes utilize the same dialog. The dialog displays the current value at a 100 millisecond refresh rate. To send the data to the server, enter the data to write in the Write Value column and then press Apply. To apply the write and close the dialog, click OK. To abort the operation, click Cancel.

Options

The OPC Quick Client is used to configure many settings. To open the Options dialog, click Tools | Options.
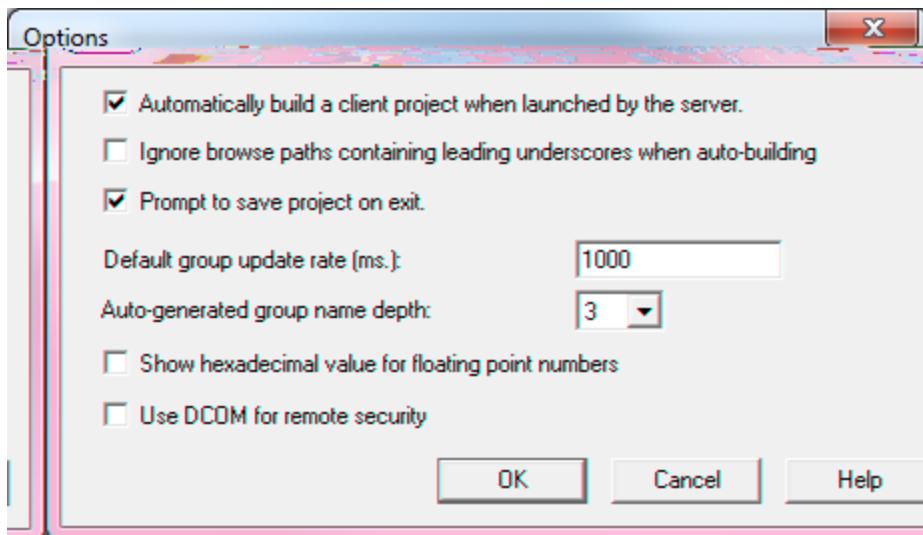
Descriptions of the parameters are as follows:

- **Automatically Build a Client Project When Launched by the Server:** When checked, the OPC Quick Client will automatically build a client project with all the tags in the OPC server when it is launched from the OPC server. The default setting is checked.
- **Ignore Browse Paths Containing Leading Underscores When Auto-Building:** When checked, items that belong to browse path segments beginning with an underscore will not be included in automatically built projects. These paths may include system or internal device tags. The default setting is unchecked (all available items will be included in the project).

- **Prompt to Save Project on Exit**: When checked, users will be prompted to save the project before exiting the application. The default setting is checked.
- **Default Group Update Rate (ms)**: This parameter specifies the default update rate for automatically generated groups. The update rate determines how often the OPC server will provide updates for changing data. The default setting is 1000 milliseconds. For more information, refer to Group.
- **Auto-Generated Group Name Depth**: This field specifies the depth level for names of automatically generated groups. The default setting is 3. For more information, refer to "Group Name Depth" below.
- **Show hexadecimal value for floating point numbers:** When enabled, this option will affect how floating point numbers are displayed and written.

    - In terms of display, this option causes Float or Double data types to display the floating point number in addition to the hexadecimal value. For example, "12.34 (OX41457OA4)".
    - In terms of writing, this option allows hexadecimal values to be written for Float and Double data types. The hexadecimal value should be in the form "Ox12345678" for Floats (8 digits) and "Ox123456789ABCDEF1" for Doubles (16 digits). Inputs that are not in a hexadecimal format will be treated as normal floating point inputs.

- **Use DCOM for remote security**: When checked, the OPC Quick Client, opens on next launch with system-level DCOM security settings. When unchecked, the OPC Quick Client opens with "wide-open" DCOM security settings. The default is unchecked.

Note: This option will also increase the Floating Point Precision to 17.

Group Name Depth

The higher the group name depth, the more sub-groups that will be created. For example, tags structured like "Channel1.Device1.Tag1" and "Channel1.Device1.Group1.Tag1" in OPC server will be added differently in OPC Quick Client depending on the Group Name depth setting.

Depth Level	Description
1	The group "Channel1" (name depth: 1) will be created. Both tags will be added to this group.
2	The group "Channel1.Device1" (name depth: 2) will be created. Both tags will be added to this group.
3	The group "Channel1.Device1" (name depth: 3. Tag names do not form a part of the group name) will be created. The tag "Channel.Device1.Tag1" will be added to this group. The group "Channel1.Device1.Group1" (name depth: 3) will be created. The tag "Channel.Device1.Group1.Tag1" will be added to this group.

