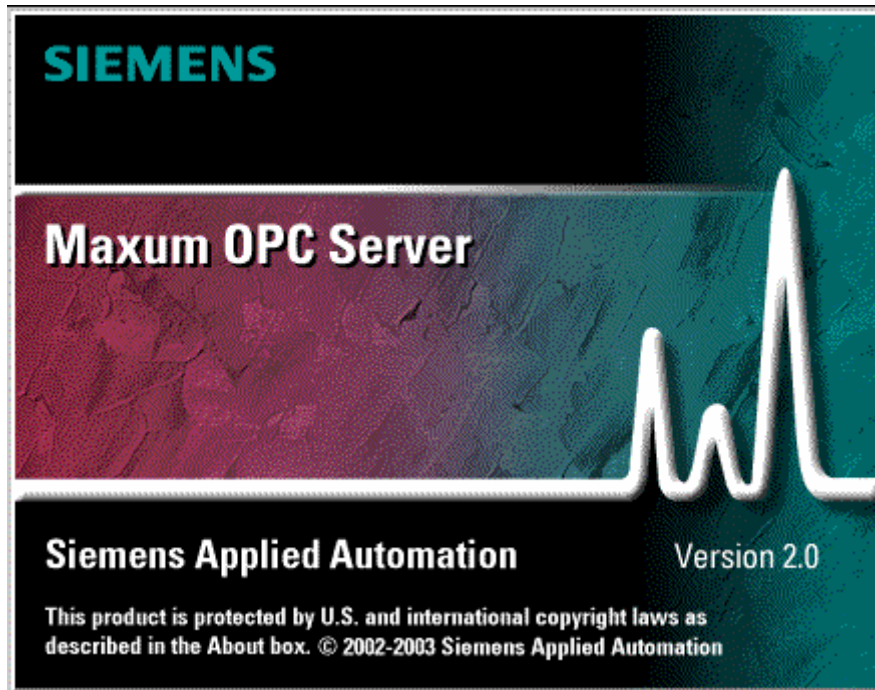


SIEMENS

Maxum OPC Server



OPERATION & TROUBLESHOOTING GUIDE



Maxum OPC Server – Operations & Troubleshooting Guide

© Copyright 2003-2006
Siemens

TABLE OF CONTENTS

- 1. INTRODUCTION..... 5**
 - 1.1. SYSTEM REQUIREMENTS 5**
 - 1.1.1. Product Interface Requirements.....5
- 2. OVERVIEW 6**
 - 2.1. OPC Overview 6**
 - 2.2. Maxum OPC Server Overview..... 6**
- 3. Getting Started 7**
 - 3.1. OPC Table Initialization:..... 7**
 - 3.2. Installation 8**
 - 3.3. COM Server Registration 14**
 - 3.4. WIN32 Service Registration 14**
 - 3.5. Standard OPC Proxy/Stubs..... 16**
 - 3.6. OPC Foundation Supplied Programs: 16**
 - 3.7. Maxum OPC Server Files 17**
 - 3.8. Default Directory Structure 17**
 - 3.9. Software Licensing..... 17**
 - 3.10. Server Tag Configuration..... 19**
- 4. Using Maxum OPC Server 26**
 - 4.1. User Interface 26**
 - 4.2. Main Menu 27**
 - 4.2.1. File Menu.....27
 - 4.2.2. Add Menu27
 - 4.2.3. Edit Menu27
 - 4.2.4. View Menu27
 - 4.2.5. Options Menu27
 - 4.2.6. Help Menu28
 - 4.3. Monitoring 28**
 - 4.4. Editing..... 28**
 - 4.4.1. Deleting29

4.4.2.	Properties	29
4.5.	Export tags	30
5.	OPERATIONAL OVERVIEW	32
5.1.	Device Connection.....	32
5.1.1.	Maxum/NAU ACTIVE QUERIES:.....	32
5.1.2.	OPTICHROM ACTIVE QUERIES:	32
5.2.	Device Updates	32
5.3.	OPC Client	33
5.3.1.	OPC and the Component Object Model	33
5.3.2.	The Registry.....	33
5.4.	OPC Server Tags.....	34
5.4.1.	Maxum, Maxum EditionII, NAU, MicroSam TAGS	34
5.4.2.	Maxum Tag Descriptions.....	37
5.4.3.	Advance Optichrom Tags	40
5.4.4.	Optichrom Tag Descriptions.....	41
6.	MAXUM DATABASE - OPC SETUP	43
6.1.	Setup Procedure	43
6.2.	OPC Data Collection Table.....	43
Appendix A - Distributed COM (DCOM).....		44
6.3.	DCOM Configuration Utility	44
6.4.	Default Properties	46
6.5.	Security Permissions	47
6.6.	Server Identity	51
6.7.	Default Protocols	52
7.	Appendix B OPC Test Client	53
8.	Appendix C Automation Client Example.....	56
8.1.	Client Code Descriptions	62
9.	Additional Information Resources.....	66
10.	Glossary	67

1. INTRODUCTION

If you are unfamiliar with any of the terms or concepts used in this document please refer to the Glossary section located at end of this document. If you wish to obtain more information on of any of the terms or concepts presented here please refer to the section titled: Additional Information Resources .

This document contains the Operation & Troubleshooting Guide for Siemens Energy & Automation, Inc. OPC Server. Maxum OPC Server allows OPC compliant clients to exchange data with the Maxum Edition II, Advance Maxum, Advance Optichrom, MicroSam, and Process Quantra analytical instruments.

Maxum OPC Server provides the following features:

- OPC DA 2.04 compliant
- Runs as Window’s application or Win32 Service
- Tag browsing using OPC Foundation List Server (OPCEnum.exe)
- Supports server self-registration
- Support for both the OPC Custom and Automation Interface (Visual Basic, VBA)
- Supports both local and network data access
- Uses free threading model for efficient COM/DCOM communications
- Supports asynchronous and synchronous communications
- Supports network communication over Ethernet, Advance Datanet, and Advance DataHiway* .

1.1. SYSTEM REQUIREMENTS

Minimum platform requirements :

- Pentium 4 with 1.8 GHz processor (or faster),
- 256 Mb RAM,
- 30 Mb free hard drive space,
- Ethernet 10/100 interface,
- Microsoft Windows NT 4.0 SP6 / Windows 2000 / Windows XP

While Maxum OPC Server may work on platforms not meeting the minimum specifications, operational reliability and efficiency will be diminished.

Note: Windows NT has a known issue with interactive services. If you plan on running Maxum OPC Server as a Win32 Service it is recommended that you upgrade to either Windows 2000 or Windows XP.

1.1.1. Product Interface Requirements

Product	Description
Advance Maxum Maxum Edition II	Can connect to the Maxum OPC Server directly through an Ethernet or DataNet network
*Advance Optichrom	Requires an (ODS) Optichrom Data Server. The ODS consists of a NAU (Network Access Unit) and either an Advance Ethernet Gateway or an Advance DataNet Gateway. The Optichrom analyzers communicate to the ODS and the ODS communicates with the OPC Server.
MicroSam	Can connect to the Maxum OPC Server directly through an Ethernet or DataNet network
Quantra	Can connect to the Maxum OPC Server directly through an Ethernet or DataNet network.

2. OVERVIEW

2.1. OPC Overview

OLE for Process Control (OPC) was designed to bridge Windows based applications and process control hardware and software applications. It is an open standard that permits a consistent method of accessing field data from industrial devices. This method remains the same regardless of the type and source of data. OPC servers provide a method for many different software packages to access data from a process control device, such as a Gas Chromatograph. Traditionally, any time an application needed to access data from a device, a custom interface, or driver, had to be written. The purpose of OPC is to define a common interface that is written once and then reused by any application supporting an OPC interface.

OPC servers use Microsoft's OLE technology (also known as the Component Object Model, or COM) to communicate with clients. COM technology permits a standard for real-time information exchange between software applications and process hardware to be defined.

2.2. Maxum OPC Server Overview

Maxum OPC Server is the ideal solution for sharing process analytical information with Distributed Control Systems, Statistical Quality Control programs, Data Historians or any custom application supporting an OPC interface. A single OPC Server can provide data to multiple client applications or a single client application may access data from multiple OPC Servers.

OPC Server handles the device specific interface issues transparently, enabling users to focus on the acquired data. Remote device connection, error detection and recovery, and all database access functions are managed automatically.

OPC Server is data change driven. Data values are sent only when there is a change in value. The Server subscribes or registers with the remote device (analyzer) as part of the power-up initialization sequence. The Server specifies the database variables it is interested in and the minimum duration between updates. The remote device automatically notifies and updates the OPC Server based on the subscription parameters. Typically device updates occur at the end of the analysis cycle.

OPC Server maintains a copy of the latest data values for all OPC Items or Tags. Each OPC Item has a set of attributes or properties that describes the item. Available attributes include data type, data value, data quality, and timestamp. Data type refers to the OPC Items Canonical data type. Supported data types are:

- VT_I2 – 2 byte integer,
- VT_I4 – 4 byte integer,
- VT_R4 - 4 byte float,
- VT_R8 - 8 byte float,
- VT_DATE – date and time,
- VT_BSTR - String,
- VT_BOOL – Boolean

Data value is the current value of the Item based on its data type.

Data quality provides an indication of the “worth” or reliability of the data. Maxum OPC Server supports the following data qualities:

Uncertain – OPC Server is not certain that the data value is current or valid.

Good - The value is current and reliable.

Bad – The value is unreliable.

Communication failure – The Server has lost communications with the remote device. The data value may or may not reflect the actual value in the analyzer database.

The timestamp includes the date and time the OPC Item was updated.

3. Getting Started

This section describes the necessary steps to install, configure, and run OPC Server.

IMPORTANT: Maxum OPC Server is compatible with Maxum database version 4.0 (or greater). Please upgrade any analyzer running an earlier version before starting the server configuration procedure. Optichrom analyzers communicate through an intermediary device called an Optichrom Data Server. The Optichrom Data Server provides the necessary database conversions between the two systems.

One of the many enhancements provided in version 4.0 is the OPC table. The OPC Table was added to facilitate data exchange between Maxum analytical devices and OPC Server. The OPC Table must be initialized and populated before a device or analyzer is added to OPC Server.

Complete the following steps in order:

Step#1 – Configure the analyzer database before adding the analyzer to the OPC Server.

Step#2 – Install the OPC Server software.

Step#3 - Configure the OPC Server Tag database.

Step#1 - Configure the Maxum Database in each analyzer before adding the analyzer to the OPC Server. The following section explains how to configure and initialize the Maxum database.

3.1. OPC Table Initialization:

- a.) In the *system_control* table set the *opcflag* attribute. When the *opcflag* attribute is set (1) the OPC Table is in enabled.
- b.) For each Application display the *stream_method* table and set the *autotrt* attribute for each stream. Repeat this step for each application and stream. The *autotrt* attribute enables OPC Table updates at the end of each analysis cycle.
- c.) Under each Application display the *result* table and set the *trtval* attribute equal to the desired result transmission sequence number. The *trtval* attribute sets the transmission order for results. Results may be sent in any order. *Trtval* values must be sequential starting with ‘1’. Out of sequence values or results with a *trtval* equal to ‘0’ will not be transmitted. Repeat this step for each application/stream.
- d.) This step is required only if the Maxum database will be receiving Optichrom transmissions (TRTs). (like an Optichrom Data Server) Data from Optichrom analyzers must be converted to Maxum database format for OPC Server compatibility. Optichrom analyzers transmit their results to an Optichrom Data Server. The OPC Server communicates with the Optichrom Data Server to obtain analyzer results and status information. The Optichrom Data Server consists of an NAU (Network Access Unit) running application software with an ANG (Advance Network Gateway). Optichrom analyzers support several different communication formats. Optichrom Data Server must

be configured to use the same format. In the Optichrom Data Server database set the *System_control* table *adh_trformat* attribute equal to the desired host computer format:
0=HCI-H, 1 = HCI-C01, 2 = HCI-C04, 3 = HCI-A

Step#2

Install Maxum OPC Server software.

IMPORTANT: During the installation process you can chose to install the Maxum OPC Server as a Windows application or as a Win32 Service. If you initially installed the MaxumOPCServer as an application you may reconfigured it to run as a Service by simply running the application from the command line and adding the program argument *-Service*. For example to change the operation mode from an application to a service select Start, Run..., then browse to the OPC Server executable (MaxumOPCServer.exe) and add the *-Service* argument (MaxumOPCServer.exe *-Service*)\

3.2. Installation

Maxum OPC Server is provided on CD-ROM. To begin the installation process, insert the CD into your drive. If the Autorun option is enabled the setup program will automatically begin operation of the Server installation program. If the Autorun option is disabled run the setup program (setup.exe) located under the main directory on the installation CD.

NOTE: Installation of the OPC Server requires administrator rights. This means the administrator or someone in the administrator group must be logged in during the installation and setup process.

The first screen displayed is the Setup Language screen. The default setting is English as shown in Figure 1.

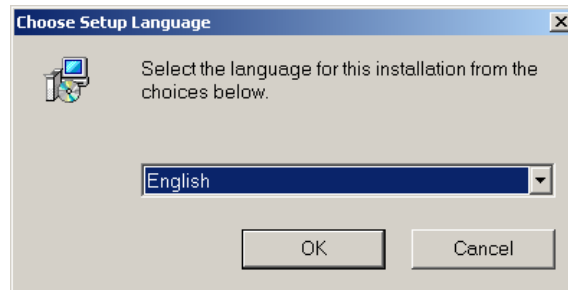


Figure 1

The next screen is the Configuration screen. This screen describes the Operating System and hardware requirements for Maxum OPC Server. (See Figure 2)

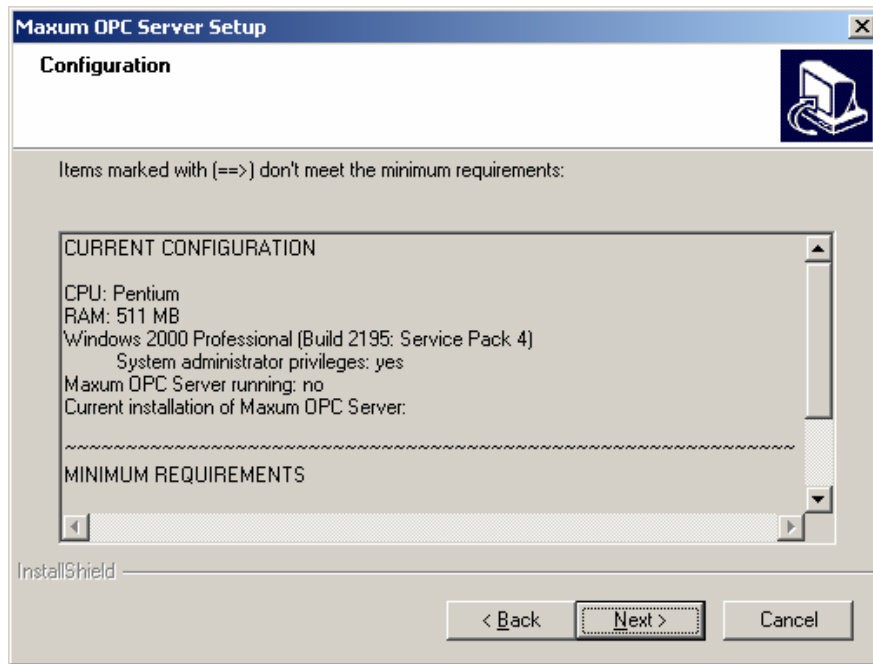
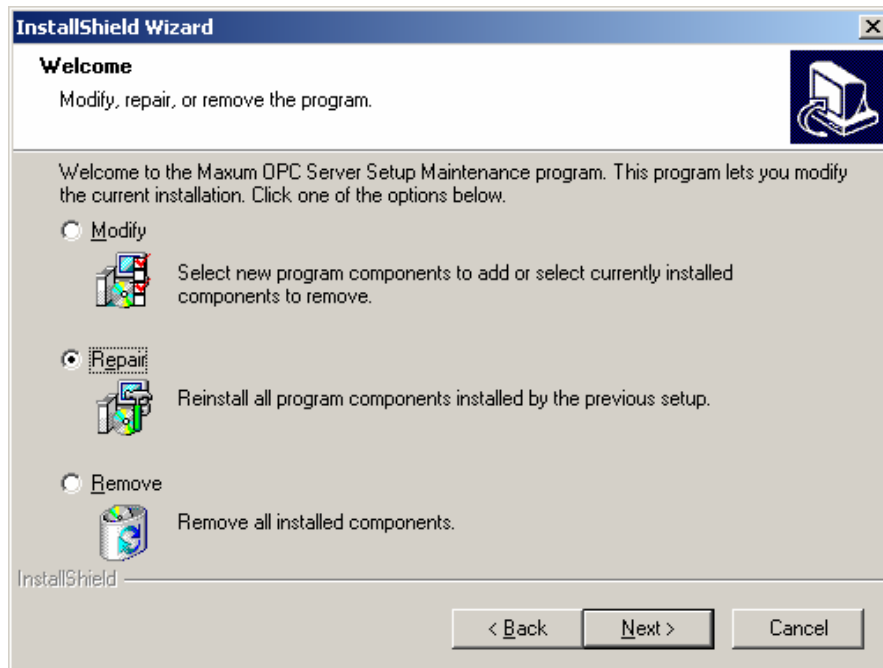


Figure 2

After reading and verifying the minimum system requirements select the Next button to continue.



The next screen is the License Agreement screen. Please read this screen carefully. If you agree to all the terms and conditions click Yes to continue. (See Figure 3)

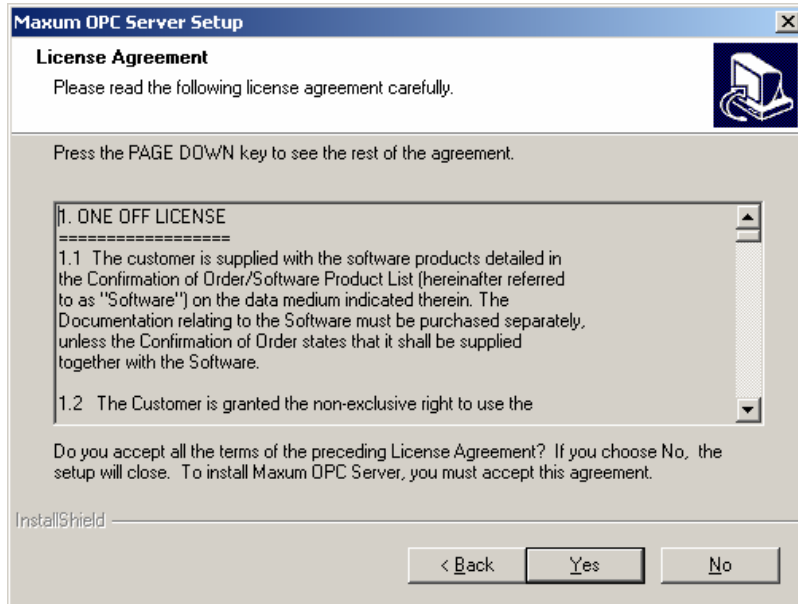


Figure 3

The next screen is the User Information and registration screen. Enter your name, your company name, and the serial number provided with your purchase. The default name and company information (if any) is displayed from the information input into Windows when it was initially installed. Enter the Software Key provided with your installation CD. If you have not purchased the Maxum OPC Server and you wish to load the program as a Demo, simply type *demo* in place of the serial number and select *Next*. (See Figure 4)

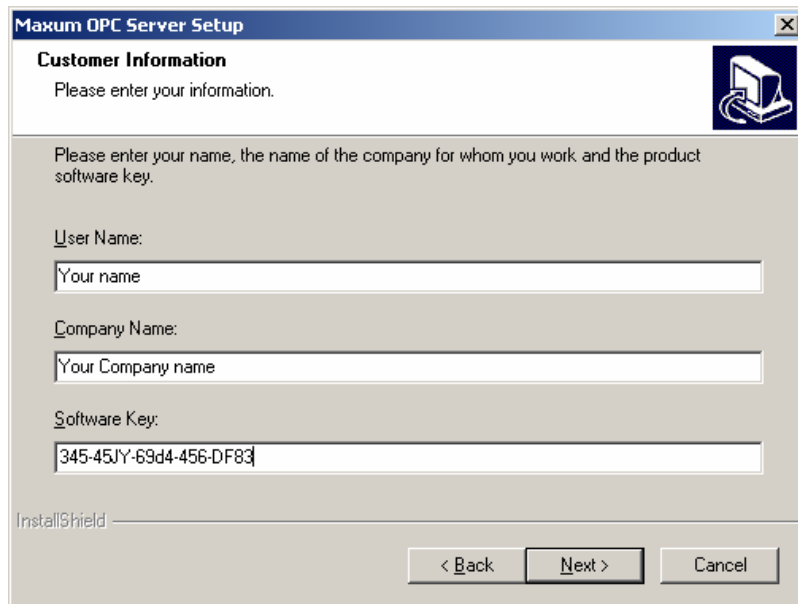


Figure 4

If you didn't enter a serial number correctly or you typed *demo*, the Error dialog box will be displayed. If you want to load the program in Demo mode select the *Yes* button.(See Figure 5)

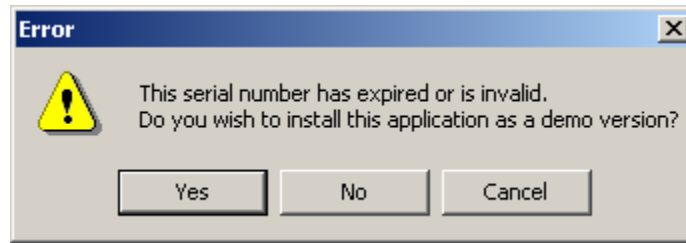


Figure 5

The next screen is the Select Components screen. This screen allows you to select the components you wish to install and to specify the destination path or folder where the Maxum OPC Server files will be installed. Note: the OPC Base components consists of the OPC Foundation support DLL's and List Server. These files are required for the proper operation of the OPC Server. Select the Browse button if you would like to install the Maxum OPC Server files in a folder other than the default folder. (See Figure 6)

Select the *Next* button to continue to the next screen.

