Integration Objects' OPC HDA Interface MS Access

OPC HDA Server for MS Access

Version 1.1 Rev.0

USER GUIDE

OPC HDA Server for MS Access User's Guide Version 1.1 Rev 0
September 2015

OPC Foundation Self-Tested for Compliance logo indicates that this product has been tested by the manufacturer to be compliant with the following OPC Specifications:
Historical Data Access 1.00
Historical Data Access 1.10
Historical Data Access 1.20

Additional information about compliance testing, logo program and a summary of test results for **OPC HDA Server for MS Access** can be found at www.opcfoundation.org.

TABLE OF CONTENTS

TABLE OF FIGURES

LIST OF TABLES

PREFACE

About this User Guide

This guide:

- Describes the main features offered by Integration Objects' OPC HDA Server for MS Access.
- Lists the system requirements for installing and running Integration Objects' OPC HDA Server for MS Access.
- Explains how to configure this OPC server for database management.
- Explains how to use and run this OPC server.
- Describes all required DCOM settings both on server and client sides.

Target Audience

This document is intended for users that have applications that require connectivity to MS Access databases. Knowledge of the basics of OPC HDA (Historical Data Access) is assumed. It's also expected that you have some prior knowledge of configuration of MS Access database and SQL queries.

Document Conventions

Convention	Description
Bold	Click/selection action required
	Information to be noted
Blue bold italics	Reference to other sections, or to other Integration Objects User Guides

Customer Support Services

Offices	Email
Houston, USA: +1 713 609 9208	Support: customerservice@integrationobjects.com
Genoa, Italy: +39 34 75 83 93 47	Sales: sales@integrationobjects.com
Tunis, Tunisia: +216 71 195 360	Online: www.integrationobjects.com

INTRODUCTION

1. Overview

Integration Objects' OPC HDA Server for MS Access is a plug and play software application that is designed to offer full access to archived data monitored by the MS Access database system to any OPC HDA client.

This chapter contains the following parts:

- First, we present a global architecture showing the interaction between this OPC server, OPC clients and the MS Access databases in a client/server environment.
- Second, we describe all OPC Historical Data Access interfaces supported by this server.
- Finally, we enumerate software and hardware system requirements.

2. System Architecture

The OPC Server for MS Access reads and updates data from/to the historian database via the ADO (Active Data Objects) technology. It can be accessed locally or remotely via DCOM by any OPC HDA client.

The following figure describes a Client/Server architecture that shows the interaction between OPC HDA clients, OPC Server and databases.

Figure 1: System Architecture

This OPC server communicates with the MS Access database via the ADO technology using the required OLE DB Provider.

3. OPC Compatibility

Integration Objects' OPC HDA Server for MS Access implements the OPC Historical Data Access (HDA) specifications version 1.1 and 1.2.

4. Features

This section describes and enumerates features offered by this OPC server including OPC HDA services and server/database configuration.

4.1. OPC Historical Data Access Capabilities

This OPC server allows any HDA client to retrieve simple and processed data from the pre-configured database. Simple data retrieving includes raw reads and reads at time. However, summary data is computed using analysis functions such as average, interpolation, etc. Historical data can also be updated. You can insert new data or replace existing rows (checking the timestamp of the HDA item). You can also delete raw(s) for a specified time or during a time range when required.
For more details, you can refer to the **"OPC HDA Fundamentals.pdf"** delivered in the package (see the "Integration Objects\Integration Objects' HDA OPC Server for MS Access\Documents" directory under the Program Files folder).

The following are the current supported HDA interfaces.

Object	Interface	Supported
OPCHDAServer	**IOPCCommon**	Yes
	IOPCHDA_Server	Yes
	IOPCHDA_SyncRead	Yes
	IOPCHDA_SyncUpdate (optional)	Yes
	IOPCHDA_SyncAnnotations (optional)	No
	IOPCHDA_AsyncRead (optional)	Yes
	IOPCHDA_AsyncUpdate (optional)	Yes
	IOPCHDA_AsyncAnnotations (optional)	No
	IOPCHDA_Playback (optional)	No
	IConnectionPointContainer	Yes
OPCHDABrowser	**IOPCHDA_Browser**	Yes

Table 1: HDA Interfaces

This server does not support the annotations and playback interfaces of the HDA standard, both of which are optional interfaces.

- **Supported attributes**

These are the standard HDA attributes supported by the server:

- OPCHDA_DATA_TYPE: Specifies the data type for the item.
- OPCHDA_ITEMID: Specifies the item id.

- **Supported aggregates**

This server supports the following standard aggregates defined by the OPC Foundation:

- OPCHDA_AVERAGE: The average data over the resample interval.
- OPCHDA_TOTAL: The total value (time integral) of the data over the resample interval.
- OPCHDA_COUNT: The number of raw values over the resample interval.
- OPCHDA_INTERPOLATIVE: Used for interpolated values.
- OPCHDA_START: The value at the beginning of the resample interval. The timestamp is the time of the beginning of the interval.

4.2. Easy Interface for Database Configuration

Integration Objects' OPC HDA Server for MS Access offers an intuitive graphical user interface to manage the supported databases. This configuration tool aims to configure tag IDs, data types, access rights, start bound, end bound, etc. Database configuration should be saved in an XML file for the next OPC server launch (configured tags are not added at runtime). The user can also Import/Export tags configuration from/to a CSV file.

At server start-up, it loads the configuration file to create tags and builds its address space. The server also retrieves all database information like tables names, matching between tags and columns names to allow transactions from server to database.

4.3. Building OPC HDA Server Address Space

The user has two options to build the OPC HDA server address space:

- **Automatic discovery:** by default, the OPC HDA Server automatically browses the configured historian tables to identify the OPC tags list and created its address space accordingly.
- **Import tags from CSV file:** The user can edit his current configuration by importing the CSV file specifying the OPC tags to be included in the address space of the OPC HDA server. The CSV file is to be generated using the OPC tag configuration tool.

5. System Requirements

- This application was successfully installed and executed under the following operating systems: Windows XP, Windows 2003, Windows Seven, Windows Server 2008, Windows 8 and Windows Server 2012
- You should install Microsoft.Jet.OLEDB.4.0 (from MDAC).
- You should have a compliant MS Access installation anywhere in your network.
- Required OPC DLLs (described in more details in the next chapter).
- An OPC HDA client compliant with OPC HDA 1.1 and 1.2 standards (see http://www.integrationobjects.com if you need to download Integration Objects OPC HDA client "OPC DA HDA Archiver").

GETTING STARTED

1. Pre-Installation Considerations

First of all, you have to install OPC Proxy/Stub DLLs. These DLLs are opccomn_ps.dll, opcproxy.dll and opchda_ps.dll.
You can download these files from the OPC Foundation web site. You can follow this link to download the OPC Core Components:
https://opcfoundation.org/developer-tools/developer-kits-classic/core-components/

These DCOM proxy/stub libraries are copied on the target system under the "system32" folder. You should register them using the **regsvr32** command.

Example (Windows XP)
/regsvr32 "C:\WINDOWS\system32\opcproxy.dll" (if your system drive is “C:”)
/regsvr32 "C:\WINDOWS\system32\opccomn_ps.dll"
/regsvr32 "C:\WINDOWS\system32\ opchda_ps.dll"

OLE DB provider for MS Access should be installed on your machine. You can get it from the Microsoft Data Access Components **(MDAC) 2.8**

2. Installing OPC Server

1. Double-click the download exe file **IntegrationObjects'OPCHDAServerforMSAccess.exe**. A WinZip Self-Extractor application starts. It automatically begins extracting the files. The installation will also start automatically.
2. Follow the installer wizard as it guides you through the rest of the installation.

The installation copies all necessary files to the target folder, creates a shortcut icon to invoke the OPC server in the start menu and makes an un-installation entry in the Add/Remove Programs Window in the Control Panel.

Click on **Start ➔ Programs ➔ Integration Objects ➔ OPC HDA Server for MS Access ➔ OPC HDA Server for MS Access**

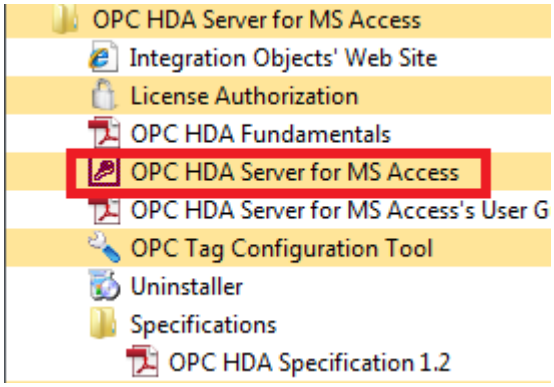


Figure 2: Start Menu

For remote connection to this OPC server, you can apply the following registration files copied in the "Integration Objects\Integration Objects' OPC HDA Server for MS Access\Reg Files" directory under the Program Files folder:

- NT_RemoteServerReg.reg: This file contains registry information for adding the necessary registry entries on a Windows NT 4.0 platform. It should be copied on a client machine then applied. This allows remote access to the OPC Server for MS Access in a distributed configuration.
- Win2K_XP_RemoteServerReg.reg: This should be applied on Windows Server 2000 or Windows XP Professional platforms.