Note: The maximum allowed group name depth is 3. In the example above, if there were any more sub-groups inside Group1, they would be added to the OPC Quick Client group "Channel1.Device1.Group1".

Test Mode

The OPC Quick Client Test Mode includes both functional and performance tests to run on projects. Projects must contain at least one server connection with one group that contains one item.

Functional Test

Functional tests perform set routines on the project. When enabled, the OPC Quick Client will run the first test in the selected list for the amount of time specified in the Test Interval parameter (which may be 1 to 360 minutes). Once the interval expires, the client will stop the first test and move onto the next. When the last test in the list has finished, the client will sit idle for the same interval before repeating the test suite.

Note: Users cannot change the project while the functional test is running.
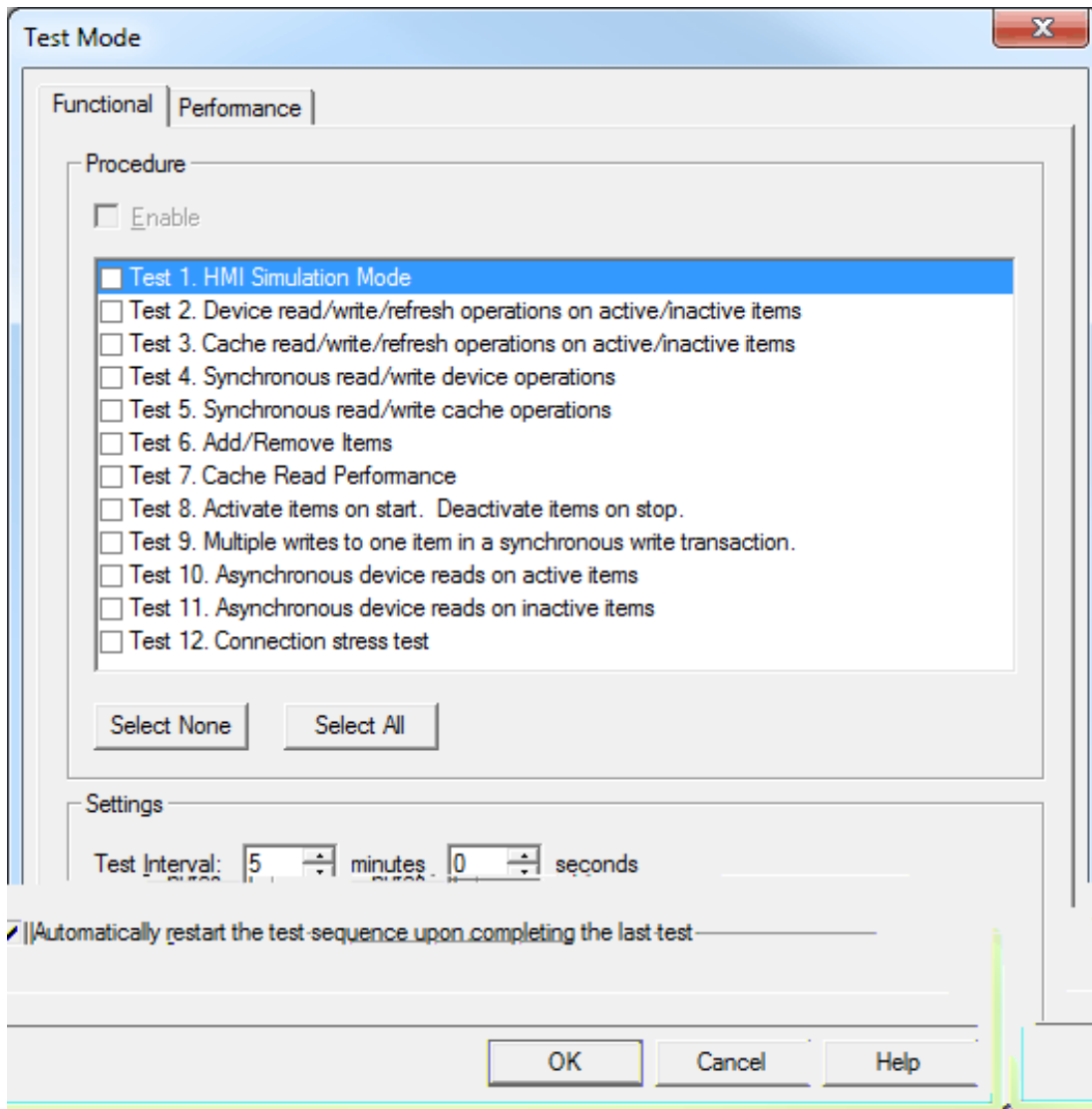
Test 1 Description

This test simulates an HMI product by treating each group as a panel. The test first sets all items to an inactive state. Then, it selects the first group in the list, sets all of the group's items active, and receives data updates for approximately 5 seconds. After the 5 second period, the active items are set to an inactive state and the test proceeds to the next group. The procedure continues cycling through all the groups and starting over when the last group is complete.