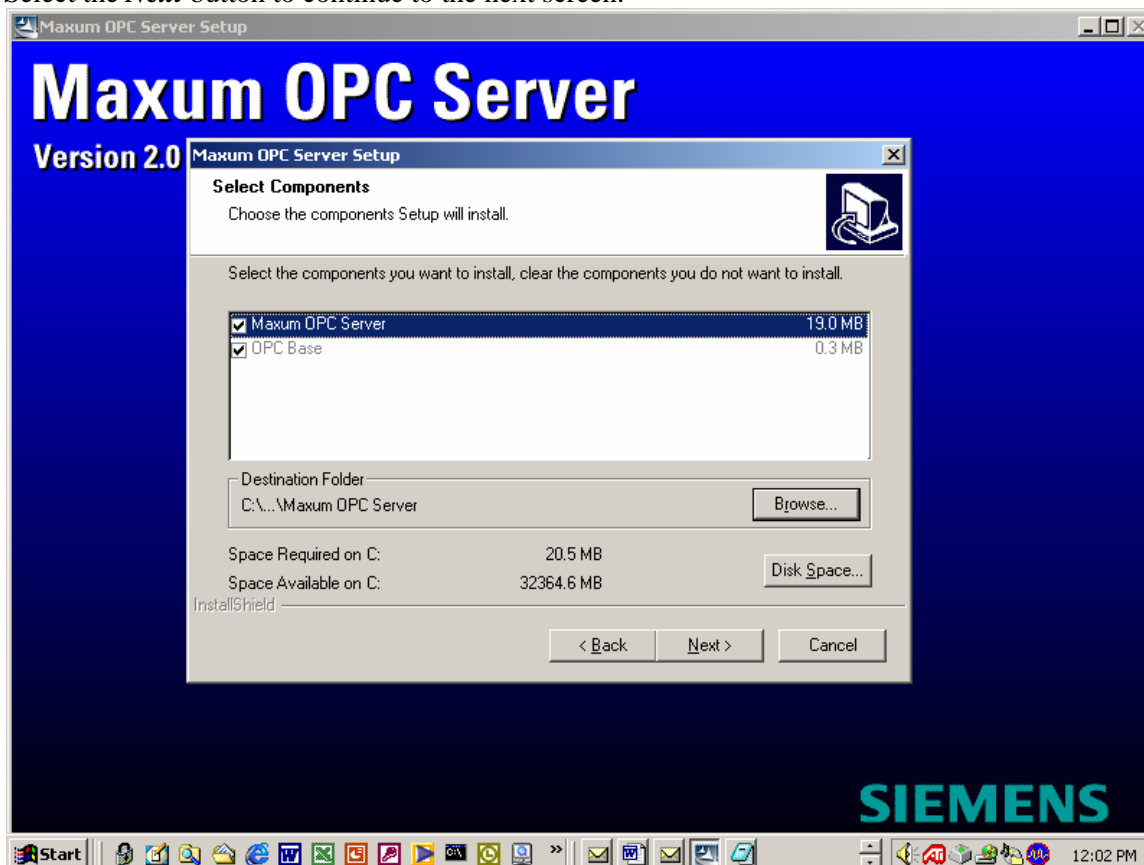


Figure 6

6

The next screen determines which folder the program icons will be placed in. The default folder (as shown in Figure 7) is the Maxum System Tools folder. Select the Next button to continue.

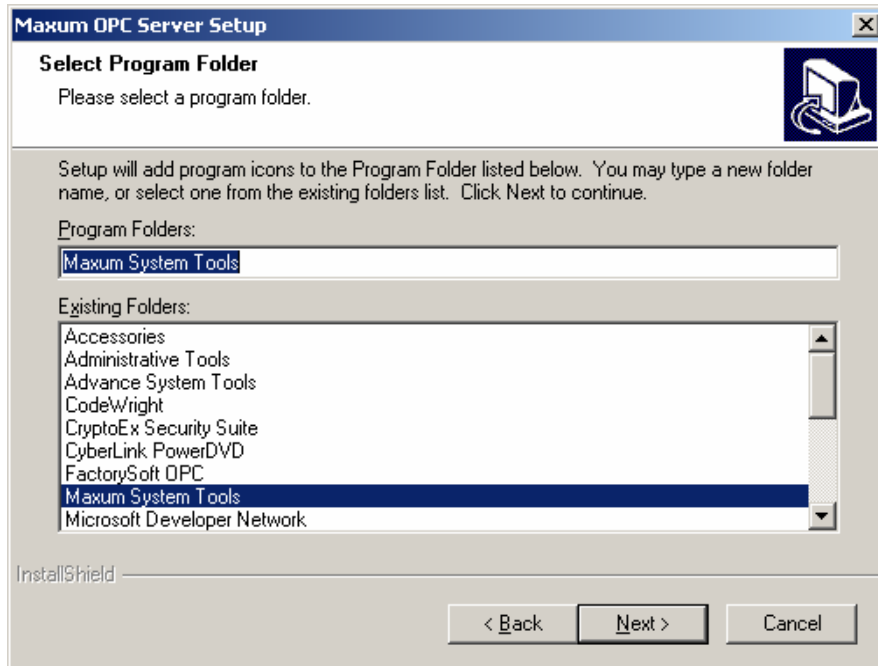


Figure 7

The Application Mode provides the user the ability to install the Maxum OPC Server as a standard Windows 32-bit application or as a Windows Interactive Service. The default mode installs the OPC Server as a Windows application. Select the desired option and click on Next

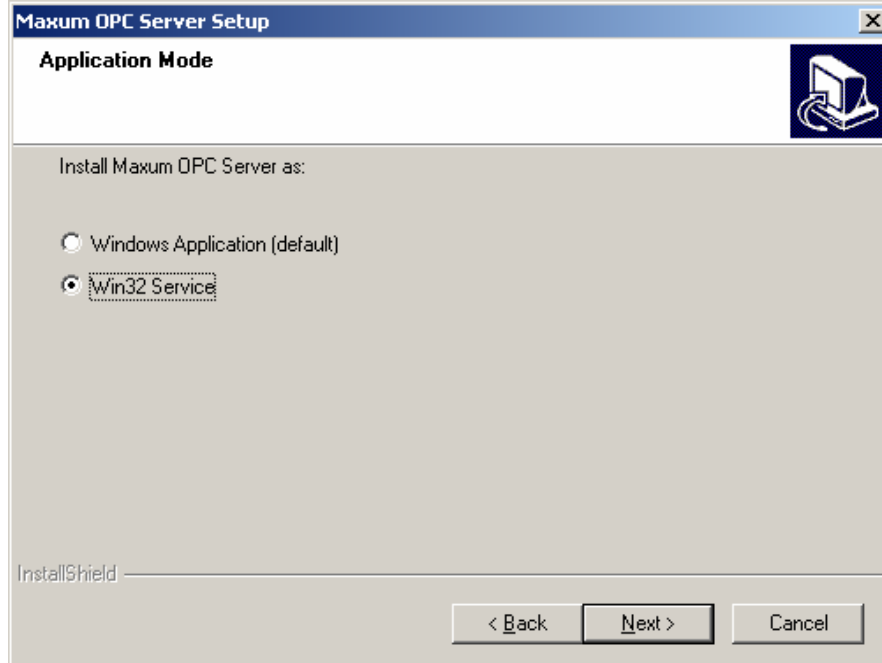


Figure 8

The Create Shortcuts dialog box provides the user the ability to choose the desired shortcut(s) for the Maxum OPC Server program. Select the desired options and click on Next.

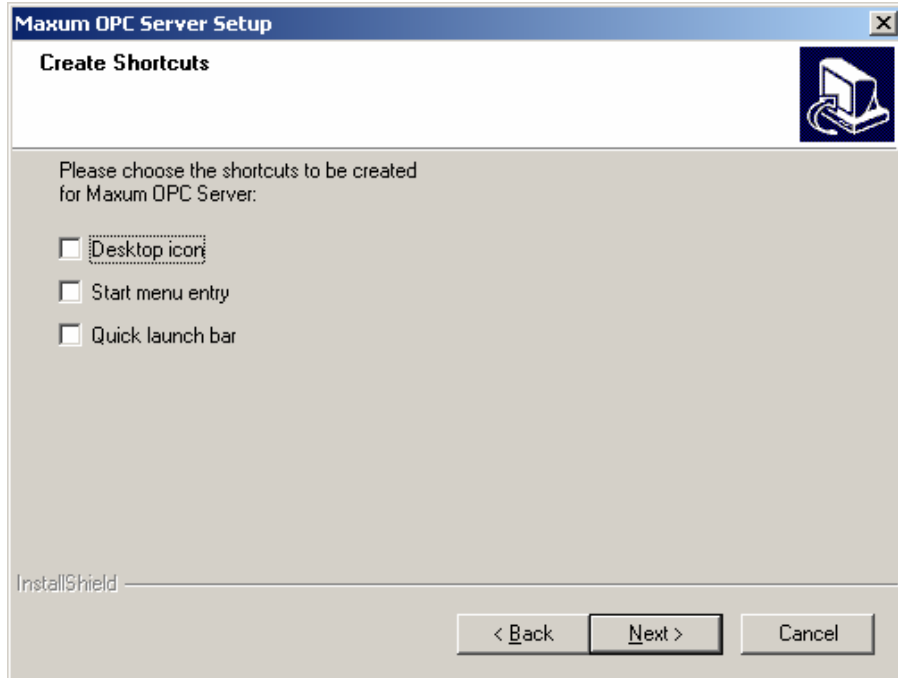


Figure 9

The next screen is the Start Copying Files screen and allows you to check your selected settings prior to the files being installed. If the Current Settings are correct select the Next button to continue. (See Figure 10)

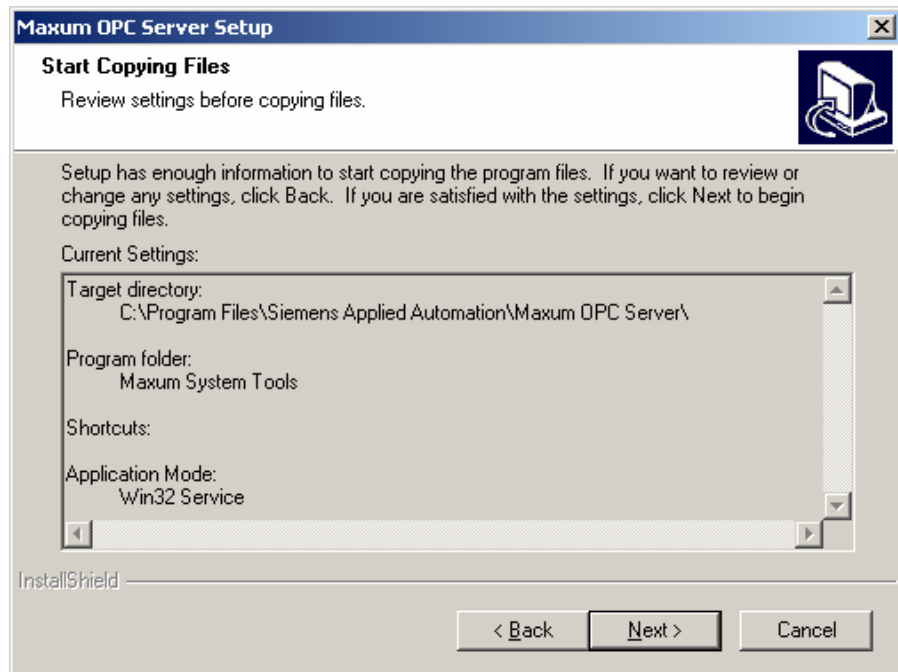


Figure 10

The next screen is the Setup Status Screen.

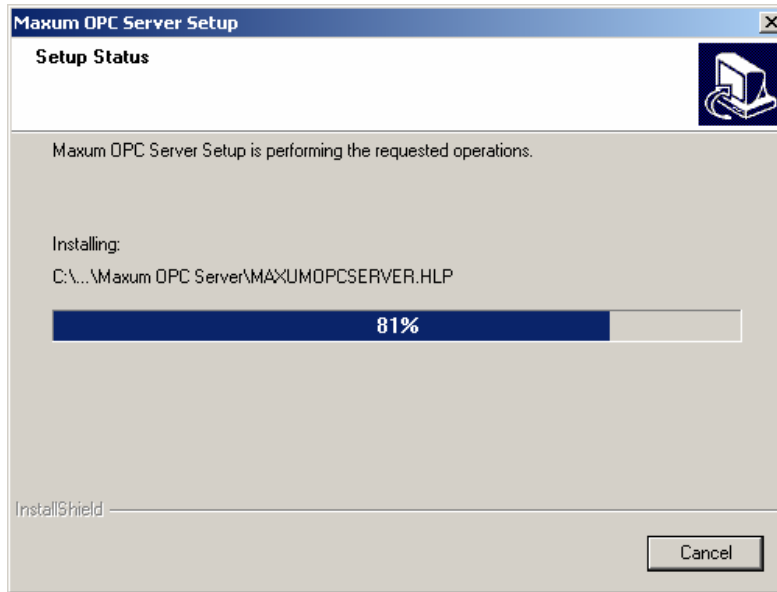


Figure 11

Assuming there is no error messages displayed this completes the installation procedure. The Server and associated OPC support programs are automatically registered and no additional steps are required.

3.3. COM Server Registration

Maxum OPC Server is automatically installed and registered during the installation process. You can use the windows program *regedit* to check the registry entries. OPC Server puts two entries under the HKEY_CLASSES_ROOT key. One is *Siemens.MaxumOPCServer* which contains the version-independent program ID (CLSID) and, one titled *Siemens.MaxumOPCServer.2* that contains the version-specific program ID (CLSID). COM/DCOM use the CLSID to determine the path to the Maxum OPC Server executable for remote launch capability.

3.4. WIN32 Service Registration

A Win32 Service is a background application that executes across logon sessions. A Win32 Service exporting a COM component must be recognized by the operating system as a COM server and as a running service. Therefore, it must be registered directly in the registry as a COM server and in the services database as a service. When Maxum OPC Server is installed with the Service option or when the program is executed with the command line option */Service* or *-Service* the OPC Server will automatically register itself as a COM server and a Windows Service.

LocalService is a registry key that merges two fundamental constructs of the Microsoft Win32® environment. It binds a Component Object Model (COM) component to a service combining the benefits of both technologies. The resulting object is available to COM clients and service controllers. Maxum OPC Server defaults to the Local System account. Services registered with *LocalService* are not automatically stopped between logon sessions. Like other services, *LocalService* continues to execute when a user session concludes.

Within a service, you assign a COM component a user or system security context. If a user context is not assigned, the security context defaults to *LocalSystem*. *LocalSystem* provides services with expanded privileges, but limits access to shared resources by restricting access to the interactive user and desktop.

The “Allow Service to interact with desktop” option is enabled in the Services database under the MaxumOPCServer LogOn tab. (See Figure 12) This option allows authorized users to view and/or edit the OPC tag database and view the built in OPC client interface. This option also enables the display of the MaxumOPCServer icon in the system tray when MaxumOPCServer is running as a service. (See Figure 13)

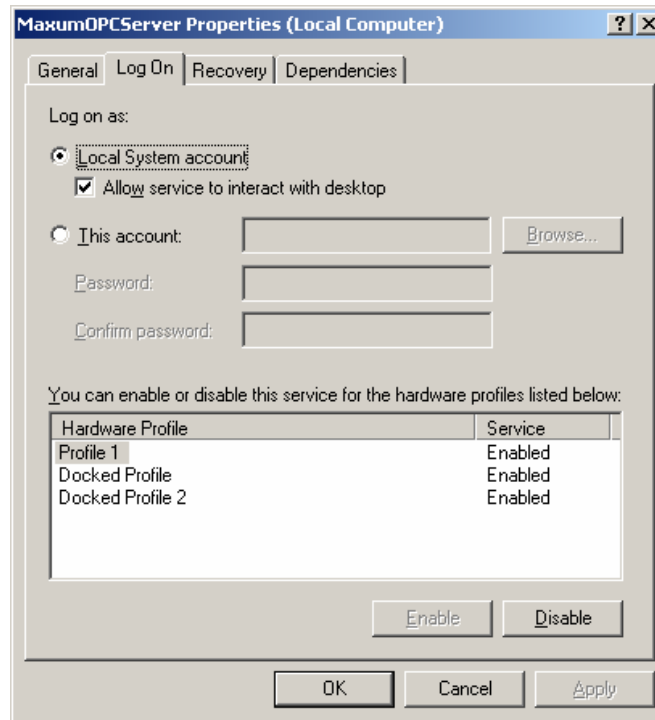


Figure 12

Note: Windows NT 4.0. has a known problem with interactive services. An interactive service is a service that interacts with the currently logged-on user through a typical Windows-based user interface. If you intend on running Maxum OPC Server as a service in Windows NT, disable the "allow service to interact with desktop" option in the Service Manager (after service installation). When this option is disabled you will not be able to see MaxumOPCServer icon in the system tray. To configure the program you will need to start a second copy of MaxumOPCServer as a regular program, Open the target Tag configuration file, make the necessary changes, Save the file then restart the MaxumOPCServer service. This problem does not exist in Windows 2000 or XP.

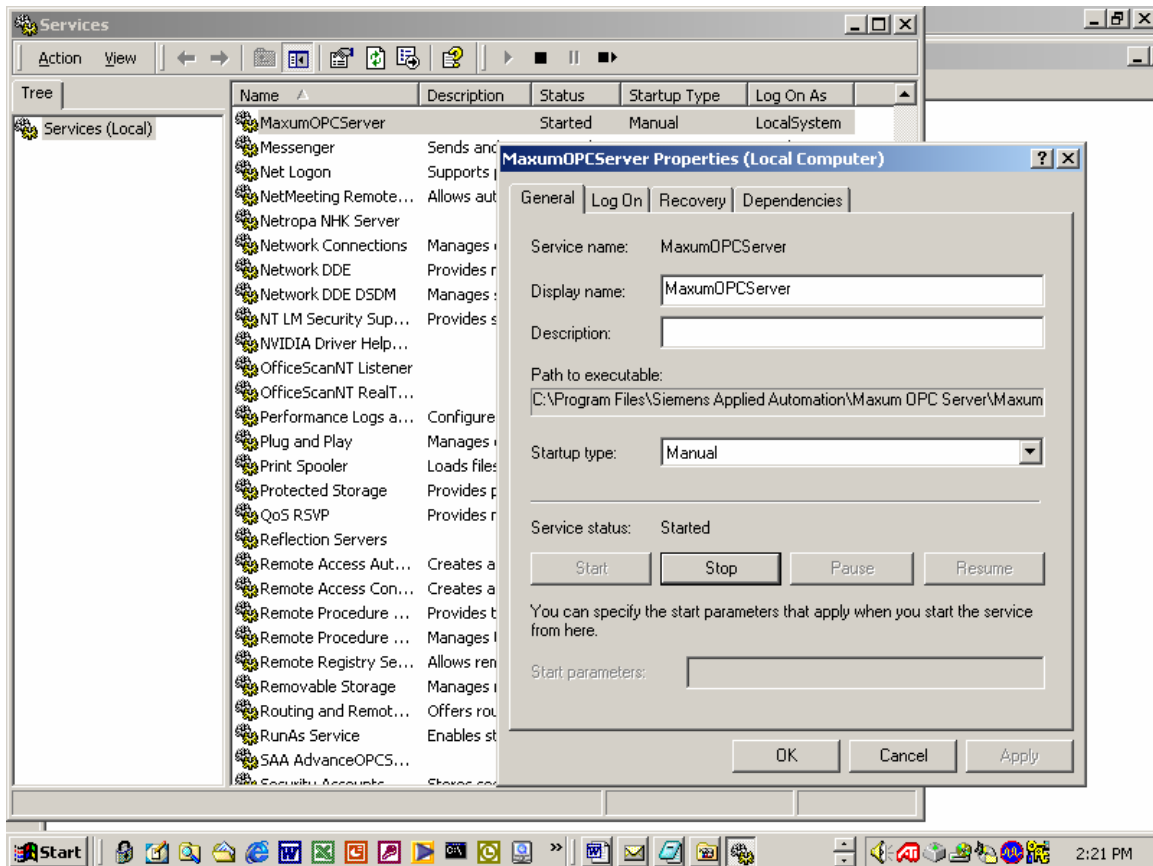


Figure 13

3.5. Standard OPC Proxy/Stubs

In compliance with OPC standards, the Maxum OPC Server installs and registers the following files in the WINNT/ SYSTEM32 directory.

- 1.) The official OPC proxy/stub DLL's, "OPCCOMN_PS.DLL and "OPCPROXY.DLL".
- 2.) The official OPC List Server (Browser) "OPCENUM.EXE".
- 3.) The Automation Component "OPCDAAUTO.DLL" wrapper which enables developers to access OPC data from client applications developed using Automation tools such as Visual Basic, VBA, and VB Script.

The version information contain in these files is used to avoid overwriting newer files with outdated ones.

3.6. OPC Foundation Supplied Programs:

Filename	Description
OPCCOMN_PS.DLL	Proxy library for OPC Common interface
OPCPROXY.DLL	Proxy library for OPC Custom interface
OPCENUM.EXE	OPC List server
OPCDAAUTO.DLL	OPC-DA automation library

3.7. Maxum OPC Server Files

The following files are installed during the installation procedure:

Filename	Description
MAXUMOPCSERVER.EXE	MOS executable
POLYHEDRA.DLL	Maxum database communication dll
OPCCCLIENT.EXE	OPC test client

3.8. Default Directory Structure

The MOS installation program creates the following default directory structure:

\Program Files

\Siemens Applied Automation

\System

\Client

\Documents

The following files are loaded under the specified directory.

\Program Files

\Siemens Applied Automation

- MaxumOPCServer.exe - main program executable
- DbPolyhedra.dll - support dll
- SwKey.dll - support dll
- MaxumOPCServer.hlp - Windows Help file
- MaxumOPCServer.cnt - Windows Help support

\System

- opcerror.log - The OPC Server log file is created the first time program is run.

\Client

- AutomationClientEx.xls - Example Excel-VBA automation client program
- OPC Client.exe - FactorySoft OPC Client test program

\Documents

- MaxumOPCServerManual.Doc - This manual
- opcda20_auto.doc - Data Access Automation Interface Standard provided by the OPC Foundation

3.9. Software Licensing

The Maxum OPC Server utilizes a software license key to enable/disable functionality and provide warranty tracking information. The Installation key is a 26-digit alphanumeric number supplied with the product. The Installation key must be entered in the Software Registration screen during installation. If no installation key is provided the program will install in the Demo mode. Product licensing information is displayed in the Windows About box. This includes the Serial Number and License type.

NOTE: If the product was installed in Demo mode and you purchased the product at a later date you do not have to re-install the product to activate the software key. The Maxum System Manager provides an Update license utility. From the Maxum System Manager select Tools->Update License. Enter the 26-digit key in the dialog box. System Manager will automatically update the license key.

There are three types of license keys supported:

1. A **Demo version** that allows the user to connect to a “local” database. A “local” database is a database that runs on the same machine as the OPC Server and is used primarily as a demonstration tool. The “local” database is started with the Maxum System Manager. From the System Manager Main Menu select *System, Analyzer – Offline*, from the Open Dialog box select the *OPCDemo.amd* file. This starts the database on the “local” machine. In the Maxum OPC Server connect to the local database using the “loop back” IP address 127.0.0.1.
2. A **Trial version** that allows the product to run in a fully functional mode for a specific period of time, typically 30 days. After the time period has expired the product reverts to the Demo version.
3. The **Full version** that provides complete and full access on a permanent basis.

There are two types of licenses available.

1. A **Single user** type that allows a single machine to run OPC Server on the same network.
2. A **Site user** type that enables multiple machines to run OPC Server on the same network.

When OPC Server connects to an analyzer it sets up three Active queries. When any of the attributes associated with an active query have a change in value the OPC Server is automatically notified and updated. The three active queries are:

Active queries

	Maxum Database Table	Table Attributes
1	OPC	lastreceived
2	OPC	Inservice Enable Mode
3	Application	Enable mode

Active query #1 is setup on the OPC table - *lastreceived* attribute. *Lastreceived* is updated at the end of each analysis cycle and contains the date and time the analysis cycle completed. This query is used to update all of the OPC tags at the end of the analysis cycle.

Active query #2 is setup on the OPC table - Inservice, Enable, and Mode. This query automatically updates the Application table - Inservice and Stream Enable tags whenever there is a change in value on either of these database attributes.

Active query#3 is setup on the Application table – enable and mode attributes. This query automatically updates the Application mode attribute whenever there is a change in value.

3.10. Server Tag Configuration

OPC Server configuration is the relatively fast and easy process of defining the set of analyzers (devices) and associated data (OPC Items) that will be available through OPC Server. OPC Clients may select all or any portion of the available OPC Items.

