

OPC Toolbox Release Notes

The “OPC Toolbox 2.0 Release Notes” on page 1-1 describe enhancements and bug fixes included in the latest version of the OPC Toolbox. The following topics are discussed in these Release Notes:

- “New Features and Enhancements” on page 1-2
- “OPC Toolbox 2.0 Installation Notes” on page 1-3
- “Upgrading from a Previous Release” on page 1-4
- “Known Software Problems” on page 1-5

The OPC Toolbox Release Notes also provide information about recent versions of the product, in case you are upgrading from an earlier version:

- “OPC Toolbox 1.1.2 Release Notes” on page 2-1
- “OPC Toolbox 1.1.1 Release Notes” on page 3-1
- “OPC Toolbox 1.1 Release Notes” on page 4-1
- “OPC Toolbox 1.0 Release Notes” on page 5-1

Printing the Release Notes

If you would like to print the Release Notes, you can link to a PDF version.

OPC Toolbox 2.0 Release Notes

1

New Features and Enhancements	1-2
Support for Simulink: OPC Blockset Library	1-2
New Item Property: QualityID	1-2
OPC Toolbox 2.0 Installation Notes	1-3
OPC Core Components	1-3
Upgrading from a Previous Release	1-4
Known Software Problems	1-5
Calling opreset in a Callback	1-5
OPC Blockset Library Dialogs Require Java	1-5
Cannot Compile OPC Blockset Library Blocks with Real-Time Workshop	1-5

OPC Toolbox 1.1.2 Release Notes

2

New Features and Enhancements	2-2
QualityID Property	2-2
Quality String Limit Status	2-2
Default Quality Value	2-2

OPC Toolbox 1.1.1 Release Notes

3

New Features and Enhancements	3-2
Timeout Property Setting	3-2
Vector Displays of Item and Group Objects	3-2
OPC Tool Enhancement	3-2
Referencing Object Properties	3-2
Major Bug Fixes	3-3
OPC Tool Memory Usage	3-3
Hyperthreading	3-3
Upgrading from a Previous Release	3-4
Loading OPC Toolbox Objects	3-4

OPC Toolbox 1.1 Release Notes

4

New Features and Enhancements	4-2
Improved Support for Large Name Spaces	4-2
Enhanced Add Item Dialog in OPCTool	4-2
Support for Public Groups	4-3
Major Bug Fixes	4-4
Upgrading from a Previous Release	4-5

OPC Toolbox 1.0 Release Notes

5

Introduction	5-2
---------------------------	-----

OPC Toolbox 2.0 Release Notes

These release notes introduce the OPC Toolbox 2.0 in the following sections:

New Features and Enhancements (p. 1-2)	Provides a list of the new features and major enhancements in this release.
OPC Toolbox 2.0 Installation Notes (p. 1-3)	Describes installation notes related to the OPC Toolbox.
Upgrading from a Previous Release (p. 1-4)	Describes issues to be aware of when upgrading from earlier versions of the toolbox.
Known Software Problems (p. 1-5)	Describes known problems with workarounds.

For more information on the OPC Toolbox, see the user documentation for the OPC Toolbox.

New Features and Enhancements

The OPC Toolbox 2.0 includes the following new features and enhancements, added since OPC Toolbox 1.1.2.

Support for Simulink: OPC Blockset Library

This release of the OPC Toolbox includes support for communicating with OPC servers from Simulink[®]. The OPC Blockset library includes blocks for reading data from an OPC server, writing data to an OPC server, and running a Simulink simulation in pseudo real time (by slowing the simulation to run at system time). For more information on using the OPC Toolbox with Simulink, see “Using the OPC Blockset Library” in the User’s Guide.

New Item Property: QualityID

The QualityID property has been added to daitem objects. The QualityID, expressed as a 16-bit unsigned integer, represents the quality of the data item when last read. To work with the QualityID property, you use the function `opcqparts` to convert the QualityID property into vendor, major, substatus, and limit status information; and the function `opcqstr` to convert the QualityID property into a string. For more information, see the QualityID property reference page.