3. Server Registration

In compliance with the OPC and COM specifications, the OPC Server for MS Access creates the following registry entries under HKEY_CLASSES_ROOT when installed on the target system. These entries are removed when the server is uninstalled:

Folder	Description
IntegrationObjects.OPCHDA.MSA	Integration Objects OPC HDA Server for MS Access; http://www.integrationobjects.com
IntegrationObjects.OPCHDA.MSA \CLSID	{***CLSID***} = {7A938F51-78FB-4ae8-A330-31A01DC16E1E}
CLSID\{***CLSID***}	Integration Objects OPC HDA Server for MS Access; http://www.integrationobjects.com
CLSID\{***CLSID***}\AppID	{***CLSID***}
CLSID\{***CLSID***}\Implemented Categories\{7DE5B060-E089-11d2-A5E6-000086339399}	Historical Data Access Specification 1.1

CLSID\{***CLSID***}\LocalServer32	Path to server executable (OPCHDAMSAccess.exe).
CLSID\{***CLSID***}\ProgID	IntegrationObjects.OPCHDA.MSA

Table 2: Folders in the Distribution

Command-Line

The OPC server can also be registered manually using command-line parameters for the server executable in a DOS Window:

[-RegServer] or [/RegServer]

This adds the necessary entries into the system registry.
For example, in the command prompt window, you type:

Prompt> OPCHDAMSAccess.exe –RegServer

Prompt is the path of the target directory where the IODBServer.exe is located.

4. Starting Up

The OPC Server for MS Access can be started manually from the OPC server's shortcut in the "Programs" menu or activated dynamically when an OPC HDA client is connecting or by the *service control manager* if it is running as NT service.
If the server is not currently running, COM will attempt to launch it when the first client attempts to connect to it.
If it runs as a local executable, when starting, a small icon appears in the tool tray at the right-hand side of the Task Bar.
In some cases, when a remotely connecting client tries to launch the OPC server, the OPC server's GUI (if it runs as executable) does not show up. This is due to your DCOM configuration. You should set the Identity for your OPC Server to "The interactive user".

Right click on the icon to display the following menu.

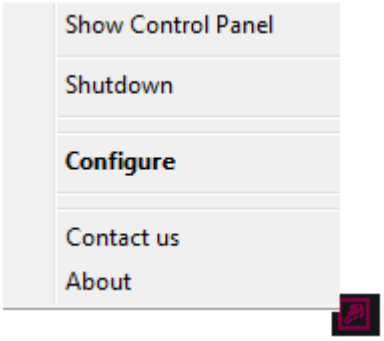


Figure 3: Server Menu

Click **Show Control Panel** to display the control panel for the server (Figure 4). The control panel displays general statistics for the server, such as the number of clients currently connected. It also displays the status of the server. The server state can be RUNNING or SUSPENDED.

Click **Shutdown** to stop the OPC server.

Click **About** from the menu to display the About Box for the server. The About Box contains the server name and version number as well as other information about the software and Integration Objects.

Click **Contact us** to get helpful links to ask for further information about this product and Integration Objects.
Click **Configure** to configure this server (the next chapter explains in detail this server functionality).

5. Removing OPC Server

You can manually remove all registry information related to this OPC server by using one of the following command-line parameters:
[-UnregServer] or [/UnregServer]

To remove the server from your machine, just click the **Uninstaller** short-cut icon in the Start menu.

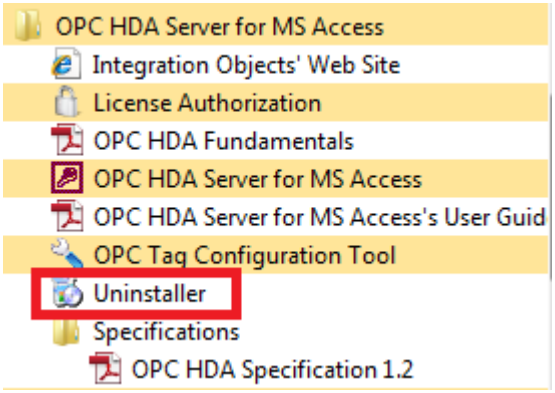


Figure 4: Menu

This OPC server can be removed manually as follows:

1. Click **Start**.
2. Click **Settings**.
3. Click **Control Panel**.
4. Click **Add/Remove Programs**.

5. In the Add/Remove Programs dialog screen select “Integration Objects OPC HDA Server for MS Access”.
6. Click **Change/Remove** then **OK**.

➔ The software will be removed with all registry entries and shortcuts created by the installation.

CONFIGURATION

1. Introduction

Users can configure this server through an intuitive GUI (Graphical User Interface) for MS Access database management.

We distinguish these main sections:

- Server Configuration: Describes how to configure server object links to ADO-Compliant database (MS Access).
  Tag Configuration: describes how to export OPC Tags list from a historian table using the OPC configuration tool.
- Alias Configuration: User has the ability to configure alias names for tags with complicated identifications.
- General Configuration: Contains server logging parameters and database recovery settings used once the connection is lost.

The main window consists of a menu bar (1), a toolbar (2) and two sub-views (a tree view (3) and a list view (4)).

Note: You should run this server with Administrator privileges.

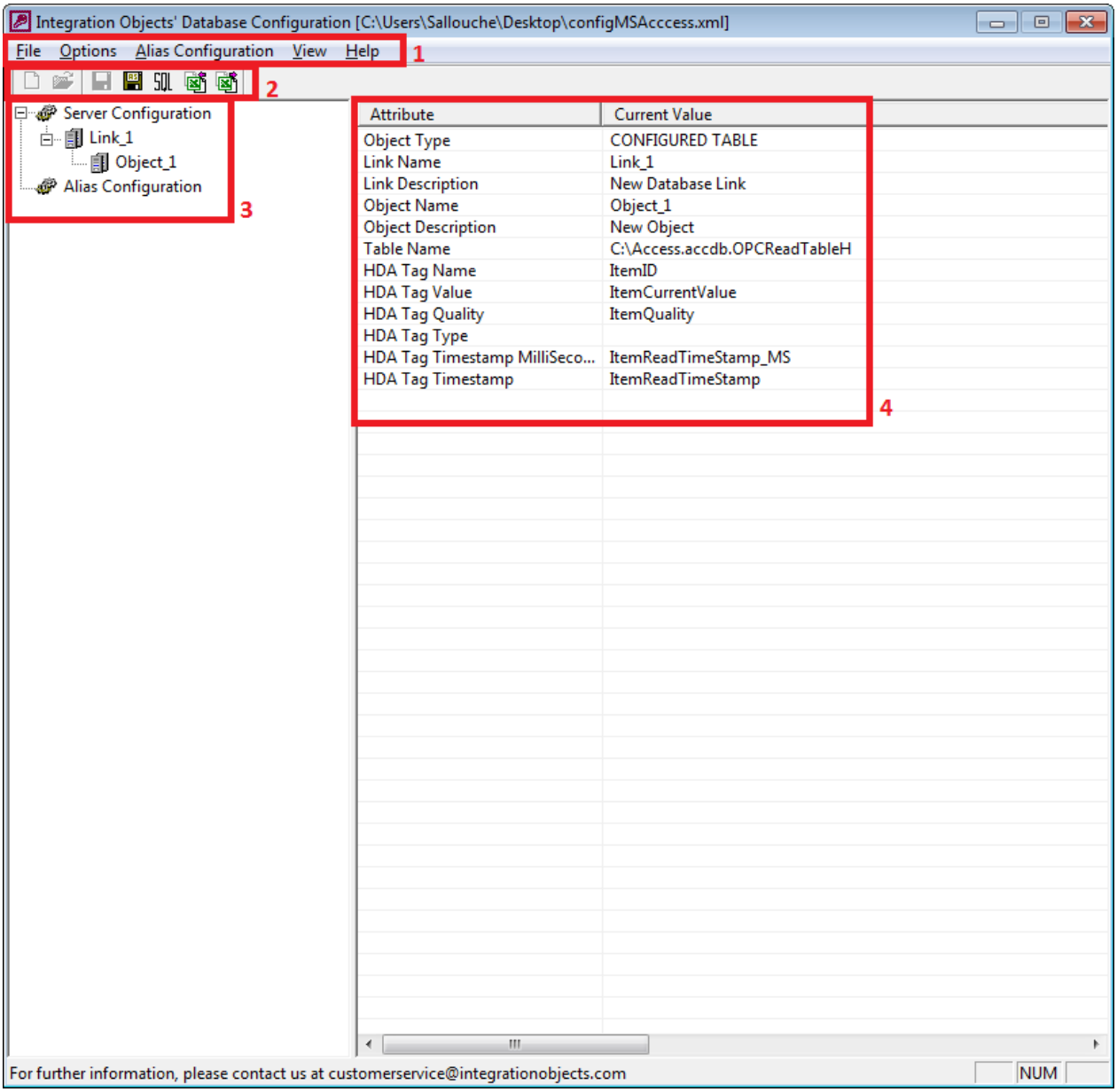


Figure 5: GUI

2. OPC Tag Configuration

Users can specify the OPC tags to be included in the OPC HDA server address space by using the OPC tag configuration tool. This tool allows you to export the selected OPC Tags list from your historian table into a specific CSV file. This file will be afterwards imported to the OPC HDA Server for SQL Server.