Configuration consists of the following steps:

- 1.) Add a new device or analyzer to OPC Server.
- 2.) Add one or more applications to the device folder.
- 3.) Add one or more streams to the application folder.
- 4.) Add one or more components to the stream folder.
- 5.) Add one or more component attributes to the component folder.

OPC Server uses the familiar “Explorer style” user interface to display a hierarchical tree of devices (analyzers) and folders (Streams, Components) with their associated OPC Items (tags). A device is a Maxum Analyzer or Network Access Unit and the folders are the logical device structures consisting of one or more applications, streams, components, and associated data variables. The necessary steps for OPC Server configurations are:

1. Choose *New Device...* from the Add menu or click on the Add New Device Icon located on the toolbar

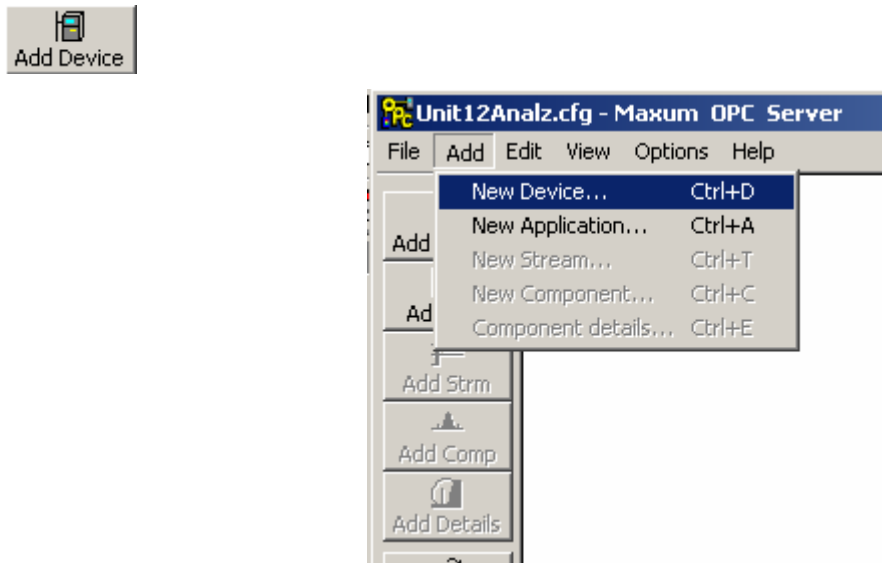


Figure 14

2. Specify the device’s network address(See Figure 15). Then select OK.

This is the device’s IP (Internet Protocol) address and is in the form of xx.xx.xx.xx. The checkbox “Select this box if the device is an Optichrom analyzer.” should be checked if the device is an Optichrom analyzer. If this box is checked the user must specify the Logical ID of the Optichrom analyzer in the “Enter the device (Anlz) Number” box. Advance Optichrom analyzers communicate with the MOS through the Advance Gateway and NAU. Therefore, the network address (IP number) entered in the Device address dialog is the same as the NAU. Multiple Advance Optichrom devices can be created, using the same IP address. This is because Optichrom analyzers communicate through an NAU, and

therefore, the Optichrom’s use the IP address of their associated NAU (Network Interface Unit). For Optichrom analyzers the IP address and the logical ID (Analyzer number) determine the device address. Advance Maxum and Maxum Edition II analyzers use the IP address for configuration purposes. The logical ID is automatically queried from the device database.

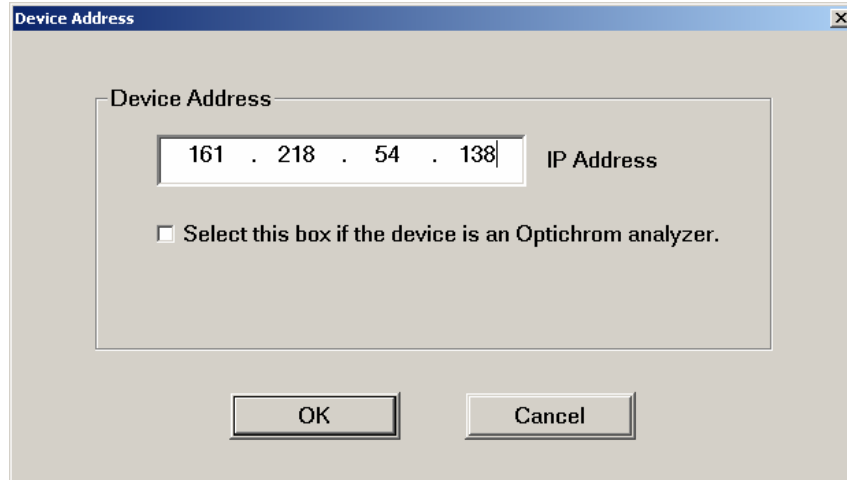


Figure 15.

3. The MOS connects with the device and sends a query to determine the available Applications and Streams. The *Device Setup* dialog box is automatically displayed. Select the specific Application(s) and Stream(s) by clicking on the appropriate checkbox under the *Enable* column or select all of the Applications and associated Streams by clicking the left mouse button on the *Select All* checkbox. (See Figure 16). Click on OK to create the Device, Application, and Stream folders with their associated Tags.

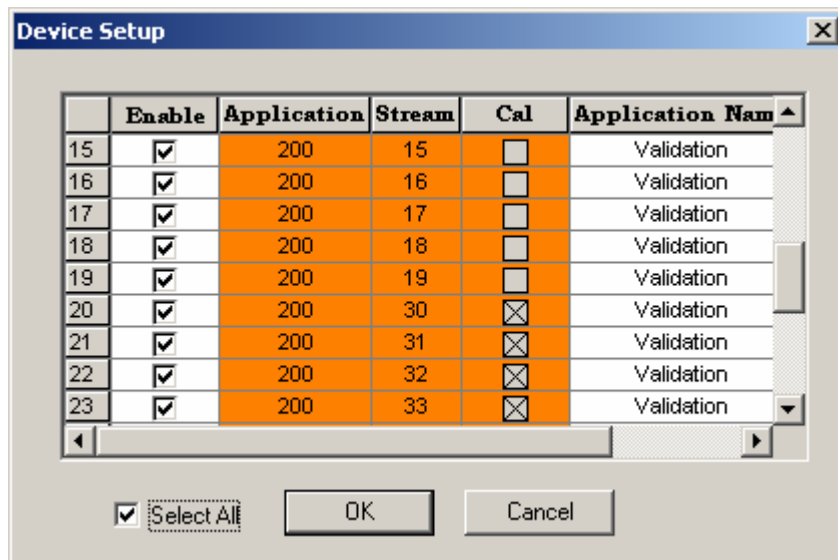


Figure 16

Notice that the Device Setup dialog box displays the available Applications ID’s, Stream ID’s, and for Maxum only whether the stream is a normal process stream or a Calibration stream and the Application name.

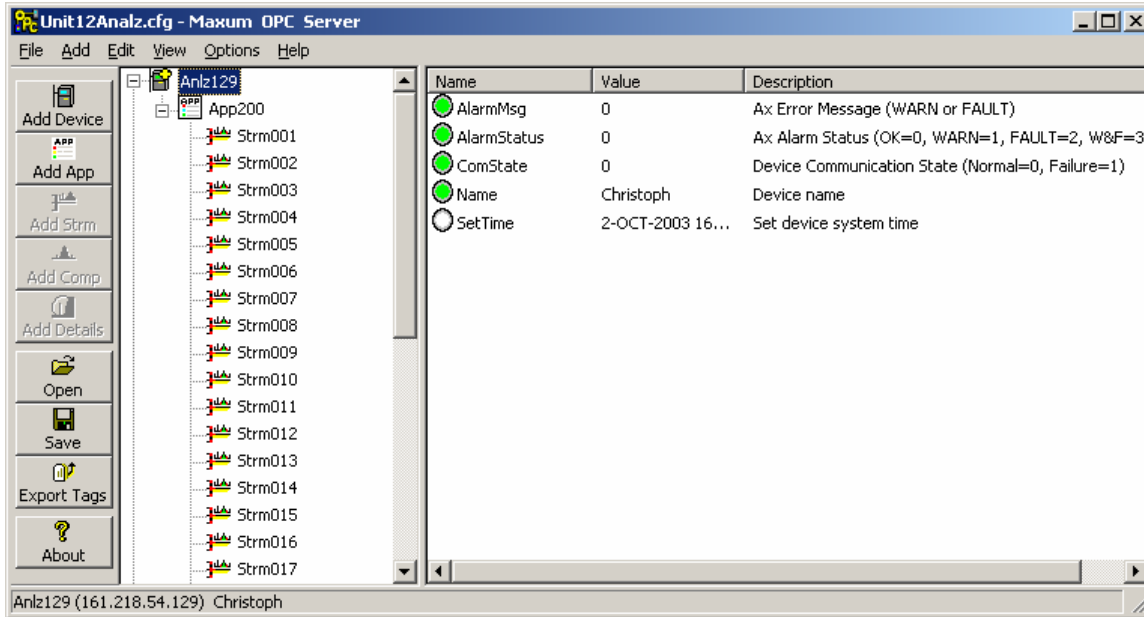


Figure 17

The configuration program automatically adds the Device Folder in the Device pane and labels the device with either “Anlzxxx” or “NAUxxx”. Where “xxx” is replaced with the device’s logical ID. In the example in Figure 4 its Anlz129. The following device level tags are automatically added to the Tag View pane under the device: AlarmMsg, AlarmStatus, ComState, Name, and SetTime. Located directly under the device folder in the Device pane is the Application folder(s). (See figure 18)

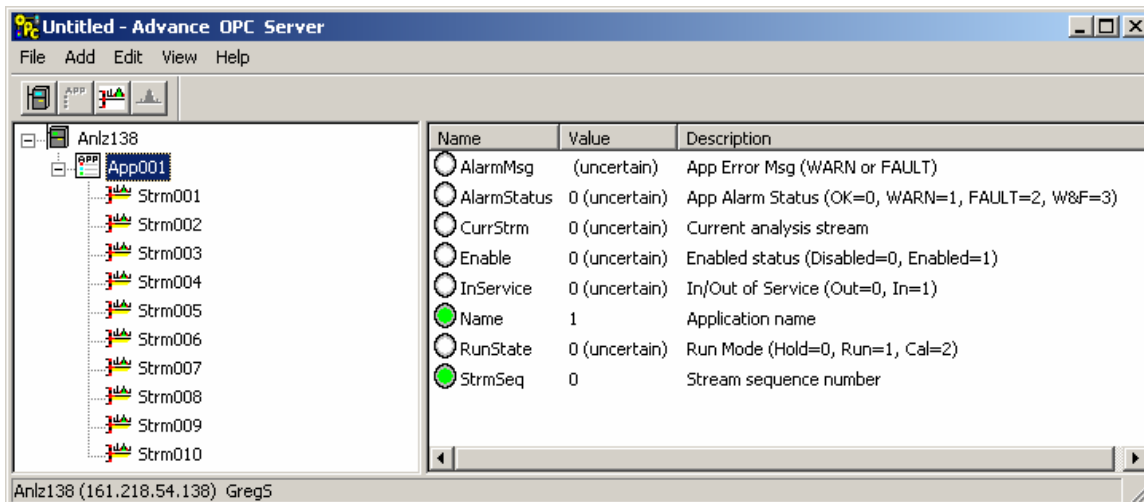


Figure 18

Each application folder is given the name “Appxxx”. Where “xxx” is replaced with the Application number. In the example in Figure 5 its App001. The following Application level tags were automatically added to the Tab View pane: AlarmMsg, AlarmStatus, CurrStrm, Enable, InService, Name, RunState, StrmSeq. Located directly under the Application folder in the Device pane is the Stream folder(s). (See figure 19)

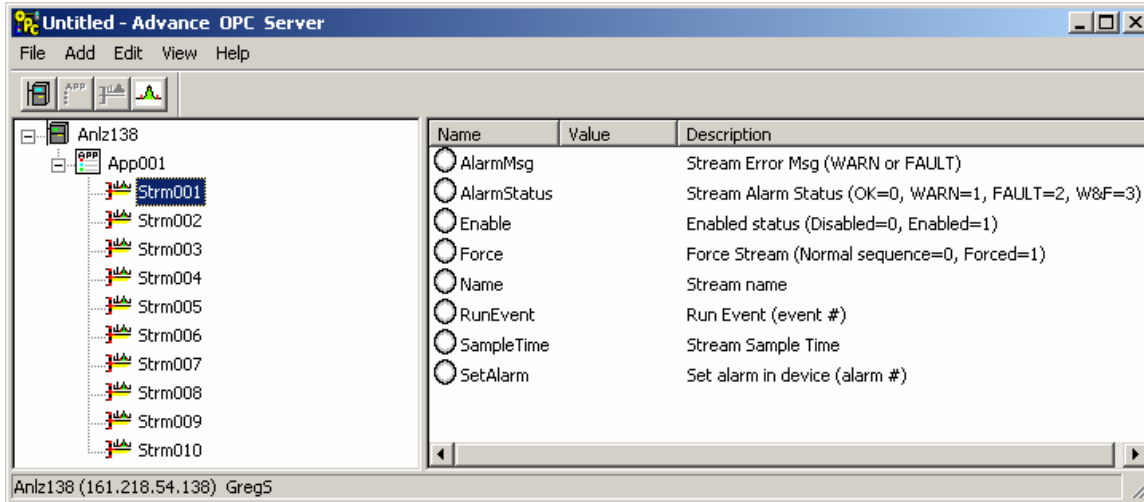


Figure 19

Each Stream folder is given the name “*Strmxxx*”. Where “xxx” is replaced with the Stream number. In the example in Figure 6 its Strm001. The following Stream level tags were automatically added to the Tab View pane: AlarmMsg, AlarmStatus, Enable, Force, Name, RunEvent, SampleTime, SetAlarm.

4. Notice that there are no stream components or results tags shown under the Stream folders. To add the stream dependent Components select “Add” from the main menu then “New Component” or click the left mouse button over the desired stream in the Device pane then right click to display the pop-up menu and select “New Component(s)” or select the Add “New Component(s)” button from the tool bar. The AOS sends a query to the device to determine the available components for the specified stream.

IMPORTANT – For Maxum and Maxum Edition II devices the components are only displayed in the Component Selection dialog box that have a sequence number defined in the Trtval attribute of the Result table.

Select the specific Components by clicking on the appropriate checkbox under the *Enable* column or select all of the stream components by clicking the left mouse button on the *Select All* checkbox. Then select OK to create the Component folder(s) and associated Tags. (See Figure 20)

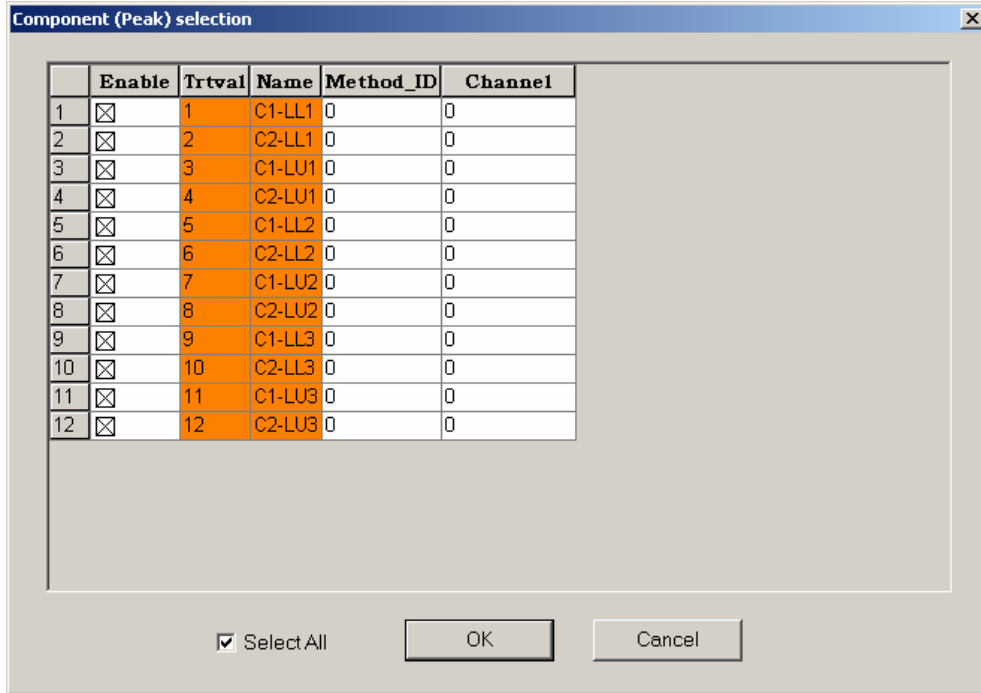


Figure 20

Each Component is given its own folder under its associated stream. Each component folder is given the name “Compxxx”. Where “xxx” is replaced with the Component number(as defined in TrtVal). In the selected example in Figure 18 its Comp001. The following Component level tags were automatically added to the Tab View pane: Name, and Value. (see Figure 20)

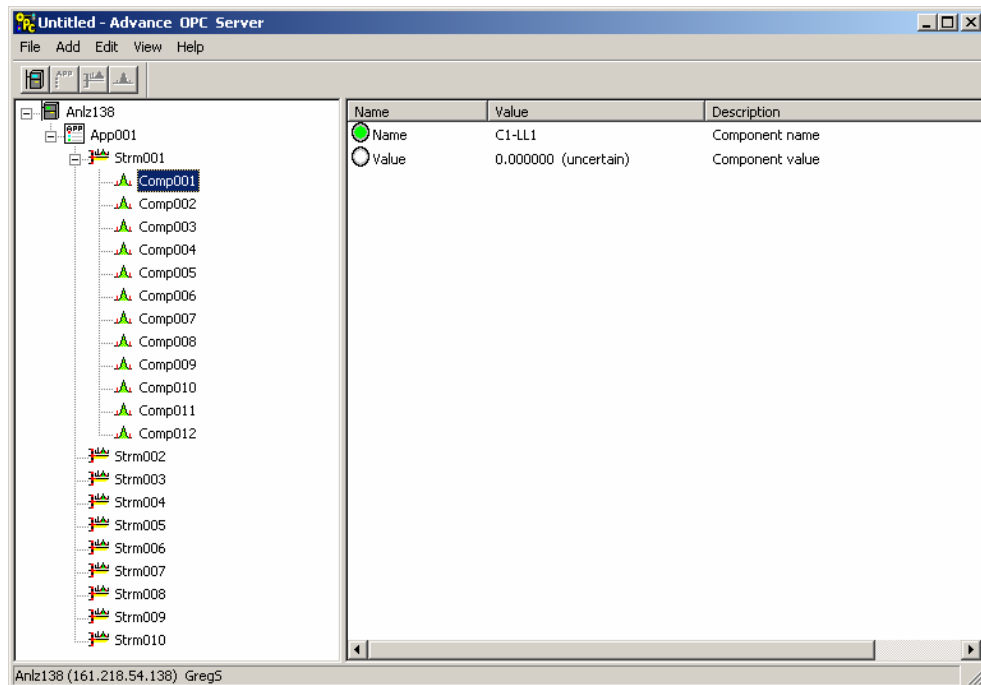


Figure 20

5. Save the Tag configuration file by selecting “File”, “Save As” from the Main Menu. (see Figure 21)

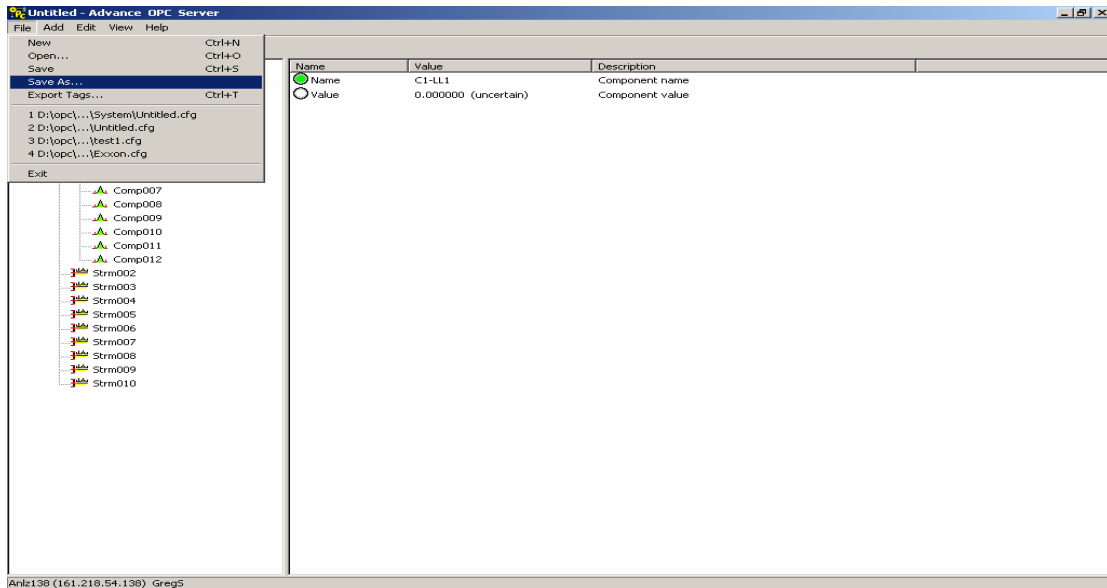


Figure 21

This results in the MOS displaying the “Save As” dialog box. Specify the desired configuration file name under the “File name:” edit box and select “Save”. This will save the Configuration file under the specified directory. This file will automatically load and run the next time the AOS program is started. (see Figure 22 & 23)

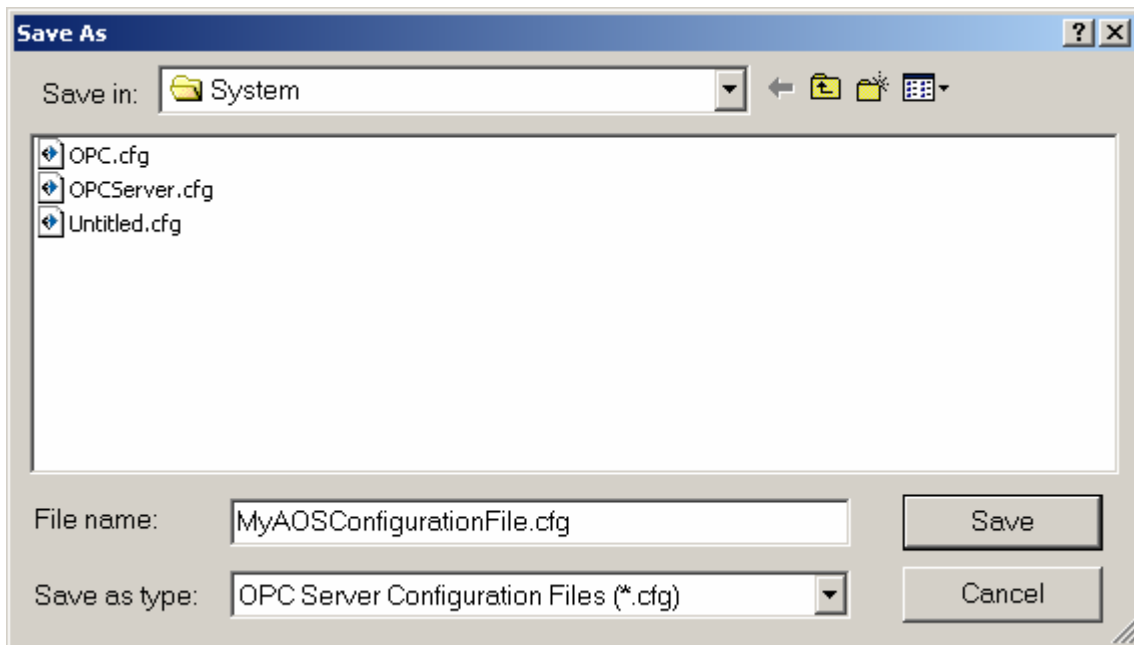


Figure 22

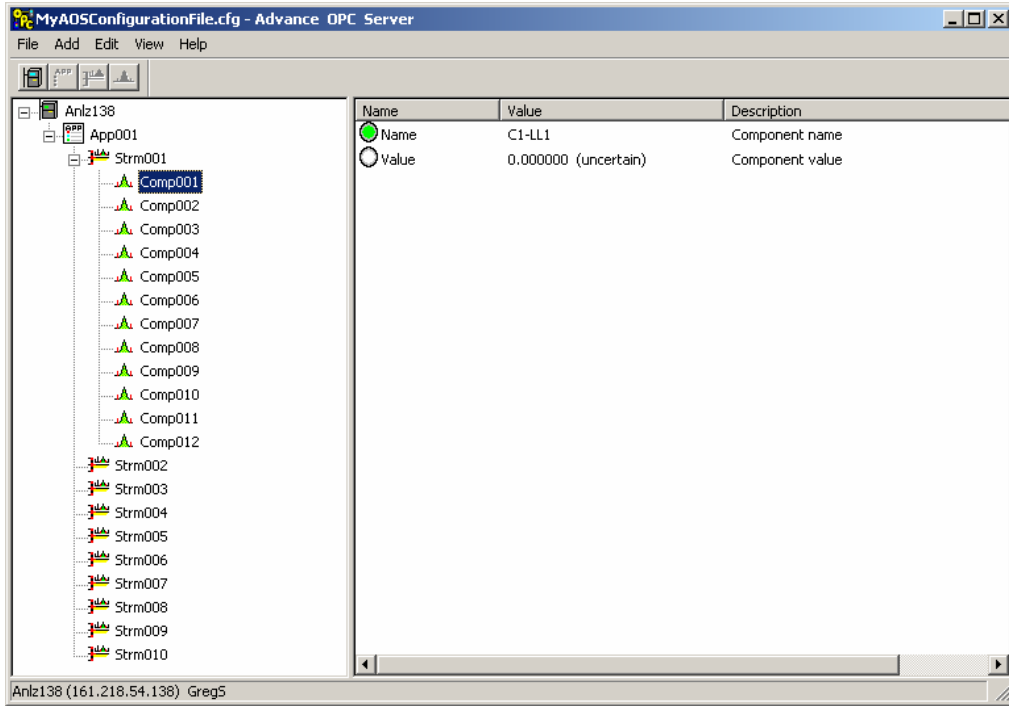


Figure 23

4. Using Maxum OPC Server

4.1. User Interface

The MOS window consists of two panels:

- 1.) The Device Group pane: used to display the top-level device (analyzer) and associated groups.
- 2.) The Tag View pane: used to display the tag name, value, and description.