OPC Toolbox 2.0 Installation Notes

OPC Core Components

The OPC Toolbox uses the OPC Data Access standard, defined by the OPC Foundation, to connect to OPC servers. As part of that standard, the OPC Foundation produces a standard set of interface files called the OPC Core Components. The Core Components are used by all OPC clients (such as the OPC Toolbox) installed on a computer.

To use the OPC Toolbox, you must install the OPC Foundation Core Components. Instructions on how to perform this task are provided in the section “Preparing to Use the OPC Toolbox” in the OPC Toolbox User’s Guide.

OPC Core Components on Windows XP

The OPC Core Components that ship with the OPC Toolbox have been updated to work with Windows XP Service Pack 2. Existing users of the OPC Toolbox do not need to update their OPC Core Components, but if you experience problems with Windows XP Service Pack 2, install the OPC Core Components from OPC Toolbox 2.0 by running the following commands in MATLAB®:

```
opcregister remove  
opcregister
```

Upgrading from a Previous Release

There are no issues when upgrading to Version 2.0 from Version 1.1.1 or 1.1.2.

If you are upgrading from Version 1.1, see “Upgrading from a Previous Release” on page 3-4.

Known Software Problems

Calling `opcreset` in a Callback

Calling `opcreset` in a callback causes a segmentation fault and crashes MATLAB. As a workaround, you should delete objects using the `delete` function. Note that deleting a client object deletes all groups and items associated with that client object.

OPC Blockset Library Dialogs Require Java

The OPC Blockset library dialogs require Java. If you use the OPC Blockset library in a MATLAB session that was started with the `-nojvm` option, MATLAB crashes when you attempt to open an OPC Read, OPC Write, or OPC Client Manager dialog.

You can start MATLAB with the `-nojvm` flag, and run the simulation without modifying any of the parameters of an OPC block. MATLAB with no JVM will not crash if you open the model and run the simulation without opening any OPC block dialogs, or if you run the simulation from the command line.

Cannot Compile OPC Blockset Library Blocks with Real-Time Workshop

You cannot use the OPC Blockset library with the Real-Time Workshop to generate a standalone executable, or to create generic C code for embedding in real-time targets such as the xPC Target or the Real-Time Windows Target.

You can use the Simulink Accelerator to run models containing OPC blocks in accelerated mode to extract performance gains from your Simulink model containing OPC blocks.

OPC Toolbox 1.1.2 Release Notes

These release notes introduce the OPC Toolbox 1.1.2 in the following sections:

New Features and Enhancements
(p. 2-2)

Provides a list of the new features and major
enhancements in this release.

For more information on the OPC Toolbox, see the user documentation for the OPC Toolbox.

New Features and Enhancements

The OPC Toolbox 1.1.2 includes the following new features and enhancements, added since OPC Toolbox 1.1.1.

QualityID Property

The QualityID property has been added to the daitem object. QualityID represents the quality of the item's data value as an integer. The opcqparts and opcqstr functions are provided to enable you to work with the QualityID property.

Quality String Limit Status

A limit status has been added to quality strings. The limit status indicates whether the data value has been limited in any way. For more information, see the reference page for the Quality property.

Default Quality Value

The default value for Quality has changed from an empty string to 'Bad: Out of Service' to reflect the true status of that data value.

OPC Toolbox 1.1.1 Release Notes

These release notes introduce the OPC Toolbox 1.1.1 in the following sections:

New Features and Enhancements (p. 3-2)	Provides a list of the new features and major enhancements in this release.
Major Bug Fixes (p. 3-3)	Describes major bug fixes provided by this release.
Upgrading from a Previous Release (p. 3-4)	Describes issues to be aware of when upgrading from Version 1.0.

For more information on the OPC Toolbox, see the user documentation for the OPC Toolbox.