To do so, you need to follow these steps:

1. Run the OPC tag configuration tool using an administrator account.
2. Configure the parameters to connect to the MS Access historian.

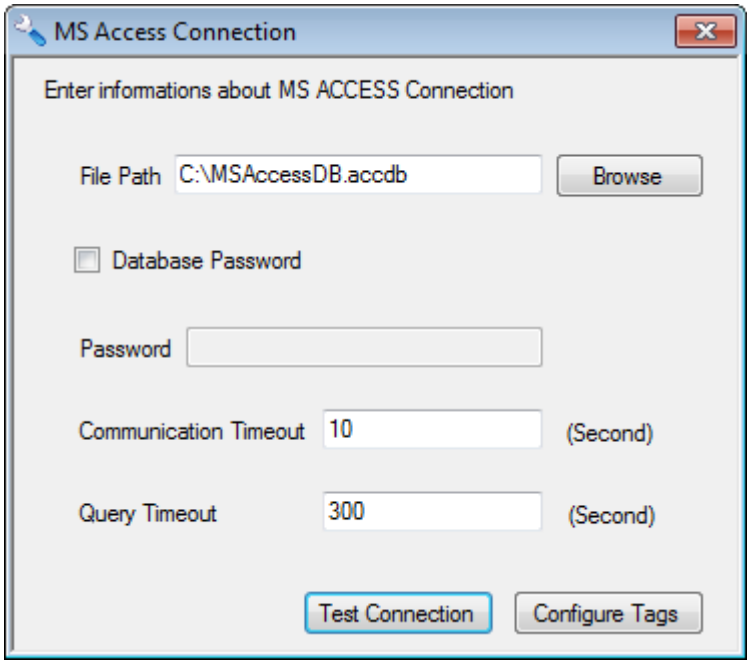


Figure 6: Configuration connection

1. Select the Runtime database and then click the **Configure Tags** button to proceed.
2. Select the historian table name.

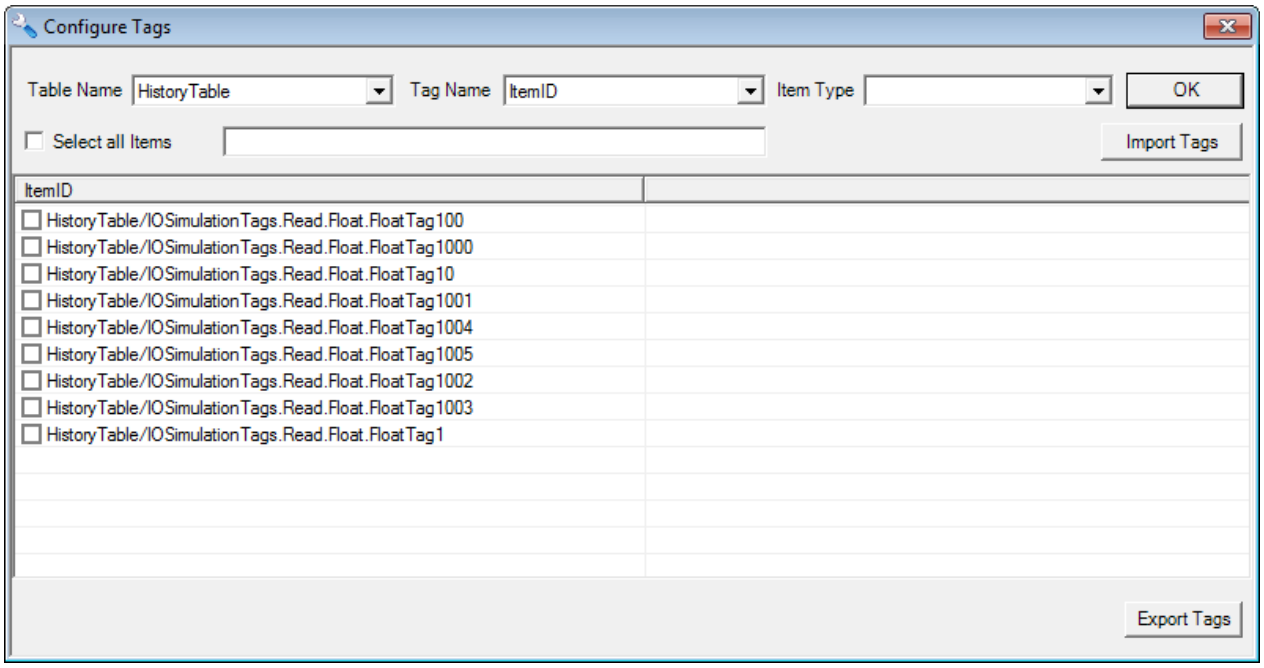


Figure 7: Configure tags

3. Select the tags you want to import into the OPC HDA Server for MS Access.
4. Click **Export Tags** to export the configuration into CSV file.

You can also use this tool to validate an existing CSV file by following the steps below:

1. Click **Import Tags** button
2. Select the CSV file from the displayed window
3. Check the needed OPC tags. Note that only the valid tags displayed with green color can be checked.

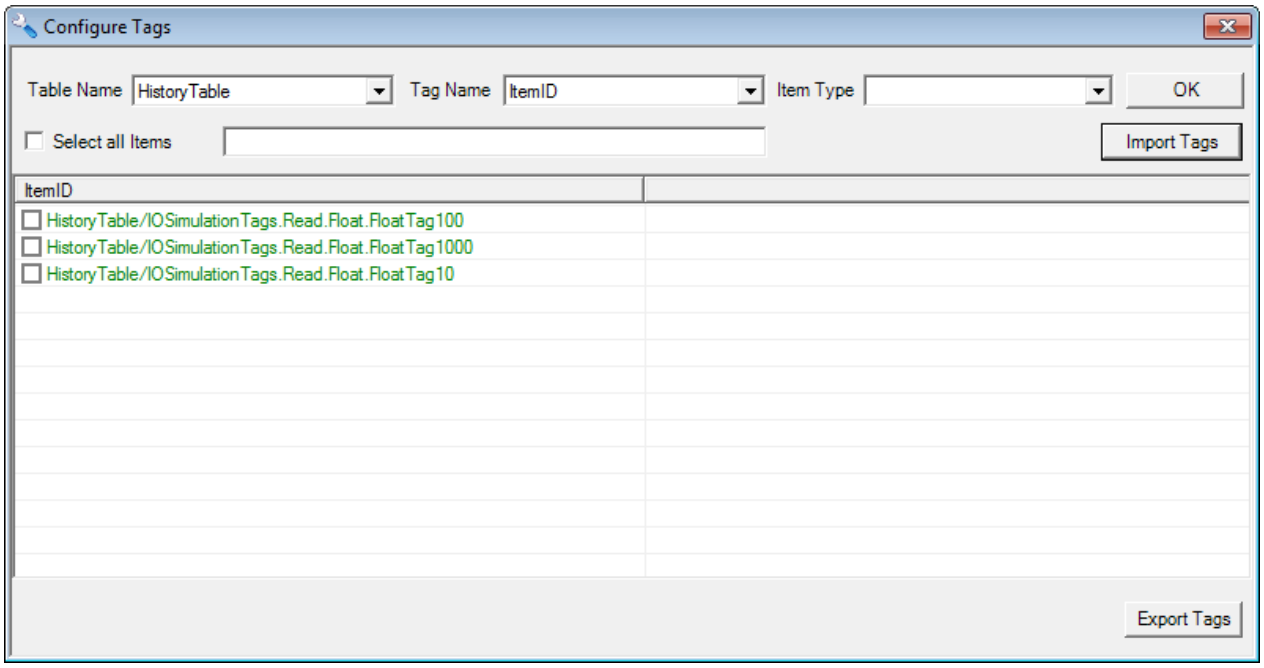


Figure 8: Validate OPC tags

4. Click **Export Tags** button.

3. Server Configuration

The main tree view for this server interface is composed of two main nodes: Server configuration and Alias configuration. Select one object under one of these nodes to display its current configuration.

3.1. Configuring the Server Link

This section describes server link management: add, remove, enable/disable, connect, view configuration and edit the current configuration

3.1.1. Adding a Server Link

Select the “Server Configuration” node, double-click and select **“new”**.