The Title bar located at the top of the screen displays the open MOS tag configuration file name. or untitled if no configuration file exists or is open.

The Status bar located at the bottom of the screen displays the selected device logical ID, IP address, and device name.

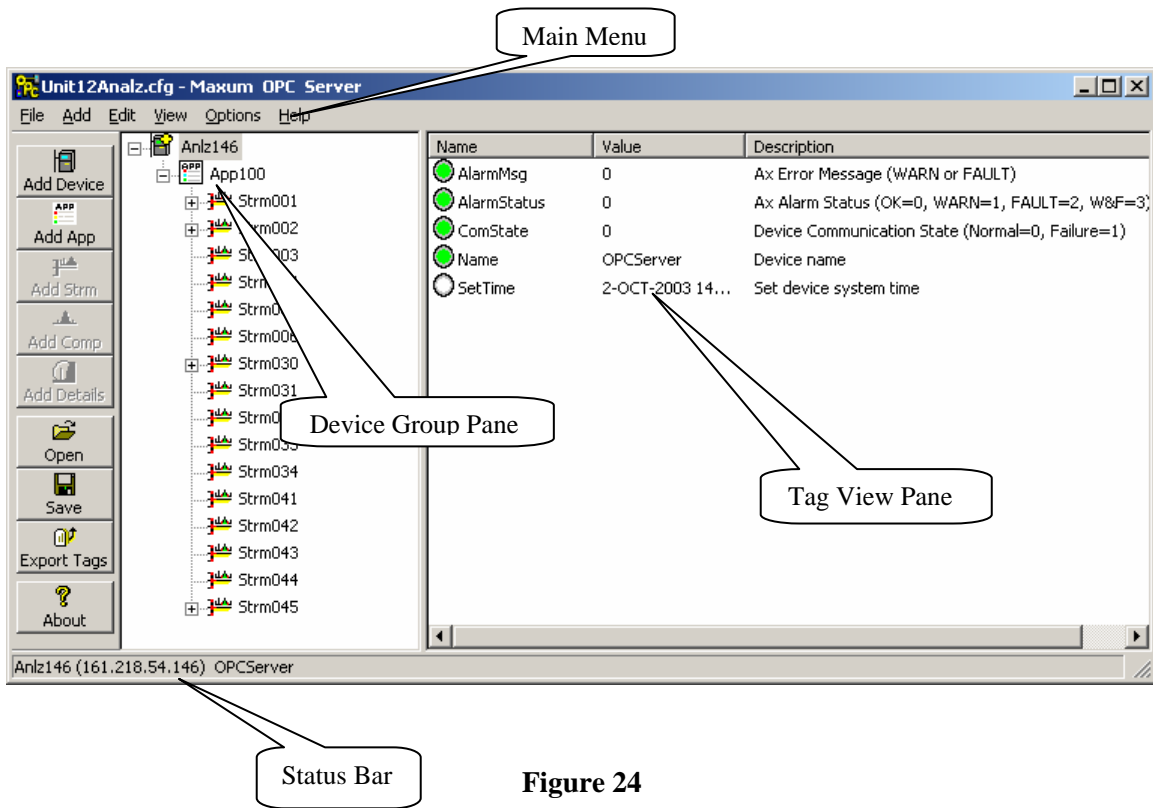


Figure 24

4.2. Main Menu

The Main menu located towards the top of the MOS screen consists of six items:
File, Add, Edit, View, Options, Help.

4.2.1. File Menu

The *File* menu offers the following choices:

- ◆ *New*: Creates a new tag configuration *.cfg file
- ◆ *Open*: Loads a configuration from a *.cfg file. See section 3.5 for a description of the configuration file format. The current MOS configuration is erased.
- ◆ *Save*: Saves the MOS configuration to a *.cfg file.
IMPORTANT: Save also sets the default startup file.
- ◆ *Save As*: Saves the MOS configuration to a *.cfg file after prompting for a file name.
- ◆ *Export Tags*: Saves the MOS device, group and tag names to a CSV file after prompting for a file name.
- ◆ *Exit*: Closes the Maxum OPC Server application.

4.2.2. Add Menu

The *Add* menu offers the following choices:

- ◆ *New Device*: Creates a new device group(Anlzxxx, Nauxxx).
- ◆ *New Application*: Creates a new application group(Appxxx) under the selected Device group.
- ◆ *New Stream*: Creates a new stream group(Strmxxx) under the selected Application group.
- ◆ *New Component*: Creates a new component group(Compxxx) under the selected Stream group.
- ◆ *Component Details*: Adds peak and group component details under the selected component.

4.2.3. Edit Menu

The *Edit* menu offers the following choices:

- ◆ *Delete*: Deletes the selected group.
IMPORTANT: The Delete menu item is grayed out when in the monitor mode. To delete a Device, Application, Stream, Component, or OPC Item de-select the Monitor command located under the View menu. This will enable the Delete command.
- ◆ *Properties*: Displays the Device, group, and tag properties.

4.2.4. View Menu

- ◆ *Monitor*: Selects or deselects the Monitor mode. When monitor mode is active the tag values are updated in the Tag View pane.
IMPORTANT: The Delete menu item is grayed out when in the monitor mode. To delete a Device, Application, Stream, Component, or OPC Item de-select the Monitor command located under the View menu. This will enable the Delete command.
- ◆ *Status Bar*: Shows or hides the Status bar. The Status bar displays the active device logical ID, IP address, and device name.

4.2.5. Options Menu

- ◆ *Short Tag Names*: The Short Tag Names command is used to limit the Tag name length. This feature is useful for legacy OPC clients that require short tag names typically 35 characters or less. When the Short Tag Names feature is enabled all new tags are created with abbreviated tag name fields. Any

existing tags are not affected by this command. If you wish to change an existing tag to a short tag name you must delete the old tag and re-create the tag with the Short Tag Name feature enabled.

4.2.6. Help Menu

- ◆ *Help Topics...*: Displays the MOS help file. Very useful information!
- ◆ *About*: Displays the MOS Version number, Serial number, License type, and Support contact information.

4.3. Monitoring

The Maxum OPC Server provides a built-in OPC Client. The Client is useful for both setup and troubleshooting purposes. The Client is enabled by selecting “View” then “Monitor” from the Main Menu. In Monitor mode, the tag values are displayed in the tag view, updating periodically. Data appears in the appropriate format for its data type. Tag data quality is shown under the name description and is shown iconically as a Green LED for “GOOD” quality, a White or transparent LED for “UNCERTAIN” quality, and a Red LED for “BAD”, “DEVICE FAILURE”, or “COMM FAILURE”. (see Figure 25)

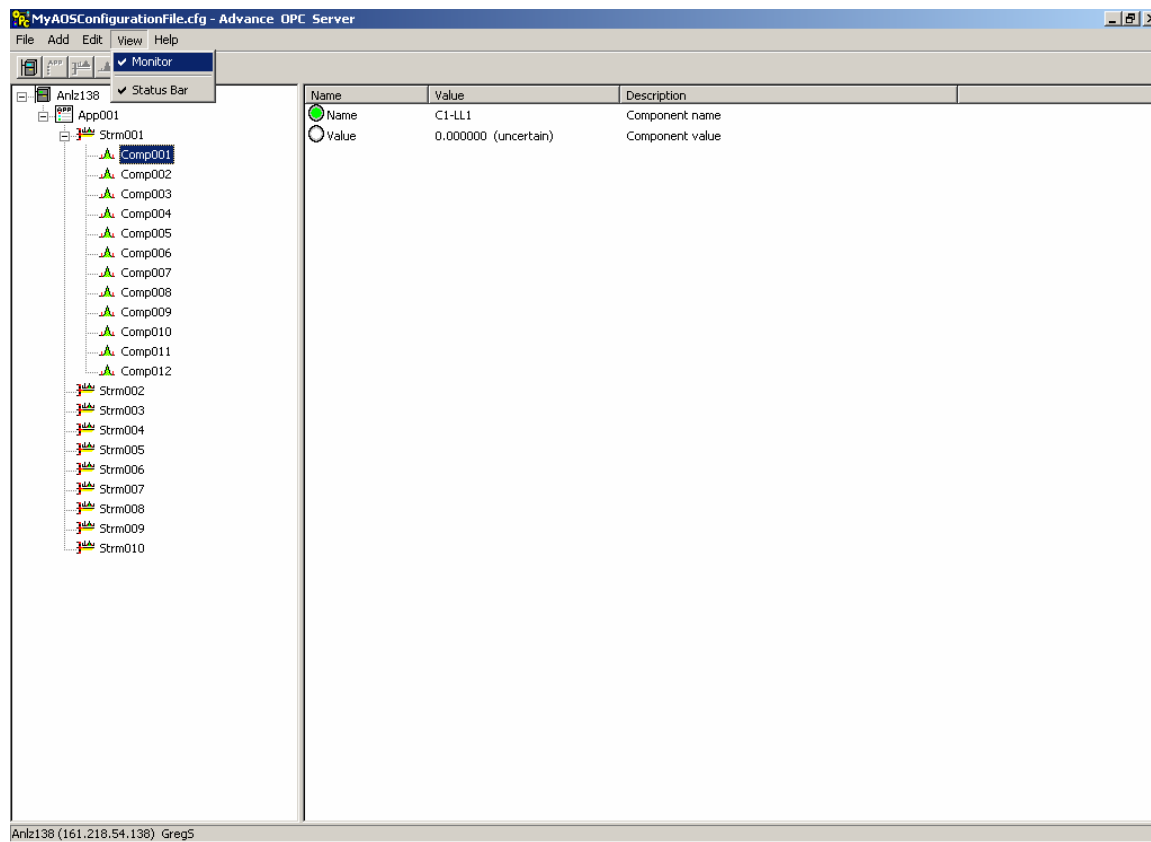


Figure 25

4.4. Editing

There are two selections under the “Edit” item on the Main Menu. “Delete” and “Properties”. “Delete” can only be selected when the AOS is not in the “Monitor” mode. (See Figure-22)

4.4.1. Deleting

Devices, Applications, Streams, Components, and individual data points or Tags can be deleted from the AOS configuration. To delete an item or folder, select the item by right clicking the mouse on the item, in either the Device pane or the Tag View pane, then select “Delete” from the “Edit” menu or push the “Del” key or right click the mouse to display the pop-up menu and select the “Delete” item from the menu. When you delete a folder (Device, Application, Stream, Component) from the Device pane you automatically delete all the folders and items that are directly underneath the item. For example if you delete a device you delete all the Application, Streams, Components, and all the associated tags. If you delete a tag from the Tag View pane only the single item is deleted from the configuration.

IMPORTANT –After deleting any item remember to save the Configuration file or the item will return the next time the MOS is re-started. (see Figure 19)

4.4.2. Properties

Device and Tag properties can be displayed with the “Properties” command. See Figure 26, 27 and 28 for an example of the Device and Tag properties dialog box.

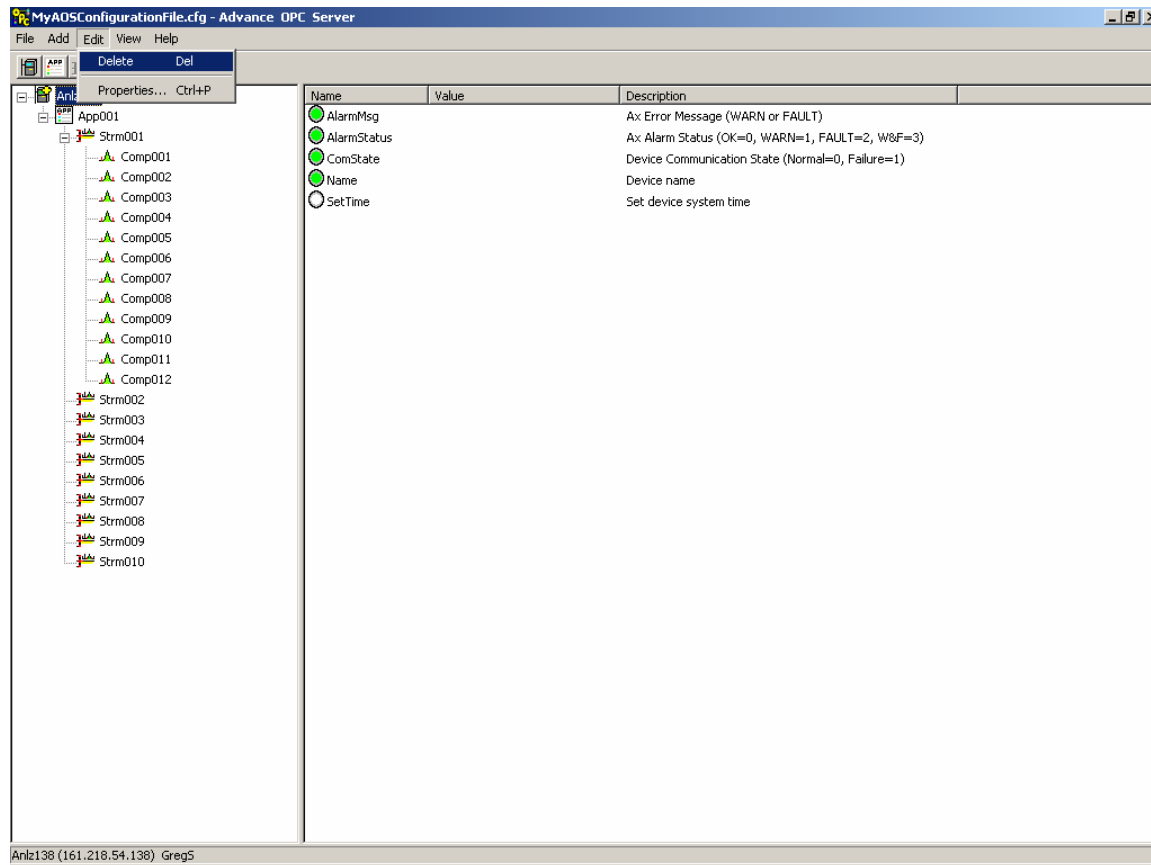


Figure 26

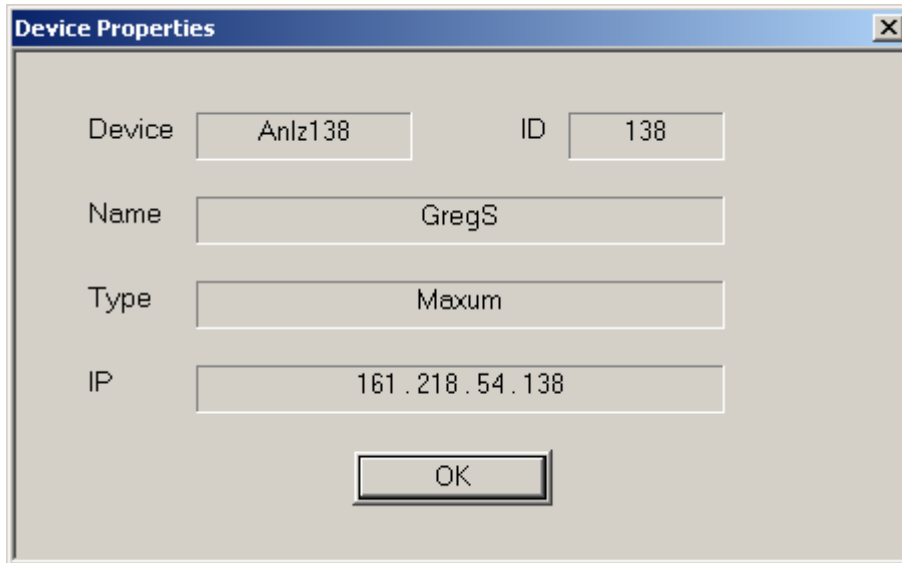


Figure 27

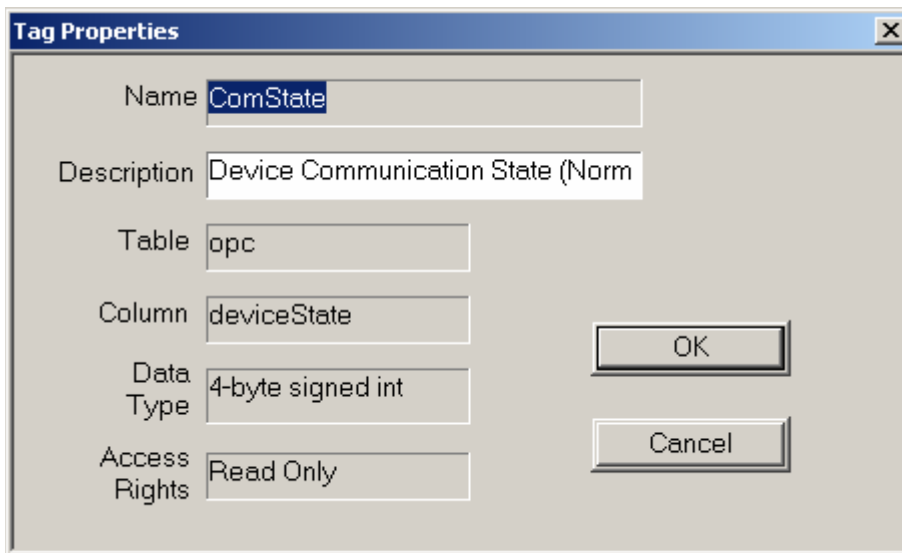


Figure 28

4.5. Export tags

MOS tags can be Exported to a *.CSV (Comma Separated Value) file for input to Microsoft Excel or other similar programs. To Export the OPC Server tags select “*Export Tags...*” from the Main Menu.

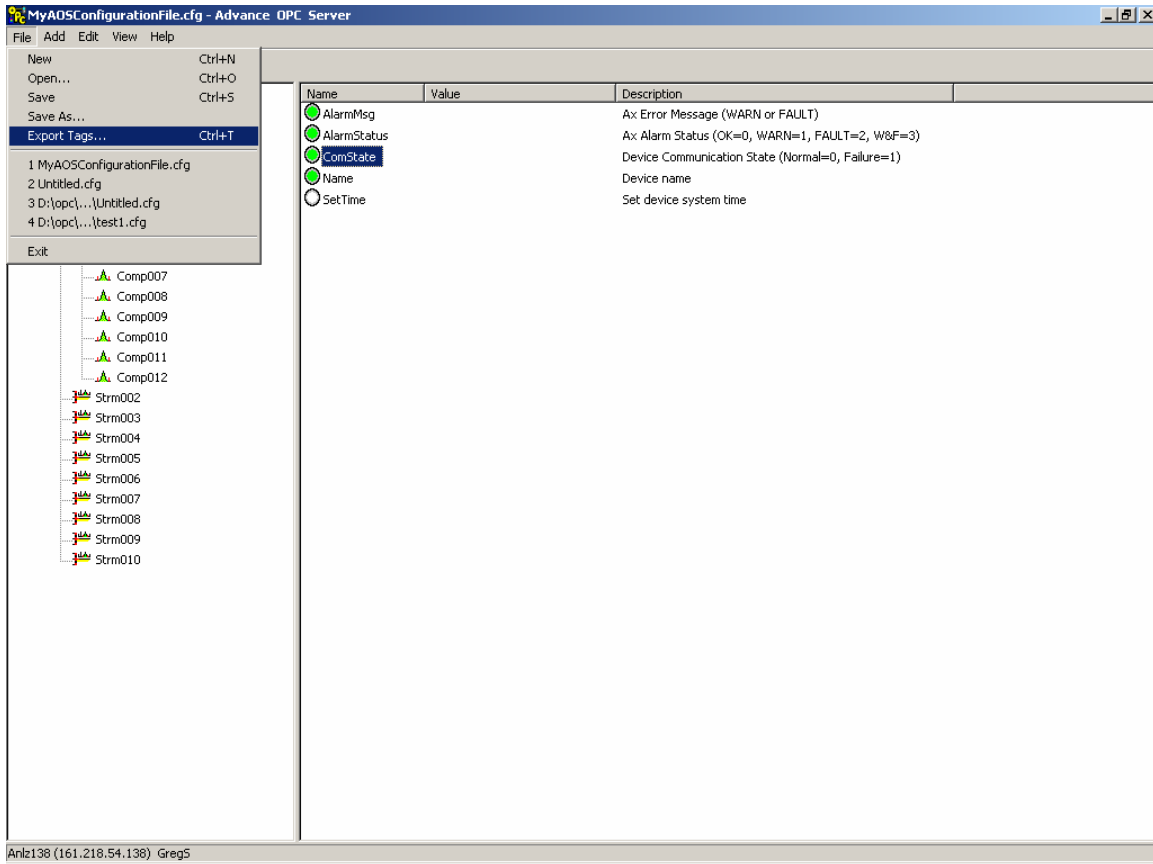


Figure 29

This results in the Export Tags dialog box being displayed.

