

## Tech Note 910

# Using the MatrikonOPC Sniffer to Trace OPC Communication

---

All Tech Notes, Tech Alerts and KBCD documents and software are provided "as is" without warranty of any kind. See the [Terms of Use](#) for more information.

Topic#: 002722

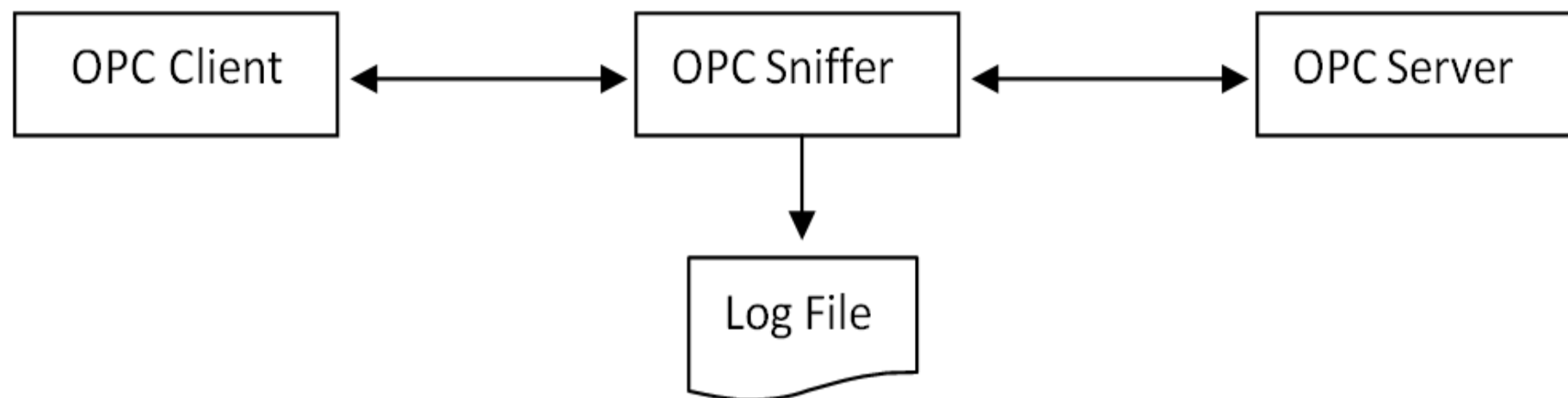
Created: March 2013

## Introduction

This *Tech Note* shows how to use the MatrikonOPC Sniffer to trace OPC communication. It provides information on some of the trace messages that the MatrikonOPC Sniffer shows. Investigating these messages could help in troubleshooting the OPC communication issues.

MatrikonOPC Sniffer can be downloaded from [www.matrikonopc.com](http://www.matrikonopc.com).

The MatrikonOPC Sniffer sits between the OPC Client and the OPC Server. OPC communication is captured by the OPC Sniffer and written to a log file. The log file can then be analyzed at a later time.



This *Tech Note* demonstrates tracing OPC communication between an OPC Server and FSGateway. FSGateway, OPC Sniffer and the OPC Server are on a single node. For this *Tech Note* we will concentrate on the following common OPC methods:

- **SetClientName:** OPC Client sends its name to the OPC Server.
- **AddGroup:** OPC Client adds a group to the OPC Server.
- **AddItems:** OPC Client adds items to a group.

**OnDataChange:** OPC Server sends exception based data changes to the OPC Client.

- **Write:** OPC Client writes values to one or more items.
- **OnWriteComplete:** OPC Server notified the OPC Client on completion of a write.
- **RemoveItems:** OPC Client removes items from a group.
- **SetActiveState:** OPC Client sets items to active or inactive.

## Application Versions

- MatrikonOPC Sniffer 1.2.0.0
- Wonderware FS Gateway 3.0

## Installing the MatrikonOPC Sniffer

[Download the MatrikonOPC Sniffer](#) (Registration required).

### To install the MatrikonOPC Sniffer

1. Double-click the installation file then click **Next**.
2. Accept the remaining defaults and click **Finish** to complete the installation.