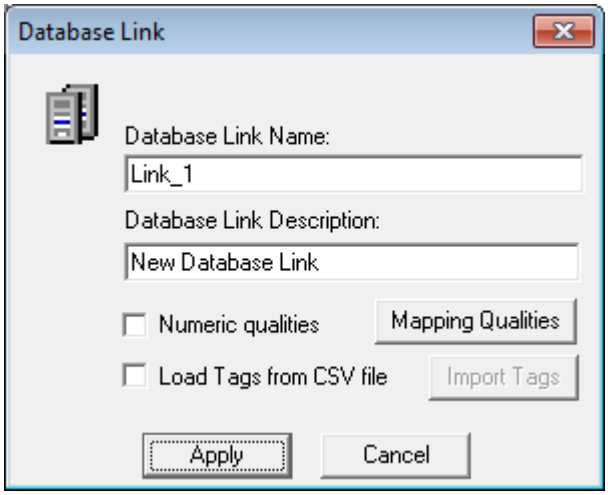


Figure 9: New Server Link

Parameter	Description
Link name	The server link name.
Link description	The server link description.
Enable server link	You should check this box to enable the server link. Server enabling lets you add an Object link and allow matching between column names and tag attributes. This option can be set later.
Numeric qualities	Configured qualities can be string or integer type (default type is string). Once this option is set, you can't modify it if you want to edit server link configuration.
Load Tags from CSV file	Use this option to specify the tags to be included in the OPC Server address space. This option overrides the automatic discovery of the tags. The CSV file is to be generated by the OPC tag configuration tool.

Table 3: Link Parameters

3.1.2. Mapping Qualities

You should click on this button to configure your own quality values. You will get the following screen:

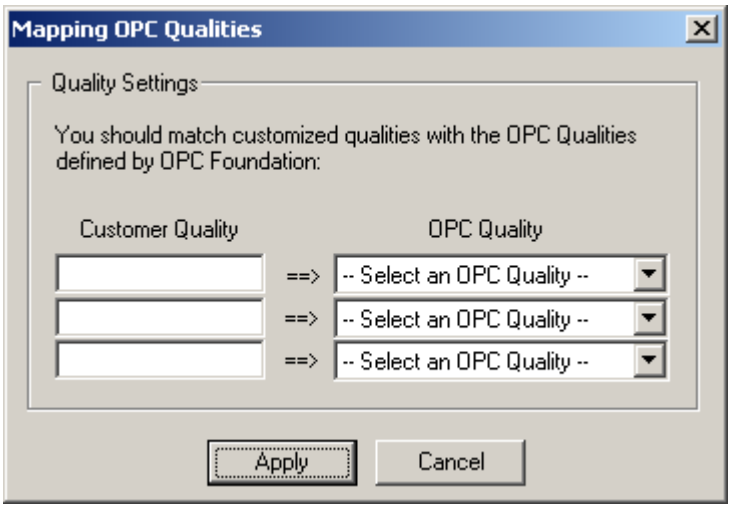


Figure 10: Mapping Qualities

Example:
Configuring customer quality with string value:

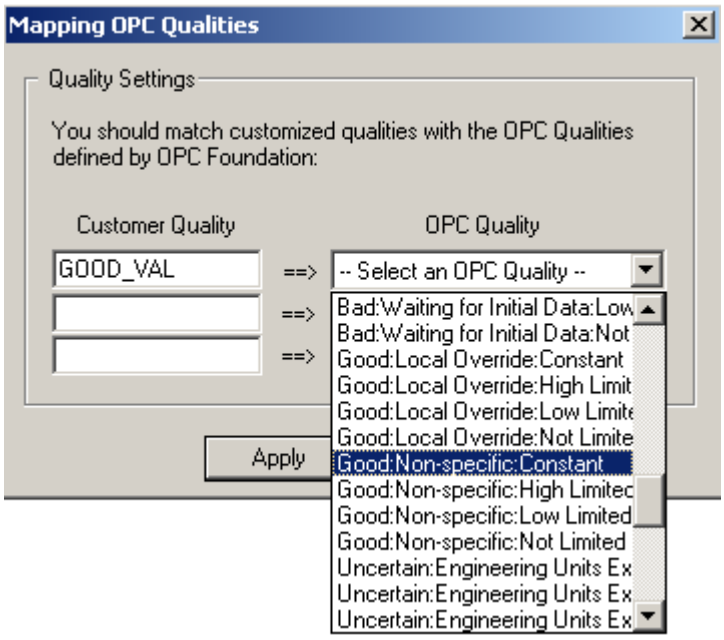


Figure 11: Mapping Qualities (Example)

After clicking the **Apply** button, the new server link is added as follows:

Figure 12: Added Server Link

Click **View** to view server link configuration.

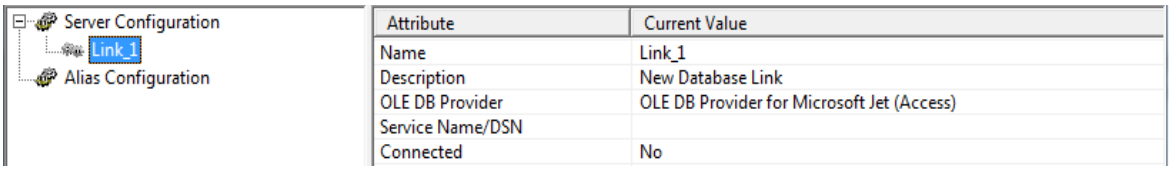


Figure 13: View Added Server Link

3.1.3. Enabling/Disabling a Server Link

To allow object links management, you should enable your configured server link. You can enable your server link in two ways:

1. Check the **"Enable server link"** option when adding the server link.
2. You may double-click on the selected server link node and click the **Enable** menu item.

To disable it, you should double-click on the selected server link node and click the **Disable** menu item.

3.1.4. Connecting a Server Link to the Database

Every server link is a link to an MS Access database using the appropriate connection string. This OPC server uses the ADO technology to manage databases.
You will get the following screen when clicking **Connect** (in the menu):

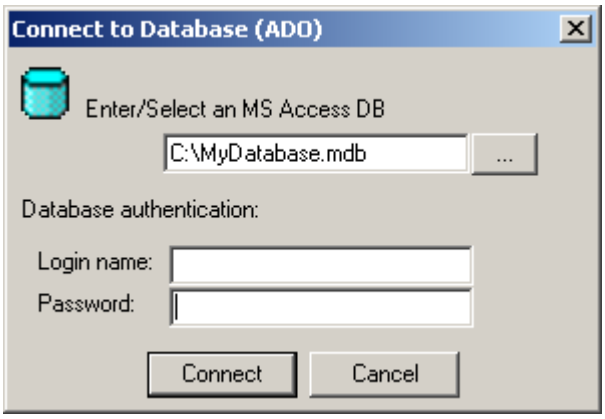


Figure 14: Connect

You should specify your database (*.mdb,*accdb).
The following are the connection parameters:

Parameter	Description
Document	The database name (*.mdb,*.accdb)
Login	The login.
Password	The password.

Table 4: Database Parameters

Once the connection to the database is established, the server link state is set to ***Connected*** as follows:

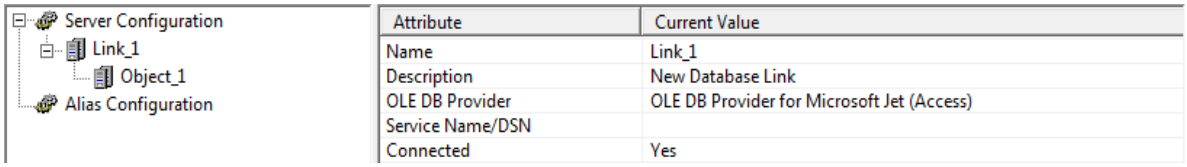


Figure 15: Connected Server

3.1.5. Editing a Server Link

To modify server settings, click **Edit** (double click menu). You will get the same screen as the "Add" one.
You can modify the following settings:

- Server link name.
- Server link description.
- Enabling/disabling the server link.
- Mapping qualities: You can't change the type of customized qualities (the type is already set when adding the server link). You can add new configured qualities.

3.1.6. Renaming a Server Link

If you only want to change the server link name, click **Rename** (double click menu). You will get the following screen:

Figure 16: Rename Server Link

3.1.7. Deleting a Server Link

To remove a server link, click **Delete** (double click menu). This will remove all object links configured under this server link.

3.1.8. Deleting All…

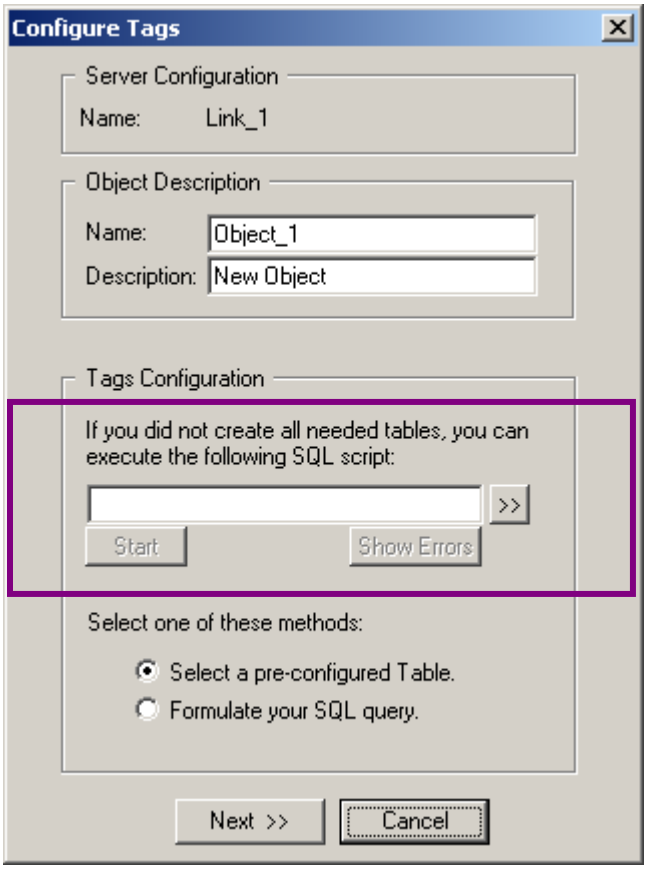
To remove all configured object links under the selected server link, click **"Delete All…"** menu item (double click menu).