New Features and Enhancements

The OPC Toolbox 1.1.1 includes the following new features and enhancements, added since OPC Toolbox 1.1.

Timeout Property Setting

The `Timeout` property can now be set to `Inf`. This disables any timeout handling in the OPC Toolbox. With a `Timeout` value of `Inf`, the toolbox will wait forever for a task to complete, or until the user breaks out of the wait with `^C` (**Ctrl+C**).

Vector Displays of Item and Group Objects

Displaying vectors of item and group objects has been enhanced. When you display a vector of group and/or item objects, additional information about each object in the group or item vector is displayed in the command window.

OPC Tool Enhancement

The **Add To Session** behavior of OPC Tool has changed. When you use the **Add to Session** option of the **File** menu, any clients that exist in the file and in the current session are merged using the following rules:

- Groups with different names are created inside the client that exists in the current session.
- Groups with the same name as a group in the current session are created with a new group name. The new group name has the text "`_merged`" appended to the group name. If that group name already exists, a unique number is added to the group name.
- Existing groups are *not* merged with groups that are loaded using the **Add to Session** option.

Referencing Object Properties

You can now reference elements of OPC Toolbox object properties. For example, the command

```
clientName = grp.Parent.Name
```

is now legal.

Major Bug Fixes

The OPC Toolbox 1.1.1 includes several bug fixes made since Version 1.1. This section describes the particularly important Version 1.1.1 bug fixes.

OPC Tool Memory Usage

OPC Tool no longer consumes large amounts of memory when being used for a long time.

Hyperthreading

Hyperthreading machines no longer crash the OPC Toolbox.

Upgrading from a Previous Release

The following are some changes to be aware of when upgrading to Version 1.1.1.

If you are upgrading from Version 1.0, see “Upgrading from a Previous Release” on page 4-5.

Loading OPC Toolbox Objects

In previous releases, if you loaded an OPC Toolbox object twice in the same session, the second load command would overwrite the properties of the objects that existed in the file, but would leave added children of the client object unchanged. No warning would be issued.

The OPC Toolbox now restores the entire client hierarchy to the state of the client that is loaded. When loading an object that exists in memory, the load command now issues a warning and takes the following actions:

- Deletes those groups or items that exist in memory but not in the file being loaded
- Creates those groups or items that exist in the file being loaded but not in memory
- Sets those properties of any objects in the file that differ from those in memory, to the values stored in the file

OPC Toolbox 1.1 Release Notes

These release notes introduce the OPC Toolbox 1.1 in the following sections:

New Features and Enhancements (p. 4-2)	Provides a list of the new features and major enhancements in this release.
Major Bug Fixes (p. 4-4)	Describes major bug fixes provided by this release.
Upgrading from a Previous Release (p. 4-5)	Describes issues to be aware of when upgrading from Version 1.0.

For more information on the OPC Toolbox, see the user documentation for the OPC Toolbox.

New Features and Enhancements

The OPC Toolbox 1.1 includes the following new features and enhancements, added since OPC Toolbox 1.0.

Improved Support for Large Name Spaces

This release provides improved support for servers with large name spaces. Previously, the name space on a server was retrieved in one step, and this would take inordinately long for a very large server. This release allows you to retrieve the name space incrementally, using *browse filters* for the `getnamespace` function. The browse filters allow you to specify a starting point for name space retrieval, and a browse depth, which controls how far “down” the tree you will be retrieving the name space.

This change has resulted in the following changes to behavior from previous versions:

- The `getnamespace` function now returns a hierarchical name space as the default. Previously, the name space was presented as a flat name space, even if the server stored the name space hierarchically. To get the same behavior as in previous versions, you can use the following code:

```
flatnamespace(getnamespace(da))
```
- The `getnamespace` function no longer accepts the 'hierarchical' flag.
- The OPCTool Graphical User Interface now loads the name space incrementally. This allows OPCTool to be used with servers that have large name spaces.
- The **Add Item** dialog for OPCTool has been reimplemented to allow for incremental loading of the server name space. Consequently, the **Available Server Items** list has been replaced by a tree view.