## Configuring the MatrikonOPC Sniffer

1. Under FSGateway, make sure you have selected the name of the MatrikonOPC Sniffer for the **Server Name** field of the OPC Object.
2. Uncheck **Activate Server Out of Proc**.

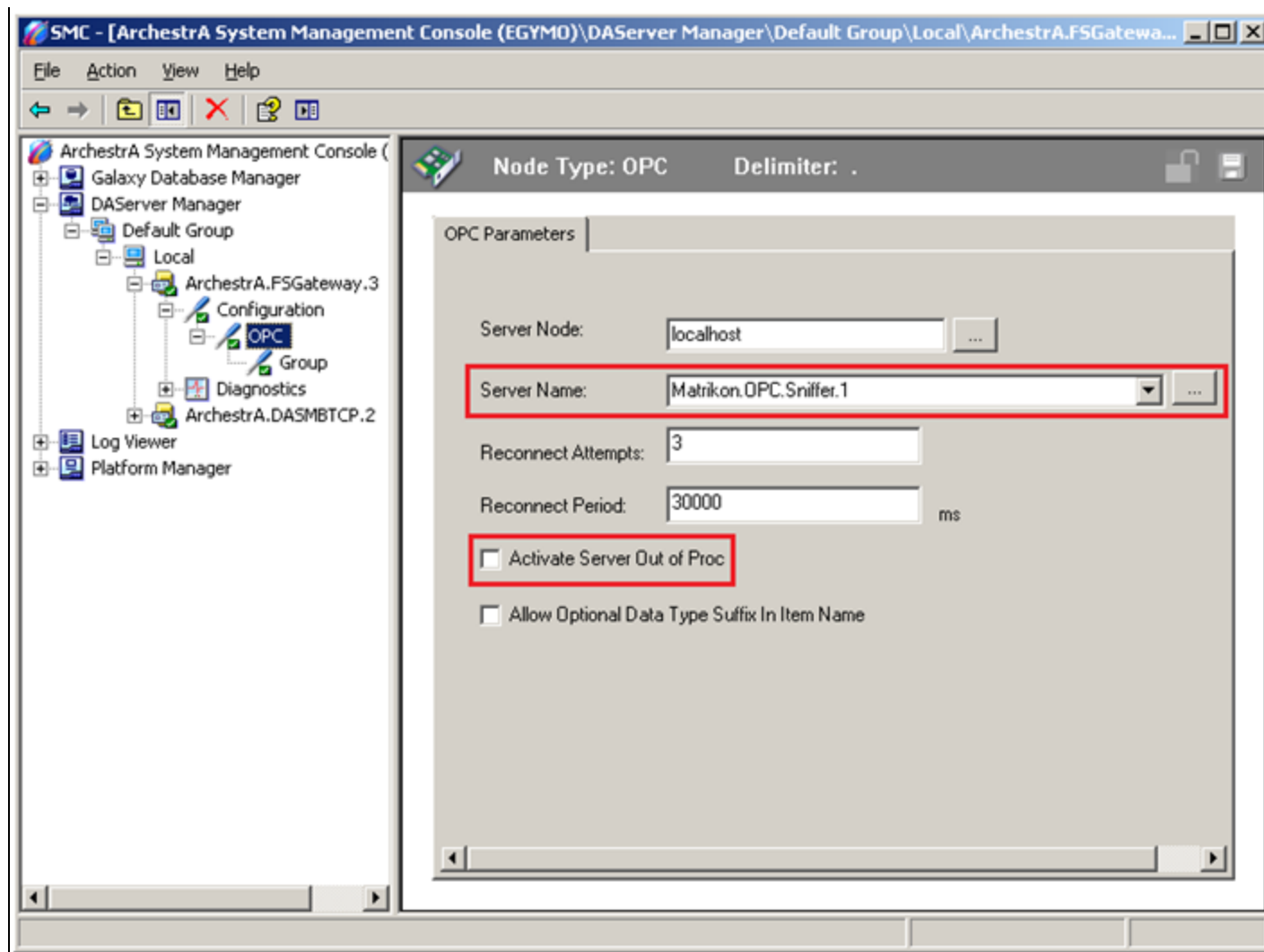


FIGURE 1: CONFIGURE OPC OBJECT

3. When InTouch WindowViewer (for example) is started with some items configured to advise data from FSGateway, FSGateway starts the MatrikonOPC Sniffer.
4. Select the target OPC Server object then click **Continue**.