3.2. Configuring the Object Link

This section describes object link management: add, remove, enable/disable, view configuration and edit the current configuration.

3.2.1. Adding an Object Link

Select the server link node to which you want add an object link (under the "Server Configuration" node), double-click and select **"new"**.



You may execute a SQL script.

Figure 17: New Object Link

1st Step: Select a method for matching column names with OPC tag attributes

Parameter	Description
Object name	The object link name.
Object description	The object link description.
Method of tags configuration	• Select a pre-configured table. • Formulate your SQL query.

Table 5: Object Parameters

2nd step: Matching column names with OPC tag attributes

There are two ways to map tag attributes with column names:

- You can you use a pre-configured table. In this case, the table should be already created.
- You can also formulate your SQL query.

Pre-configured table

You should select the first option “Select a Pre-configured Table” under the “Tags Configuration” section.

At the beginning, all fields are disabled until you select a table name from the list box or you enter a known table name.

You can then match OPC tag attributes including ‘Tag Name’, ‘Tag Value’, ‘Tag Timestamp’ and ‘Tag Quality’ with column names of the selected table.

Note that filling in the ‘Tag Name’, ‘Tag Value’ and ‘Tag Timestamp’ fields is mandatory. The selected columns should also be distinct. Otherwise, the object is rejected.

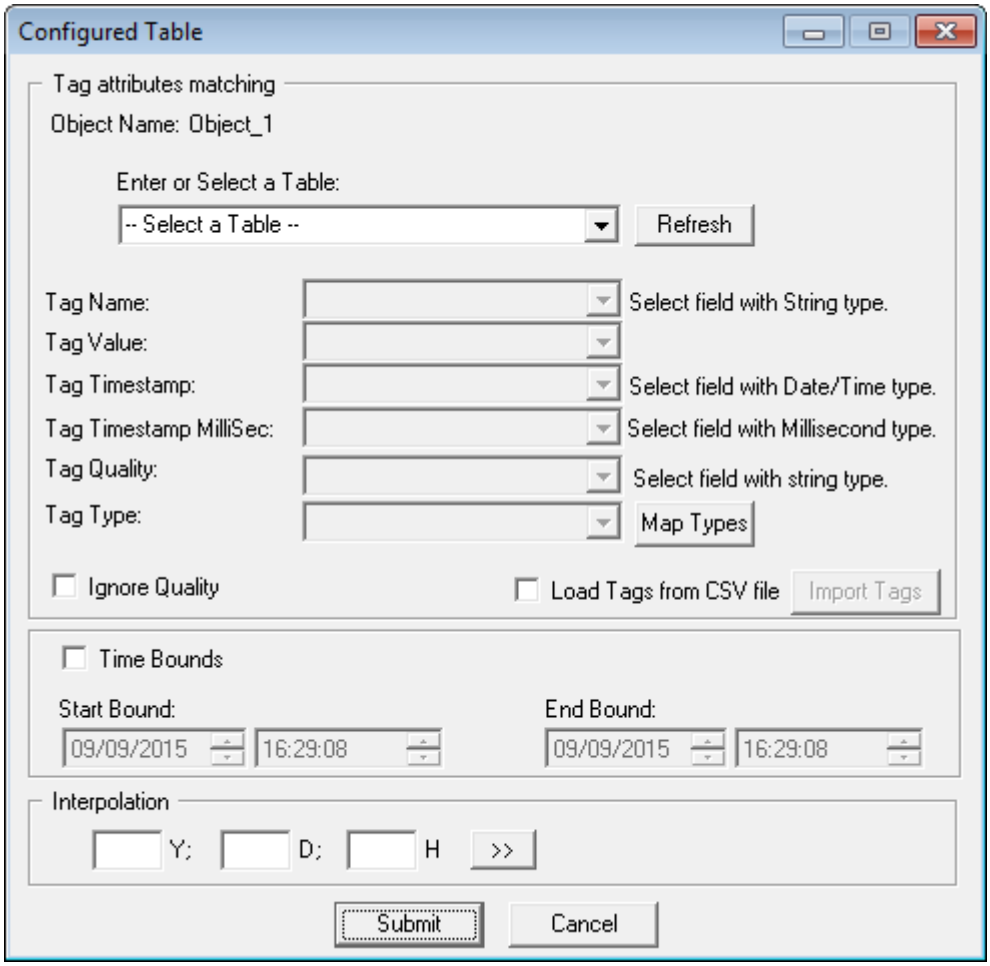


Figure 18: Pre-configured Table

Parameter	Description
Table name	The selected table name.
Tag Name	Select a column from the selected table from which the tag name will be collected.
Tag Value	Select a column from the selected table from which the tag value will be collected.
Tag Timestamp	Select a column from the selected table from which the tag timestamp will be collected.
Tag Timestamp Millisec	Select a column from the selected table from which the tag timestamp millisecond will be collected.
Tag Quality	Select a column from the selected table from which the tag quality will be collected.
Start Bound	The start bound used in HDA reads. This is an

	optional configuration parameter.
End Bound	The end bound used in HDA reads. This is an optional configuration parameter.

Table 6: Table Fields

Example:

After selecting a table name, all fields will be activated. For each tag attribute, you can select the column name that you want. The following screen shows an example:

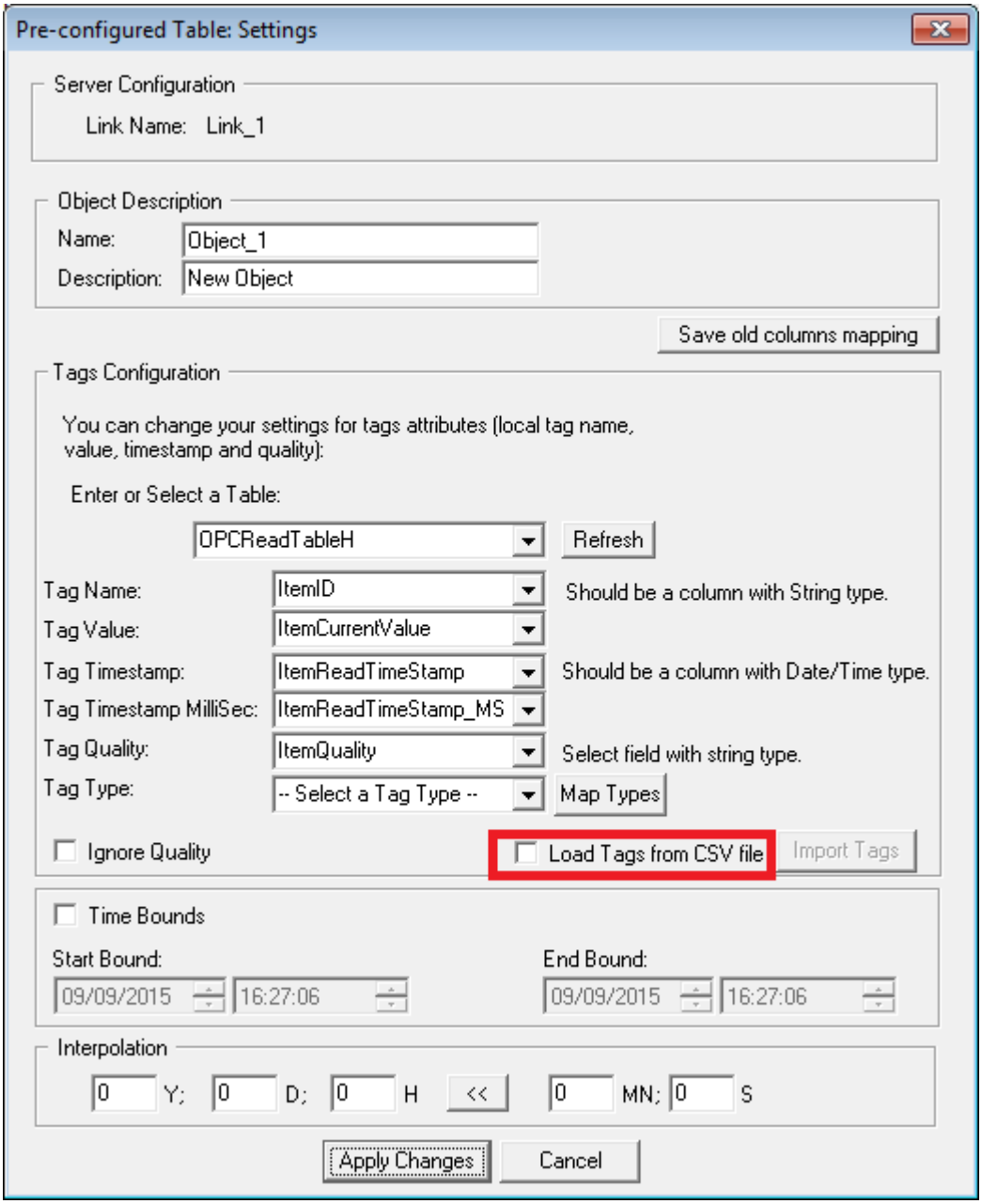


Figure 19: Pre-configured Table (example)

You can load OPC tags from CSV file by checking the Load Tags from CSV File button (highlighted in red in the screenshot above), click Import tags and then select the generated CSV file using the OPC Tag Configuration Tool from the displayed window. Using this option, the OPC HDA Server will not use the automatic discovery feature to build its address space. It will only load the tags included in the CSV file. For more details, refer to the OPC Tag Configuration section.