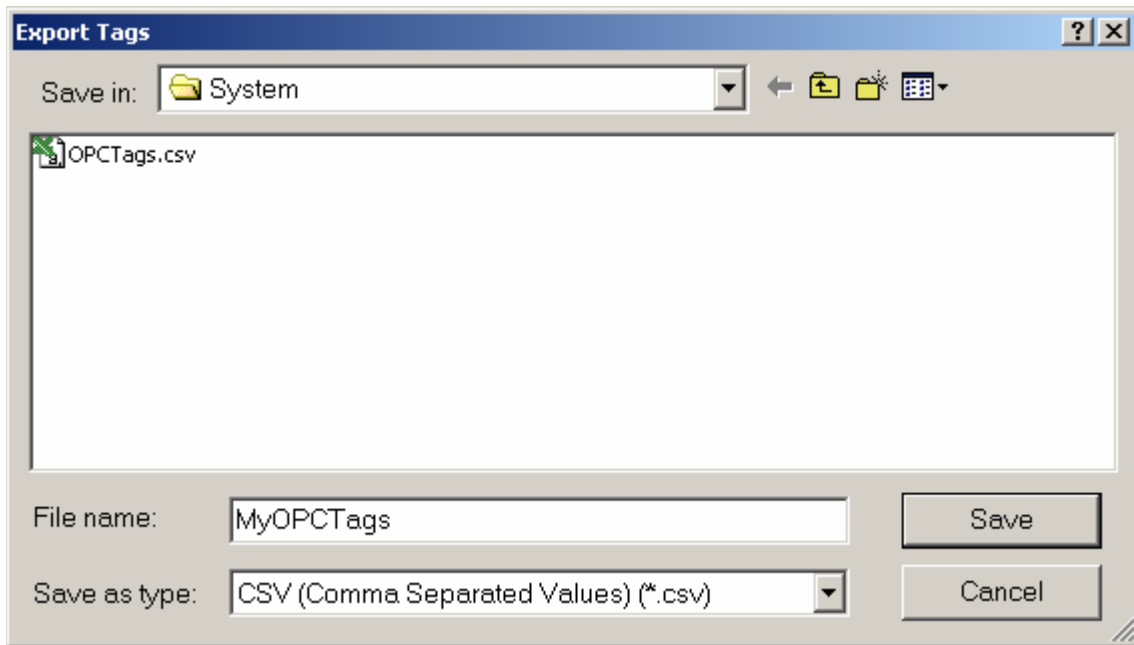


Figure 30

Specify a File name and a directory if you want the file to be saved in a directory other than the default directory. Then select “Save” and the file will be saved as a *.csv file. The Tag export file has the following format:

TagName	DataType	AccessRights
Anlz138.ComState	3	1
Anlz138.Name	8	1
Anlz138.SetTime	7	3
Anlz138.App001.CurrStrm	3	1
Anlz138.App001.Name	8	1
Anlz138.App001.RunState	3	1
Anlz138.App001.Strm001.AlarmMsg	8	1
Anlz138.App001.Strm001.AlarmStatus	3	1

5. OPERATIONAL OVERVIEW

5.1. Device Connection

OPC Server uses TCP/IP to communicate with Advance Maxum, Maxum Edition II, NAU, MicroSam, and Advance Optichrom analyzers. When a new device is added to OPC Server a connect message is sent to the remote device. If the connection is successful, the server sends a query to determine the device firmware version number. The firmware version must be equal to or greater than 4.0 for the connection process to continue. Once the server has verified the device version number the server sets up (registers) three Active queries (subscribes) for a Maxum device or two Active queries for an Optichrom device.(See tables below) This completes the connection process.

5.1.1. Maxum/NAU ACTIVE QUERIES:

Table Name	Table Attribute(s)
OPC	lastreceived
OPC	enable, runstatus
Application	enable, inservice

5.1.2. OPTICHROM ACTIVE QUERIES:

Table Name	Table Attributes(s)
OPC	lastreceived
OPC	enable, runstatus

5.2. Device Updates

At the end of each analytical cycle the analysis results are updated in the device (analyzer) database. The *lastreceived* attribute is updated to reflect the time the analysis cycle completed. Any change to the *lastreceived* attribute activates an Active query forcing the event notification to be sent to OPC Server. OPC Server responds with a query for the analysis results. The Server overwrites the previous analysis cycle results and updates any interested OPC Clients with the new data.

5.3. OPC Client

A client is defined as a requester of services and a server is defined as the provider of services. A single machine can be both a client and a server depending on the software configuration.

An OPC client is any OPC compatible application that wants to use the information stored in Maxum OPC Server. A typical client for Maxum OPC Server is a DCS (Distributed Control System) or SQC program that supports an OPC interface.

OPC clients connect to OPC Server by creating an OPC Group object and specifying the OPC Items (tags) to be included in the group. The client registers and subscribes to server data by creating an advise sink which receives data change notifications from the server. OPC is change-driven. Data is only sent when there is a change in value.

OPC Groups have several properties that must be specified when the group is created in the OPC Server. One property of an OPC Group is the update rate. The update rate is the rate at which data changes should be evaluated and sent to the client.

5.3.1. OPC and the Component Object Model

COM is an “industry-standard” software architecture supported by Microsoft and many other companies. The problem that COM solves is that there’s no standard way for Windows program modules(applications) to communicate with each other. COM is a protocol that connects one software module(application) with another and then drops out of the picture.

COM is a binary standard that specifies how objects interact with each other. A COM object is accessed exclusively through one or more sets of interfaces. An interface is a set of functions or methods. The COM standard not only specifies the binary object standard, it also defines a series of standard interfaces that provide common functionality. The OPC-DA specification defines several COM objects and their associated interfaces. These objects and interfaces provide a standard for real-time information exchange between software applications and process hardware.

The Maxum OPC Server uses Microsoft COM/DCOM/OLE technology to communicate with clients.

5.3.2. The Registry

OPC clients locate OPC Servers on the local or remote machine through a set of entries in the Windows Registry. Chapter 6 of the OPC Specification discusses the registry entries necessary for an OPC server. A COM class is identified through a CLSID (ClassID). The operating system maintains a database of all registered CLSIDs in the Registry. The registry entries for a server’s CLSID specify the location of the server, its type (local , remote, in-process), threading model, etc... Maxum OPC Server is self-registering. The first time OPC Server is executed a set of entries is installed in the Registry.

Registration entries are made under

HKEY_LOCAL_MACHINE\CLASSES\Siemens.MaxumOPCServer.1. Referred to as the ProgID. A subkey located under this key contains the CLSID for Maxum OPC Server. Clients use the registration information to find the server path.

5.4. OPC Server Tags

5.4.1. Maxum, Maxum Edition II, NAU, MicroSam TAGS

TAG LEVEL	TAG NAME	VARIANT DATATYPE	Access Type	DATABASE LOCATION
Device	AlarmMsg	VT_BSTR	Read Only	The AlarmMsg Tag comes from the OPC table “sysmsg” attribute.
Device	AlarmStatus	VT_I4	Read Only	The AlarmStatus Tag comes from the OPC table “sysalmstatus” attribute
Device	ComState	VT_I4	Read Only	The ComState Tag is set by the AOS based on the connection state between the AOS and the device
Device	Name	VT_BSTR	Read Only	The Name Tag comes from the System_Control table “name” attribute.
Device	SetTime	VT_DATE	Write Only	The SetTime Tag comes from the OPC table “settime” attribute.
Application	AlarmMsg	VT_BSTR	Read Only	The AlarmMsg Tag comes from the Application table "app_status_msg" attribute.
Application	AlarmStatus	VT_I4	Read Only	The AlarmStatus Tag comes from the Application table "app_status" attribute.
Application	CurrStrm	VT_I4	Read Only	The CurrStrm Tag comes from the OPC table "nextstream" attribute.
Application	Enable	VT_BOOL	Read & Write	The Enable Tag comes from the Application table “app_status” attribute.
Application	InService	VT_I4	Read Only	The InService Tag comes from the Application table "inservice" attribute
Application	Name	VT_BSTR	Read Only	The Name Tag comes from the Application table "application_name" attribute.
Application	RunState	VT_I4	Read & Write	The RunState Tag comes from the OPC table "runstatus" attribute
Application	StrmSeq	VT_I4	Read & Write	The StrmSeq Tag comes from the Application table "active_sequence" attribute.
Stream	AlarmMsg	VT_BSTR	Read	The AlarmMsg Tag comes from the OPC table "alarmmsg" attribute.
Stream	AlarmStatus	VT_I4	Read	The AlarmStatus Tag comes from the OPC table "alarmstatus" attribute.
Stream	Enable	VT_BOOL	Read & Write	The Enable Tag comes from the OPC table "enable" attribute.
Stream	Force	VT_BOOL	Read &	The Force Tag comes from the

Stream	Name	VT_BSTR	Write Read	OPC table "force" attribute. The Name Tag comes from the OPC table "streamname" attribute.
Stream	RunEvent	VT_I4	Read & Write	The RunEvent Tag comes from the OPC table "runevent" attribute.
Stream	SampleTime	VT_BSTR	Read	The SampleTime Tag comes from the OPC table "lastreceived" attribute.
Stream	SetAlarm	VT_I4	Read & Write	The SetAlarm Tag comes from the OPC table "setalarm" attribute.
Component	Name	VT_BSTR	Read	The Name Tag comes from the Result table "result_name" attribute.
Component	Value	VT_R4	Read	The Value Tag comes from the OPC table "results" attribute.
Component	PkArea	VT_R8	Read	The PkArea Tag comes from the ezchrom_result_chrompeak table "corrected_area" attribute.
Component	PkArea%	VT_R8	Read	The PkArea% Tag comes from the ezchrom_result_chrompeak table "area_percent" attribute.
Component	PkAsym	VT_R8	Read	The PkAsym Tag comes from the ezchrom_result_chrompeak table "asymmetry" attribute.
Component	PkAsym10	VT_R8	Read	The PkAsym10 Tag comes from the ezchrom_result_chrompeak table "asym10" attribute.
Component	PkHeight	VT_R8	Read	The PkHeight Tag comes from the ezchrom_result_chrompeak table "peak_height" attribute.
Component	PkHeight%	VT_R8	Read	The PkHeight% Tag comes from the ezchrom_result_chrompeak table "height_percent" attribute.
Component	PkNoise	VT_R8	Read	The PkNoise Tag comes from the ezchrom_result_chrompeak table "peak_noise" attribute.
Component	PkResolution	VT_R8	Read	The PkResolution Tag comes from the ezchrom_result_chrompeak table "resolution" attribute.
Component	PkRetTime	VT_R8	Read	The PkRetTime Tag comes from the ezchrom_result_chrompeak table "retention_time" attribute.
Component	PkSignaltoNoise	VT_R8	Read	The PkSignaltoNoise Tag comes from the ezchrom_result_chrompeak table "peak_signal_to_noise" attribute.

Component	PkStartFlg	VT_BSTR	Read	The PkStartFlg Tag comes from the ezchrom_result_chrompeak table “start_flag” attribute.
Component	PkStartTime	VT_R8	Read	The PkStartTime Tag comes from the ezchrom_result_chrompeak table “front_time” attribute.
Component	PkStopFlg	VT_R8	Read	The PkStopFlg Tag comes from the ezchrom_result_chrompeak table “stop_flag” attribute.
Component	PkStopTime	VT_R8	Read	The PkStopTime Tag comes from the ezchrom_result_chrompeak table “back_time” attribute.
Component	PkType	VT_R8	Read	The PkType Tag comes from the ezchrom_result_chrompeak table “peak_type” attribute.
Component	PkUSPWidth	VT_R8	Read	The PkUSPWidth Tag comes from the ezchrom_result_chrompeak table “usp_peak_width” attribute.
Component	PkWidth	VT_R8	Read	The PkWidth Tag comes from the ezchrom_result_chrompeak table “peak_base_width” attribute.
Component	PkWidth10	VT_R8	Read	The PkWidth10Tag comes from the ezchrom_result_chrompeak table “peak_width10” attribute.
Component	PkWidth5%	VT_R8	Read	The PkWidth5%comes from the ezchrom_result_chrompeak table “peak_width5” attribute.
Component	PkWidth50	VT_R8	Read	The PkWidth50 comes from the ezchrom_result_chrompeak table “peak_width50” attribute.
Component	PkTheorPlates	VT_R8	Read	The PkTheorPlates comes from the ezchrom_result_chrompeak table “theor_plates” attribute.
Component	GrpArea	VT_R8	Read	The GrpArea comes from the ezchrom_result_group table “corrected_area” attribute.
Component	GrpArea%	VT_R8	Read	The GrpArea% comes from the ezchrom_result_group table “area_percent” attribute.
Component	GrpHeight	VT_R8	Read	The GrpHeight comes from the ezchrom_result_group table “group_height” attribute.
Component	RespFactL(0-9)	VT_R8	Read	The RespFactL(0-9) comes from the ezchrom_result_group table “response_factor” attribute.
Component	RespFactL(0-9)	VT_R8	Read	The RespFactL(0-9) comes from the calibration_level table “response factor” attribute.

5.4.2. Maxum Tag Descriptions

Anlzxxx.AlarmMsg –

The AlarmMsg tag is a read-only, string value that displays the highest priority device level Alarm message.

Anlzxxx.AlarmStatus –

The AlarmStatus tag is a read-only, 4-byte integer value that displays the device alarm status.

0 = OK, Normal

1 = WARNING ALARM

2 = FAULT ALARM

3 = WARNING & FAULT ALARM

Anlzxxx.ComState –

The ComState tag is a read-only, 4-byte integer value that displays the current status of the communication link between the AOS and the device.

0 = OK, Normal

1 = COMMUNICATION FAILURE between device and server

Anlzxxx.Name –

The Name tag is a read-only, string that contains the device name.

Anlzxxx.SetTime –

SetTime is a Write only, Ole Date tag that is used to set the device and/or system time. A Maxum analyzer or NAU can be configured as a Time server. The SetTime tag is used to update the time/date in the designated Time server. The other devices in the system must be configured with the IP address of the designated Time Server. Maxum devices automatically synchronize their time base once a day at 12:20am.

write format: Time may be entered in 12 or 24 hr format

06/11/2003 01:28 pm or 6-11-2003 13:28

Appxxx.AlarmMsg –

AlarmMsg is a read-only, string tag that contains the current application error message.

AlarmMsg always contains the highest priority FAULT or WARNING alarm message.

Displays a NULL string for no alarm.

Appxxx.AlarmStatus –

AlarmStatus is a read-only, 4-byte integer that contains the current application alarm status.

0 = OK, Normal

1 = WARNING ALARM

2 = FAULT ALARM

3 = WARNING & FAULT ALARM

Appxxx.CurrStrm –

CurrStrm is a read-only, 4-byte integer that contains the number of the stream currently being analyzed by the application.

Appxxx.Enable –

Enable is a read/write, Boolean that is used to enable or disable this application.

0 = This application DISABLED

1 = This application ENABLED

Write format: 0 or 1

Appxxx.InService –

InService is a read/write, Boolean that contains the current service state for this application.

0 = This application is OUT OF SERVICE

1 = This application is IN SERVICE

Write format: 0 or 1

Appxxx.Name –

Name is a read-only, string that contains the name of this application.

Appxxx.RunState –

RunState is a read/write, 4-byte integer that sets the state of this application.

0 = This application is in HOLD

1 = This application is in the RUN state

2 = This application is in the CALibration state

Other values are not allowed.

Write format: range 0 - 2

0 = Sets application to HOLD state

1 = Sets application to RUN state

2 = Sets application to CAL state

Appxxx.StrmSeq –

StrmSeq is a read/write, 4-byte integer that sets the application stream sequence.

Write format:

positive integer greater than zero

Strmxxx.AlarmMsg –

AlarmMsg is a read-only, string tag that contains the highest priority WARNING or FAULT error message for this stream.

Strmxxx.AlarmStatus –

AlarmStatus is a read-only 4-byte integer tag that contains the current stream alarm status.

0 = OK, Normal

1 = WARNING ALARM this stream

2 = FAULT ALARM this stream

3 = WARNING & FAULT ALARM this stream

Strmxxx.Enable –

Enable is a read/write, Boolean tag used to set this stream ENABLE/DISABLE state.

0 = This stream DISABLED

1 = This steam ENABLED

Write format: 0 or 1

Strmxxx.Force –

Force is a read/write, Boolean tag used to force this stream to run repeatedly.

0 = Normal Stream Sequencing

1 = Force this stream to run next

Write format: 0 or 1

Strmxxx.Name –

Name is a read-only, string tag that contains the name of this stream.

Strmxxx.RunEvent –

RunEvent is a read/write, 4-byte integer used to run a stream dependent event. Example to run EVT32 in the analyzer, write a 32 to this tag.

Write format: Positive integer in the range 1- 32535

Strmxxx.SampleTime –

SampleTime is a read-only, Date tag that contains the time the sample was injected for the associated components (results) on this stream..

Strmxxx.SetAlarm –

SetAlarm is a read/write, 4-byte integer that is used to set an alarm in the device.

Write format: Positive integer in the range 1- 32,535

Example, to force an Alarm 54 in the device write a 54 to this tag.

Compxxx.Name –

Name is a read-only, string tag that contains the name of this component.

Compxxx.Value –

Value is a read-only, 4-byte float(floating-point) tag that contains the value of the component.

***** **The following tags were added in Version 2.0** *****

- Compxxx.**PkArea** –
PkArea is a read-only, 4-byte real tag that contains the corrected peak area for this component.
- Compxxx.**PkArea%** –
PkArea% is a read-only, 4-byte real tag that contains the percent area for this component.
- Compxxx.**PkAsym** –
PkAsym is a read-only, 4-byte real tag that contains the peak asymmetry for this component.
- Compxxx.**PkAsym10**–
PkAsym10 is a read-only, 4-byte real tag that contains the Asymmetry at 10% peak height for this component.
- Compxxx.**PkHeight**–
PkHeight is a read-only, 4-byte real tag that contains the maximum peak height for this component.
- Compxxx.**PkHeight%**–
PkHeight% is a read-only, 4-byte real tag that contains the percent height for this component.
- Compxxx.**PkNoise** –
PkNoise is a read-only, 4-byte real tag that contains the peak noise for this component.
- Compxxx.**PkResolution** –
PkResolution is a read-only, 4-byte real tag that contains the peak resolution relative to previous peak for this component.
- Compxxx.**PkRetTime** –
PkRetTime is a read-only, 4-byte real tag that contains the retention time at peak apex for this component.
- Compxxx.**PkSignaltoNoise** –
PkSignaltoNoise is a read-only, 4-byte real tag that contains the peak signal to noise for this component. (Peak height / rms noise)
- Compxxx.**PkStartFlg** –
PkStartFlg is a read-only, string tag that contains the peak start flag for this component.
- Compxxx.**PkStartTime** –
PkStartTime is a read-only, 4-byte real tag that contains the retention time at start of peak for this component.
- Compxxx.**PkStopFlg** –
PkStopFlg is a read-only, string tag that contains the peak stop flag for this component.
- Compxxx.**PkStopTime** –
PkStopTime is a read-only, 4-byte real tag that contains the retention time at end of peak for this component.
- Compxxx.**PkType** –
PkType is a read-only, 4-byte real tag that contains the peak type this component.
- Compxxx.**PkUSPWidth** –
PkUSPWidth is a read-only, 4-byte real tag that contains the peak USP width (U.S. Pharmacopeia) for this component.
- Compxxx.**PkWidth** –
PkWidth is a read-only, 4-byte real tag that contains the peak width at base for this component.
- Compxxx.**PkWidth10** –
PkWidth10 is a read-only, 4-byte real tag that contains the peak width at 10% height for this component.
- Compxxx.**PkWidth5%** –
PkWidth5% is a read-only, 4-byte real tag that contains the peak width at 5% height for this component.

Compxxx.**PkWidth50** –

PkWidth50 is a read-only, 4-byte real tag that contains the peak width at 50% height for this component.

Compxxx.**PkTheorPlates** –

PkTheorPlates is a read-only, 4-byte real tag that contains the Theoretical plates for this component.

Compxxx.**GrpAre** –

GrpArea is a read-only, 4-byte real tag that contains the corrected peak area (for the group) for this component.

Compxxx.**GrpArea%** –

GrpArea% is a read-only, 4-byte real tag that contains the peak area percent (for the group) for this component.

Compxxx.**GrpHeight** –

GrpHeight is a read-only, 4-byte real tag that contains the maximum peak height (for the group) for this component.

Compxxx.**RespFactL(0-9)** –

RespFactL(0-9) is a read-only, 4-byte real tag that contains the calibration level response factor for levels 0 – 9 if applicable for this component.

5.4.3. Advance Optichrom Tags

TAG LEVEL	TAG NAME	VARIANT DATATYPE	Access Type	DATABASE LOCATION
Device	ComState	VT_I4	Read Only	The ComState Tag is set by the AOS based on the connection state between the AOS and the NAU (device)
Device	Name	VT_BSTR	Read Only	The Name Tag for OPTICHRM is always Anlzxxx. Where xxx is the logical ID.
Device	SetTime	VT_DATE	Write Only	The SetTime Tag comes from the OPC table “settime” attribute.
Application	CurrStrm	VT_I4	Read Only	The CurrStrm Tag comes from the OPC table "nextstream" attribute.
Application	Name	VT_BSTR	Read Only	The Name Tag for OPTICHRM is always Appxxx. Where xxx is the application number.
Application	RunState	VT_I4	Read & Write	The RunState Tag comes from the OPC table "runstatus" attribute
Stream	AlarmMsg	VT_BSTR	Read	The AlarmMsg Tag comes from the OPC table "alarmmsg" attribute.
Stream	AlarmStatus	VT_I4	Read	The AlarmStatus Tag comes from the OPC table "alarmstatus" attribute.
Stream	Enable	VT_BOOL	Read & Write	The Enable Tag comes from the OPC table "enable" attribute.
Stream	Force	VT_BOOL	Read &	The Force Tag for comes from the

Stream	Name	VT_BSTR	Write Read	OPC table "force" attribute. The Name Tag for OPTICHROM is always Strmxxx. Where xxx is the stream ID.
Stream	RunEvent	VT_I4	Read & Write	The RunEvent Tag comes from the OPC table "runevent" attribute.
Stream	SampleTime	VT_BSTR	Read	The SampleTime Tag comes from the OPC table "lastreceived" attribute.
Component	Name	VT_BSTR	Read	The Name Tag for OPTICHROM is always Compxxx. Where xxx is the component ID.
Component	Value	VT_R4	Read	The Value Tag comes from the Result table "results" attribute.

5.4.4. Optichrom Tag Descriptions

Anlzxxx.ComState –

The ComState tag is a read-only, 4-byte integer value that displays the current status of the communication link between the AOS and the device.

0 = OK, Normal

1 = COMMUNICATION FAILURE

Anlzxxx.Name –

The Name tag is a read-only, string that contains text “Anlzxxx”. Where xxx is the analyzer’s logical ID.