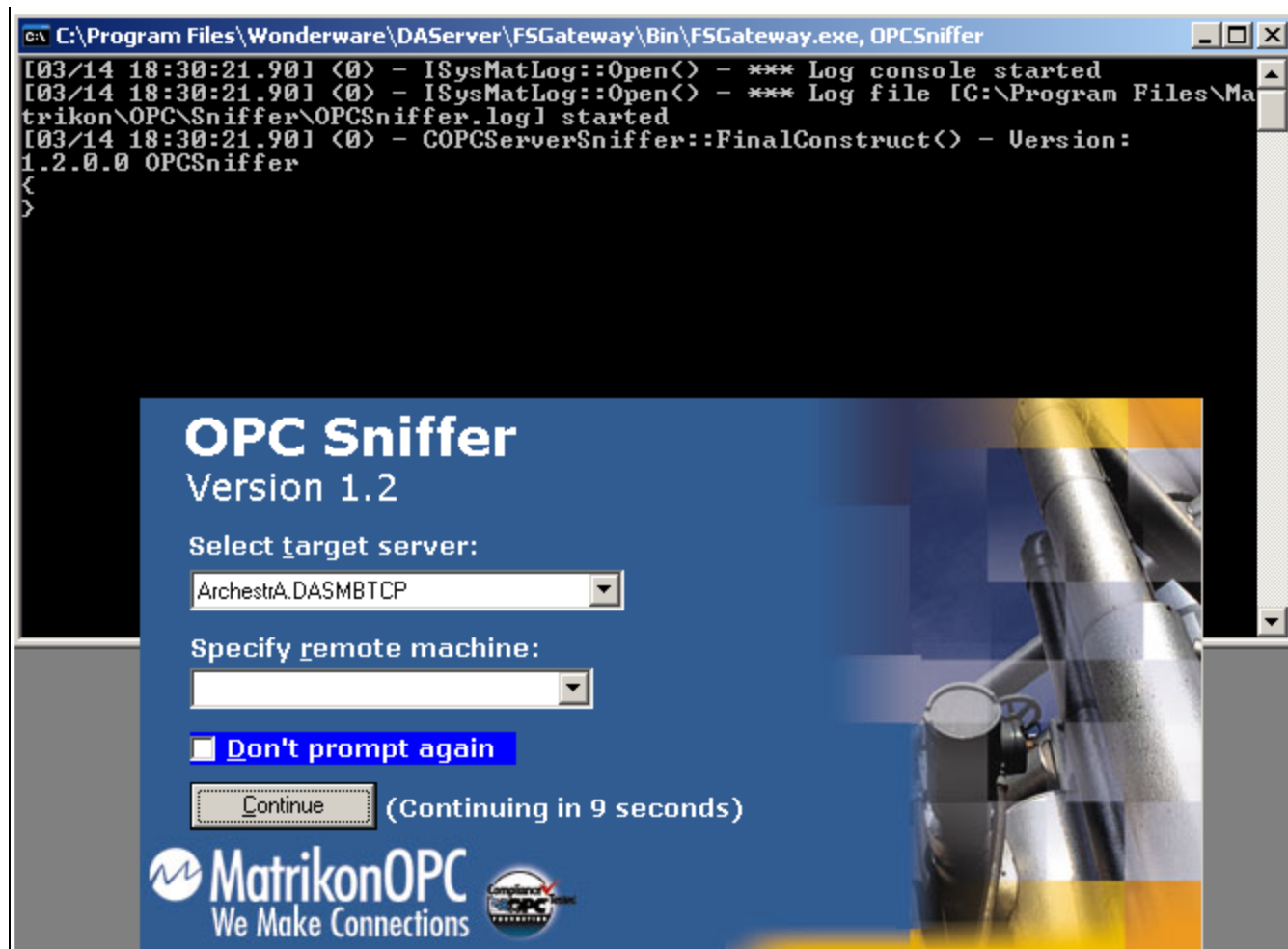
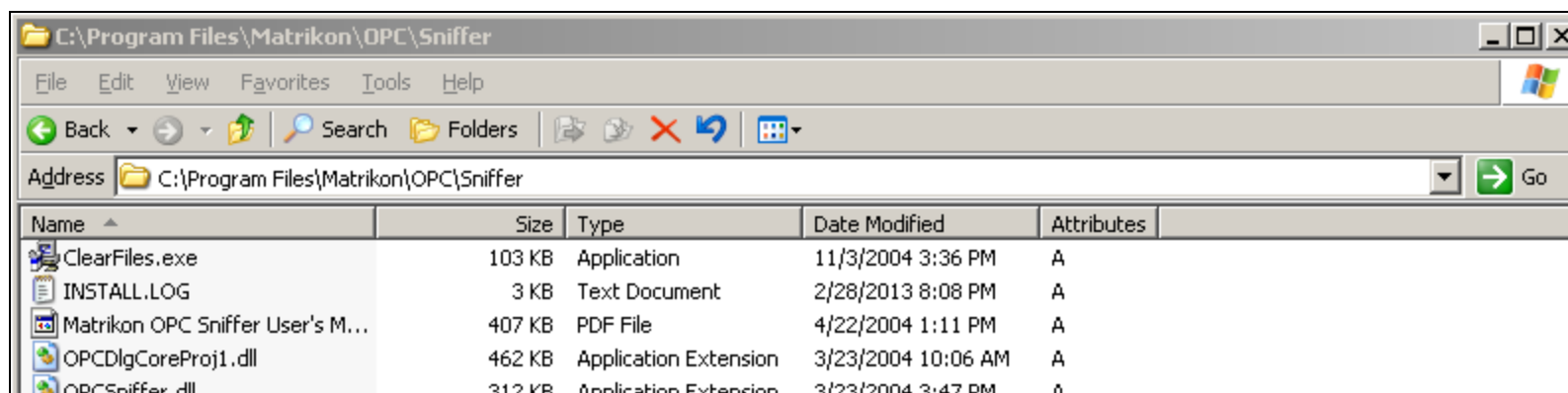