After applying your changes, you need to save the current configuration.

Refresh table names list

To refresh the table names available on your machine, click the **Refresh** button.

Bounding

This is optional.
You can select the start and end time that will be used in HDA reads.

Interpolation

This is optional.
You can customize the time range for Read at time requests concerning interpolated values. The time range is [ftReadAtTime – Interpol, ftReadAtTime + Interpol] with ftReadAtTime the filetime argument passed in the ReadAtTime request and Interpol is the configured time range.
Interpol = x_1**Y** + x_2**D** + x_3**H** + x_4**MN** + x_5**S**.

Y: Year.
D: Day.
H: Hour.
MN: Minute.
S: Second.

Interpolation parameters:

Example:
Interpol = 0Y+ 0D + 1H + 0MN + 0S.

3.2.2. SQL Query

You should select the second option “Formulate your SQL query” under the “Tags Configuration” section.

You can enter any type of SQL query (CREATE, INSERT, update, select …). Only “SELECT” queries are considered for column mapping. Note that conditions in the SELECT query are ignored (example: SELECT * FROM Test WHERE (condition1) ➔ condition1 is ignored). You can then match OPC tag attributes including ‘Tag Name’, ‘Tag Value’, ‘Tag Timestamp’ and ‘Tag Quality’ with the selected column names.
You enter your SQL query in the edit text. Then, click the Execute button. Error messages are displayed in the edit text (“Success” for succeeded queries).

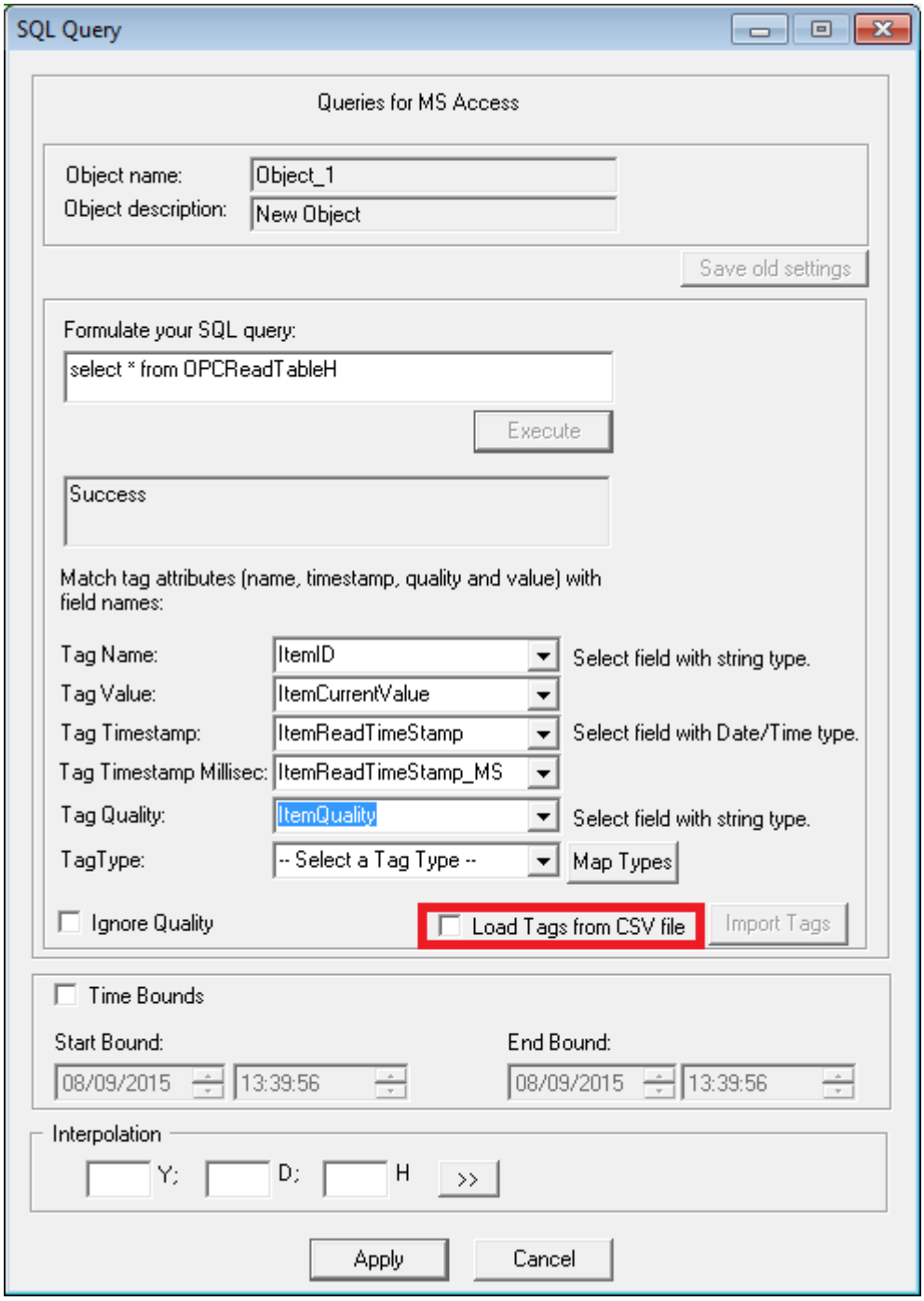


Figure 20: SQL Query

You can load OPC tags from CSV file by checking the Load Tags from CSV File button (highlighted in red in the screenshot above), click Import tags and then select the generated CSV file using the OPC Tag Configuration Tool from the displayed window.

Using this option, the OPC HDA Server will not use the automatic discovery feature to build its address space. It will only load the tags included in the CSV file. For more details, refer to the OPC Tag Configuration section.

After applying your changes, you need to save the current configuration.

Example:
SQL query = **"SELECT * FROM** OPCReadTableH".

3.2.3. Enabling/Disabling an Object Link

To allow HDA reads/updates to tags related to an object link, enable your configured object link. You may double-click the selected object link node and click the **Enable** menu item. To disable it, you should double-click on the selected object link node and click the **Disable** menu item.

3.2.4. Editing Object Link

You may change your configuration for an Object link. You should click **Edit** (double click menu) of the selected Object link.

- **Pre-configured table**

You can modify your configuration for an Object link created using a pre-configured table. You will get the following screen:

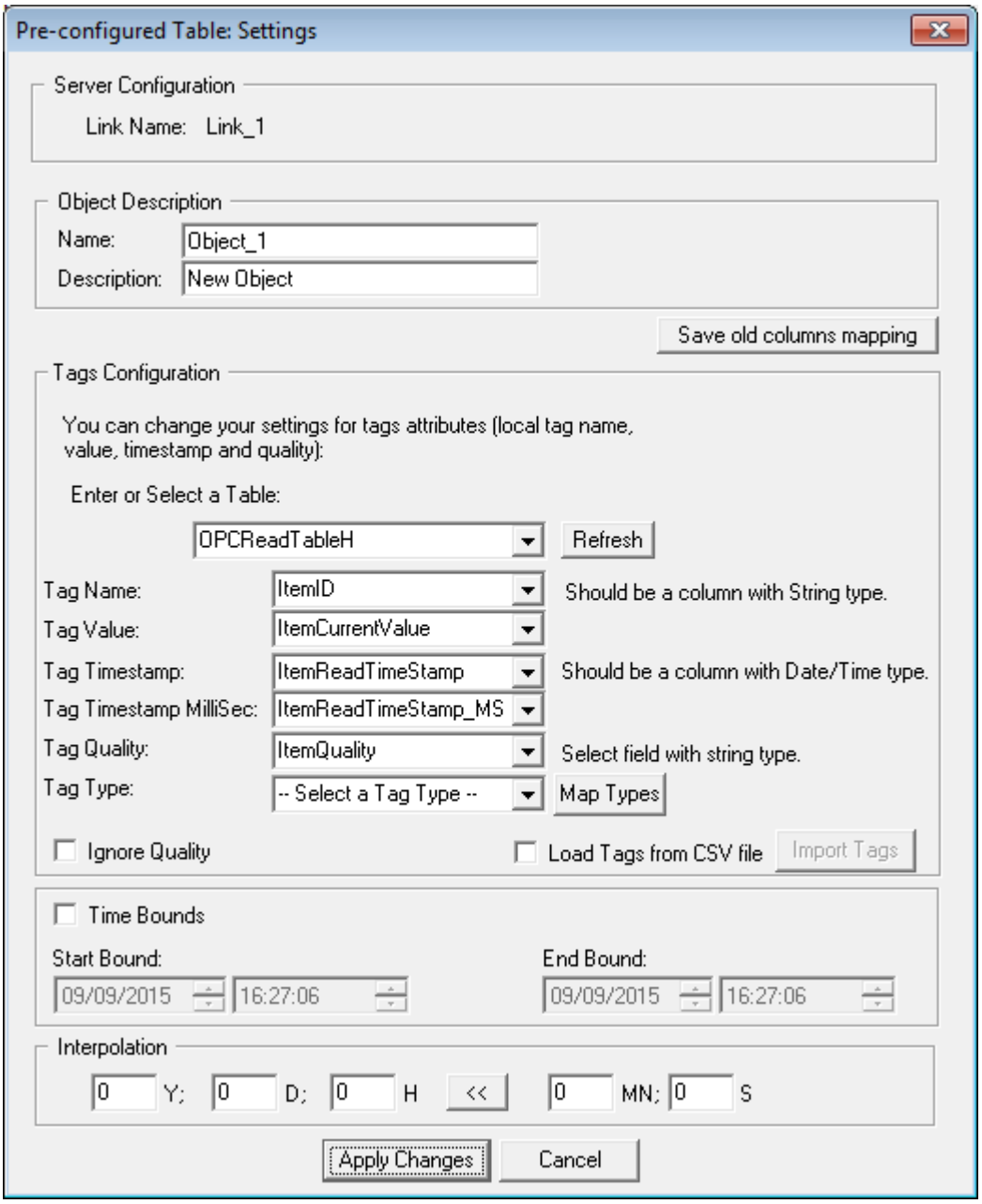


Figure 21: Pre-configured Table (Editing)

○ SQL query

You can modify your configuration for an object link created using an SQL query. You will get the same dialog screen as adding an object link with one difference: the **Save Old Settings** button is activated.

Saving Old Columns mapping

You may keep your old settings for mapping tag attributes with column names.

3.3. Item IDs syntax

The slash "/" character is the separator used for item IDs in this OPC server.
A full tag name is defined as follows:

Server_Link**/**Type_Of_Method**/**Object_Link**/**Column_Name_Value

Possible values for "Type_Of_Method" are "QUERY" and "CONFIGUREDTABLE".

(Ex. Link_1/QUERY/Object_1/tag1).

4. Alias Configuration

Alias configuration is an option offered by Integration Objects' OPC HDA Server for MS Access to avoid the use of complicated full names for tags or to customize some tags' attributes such as access rights.
Each alias name matches an item path in the configured tag tree for the "Server Configuration" node. You may configure one or more alias names for the same item path.

4.1. Configuring the Alias Group

Alias names are gathered in groups.

4.1.1. Adding a Group Alias

To add a new group alias, select the "Alias Configuration" node from the tree view. Then, click the **New** menu item (double click menu). You will get the following dialog screen:

Figure 22: New Alias Group

Parameter	Description
Group Name	The group alias name.

Table 7: Group Name

4.1.2. Renaming Group Alias

If you only want to rename the group, click **Rename** (double click menu). You will get the following screen:

Figure 23: Rename Group Alias

4.1.3. Deleting Group Alias

To remove a group alias, click **Delete** (double click menu). This will remove all aliases configured under this group.

4.1.4. Deleting All…

To remove all configured aliases under the selected group alias, click the **Delete All…** menu item (double click menu).

4.2. Configuring Aliases

4.2.1. Adding an Alias

Select the group alias node to which you want to add an alias (under the “Alias configuration” node), double-click and select **“new”**. You will get the following dialog screen:

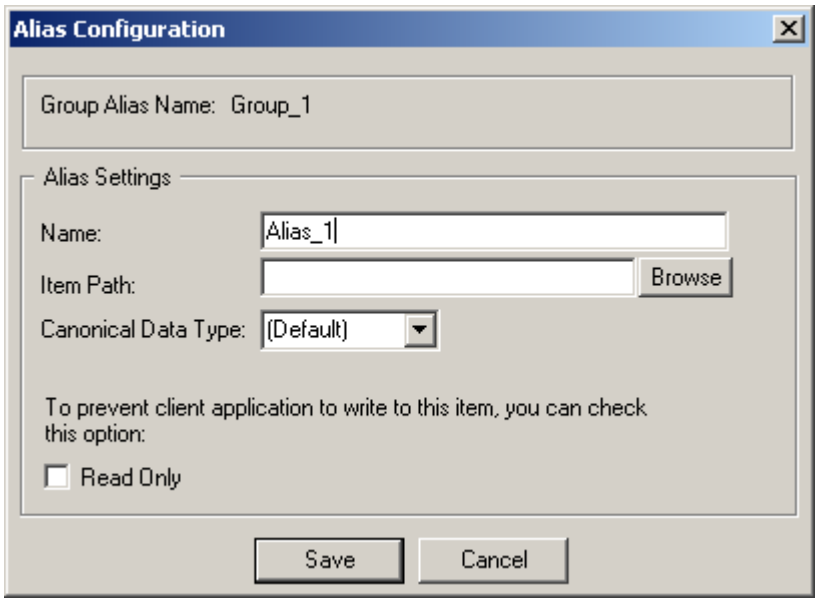


Figure 24: New Alias

Browse configured items tree

To select an item path from the available item tree for the configured alias, click on the **Browse** button. You will get the following screen:

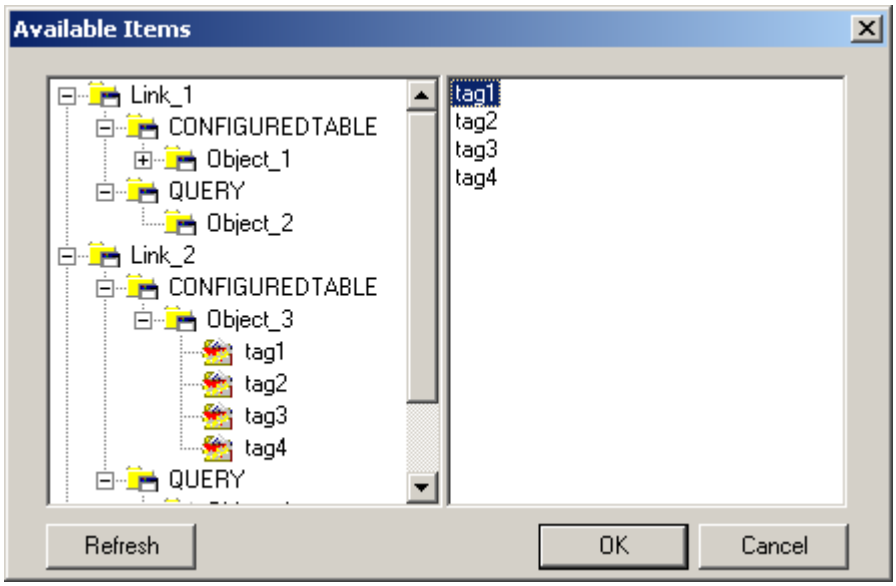


Figure 25: Select an Item

You should double click on the object's name to get the items' list.

Refresh table names list

To refresh the available items tree, click the **Refresh** button.

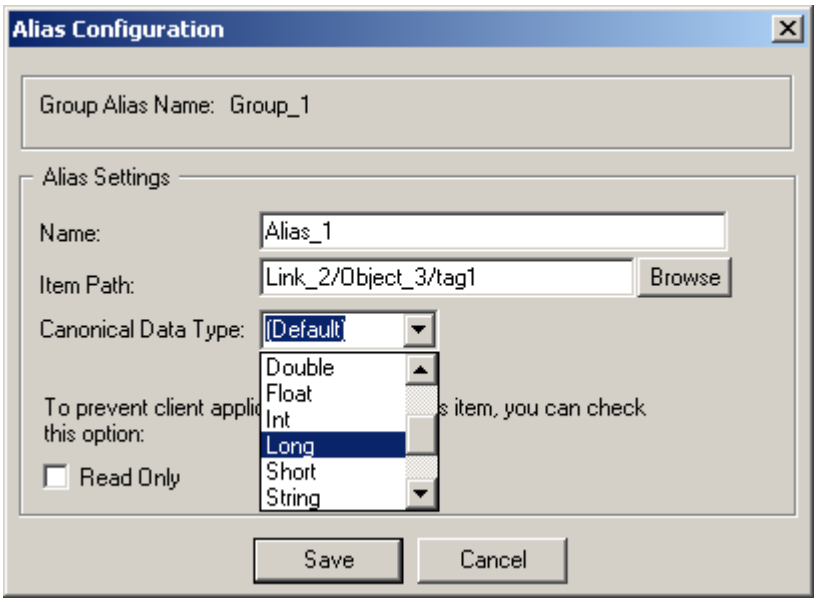


Figure 26: Select Canonical Data Type

Parameter	Description
Alias name	The alias name.
Item Path	The item path.
Canonical data type	Item values returned to the OPC client will be obliged to use this data type.
Read Only	If you want to prevent OPC client applications from writing to this tag, you can check this option.