Anlzxxx.SetTime –

SetTime is a read/write, Ole Date tag that is used to set the device and/or system time. Writing the SetTime tag for an Optichrom device results in a Optichrom TOD(Time of Day) broadcast. This will synchronize the date/time on all the Optichrom devices connected to the network.

write format: Time may be entered in 12 or 24 hr format

06/11/2003 01:28 pm or 6-11-2003 13:28

Appxxx.CurrStrm –

CurrStrm is a read-only, 4-byte integer that contains the number of the stream currently being analyzed by this application.

Appxxx.Name –

Name is a read-only, string that contains the text “Appxxx”. Where xxx is the application ID. For Optichrom its always a one.

Appxxx.RunState –

RunState is a read/write, 4-byte integer that sets the state of this application.

0 = This application is in HOLD

1 = This application is in the RUN state

2 = This application is in the CALibration state

4 = Optichrom loss of communication with OPC Server

Write format: range 0 – 2

0 = Sets application to HOLD state

1 = Sets application to RUN state (Write 7, Read 1)

2 = Sets application to CAL state

Strmxxx.AlarmMsg –

AlarmMsg is a read-only, string tag that contains the highest priority WARNING or FAULT error message for this stream.

Strmxxx.AlarmStatus –

AlarmStatus is a read-only 4-byte integer tag that contains the current stream alarm status.

0 = OK, Normal

1 = WARNING ALARM this stream

2 = FAULT ALARM this stream

3 = WARNING & FAULT ALARM this stream

Strmxxx.Enable –

Enable is a read/write, Boolean tag used to set this stream ENABLE/DISABLE state.

0 = This stream DISABLED

1 = This stream ENABLED

Write format: 0 or 1

vStrmxxx.Force –

Force is a read/write, Boolean tag used to force this stream to run next.

0 = Normal Stream Sequencing

1 = Force this stream to run next

Write format: 0 or 1

Strmxxx.Name –

Name is a read-only, string tag that contains the text “Strmxxx”. Where xxx is the stream ID.

Strmxxx.RunEvent –

RunEvent is a read/write, 4-byte integer used to run an event in the device. Example to run EVT32 in the analyzer, write a 32 to this tag.

Write format: Positive integer in the range 1- 32535

Strmxxx.SampleTime –

SampleTime is a read-only, Date tag that contains the time the current analysis was injected for this stream.

Compxxx.Name –

Name is a read-only, string tag that contains the name of this component.

Compxxx.Value –

Value is a read-only, 4-byte real tag that contains the value of this component.

6. MAXUM DATABASE - OPC SETUP

Maxum OPC Server accesses attributes in the following Maxum database tables:

- 1.) System_control Table
- 2.) Stream_method Table
- 3.) Application Table
- 4.) Result Table
- 5.) OPC Data Collection Table

Certain attributes in these tables must be setup correctly before OPC Server configuration can be completed. Failure to complete the necessary database configuration steps can result in the inability to properly configure OPC Server.

IMPORTANT: The Maxum OPC Server is compatible with Maxum version 4.0 or greater.

6.1. Setup Procedure

- 1.) Select and set the *System_control* table *opcflag* attribute. This enables the OPC Data Collection table.
- 2.) Select and set the *Stream_method* table *autotrt* attribute.
- 3.) In the *Result* table in the *trtval* attribute set the transmission order for the results that are to be sent. This needs to be set for each stream on each application. **Note:** Results (components) that do not have the *trtval* set will not show up in the AOS component setup dialog. *Trtval* order numbers must be sequential beginning with one for each stream.
- 4.) If the Maxum database will be receiving Optichrom transmissions (TRTs) set the *System_control* table *adh_trt* attribute equal to the desired host computer format:
0=HCI-H, 1 = HCI-C01, 2 = HCI-C04, 3 = HCI-A

6.2. OPC Data Collection Table

The OPC Data Collection Table is made up of the following attributes:

Anlz - 0 for local analyzer , > 0 the analyzer number of the Optichrom analyzer

Application - 1 for Optichrom or application_id for Maxum

Stream - Stream #

Streamname - For local analyzer only

Enable - Stream enable status, or used to enable/disable a stream from the OPC server

Force - Used to select a stream to run continuously from the OPC server

Nextstream - Next stream to run

Settime - Used to set the time from the OPC server

Runstatus - 0=hold, 1=run, 2=cal; or used from the OPC server to set the status

Binversion - Results binary version

Numresults - Number of results in binary

Lastreceived -Cycle time of Maxum results, received time for Optichrom results

Xmit_count - Keeps count of the # of results transmissions received

Results - Binary format of results

Runevent - Used to run an event from the OPC server

Setalarm - Used to post an alarm from the OPC server

Alarmstatus - Alarms on the stream 0=none, 1=warnings, 2=faults, 3=faults and warnings

Alarmmsg - Text message of most serious and recent stream alarm

Sysalmstatus - Alarms on the analyzer 0=none, 1=warnings, 2=faults, 3=faults and warnings
sysmsg - Text message of most serious and recent analyzer alarm

Appendix A - Distributed COM (DCOM)

(Component Object Model) is an integral part of Microsoft Windows NT/2000/XP. Several Windows features are built on COM. **DCOM** (Distributed Component Object Model) is a protocol that enables COM objects to communicate over a reliable, secure, and efficient network link. The **Maxum OPC Server** uses COM/DCOM to communicate with remote OPC clients. If you plan on using the OPC Server with one or more remote OPC clients then the DCOM access permissions must be configured properly on both the OPC Server and Client machines.

IMPORTANT: *Inadvertently changing DCOM settings can potentially make your PC inoperable. Document any changes you make to your DCOM settings. If you have problems restore the original settings. Read through this document before you make any changes to your DCOM Configuration Properties. This document is intended to provide guidance for proper DCOM setup. The settings suggested here may need to be varied for your situation depending on your OPC Client application and network security requirements. It is suggested that you review your DCOM settings with your network Administrator prior to making any changes.*

Two types of permissions are discussed in this document :

Access – this type of permission allows a client machine to connect to the OPC Server to obtain a list of OPC servers, connect to an OPC Server, and to register a callback with the OPC Server, known as a data subscription.

Launch – this type of permission allows an OPC client to start the OPC Server or "launch" the server on a remote machine.

If the permissions are not correct on both ends you may have one or more of the following symptoms:

1. The OPC client cannot browse the Server machine for a list of available OPC Servers.
2. The OPC client can successfully browse the Server machine but is unable to connect to the Maxum OPC Server application. (Siemens MaxumOPCServer)
3. The OPC client can successfully browse and connect to the MaxumOPCServer yet no OPCItems or data tags are available.

6.3. DCOM Configuration Utility

Windows comes with the DCOM configuration utility “DCOMCNFG.EXE”. Use this utility to check and configure your DCOM settings prior to using the Maxum OPC Server.

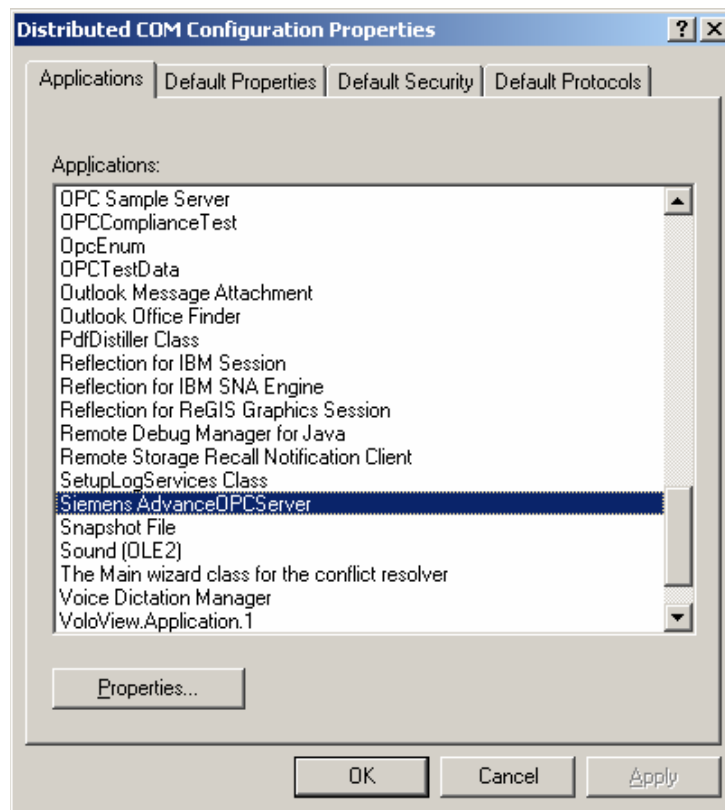
IMPORTANT: *Inadvertently changing DCOM settings can potentially make your PC inoperable. Document any changes you make to your DCOM settings. If you have problems restore the original settings. Read through this document before you make any changes to your DCOM Configuration Properties. This document is intended to provide guidance for proper DCOM setup. The settings suggested here may need to be varied for your situation depending on your OPC Client application and network security requirements. It is suggested that you review your DCOM settings with your network Administrator prior to making any changes.*

Make sure you are logged in as the Administrator or on an account with Administrator rights before attempting to modify the DCOM settings. Start the DCOM configuration utility by selecting *Start, Run..., DCOMCNFG* or from the command-line by typing *DCOMCNFG*. Answer "Yes" to any message boxes that appear, allowing the utility to assign application ID entries to those servers that don't already have them.

The main window for *DCOMCNFG* allows the user to either configure default settings for all COM servers or else to configure settings for a specific server chosen from the list. The former will affect all servers configured to use the default settings. The latter will affect the selected server only.

NOTE: DCOM settings are stored in the registry and loaded by Maxum OPC server at start-up. Therefore, server processes must be shut down and restarted in order for changes to take effect.

Start the *DCOMCNFG* utility. The first screen contains four tabs: Applications, Default Properties, Default Security, and Default Protocols.



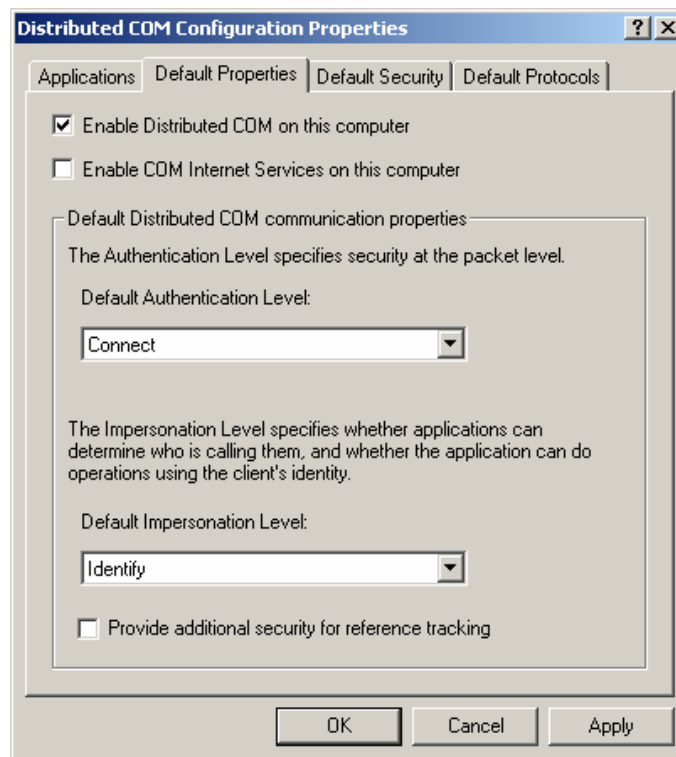
Appendix A1

6.4. Default Properties

The Default Properties tab contains settings that affect all DCOM communication on the machine. First of all, ensure that the "Enable Distributed COM on this computer" is selected. This makes the machine available to others via DCOM. Typically, the other settings do not need to be changed.

The Authentication Level specifies when COM should authenticate the identity of calling clients (each call, each packet, etc). Normally, it should be set to "Connect", indicating that COM should authenticate clients when they first connect to a server.

The Impersonation Level specifies whether servers can ascertain the identity of calling clients and whether they can then perform operations on the client's behalf (as if the server is the client). Use the default setting "Identify", allowing the server to identify the calling client to see if it is allowed access to a certain resource but not to actually access any of these resources as the client.

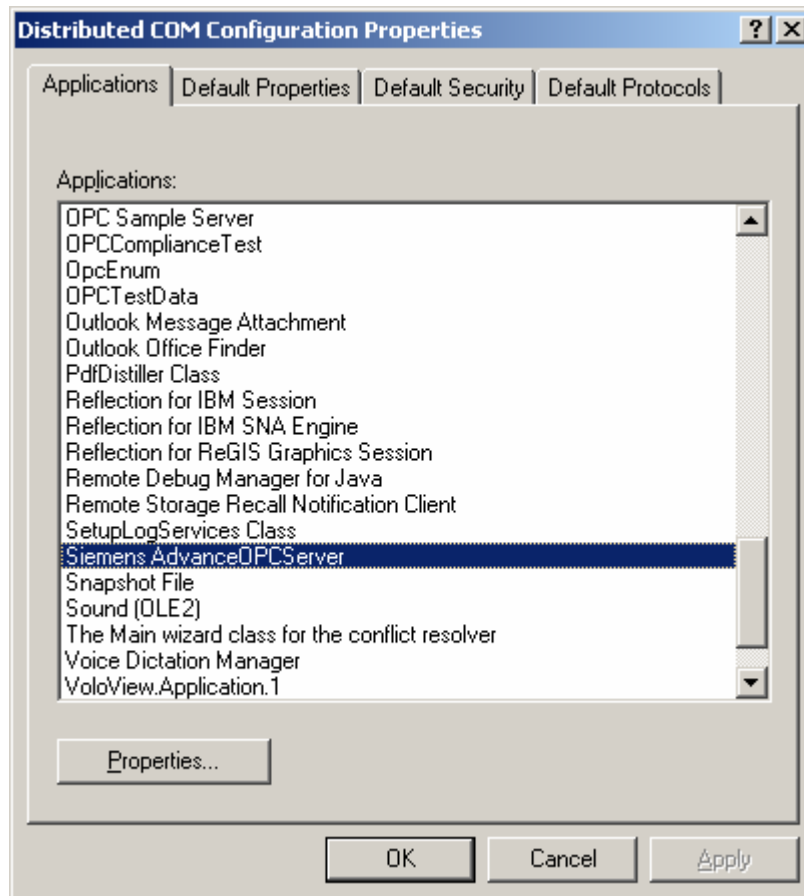


Appendix A2

6.5. Security Permissions

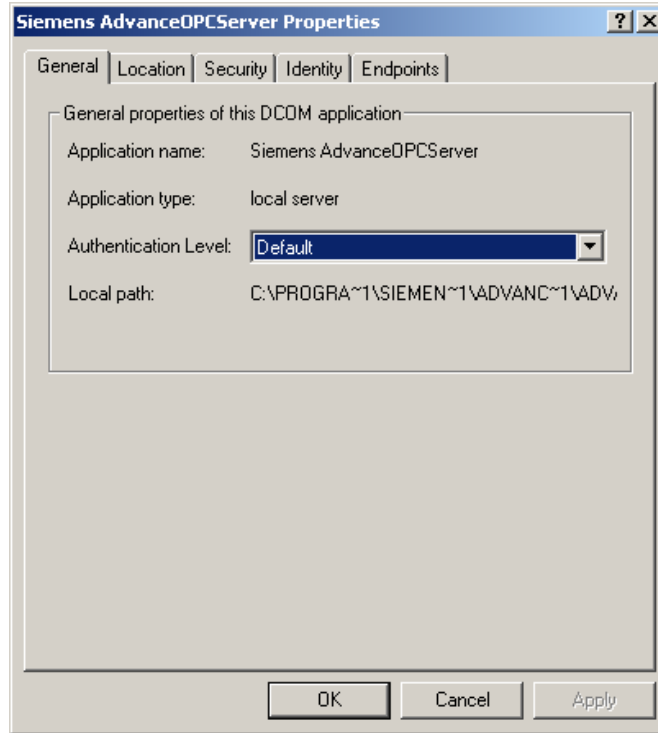
The most important DCOM settings for an OPC server are the security permissions. There are two ways to change the security permissions.

- 1.) Change the default settings (not recommended) and make sure that the OPC server will use these settings. To change the default settings select the Default Security tab on the main Configuration Properties dialog box.
- 2.) Change the specific settings for the Maxum OPC Server (recommended). To set the security permissions for The Maxum OPC Server first open the DCOM configuration utility. After it is open click on the Siemens MaxumOPCServer.



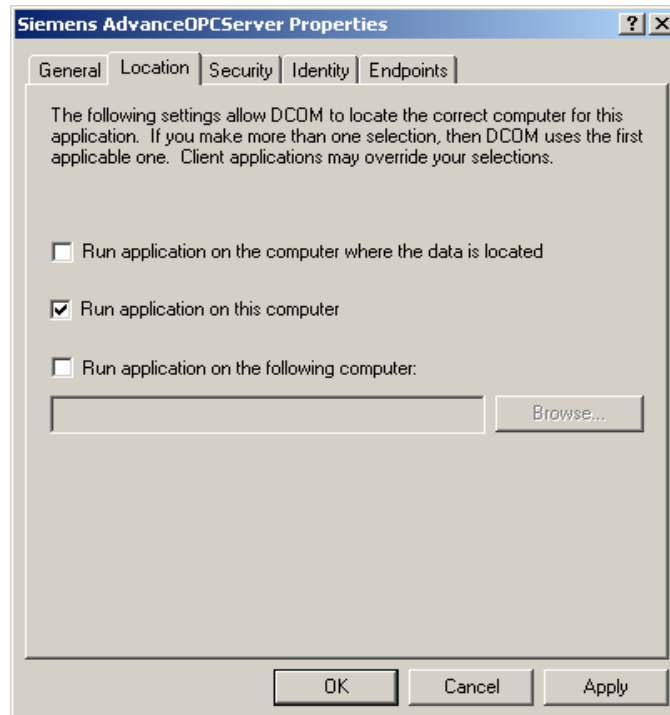
Appendix A3

On the General tab verify that the Authentication Level is set to Default.



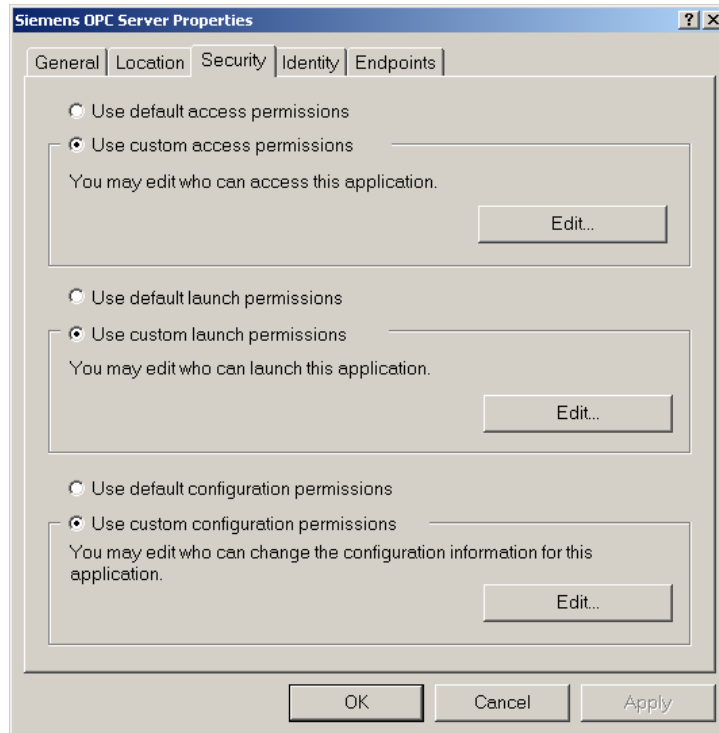
Appendix A4

Click on the *Location* tab and check Run application on this computer



Appendix A5

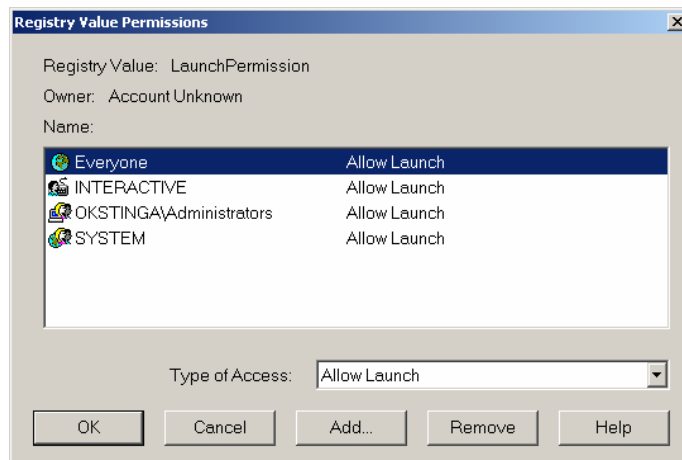
Click on the Security tab to set the Security for the Server.



Appendix A6

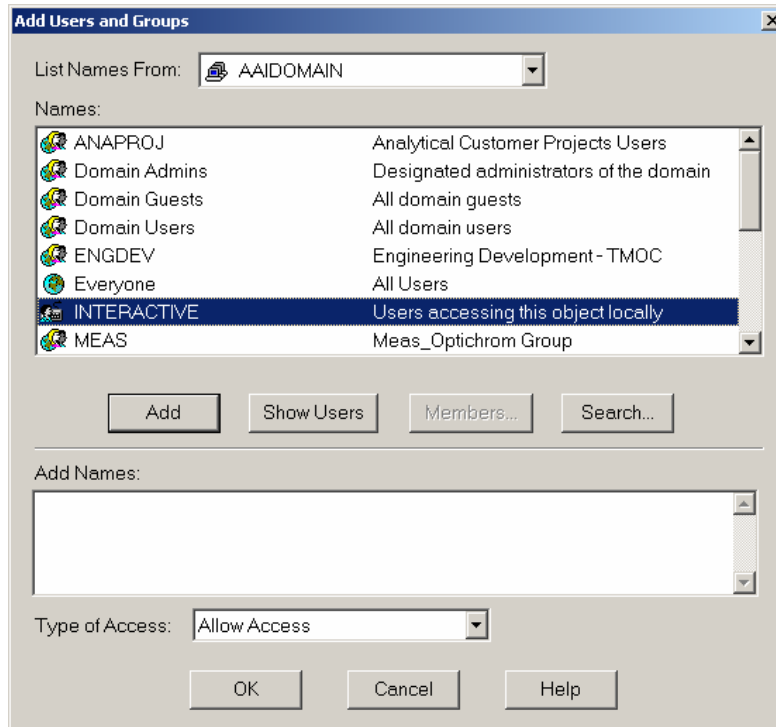
The Access Permissions contain an Access Control List of principals that are allowed to interact with objects supplied by a server. The Launch Permissions contain an Access Control List of principals that are allowed to start up a server process or service. Include the names of users or user groups from trusted domains that you wish to be able to use the OPC server on this machine. In the initial startup phase it is suggested that you add the Everyone group to the OPC Server and Client end permissions. After the system is working you can remove the Everyone group and refine the permissions list.

To set the Access permissions you would click “Use custom access permissions” and then click edit. A screen similar to the following will appear:



Appendix A7

To add a user to the list click on Add.



Appendix A8

If the user you wish to add does not appear then click Show Users. When you are done adding users click OK to close “Add Users and Groups”. You can then choose to “Allow Access” or “Deny Access” for each user.

The procedure to set the launch permissions is similar to the above, but instead of choosing “Allow Access” for a user you would choose “Allow Launch”.