FIGURE 2: OPEN MATRIKONOPC SNIFFER

- After you configure the MatrikonOPC Sniffer to connect to the OPC server, the **OPCSniffer.log** file is created at **C:\Program Files\Matrikon\OPC\Sniffer**.



```

[03/04 15:11:20.50] (0) - ISysMatLog::open() - *** Log file [C:\Program Files\Matrikon\OPC\Sniff
[03/04 15:11:20.50] (0) - COPCServerSniffer::FinalConstruct() - Version:
1.2.0.0 OPCSniffer
{
[03/04 15:11:27.78] (7) - COPCServerSniffer::FinalConstruct() - Function returning [hr]=[0x0000
[03/04 15:11:27.78] (6) - COPCServerSniffer::_InternalQueryInterface() - iid={{39C13A4D-011E-11
[03/04 15:11:27.78] (7) - COPCServerSniffer::_InternalQueryInterface() - Function returning [hr
[03/04 15:11:27.78] (6) - COPCServerSniffer::_InternalQueryInterface() - iid={{39C13A4D-011E-11
[03/04 15:11:27.78] (7) - COPCServerSniffer::_InternalQueryInterface() - Function returning [hr
[03/04 15:11:27.78] (6) - COPCServerSniffer::_InternalQueryInterface() - iid={{F31DFDE2-07B6-11
[03/04 15:11:27.78] (7) - COPCServerSniffer::_InternalQueryInterface() - Function returning [hr
[03/04 15:11:27.78] (6) - COPCServerSniffer::SetClientName() - szName=[Wonderware FSGateway]
[03/04 15:11:27.78] (0) Wonderware FSGateway - ISysMatLog::Prefix - Prefix is now [Wonderware F
[03/04 15:11:27.78] (7) Wonderware FSGateway - COPCServerSniffer::SetClientName() - Function re
[03/04 15:11:27.78] (6) Wonderware FSGateway - COPCServerSniffer::_InternalQueryInterface() - i
[03/04 15:11:27.78] (7) Wonderware FSGateway - COPCServerSniffer::_InternalQueryInterface() - F
[03/04 15:11:27.78] (6) Wonderware FSGateway - COPCServerSniffer::IConnectionPointContainer::Fi
[03/04 15:11:27.78] (7) Wonderware FSGateway - CServerShutdownObjectsSniffer::Init() - Function i
[03/04 15:11:27.78] (6) Wonderware FSGateway - CServerShutdownObjectsSniffer::_InternalQueryInter
[03/04 15:11:27.78] (7) Wonderware FSGateway - CServerShutdownObjectsSniffer::_InternalQueryInter
[03/04 15:11:27.79] (7) Wonderware FSGateway - COPCServerSniffer::IConnectionPointContainer::Fi
[03/04 15:11:27.79] (7) Wonderware FSGateway - CClientShutdownObjectsSniffer::Init() - Function i
[03/04 15:11:27.79] (6) Wonderware FSGateway - CClientShutdownObjectsSniffer::_InternalQueryInter
[03/04 15:11:27.79] (7) Wonderware FSGateway - CClientShutdownObjectsSniffer::_InternalQueryInter