Test 2 Description

This test performs read, write, and refresh operations on active and inactive items. The test first sets all items to an inactive state. Then, it proceeds through the following sequence:

1. Select the next group in the list.

2. Perform a synchronous read from device for all items (all group items are active).

3. Perform a synchronous write to all items (all group items are active).

4. Perform a synchronous read from device for all items (all group items are inactive).

5. Perform a synchronous write to all items (all group items are inactive).

6. Perform an asynchronous read from device for all items (all group items are active).

7. Perform an asynchronous write to all items (all group items are active).

8. Perform an asynchronous read from device for all items (all group items are inactive).

9. Perform an asynchronous write to all items (all group items are inactive).

10. Perform an asynchronous refresh from device (all group items are active).

The sequence will repeat until the test interval time is reached.

Test 3 Description
This test duplicates Test 2, except all operations are done from cache instead of device.

Test 4 Description
This test performs synchronous read and write operations on active items. The test first sets all items in the project to an active state. Then, it performs a synchronous write every 2 seconds and a synchronous read from device every 5 seconds. If there is more than one group, the test performs 2 writes to 1 read, and then moves onto the next group.

Test 5 Description
This test duplicates Test 4, except that all operations are done from cache instead of device.

Test 6 Description
This test performs the adding and removal of all items in the project. The test first removes all items in the project. After a 2 second period the test re-adds all the items in the project. The test continues to transition between the two states every 2 seconds.

Test 7 Description
This test shows the performance of the server when reading items from cache. The test first activates all items in the project, which is a requirement for cache reads. Then, it proceeds to the first group in the project and performs as many synchronous reads as it can in 1 second. The number of items read in 1 second is displayed in the message log. The test continues by selecting the next group and proceeding with the same tests.

Note: If a group contains multiple items, each synchronous read request includes all the items in one call to the server.

Test 8 Description
This test activates all items in a project (including all connections and groups). It updates the display window, pauses for the test interval selected by the user, and then inactivates all items.

The sequence will complete once per test interval.

Test 9 Description
This test activates all items in a project (including all connections and groups). It performs a random number of synchronous writes (between 5 and 100) to all items in a project in the following sequence:

1. Get the first connection, first group.

2. Perform a random number of synchronous writes (between 5 and 100) to each item in the group.

3. Pause for 1 second between each item in the group.

4. Pause for 10 seconds between groups.

5. Repeat for each connection in the project.

The sequence will repeat until the test interval time is reached.

Test 10 Description

This test activates all items in a project (including all connections and groups). It performs an asynchronous read for all items in a project in the following sequence:

1. Get the first connection, first group.

2. Perform an asynchronous read for all items in the group.

3. Pause for 1 second between each item in the group.

4. Pause for 10 seconds between groups.

5. Repeat for each connection in the project.

The sequence will repeat until the test interval time is reached.

Test 11 Description
This test inactivates all items in a project (including all connections and groups). It performs an asynchronous read for all items in a project in the following sequence:

1. Get the first connection, first group.

2. Perform an asynchronous read for all items in the group.

3. Pause for 1 second between each item in the group.

4. Pause for 10 seconds between groups.

5. Repeat for each connection in the project.

The sequence will repeat until the test interval time is reached.

Test 12 Description
This test is a server connection stress test. It creates the maximum allowable number of server connections. In a separate thread for each connection, it automatically creates tags and performs random sync and async Read and Write requests to all items in all non-system groups. The test will verify that incomplete transactions are handled correctly when the client disconnects with a transaction pending. After writing to each non-system group, each thread in Test 12 will close its connection.

The sequence will end when all connections are closed or the test interval time is reached.