Enhanced Add Item Dialog in OPCTool

The **Add Item** dialog in OPCTool includes the following enhancements:

- You can add multiple items to a group more easily. You can now use the **All below** button to add all items below a particular node to the **Selected server items** list.

- The Available server items list is now a tree view. If an item has already been added to the group, or is selected for addition, the item will appear greyed out in the tree view. This allows you to see, in the context of the name space, which items have been added to the group already.
- If you know the Item ID of the item you wish to add to a group, you can now type that ItemID in the **Enter ItemID(s)** field.

Support for Public Groups

The OPC Data Access Specification provides functionality to allow multiple clients to share group configuration, through the use of *public groups*. One client may configure a group with a number of items, and share that configuration with other clients by making that group public.

This release adds support for public groups to the OPC Toolbox, allowing you to create public groups, connect to existing public groups on a server, and remove public groups from the server.

For additional information on public groups, see “Working with Public Groups” in the “Using OPC Toolbox Objects” chapter of the User’s Guide.

Major Bug Fixes

The OPC Toolbox 1.1 includes several bug fixes made since Version 1.0. This section describes the particularly important Version 1.1 bug fixes.

- `opcreset` no longer removes objects from the OPCTool GUI. An error is generated if you attempt to call `opcreset` while OPCTool is running. You must close OPCTool before you call `opcreset`.
- The `LanguageID` property of a `dagroup` object has been removed. Previous versions of the OPC Toolbox provided this property, but did not use the property at any time.

Upgrading from a Previous Release

The following are some changes from previous releases.

- Substantial changes to the `getnamespace` function are noted in “Improved Support for Large Name Spaces” on page 4-2.
- The `LanguageID` property has been removed. This may cause errors in any code that tries to access this property. Removal of this property has no effect on toolbox internal code behavior.
- The OPC Toolbox no longer supports subscribing into OPC Toolbox properties. Previously, using dot-notation referencing allowed you to subscript into properties. This functionality has been removed, as the subscribing operation did not work correctly for all cases. As a workaround, you must first get the property, then use subscribing on that property. For example, instead of using

```
grpName = da.Group.Name;
```

you must now use

```
grp = da.Group;  
grpName = grp.Name;
```


OPC Toolbox 1.0 Release Notes

These release notes introduce the OPC Toolbox 1.0 in the following sections:

Introduction (p. 5-2)

Provides a brief overview of the OPC Toolbox.

For more information on the OPC Toolbox, see the user documentation for the OPC Toolbox.

Introduction

This section describes the capabilities of the OPC Toolbox 1.0.

The OPC Toolbox is a collection of M-file functions and MEX-file dynamic link libraries (DLLs) built on the MATLAB technical computing environment. The OPC Toolbox allows you to connect to an OPC server directly from MATLAB and gather plant information and log data, so that you can perform analysis in MATLAB and write values back to the OPC server from MATLAB. The OPC Toolbox also allows you to log data to disk and memory, so that you can get a history or time series of data to analyze and process in MATLAB.

The toolbox provides you with these main features:

- A framework for interacting with one or more OPC servers directly from MATLAB.
- Compliance with the OPC Foundation Data Access Standard, version 2.05a.
- Event-driven interaction with OPC servers.
- An object-oriented hierarchical structure for managing connections to OPC servers, creating and interacting with groups of OPC server items (also known as tags), and communicating with servers in a flexible and efficient way.
- The ability to log data from OPC Data Access servers without the need for Historical Data Access support on the OPC Server.
- An intuitive graphical user interface (GUI) for browsing OPC servers, creating OPC Toolbox objects, and initiating logging tasks.

For more information on OPC, and particularly the capabilities of the OPC Toolbox 1.0, consult the documentation on the OPC Toolbox.