[03/04 15:11:27.79] (6) Wonderware FSGateway - CClientShutdownObjectsSniffer::_InternalQueryInter
[03/04 15:11:27.79] (4) Wonderware FSGateway - CClientShutdownObjectsSniffer::_InternalQueryInter
[03/04 15:11:27.79] (4) Wonderware FSGateway - CClientShutdownObjectsSniffer::_InternalQueryInter
[03/04 15:11:27.79] (6) Wonderware FSGateway - CClientShutdownObjectsSniffer::_InternalQueryInter
[03/04 15:11:27.79] (6) Wonderware FSGateway - - 0x80004002 [No such interface supported]
[03/04 15:11:27.79] (6) Wonderware FSGateway - CClientShutdownObjectsSniffer::_InternalQueryInter
[03/04 15:11:27.79] (4) Wonderware FSGateway - CClientShutdownObjectsSniffer::_InternalQueryInter
[03/04 15:11:27.79] (4) Wonderware FSGateway - CClientShutdownObjectsSniffer::_InternalQueryInter
[03/04 15:11:27.79] (6) Wonderware FSGateway - CClientShutdownObjectsSniffer::_InternalQueryInter
[03/04 15:11:27.79] (6) Wonderware FSGateway - - 0x80004002 [No such interface supported]
[03/04 15:11:27.79] (6) Wonderware FSGateway - CClientShutdownObjectsSniffer::_InternalQueryInter
[03/04 15:11:27.79] (4) Wonderware FSGateway - CClientShutdownObjectsSniffer::_InternalQueryInter
[03/04 15:11:27.79] (4) Wonderware FSGateway - CClientShutdownObjectsSniffer::_InternalQueryInter
[03/04 15:11:27.79] (6) Wonderware FSGateway - CClientShutdownObjectsSniffer::_InternalQueryInter
[03/04 15:11:27.79] (6) Wonderware FSGateway - - 0x80004002 [No such interface supported]
[03/04 15:11:27.79] (6) Wonderware FSGateway - CClientShutdownObjectsSniffer::_InternalQueryInter

```

FIGURE 3: SNIFFER LOG FILE IN NOTEPAD

## Using the MatrikonOPC Sniffer

This section demonstrates some common trace messages that the MatrikonOPC Sniffer provides when you advise an item, write a value to an item, and Activate/Deactivate an item. InTouch WindowViewer is used for this example.

## Advise an Item

### FSGateway sends its name to the OPC Server after a connection is established (SetClientName)

```
[03/13 13:00:31.70] (6) - COPCServerSniffer::SetClientName() - szName=[Wonderware FSGateway]
[03/13 13:00:31.70] (7) Wonderware FSGateway - COPCServerSniffer::SetClientName() - Function returning [hr]=[0x00000000] at
line [676] in [.\OPCServerSniffer.cpp]
```

- **szName:** FSGateway sends its name to the OPC Server.
- **[hr]:** HRESULT from OPC Server returns OK.