Performance Test
Performance tests measure the amount of time it takes to execute an operation on each item configured in the OPC Quick Client.
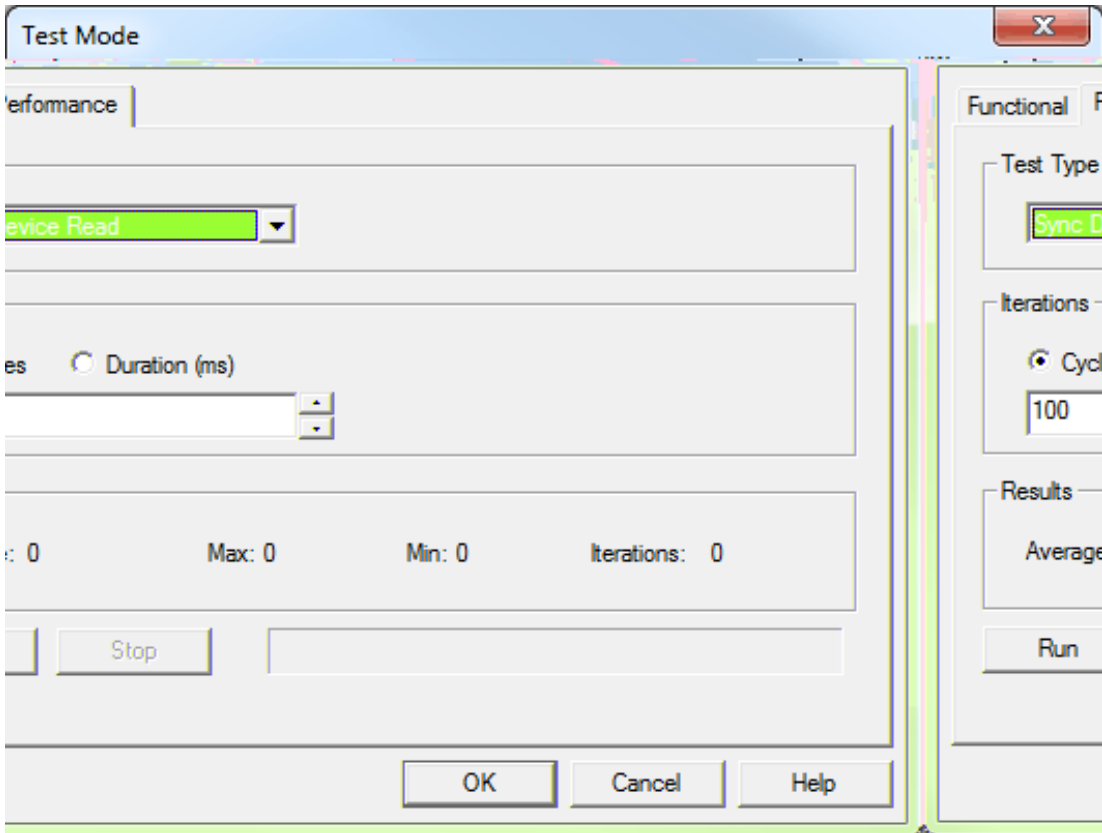
Descriptions of the parameters are as follows:

- **Test Type**: This parameter specifies the test type. Tests are named after the operation that they perform. Options include Sync Device Read, Sync Cache Read, and Sync Write (which writes a random value to each item). The default setting is Sync Device Read.
- **Iterations**: This parameter specifies the iteration. An iteration is complete when the test has performed once on each item. Options include Cycles and Duration. The default setting is 100 cycles. Descriptions of the options are as follows:

  - **Cycles**: The test will iterate for as many times as entered in the field.
  - **Duration (ms)**: The test will run for as many milliseconds as entered in the field.
- **Results**: This parameter displays the results of the performance test. The following fields will be displayed:

  - **Average**: This specifies the median time per iteration in milliseconds.
  - **Max**: This specifies the maximum time for an iteration in milliseconds.
  - **Min**: This specifies the minimum time for an iteration in milliseconds.
  - **Iterations**: This specifies the number of iterations.

Running a Performance Test
For more information, refer to the instructions below.

1. To start, use the drop-down menu to specify the test type.

2. Next, select the type of iteration. Then, specify the amount.

3. Click **Run**.

Note: Users can stop a test at any time by clicking **Stop**. A progress bar will display how far the test has progressed.

License Agreement

License and Warranty

The license and warranty refer to the OPC Quick Client provided with the OPC Server. It does not refer to any other software provided by the supplier.

The OPC Quick Client Software accompanying this license agreement is the property of supplier and is protected by copyright law. A user's rights and obligations in its use are described as follows:

1. Users may not modify or attempt to reverse engineer the software, or make any attempt to change or examine the source code of the Software.

2. Users may not give or distribute copies of the Software or written materials associated with the Software to others.

3. Users may not sub-license, sell or lease the Software to any person or business.

Warranty

The Supplier does not guarantee that the Software will satisfy application needs or be error free.

Disclaimer of Damages

Even if the Software does not satisfy the customer's needs or is viewed as failing to perform, in no event will the Supplier be liable to users for any direct or indirect damages, including lost profits or lost data. The Supplier's liability shall never exceed the purchase price for the Software.

Index

M

O

P

R

S

T

U