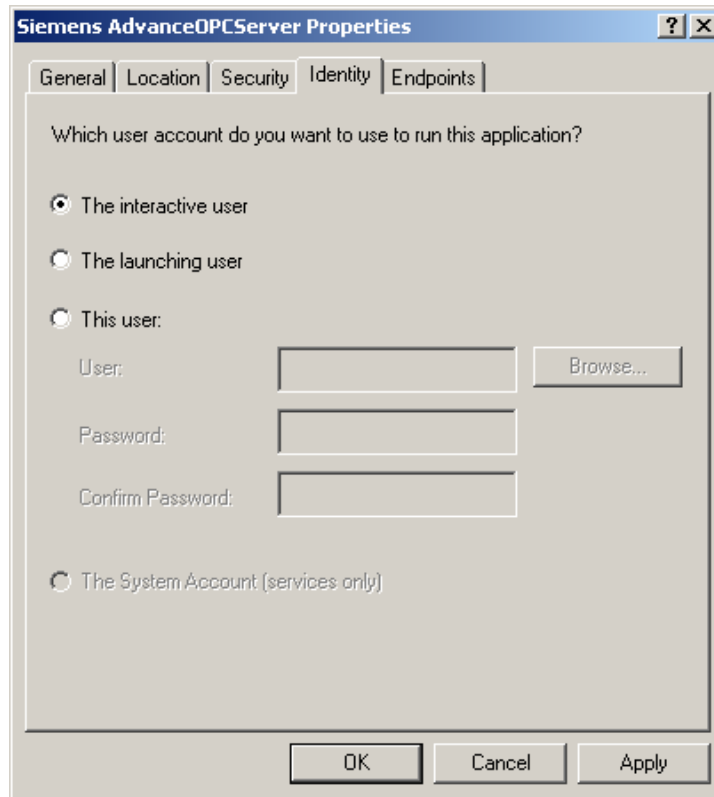
The Configuration Permissions contain an Access Control List of principals that are allowed to modify the configuration information for a server. In other words, it indicates who is allowed to alter the registry entries for installed servers as well as who is able to access the registry for the purposes of installing new servers. It is usually simpler to install and configure servers as a user with local administrative rights.

To connect to an OPC server from outside of the domain create a local user account on both the server and the client machine with identical username and password. Then add the local user on the OPC server to the DCOM permissions. Then use the local account on the client machine to connect to the OPC server.

6.6. Server Identity

The Identity tab for a selected COM (or OPC) server specifies the user account that should be used when COM starts up the process.

Servers that run as local executables have the option of running as the launching user (the calling client –this is the default), the interactive user (the one currently logged onto the machine), or a specified user. Don't use the default setting because it will launch a new copy of the AOS each time a client attempts to connect to the server. Use *The interactive user* or *This User* setting. When using the *This User* option its suggested that you create a group and specify the group as the User. A group can be created under Settings->Control Panel->Users and Passwords->Advanced tab->Advanced button, select the Groups folder then right click and select New Group...



Appendix A9

6.7. Default Protocols

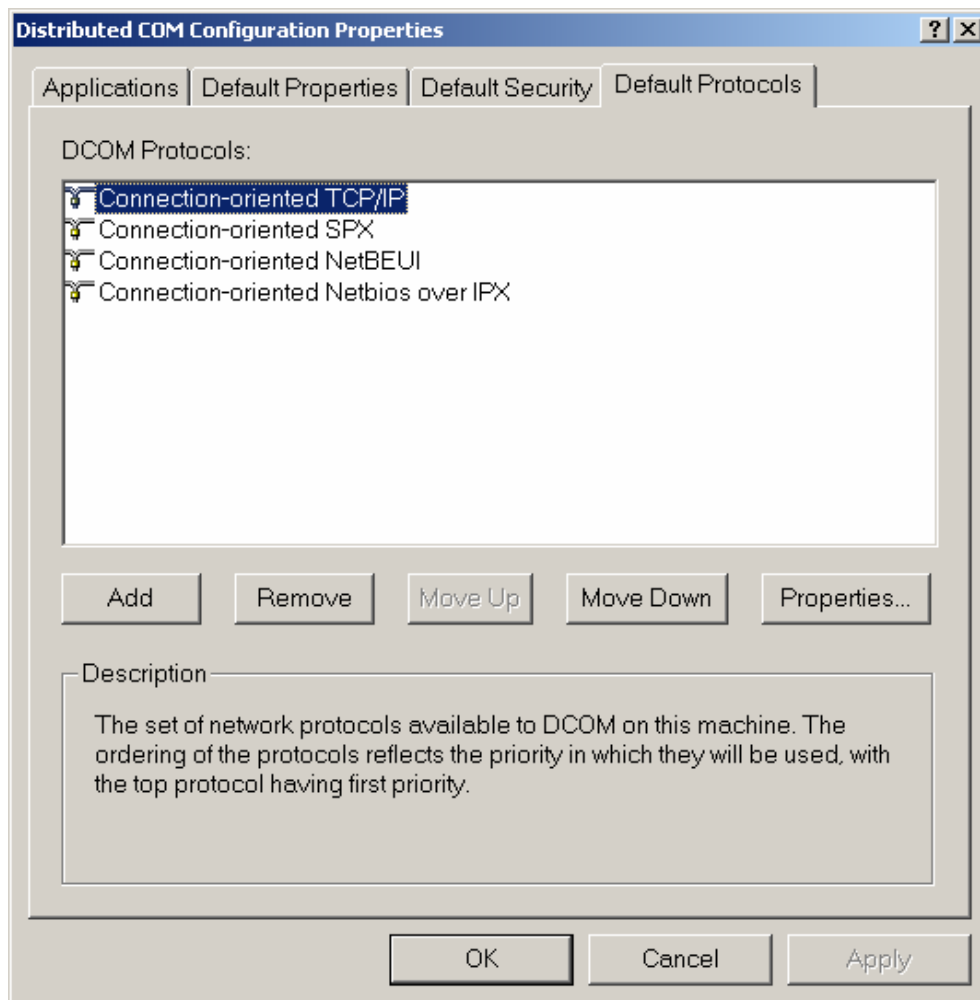
The Default Protocols tab specifies the communication protocols available to DCOM. The order that protocols appear in the list indicates the priority in which they will be used (the topmost having the highest priority).

Note: UDP/IP protocol does not work correctly for DCOM and should be removed from the list

The more protocols that appear in the list, the better the chances of connecting to an OPC server on an unknown remote machine. However, it may also take longer for DCOM to time out if a server machine is not present since it has to try each protocol in turn.

For most situations it is best to remove all unused protocols from the list and only include those that are necessary for your network.

The recommended protocol for the Maxum OPC Server is the "Connection-oriented TCP/IP" protocol. Contact your IT personnel for more information about your network.



Appendix A8

7. Appendix B OPC Test Client

The FactorySoft OPC Client program is included with the Maxum OPC Server. The Client program can be used as a verification and/or troubleshooting tool. The OPC Client executable (OPC Client.exe) is installed in the \Program Files\Siemens Applied Automation\Client directory. You must add a shortcut to the OPC Client program or select Start, Run..., Browse to the OPC Client.exe, and then click OK to start the program. Connect to the OPC Server by selecting OPC->Connect... from the main menu. (See Figure B3)

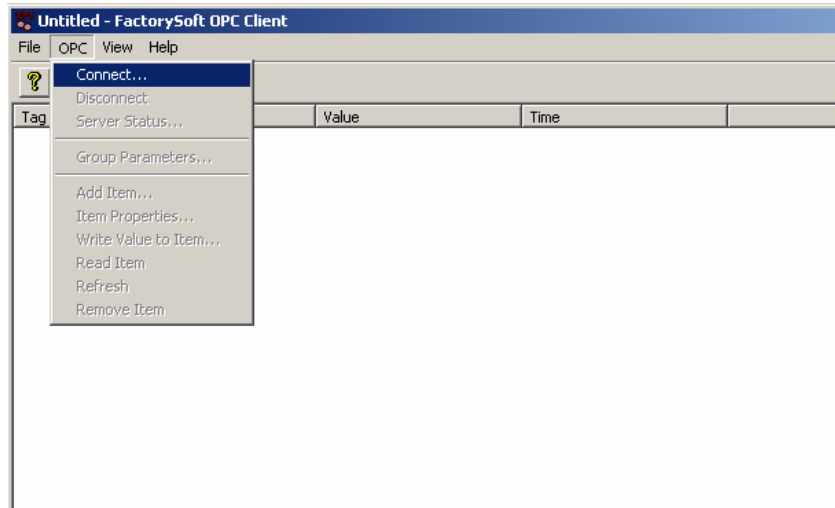


Figure B3

When the Select OPC Server dialog box is displayed enter the IP address of the remote device in the **Server Node** combo box (if the OPC Server and OPC Client are running on the same machine leave the box empty) then click on the Refresh List box. (See Figure B2) The OPC Client launches the OPC Foundation List Server on the OPC Server machine. If the command is successful a list of the available OPC Servers is displayed in the **Available Servers** list box. If an *Access is Denied* error message is displayed it could mean that the DCOM Security settings are not properly setup on either the OPC Server or OPC Client machine. Refer to the appropriate section of this manual for more information on DCOM settings.

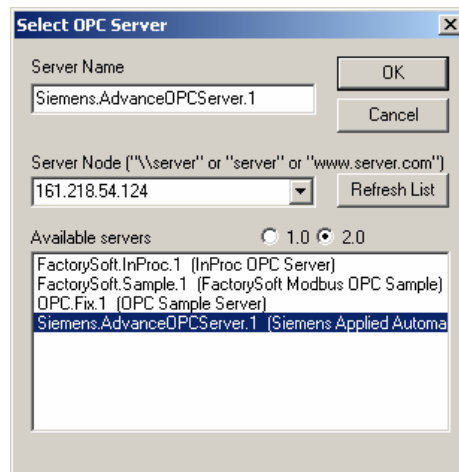


Figure B2

To connect to the OPC Server double click on the Server name in the **Available servers** list box or select the Server name and click on OK. If no error messages are returned the connection is complete and the OPC group object was automatically created. The next step is to define which of the OPC Items available in the OPC Server to add to the Client group. From the main menu select OPC->Add Item. (See Figure B3)

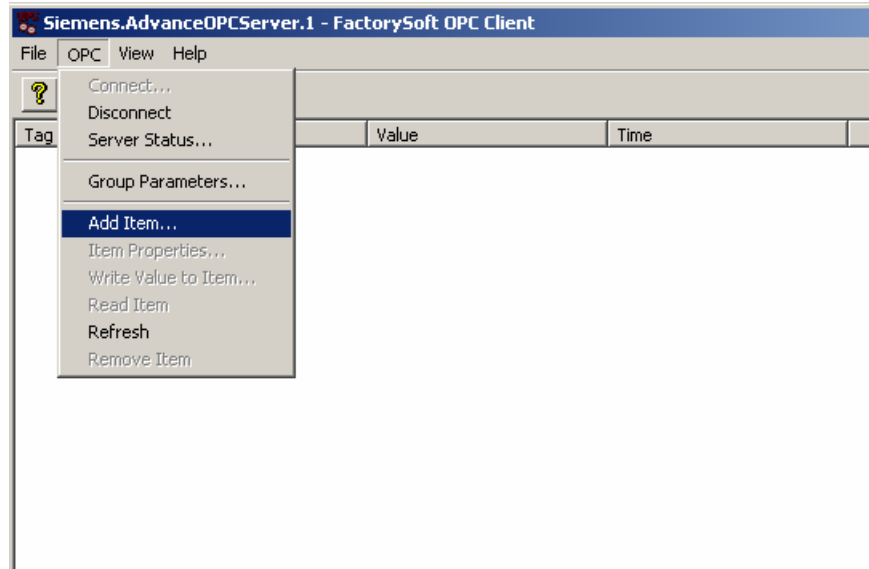


Figure B3

The Add Item dialog box is displayed with the defined devices, groups, and tag in the **Browse items:** list box. (See Figure B4)

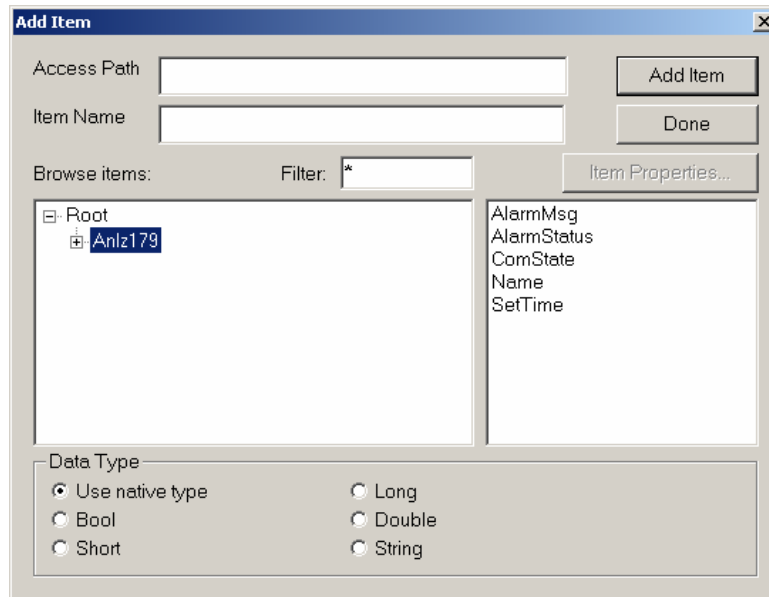


Figure B4

Select a device or folder in the **Browse items:** list box and the associated tags are displayed in the right hand pane. Select each tag to add it to the OPC Client group and click on Add Item button or double click on each item. Click on the Done button to complete this action.

The OPC Items selected are displayed in the main window of the OPC Client. (See Figure B5)

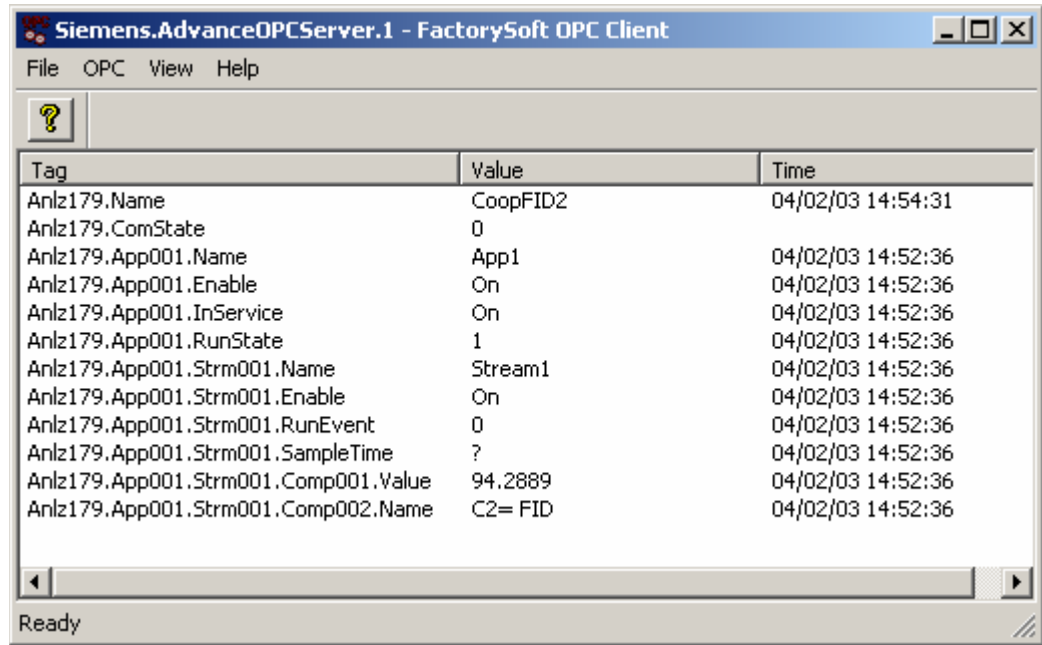


Figure B5

The OPC Client has subscribed to the tags displayed in the main window and the values will be automatically updated based on a change in data value.

To determine an OPC Item property, to force a read refresh, or to write an OPC Item supporting write access right click on the OPC item and select the appropriate command from the popup menu. (See Figure B6)

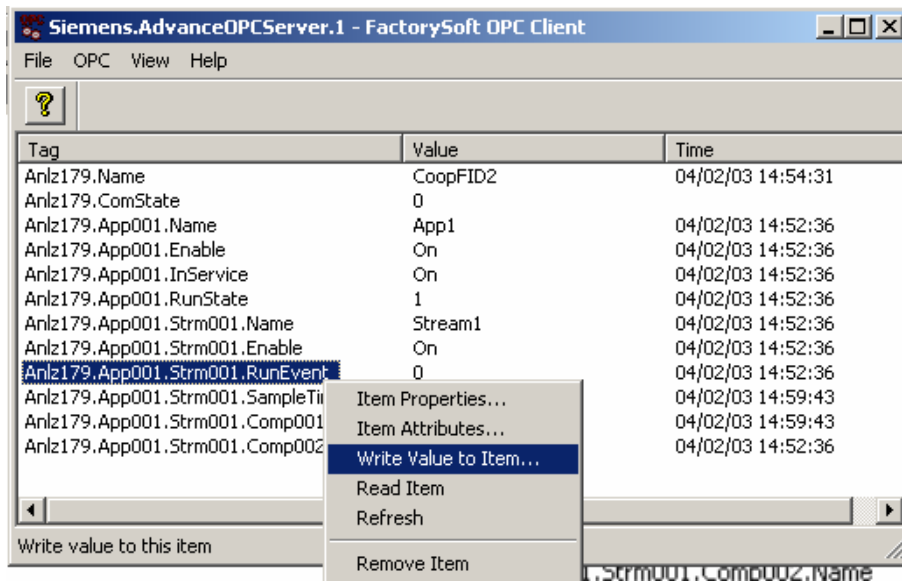


Figure B6

8. Appendix C Automation Client Example

A working example of an Automation Client program is included with the Maxum OPC Server installation program. The Client program is written in Excel – VBA and can be used as a starting point in writing your own OPC Client application. The Excel Client program is located in:

|Program Files|Siemens Applied Automation|Client and is titled **AutomationClientEx.xls**. The program is an Excel macro and therefore requires that the Microsoft Excel program is loaded on your machine.

Background:

The OPC Data Access Automation Interface defines a standard by which automation applications can access process data. This interface provides the same functionality as the custom interface, but in an “automation friendly” manner. The Automation Client can be written in Visual Basic or Visual Basic for Application (VBA). The Automation Client uses a ‘wrapper’ DLL provided by the OPC Foundation (OPCDAuto.DLL) to communicate with the Custom interface provided by the Maxum OPC Server. The OPC Foundation specification for the Data Access Automation Interface Standard is included with the Maxum OPC Server installation and the OPCDAuto.DLL was installed automatically with the OPC Server installation. The Automation Interface document defines the methods and parameters plus some examples to write an Automation Client. Use the example client program as is or as a guide in writing your own OPC Client.

How to run the example Automation Client:

First, execute the AutomationClientEx.xls program. This should start the Microsoft Excel program on your PC. The following dialog box will be displayed in Excel:(See Appendix B1)

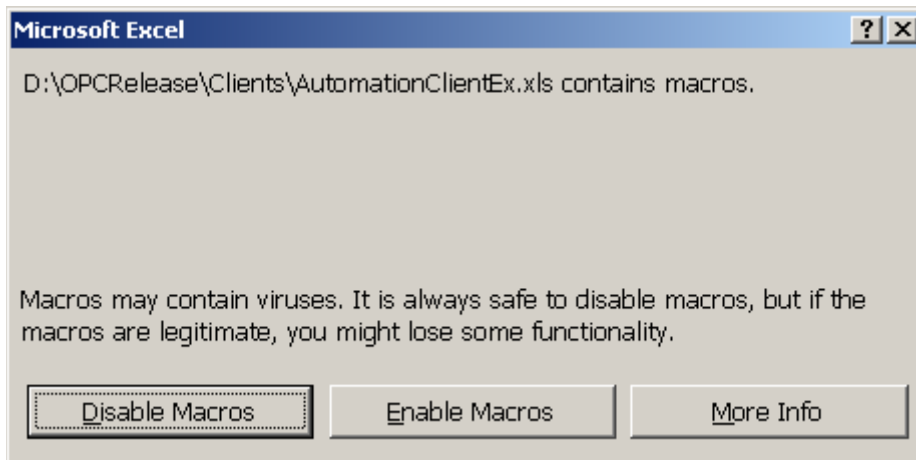


Figure B1

Click on the Enable Macros button.