### FSGateway Adds a Group to the OPC Server (AddGroup)

```
[03/13 13:00:31.76] (6) Wonderware FSGateway - COPCServerSniffer::AddGroup() - szName=[opc.group], bActive=1,
dwRequestedUpdateRate=1000, hClientGroup=0x01000000, dwLCID=0x00000000
[03/13 13:00:31.76] (6) Wonderware FSGateway - COPCServerSniffer::AddGroup() - pTimeBias=NULL
[03/13 13:00:31.76] (6) Wonderware FSGateway - COPCServerSniffer::AddGroup() - pPercentDeadband=NULL
[03/13 13:00:31.76] (6) Wonderware FSGateway - COPCServerSniffer::AddGroup() - riid=[{ 39C13A54-011E-11D0-9675-
0020AFD8ADB3-IOPCItemMgt}]
[03/13 13:00:31.81] (6) Wonderware FSGateway - COPCServerSniffer::AddGroup() - *phServerGroup=0x01000003,
*pRevisedUpdateRate=1000
[03/13 13:00:31.82] (7) Wonderware FSGateway - COPCServerSniffer::AddGroup() - Function returning [hr]=[0x00000000] at line
[294] in [.\OPCServerSniffer.cpp].
```

- **szName:** The Group name is the FSGateway OPC object name and the OPC Group object name separated by a period.
- **bActive:** FSGateway creates the group as active. (1 = Active, 0=Inactive).
- **dwRequestedUpdateRate:** FSGateway specifies the rate at which data changes may be sent from the OPC Server. This value is in milliseconds.
- **hClientGroup:** FSGateway provides the client handle for this group. The OPC Server uses this handle to refer to the client group.
- **phServerGroup:** The OPC Server generated handle to the group. The client (FSGateway) will use this handle for functions such as adding items to the group.
- **pRevisedUpdateRate:** The OPC server returns the value it will actually use for the update rate. This may be different than the requested update rate.
- **[hr]:** HRESULT from OPC Server returns OK.

### FSGateway Subscribes to Items (AddItems)

FSGateway subscribes to the item by adding the item to the group that was just created.

```
[03/13 13:00:31.87] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::AddItems() - dwCount=1
```

```
[03/13 13:00:31.87] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::AddItems() - [ 0] - szAccessPath=[],
szItemID=[port.plc.400001], bActive=1, hClient=0x01000000, dwBlobSize=0, vtRequestedDataType=0x0000
[03/13 13:00:31.87] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::AddItems() - [ 0] - ppErrors=0x00000000
[03/13 13:00:31.87] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::AddItems() - hServer=0x01000004,
vtCanonicalDataType=0x0012, dwAccessRights=0x00000003, dwBlobSize=0
[03/13 13:00:31.87] (7) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::AddItems() - Function returning


---


[hr]=[0x00000000] at line [371] in [.\OPCGroupSniffer.cpp].
```

- **dwCount**: The number of items to be added.
- **szAccessPath**: There is no access path defined in this example.
- **szItemID**: The name of the item in the OPC Server that FSGateway wants to subscribe to.
- **bActive**: FSGateway adds the item as active. (1 = Active, 0 = Inactive).
- **hClient**: FSGateway sends its item handle to the OPC Server.
- **vtRequestedDataType**: FSGateway requests the data type as not specified (VT\_EMPTY). Passing VT\_EMPTY means FSGateway will accept the OPC Server's data type that will be returned. See below for information on specifying the requested data type.
- **hServer**: OPC Server sends its item handle to FSGateway.
- **vtCanonicalDataType**: The native data type within the OPC Server for this item. (VT\_UI2 - 2 byte unsigned integer).
- **dwAccessRights**: The access rights for this item. (3 = Read/Write).

### FS Gateway (Requesting specific data types)

A new [optional] item name suffix has been added to FSGateway v3.0 to allow selection of the requested data type in the OPCServer which FSGateway is connected to as a data source.

The following are the supported suffixes:

<item name><space>/<suffix>

Suffix	Description
VT_BOOL	Boolean; True=-1, False=0
VT_BSTR	Automation string
VT_I1	Char
VT_I2	2 byte signed int
VT_I4	4 byte signed int

VT\_UI1 unsigned char

VT\_UI2 2 byte unsigned int

VT\_UI4 4 byte unsigned int

VT\_UIINT Unsigned machine int

VT\_INT Signed machine int

VT\_R4 4-byte real

VT\_R8 8-byte real

VT\_DATE time stamp

**For example:** item /VT\_I1

This item will use the RequestedDatatype of **VT\_I1**. In this case the OPC Server will use the requested Data type.

### OPC Server Pushes Data Changes to FSGateway (OnDataChange)

After an item has been added and is active, the OPC Server will push data changes to FSGateway.

```
[03/13 13:00:32.76] (6) Wonderware FSGateway - CClientDataObjectSniffer::OnDataChange() - dwTransid=0x00000000,
hGroup=0x01000000, hrMasterQuality=0x00000000, hrMasterError=0x00000000, dwCount=1
[03/13 13:00:32.76] (6) Wonderware FSGateway - CClientDataObjectSniffer::OnDataChange() - [ 0] - pErrors=0x00000000
[03/13 13:00:32.78] (6) Wonderware FSGateway - CClientDataObjectSniffer::OnDataChange() - hClientItem=0x01000000
ftTimeStamp=0x01ce2025-0x6abdde70 [3/13/2013 8:00:31 PM] wQuality=0x00c0 vDataValue: vt=0x0012 [48]
[03/13 13:00:32.78] (7) Wonderware FSGateway - CClientDataObjectSniffer::OnDataChange() - Function returning [hr]=[0x00000000]
at line [188] in [.\ClientDataObjectSniffer.cpp].
```

- **hGroup:** Client group handle to which the item belongs.
- **hrMasterQuality:** OPC Server indicates that all item qualities are good.
- **hrMasterError:** OPC Server indicates no errors.
- **dwCount:** The number of items.
- **pErrors:** List of hResults for the items. If the quality of an item is not good, this field will list specific errors for that item.
- **hClientItem:** List of client handles for the items that have changed.



**ftTimeStamp**: List of Timestamps.

- **wQuality**: List of Quality values.
- **vDataValue**: Contains the data type (vt) and values [48] for the items that have changed.

## Writing Values

### FSGateway Adds the Item (AddItems)

The item is added to the OPC Server even if it is already subscribed. Creating a new item for writing is done to prevent the value from bouncing between the read value and the written value in the client application.

```
[03/13 13:00:45.64] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::AddItems() - dwCount=1
[03/13 13:00:45.64] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::AddItems() - [ 0] - szAccessPath=[],
szItemID=[port.plc.400001], bActive=0, hClient=0x01000001, dwBlobSize=0, vtRequestedDataType=0x0000
[03/13 13:00:45.64] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::AddItems() - [ 0] - ppErrors=0x00000000
[03/13 13:00:45.64] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::AddItems() - hServer=0x01000005,
vtCanonicalDataType=0x0012, dwAccessRights=0x00000003, dwBlobSize=0
[03/13 13:00:45.64] (7) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::AddItems() - Function returning
[hr]=[0x00000000] at line [371] in [.\OPCGroupSniffer.cpp].
```

The item is added as inactive (**bActive=0**) since FSGateway only wants to write a value to the item and does not expect data changes.

### FSGateway Writes to the Item (Write)

```
[03/13 13:00:45.65] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCAsyncIO2::Write() - dwCount=1,
dwTransactionID=0x01000000
[03/13 13:00:45.65] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCAsyncIO2::Write() - [ 0] - hServer=0x01000005
vDataValue: vt=0x0012 [24]
[03/13 13:00:45.65] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCAsyncIO2::Write() - *pdwCancelID=0x01000000
[03/13 13:00:45.65] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCAsyncIO2::Write() - [ 0] - ppError=0x00000000
[03/13 13:00:45.65] (7) Wonderware FSGateway - COPCGroupSniffer::IOPCAsyncIO2::Write() - Function returning [hr]=[0x00000000]
at line [1307] in [.\OPCGroupSniffer.cpp].
```

- **dwCount**: The number of items to be written.
- **dwTransactionID**: This transaction ID is used in the write complete information returned from the OPC Server.
- **hServer**: Server handle of the item to be written.
- **vDataValue**: Data Type (vt) and Value to be written [24].
- **ppError**: Errors returned by the OPC Server.