The Advance Automation Client screen will be displayed similar to the figure B2

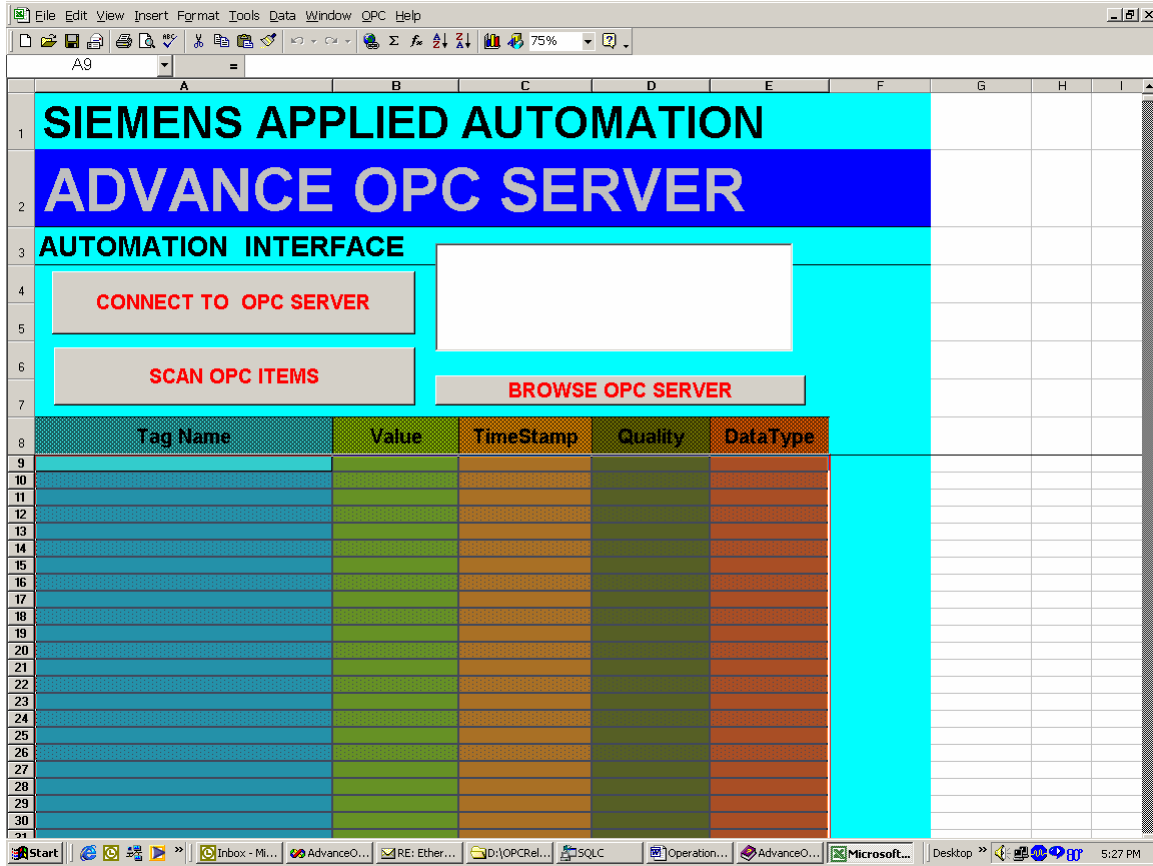
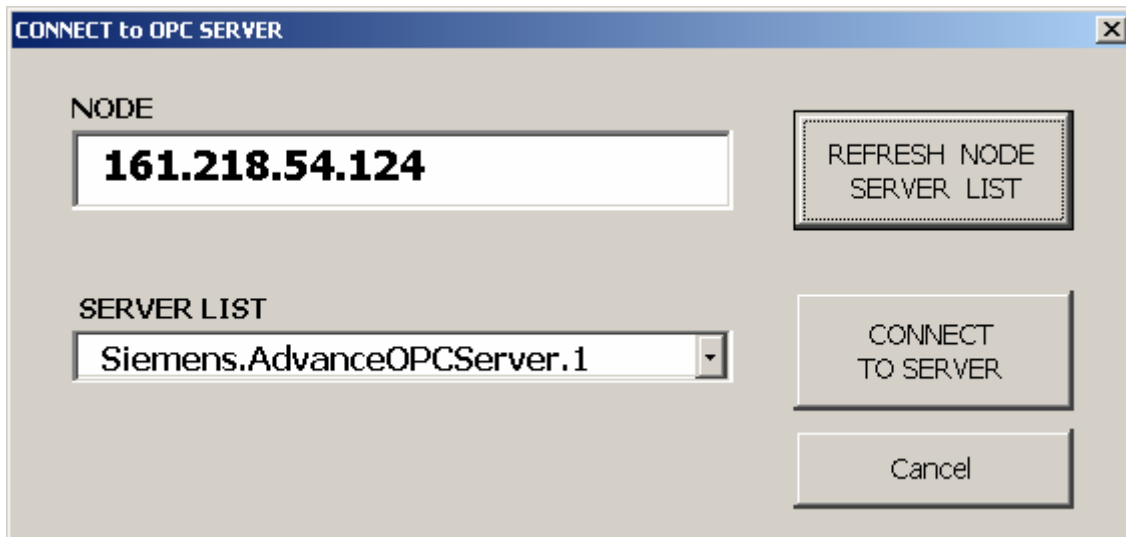


Figure B2

Click on the CONNECT TO OPC SERVER button. The CONNECT to OPC Server Dialog box will be displayed. (See Figure B3) Enter the IP address of the Maxum OPC Server then press the REFRESH NODE SERVER LIST button. The client program will use the OPC List Server located on the PC where the Maxum OPC Server is running to display a list of all the available OPC Servers in the SERVER LIST list box. Select Siemens.MaxumOPCServer.1 and click on the CONNECT TO SERVER button.



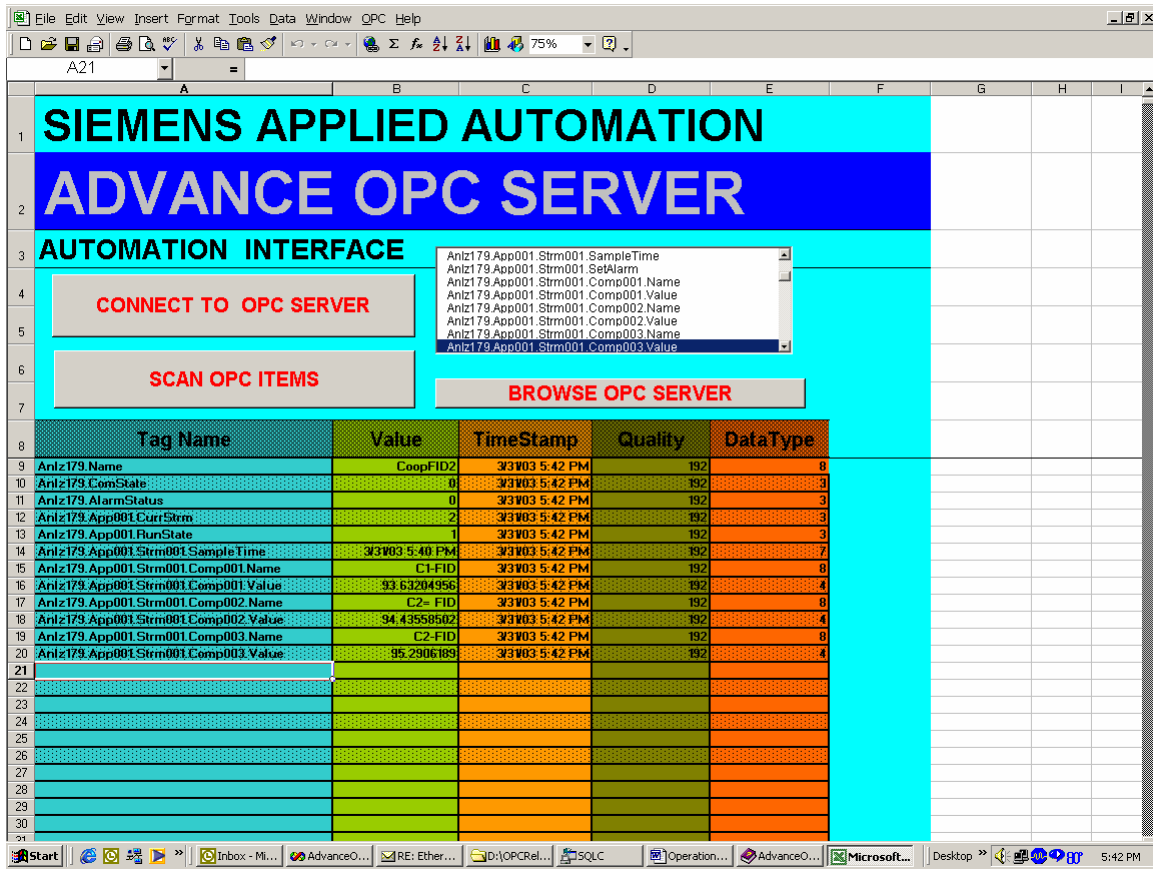


Figure B6

The selected items will be updated at the end of each analyzer cycle.

From the main menu select Tools->Macro->Visual Basic Editor to display the AutomationClientEx macro.

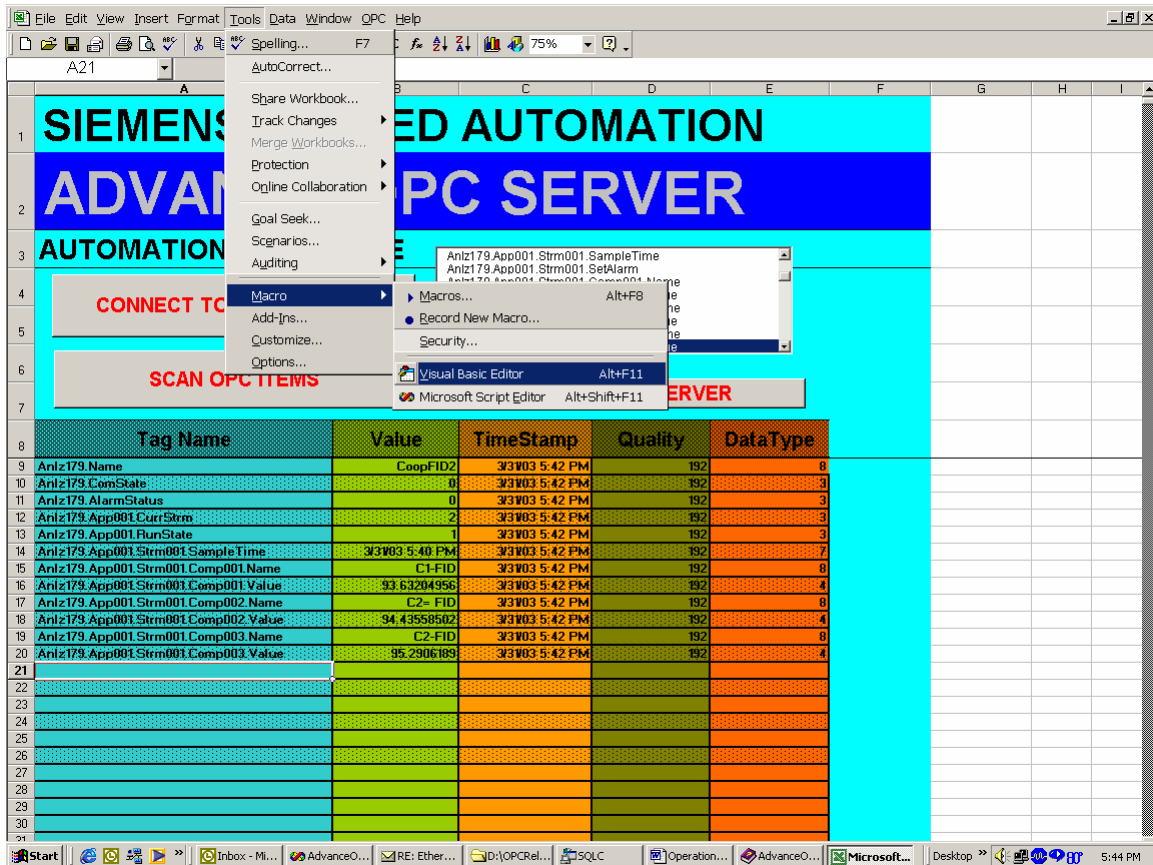
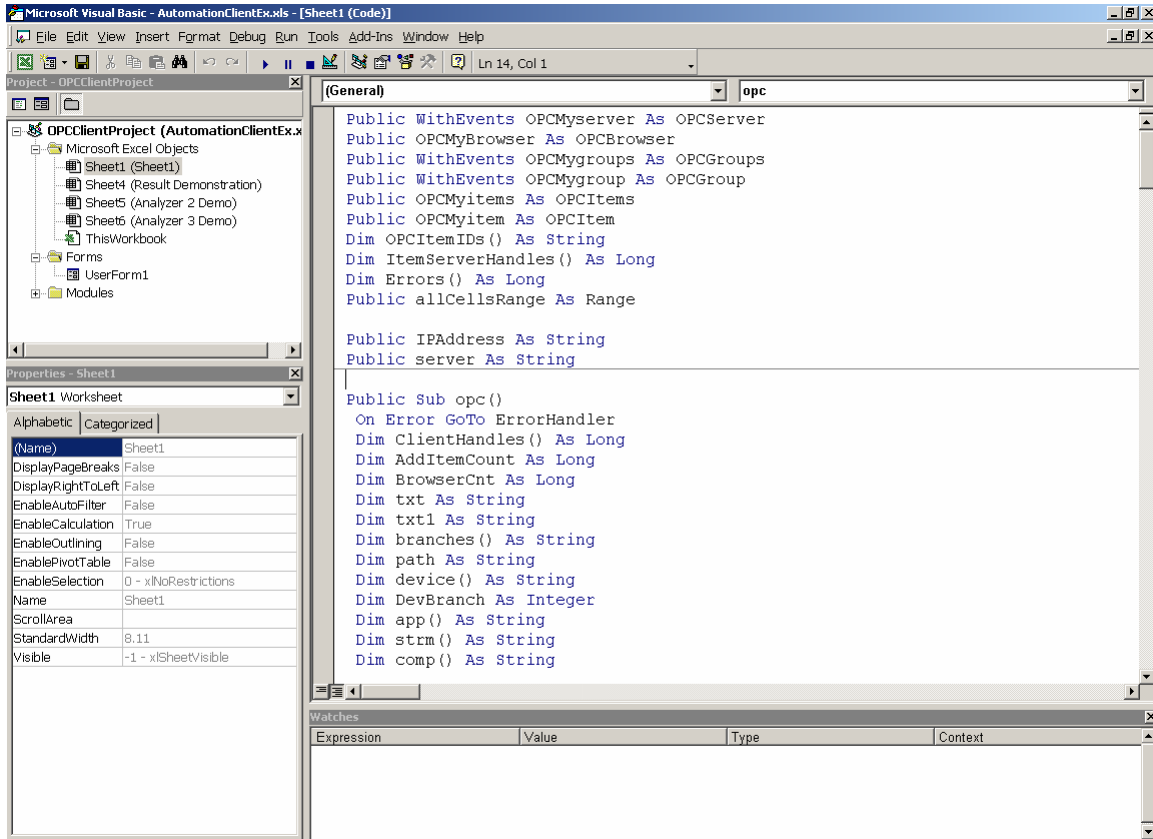


Figure B7



8.1. Client Code Descriptions

Please refer to the OPC Foundation document titled “**Data Access Automation Interface Standard**”. This document was installed under the Maxum OPC Server directory.
 \Siemens Applied Automation\DOC\OPCDAAuto.doc or it can be obtained from the OPC Foundation web site at www.OPCFoundation.org. The following example was written using VBA (Visual Basic Applications)

Define the Server/Group/Item Objects

First define the OPCServer Automation object, the OPCGroups and OPCGroup objects, and the OPCItems and OPCItem objects. The WithEvents syntax enables the object to support the declared events for the particular object.

```

Public WithEvents OPCMyserver As OPCServer
Public OPCMyBrowser As OPCBrowser
Public WithEvents OPCMygroups As OPCGroups
Public WithEvents OPCMygroup As OPCGroup
Public OPCMyitems As OPCItems
Public OPCMyitem As OPCItem
Dim OPCItemIDs() As String
Dim ItemServerHandles() As Long
Dim Errors() As Long
Public allCellsRange As Range
Public IPAddress As String
Public server As String
    
```

Create the OPCServer object.

Create a new OPCServer object and make a list of all the available OPC Servers located on the specified machine using the OPCServer object method GetOPCServers. (Sheet1.OPCMyserver.GetOPCServers(Sheet1.IPAddress))

```
Public Sub CommandButton1_Click()
    Dim servers() As String
    Set Sheet1.OPCMyserver = New OPCServer
    If UserForm1.TextBox1.value <> "" Then
        Sheet1.IPAddress = UserForm1.TextBox1.value
    Else
        Sheet1.IPAddress = "127.0.0.1"
    End If
    On Error GoTo ErrorHandler
    servers = Sheet1.OPCMyserver.GetOPCServers(Sheet1.IPAddress)
    UserForm1.ComboBox1.Clear
    Sheet1.ListBox1.Clear
    For I = 1 To UBound(servers)
        UserForm1.ComboBox1.AddItem servers(I)
    Next I
    UserForm1.ComboBox1.DropDown
    UserForm1.ComboBox1.SelText = UserForm1.ComboBox1.List(0, 0)

    If UserForm1.ComboBox1.value <> "" Then
        Sheet1.server = UserForm1.ComboBox1.value
    End If

Exit Sub
ErrorHandler:
MsgBox ("Can't connect to OPC Server, check IP address")

End Sub
```

Connect to the specified OPC Server and add a Group object.

Using the server object (OPCMyserver) execute the *connect* method with input parameters server name (Sheet1.server) and server IP address (Sheet1.IPAddress) . Then *Add* an OPCGroup object (Group1) using the OPCGroups property of the server object. Create the Browser object using the *CreateBrowser* method of the server object.

```
Public Sub CommandButton2_Click()
    If Sheet1.server = "" Then GoTo end1
    If Sheet1.IPAddress = "" Then GoTo end1
    Sheet1.OPCMyserver.connect Sheet1.server, Sheet1.IPAddress
    On Error GoTo ErrorHandler
    Set Sheet1.OPCMygroups = Sheet1.OPCMyserver.OPCGroups
    Set Sheet1.OPCMygroup = Sheet1.OPCMygroups.Add("Group1")
    Set Sheet1.OPCMyBrowser = Sheet1.OPCMyserver.CreateBrowser
    Sheet1.allCellsRange.Select
    Sheet1.allCellsRange.ClearContents

    UserForm1.Hide
end1:
Exit Sub

ErrorHandler:
MsgBox ("Access denied, check DCOM access settings on both server and client.")
```

End Sub

Create and Add Items to the OPCItems object

Using the OPCGroup object (OPCMyGroup) method OPCItems create the OPCItem object and add OPCItems to the group using the AddItems method.

```
Public Sub opc()  
    On Error GoTo ErrorHandler  
    Dim ClientHandles() As Long  
    Dim AddItemCount As Long  
    Dim BrowserCnt As Long  
    Dim txt As String  
    Dim txt1 As String  
    Dim branches() As String  
    Dim path As String  
    Dim device() As String  
    Dim DevBranch As Integer  
    Dim app() As String  
    Dim strm() As String  
    Dim comp() As String  
    Dim branchCnt As Long  
    Dim ret As Boolean  
  
    If OPCMygroup.OPCItems.count > 0 Then  
        OPCMygroup.IsActive = False  
        OPCMygroup.IsSubscribed = False  
        OPCMyitems.Remove OPCMygroup.OPCItems.count, ItemServerHandles, Errors  
    End If  
  
    ReDim ClientHandles(UBound(OPCItemIDs))  
    For X = 1 To CInt(UBound(OPCItemIDs))  
        ClientHandles(X) = X + 1  
    Next X  
  
    Set OPCMyitems = OPCMygroup.OPCItems  
    OPCMyitems.AddItems UBound(OPCItemIDs), OPCItemIDs, ClientHandles, ItemServerHandles, Errors  
    If Errors(1) <> 0 Then  
        MsgBox "Error"  
    End If
```

Set the Group Active and Subscribe to the Group Data

When the IsActive Groupproperty is set to TRUE , the OPC server reads out registered items. When the IsSubscribed property is set to TRUE, the DataChange event is called back on data change. The OPCItem object method **READ** is used to read the item values into the group variables.

```
OPCMygroup.IsActive = True  
OPCMygroup.IsSubscribed = True  
  
Private Sub OPCMygroup_DataChange(ByVal TransactionID As Long, ByVal NumItems As Long,  
    ClientHandles() As Long, ItemValues() As Variant, Qualities() As Long, TimeStamps() As Date)  
    Dim anItem As OPCItem
```


Dim row As Integer
Dim col As Integer

Worksheets("Sheet1").Activate

For X = 1 To NumItems
Range("A9:E9").Select
Range("A9").Activate
Set anItem = OPCMyitems.Item(ClientHandles(X) - 1)
anItem.Read OPCDevice
row = ClientHandles(X) - 2
col = 0
ActiveCell.Offset(row, col).Activate
ActiveCell.NumberFormat = "General"
ActiveCell.value = anItem.ItemID
ActiveCell.Offset(0, 1).Activate

Select Case anItem.CanonicalDataType
Case 3
ActiveCell.NumberFormat = "General"
Case 7
ActiveCell.NumberFormat = "m/d/yy h:mm AM/PM"
Case 8
ActiveCell.NumberFormat = "General"
ActiveCell.HorizontalAlignment = xlRight

End Select

ActiveCell.value = anItem.value
ActiveCell.Offset(0, 1).Activate
ActiveCell.NumberFormat = "m/d/yy h:mm AM/PM"
ActiveCell.value = anItem.TimeStamp
ActiveCell.Offset(0, 1).Activate
ActiveCell.NumberFormat = "General"
ActiveCell.value = anItem.Quality
ActiveCell.Offset(0, 1).Activate
ActiveCell.NumberFormat = "General"
ActiveCell.value = anItem.CanonicalDataType
Next X
Range("A9").Activate
End Sub

9. Additional Information Resources

1. The OPC Foundation web sites located at www.opcfoundation.org or www.opceurope.org
2. Book, “OPC - Fundamentals, Implementation and Application”, 2., published by Huthig Fachverlag ISBN-3-7785-2883-1, English, Amazon.com
3. Book, “Inside COM”, Dale Rogerson / Microsoft Press, ISBN:1572313498, Amazon.com
4. Web site, www.microsoft.com/com/wpaper/, site includes whitepapers on COM/DCOM

10. Glossary

COM - Component Object Model.

COM is an "industry-standard" software architecture supported by Microsoft and many other companies. COM solves is the standard way for Windows program modules (applications) to communicate with each other. COM is a protocol that connects one software module (application) with another and then drops out of the picture.

COM is a binary standard that specifies how objects interact with each other. A COM object is accessed exclusively through one or more sets of interfaces. An interface is a set of functions or methods. The COM standard not only specifies the binary object standard, it also defines a series of standard interfaces that provide common functionality. The OPC-DA specification defines several COM objects and their associated interfaces. These objects and interfaces provide a standard for real-time information exchange between software applications and process hardware.

The Maxum OPC Server uses Microsoft COM/DCOM/OLE technology to communicate with clients.

DCOM - The Distributed Component Object Model

DCOM extends COM to networks (remote objects). It is a new, highly optimized protocol, where remote components appear to be local.

Client –

In the Client / Server model the client is defined as a requestor of services. The Server is defined as a provider of services.

Server –

In the Client / Server model the Server is defined as a provider of services. The Client is defined as a requestor of services.

OPC - OPC (OLE for Process Control)

OPC is based on Microsoft's OLE, COM (component object model) and DCOM (distributed component object model) technologies, OPC consists of a standard set of interfaces, properties, and methods for use in process-control and manufacturing-automation.

OPC is an industry standard created by the OPC Foundation in collaboration with Microsoft corporation. The OPC Foundation is an organization with over 300 members world-wide consisting of nearly all the major control systems, instrumentation, and process control systems providers.

The objective of the OPC Foundation is to develop an open, flexible, plug-and-play standard that allows instrumentation suppliers to develop standard form device drivers that "serve" the data to any application supporting an OPC client interface.

Component –

Typically a component is defined as an analysis peak result. For Maxum and Advance Optichrom analyzers analysis peak values are stored in the Result table. Every Component or Result has certain attributes associated with the Result value. For the OPC Server every Component must have a name, value and trtval defined. The trtval attribute determines if the result is visible to the OPC Server and defines the result's reference number.

User(s) –

In Windows NT/2000/XP security terms, a User is a login name + password combination used to gain access(sign on) to the Windows machine.

NOTE: Any database variable can be placed in the Result table and assigned a name and trtval.

Group(s) -

In Windows NT/2000/XP security terms, a Group contains one to many Users. A Group account is easier to maintain when user names or access permissions will change frequently. Its easier to add/delete Users from a Group when granting temporary access for maintenance reasons. Proper use of Groups is a good way to reduce long-term maintenance costs.

Domain –

In Windows NT/2000/XP network security terms, a Domain is a group of machines that have reason to work together. Examples include Sales domain, Engineering domain, Account domain. Typically a Domain Server is setup that provides name and password authentication for the Users and Groups in the Domain list. Multiple domains may or may not be able to access one another unless a special relationship called a Domain Trust Relationship has been setup. A Domain user consists of the Domain name + user name.

Local User/Groups –

In Windows NT/2000/XP network security terms, a Local User or Group account is known only to the machine the account is setup on. Local accounts cannot access another machine unless the identical Local account (username plus password) was setup on both machines. Its typically easier to setup and use a Domain account with OPC Server and remote Clients.

Domain User/Groups –

In Windows NT/2000/XP network security terms, a Domain User or Group account can be used on any computer that is a member of the Domain. Authentication of the user name plus password is handled by the Primary Domain controller. It's easier to setup the DCOM permissions on both the OPC Server and Client with a Domain account as opposed to a Local User/Group account.



Siemens Energy & Automation, Inc.
7101 Hollister Road, Houston, TX 77040
Phone +1 (713) 939-7400, Fax +1 (713) 939-9050