### OPC Server Notifies Client of Write Complete (OnWriteComplete)

```
[03/13 13:00:45.67] (6) Wonderware FSGateway - CClientDataObjectSniffer::OnWriteComplete() - dwTransid=0x01000000,
hGroup=0x01000000, hrMasterError=0x00000000, dwCount=1
[03/13 13:00:45.67] (6) Wonderware FSGateway - CClientDataObjectSniffer::OnWriteComplete() - [ 0] - pErrors=0x00000000
[03/13 13:00:45.67] (6) Wonderware FSGateway - CClientDataObjectSniffer::OnWriteComplete() - hClientItem=0x01000001
[03/13 13:00:45.67] (7) Wonderware FSGateway - CClientDataObjectSniffer::OnWriteComplete() - Function returning
[hr]=[0x00000000] at line [316] in [.\ClientDataObjectSniffer.cpp].
```

- **dwTransid**: The Transaction ID returned to the client.
- **hrMasterError**: OPC Server indicates no errors.
- **dwCount**: Number of items.
- **pErrors**: Item errors returned.
- **hClientItem**: Client item handle for the item that was written.

### FSGateway Removes the Item (RemoveItems)

```
[03/13 13:00:53.95] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::RemoveItems() - dwCount=1
[03/13 13:00:53.95] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::RemoveItems() - [ 0] - phServer=0x01000005
[03/13 13:00:53.95] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::RemoveItems() - [ 0] - ppErrors=0x00000000
[03/13 13:00:53.95] (7) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::RemoveItems() - Function returning
[hr]=[0x00000000] at line [469] in [.\OPCGroupSniffer.cpp].
```

- **dwCount**: Number of items.
- **phServer**: Server Item Handle for the item to be removed.
- **ppErrors**: Item errors returned.

### Deactivating and Activating an Item

In InTouch or Application Server, items can be activated or de-activated. For example, If there is an item on an InTouch window and that's the only place the item is used, InTouch will activate the item when the window is showing and deactivate the item when the window is closed. This is so the Server doesn't have to poll for points that clients aren't interested in. FSGateway will in-turn tell the OPC server to activate or deactivate the item.

### FSGateway Deactivates an Item (SetActiveState)

```
[03/13 13:00:53.95] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::SetActiveState() - dwCount=1, bActive=0
[03/13 13:00:53.95] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::SetActiveState() - [ 0] - phServer=0x01000004
[03/13 13:00:53.95] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::SetActiveState() - [ 0] - ppErrors=0x00000000
[03/13 13:00:53.95] (7) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::SetActiveState() - Function returning
[hr]=[0x00000000] at line [506] in [.\OPCGroupSniffer.cpp].
```

- **dwCount**: – Number of items to be affected.

- **bActive**: FSGateway deactivates the item. (1 = Activate, 0=Deactivate).
- **phServer**: Server Item Handle.
- **ppErrors**: Item errors returned.

### FSGateway Activates an Item (SetActiveState)

```
[03/13 13:00:59.39] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::SetActiveState() - dwCount=1, bActive=1  
[03/13 13:00:59.39] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::SetActiveState() - [ 0] - phServer=0x01000004  
[03/13 13:00:59.39] (6) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::SetActiveState() - [ 0] - ppErrors=0x00000000  
[03/13 13:00:59.39] (7) Wonderware FSGateway - COPCGroupSniffer::IOPCItemMgt::SetActiveState() - Function returning  
[hr]=[0x00000000] at line [506] in [.\OPCGroupSniffer.cpp].
```

Activating an item is the same as deactivating an item but **bActive** is set to 1.

For more information about the MatrikonOPC Sniffer, please visit [www.matrikonopc.com](http://www.matrikonopc.com).

Click the following icon to view this file in .pdf format:



A. Ibrahim, G. Alldredge

*Tech Notes* are published occasionally by Wonderware Technical Support. Publisher: Invensys Systems, Inc., 26561 Rancho Parkway South, Lake Forest, CA 92630. There is also technical information on our software products at [Wonderware Technical Support](#).

For technical support questions, send an e-mail to [wwsupport@invensys.com](mailto:wwsupport@invensys.com).



[Back to top](#)

©2013 Invensys Systems, Inc. All rights reserved. No part of the material protected by this copyright may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying, recording, broadcasting, or by any information storage and retrieval system, without permission in writing from Invensys Systems, Inc.

[Terms of Use](#).