Table 8: Alias

Canonical data type

Only simple types are available for the current version of this OPC server. You may select one of the following data types:

Displayed data type	VARIANT
Byte	VT_UI1
Short	VT_I2
Long	VT_I4
Float	VT_R4

Date	VT_DATE
Double	VT_R8
String	VT_BSTR
Int	VT_INT
Unsigned Int	VT_UINT
Unsigned short	VT_UI2
Unsigned long	VT_UI4
Boolean	VT_BOOL
Char	VT_I1

Table 9: Tag type

"(Default)" type is the default type of the item path.

Once the add operation succeeds, the tree view and the list view are updated as follows:

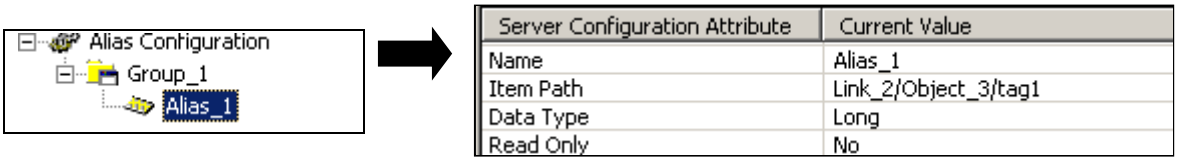


Figure 27: View Alias

4.2.2. Renaming Alias

If you only want to rename the alias, click **Rename** (double click menu). You will get the following screen:

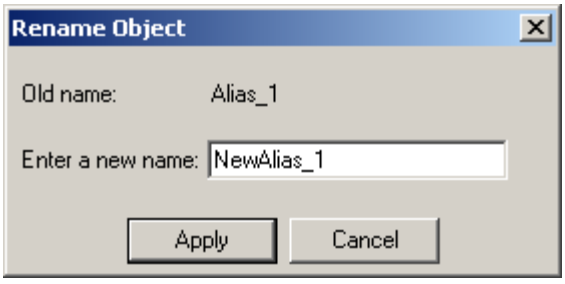


Figure 28: Rename Alias

4.2.3. Editing Alias

You may change configuration parameters for an alias. To do this, click the **Edit** menu item for the selected alias. You will get the same dialog screen as adding an alias.

4.2.4. Deleting Alias

To remove an alias, click the **Delete** menu item.

4.3. Importing/Exporting Alias Configuration

You may import/export your alias configuration stored in a CSV file. To do so, click on the **Alias Configuration** menu.

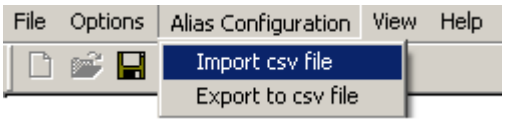


Figure 29: Alias Configuration Menu

Or, you can click on the toolbar as follows:

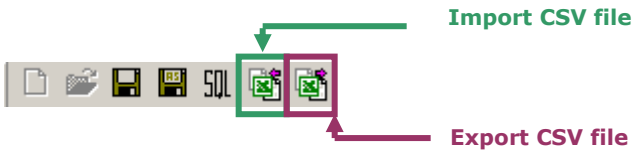


Figure 30: Alias Configuration Toolbar

When importing a CSV file, your current alias configuration will be replaced by the loaded one.

CSV File format

Comma-delimited text file (CSV) contains, in each line, a record that's composed of a set of fields. Fields are separated by a Comma mark. Each field is an alias attribute.

- Group,
- Alias,
- ItemPath,
- DataType,
- ReadOnly.

4.4. Item IDs syntax

Remember that the slash "/" character is the separator used for item IDs in this OPC server.
A tag is defined as follows:
Configured Aliases**/**Group_Alias**/**Column_Name_Value
(Ex. Configured Aliases /Group_1/tag1).

5. General Configuration

Integration Objects' OPC HDA Server for MS Access provides you with a general configuration screen. General configuration includes the following main features:

- You may need to execute some SQL script to create your tables and populate them with rows, before configuring server and object links.
- The database connection may be lost at server start-up or during server running. This OPC server incorporates a database recovery mechanism to restore lost connections.
- This OPC server supports event logging. This feature can be configurable. You may set some logging parameters.

5.1. Executing SQL script

You can execute any SQL script to create all needed tables. To do so, click on the **Options** menu and then on the **Execute SQL script** menu item (Figure 29).

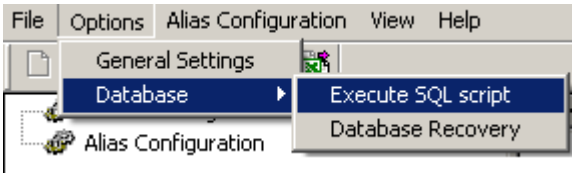


Figure 31: Execute SQL Script Menu

Or, you can click on the toolbar as follows:

Figure 32: Execute SQL Script Toolbar

You will get the following dialog screen:

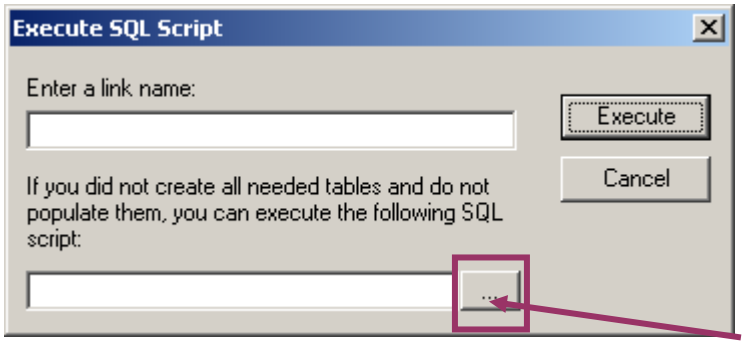


Figure 33: Execute SQL Script

First of all, you must enter a valid link name (already connected to a database).
Then, you may enter the full path of an SQL script (*.sql) or click the browse button to show a file dialog screen that allows you to select the script file.

Then, click the **Execute** button to execute the SQL script file. If errors occur while executing this script, you will get a dialog with an error icon like the following:

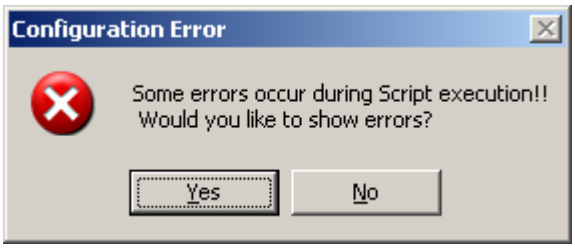


Figure 34: Configuration Error in Executing SQL Script

If you want to view the log event file for script execution, click **Yes**.

5.2. Database Recovery

You can customize database recovery parameters. To do so, click on the "Options" menu ➔ "Database" menu item ➔ "Database Recovery" sub menu item (Figure 33).

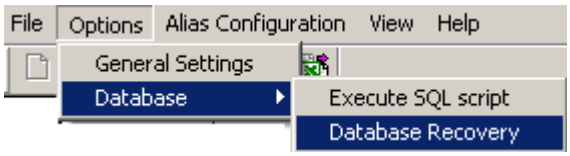


Figure 35: Database Recovery Menu

You will get the following dialog screen:

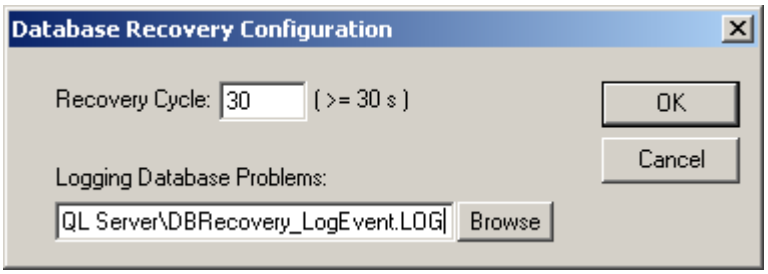


Figure 36: Database Recovery Settings

Parameter	Description
Recovery cycle	The amount of time (in seconds) that the server uses to check for database connection status. If there is any connection problem, the server application tries to restore this connection (at least 30 s).

Log event file path	This is the full path of log event file used to trace database connections problems.

Table 10: Database parameters

5.3. Server/Log Settings

You can configure some general server parameters like server frequency. You can also configure your logging parameters. To do so, click the **Options** menu and then the **General Settings** menu item (Figure 35).

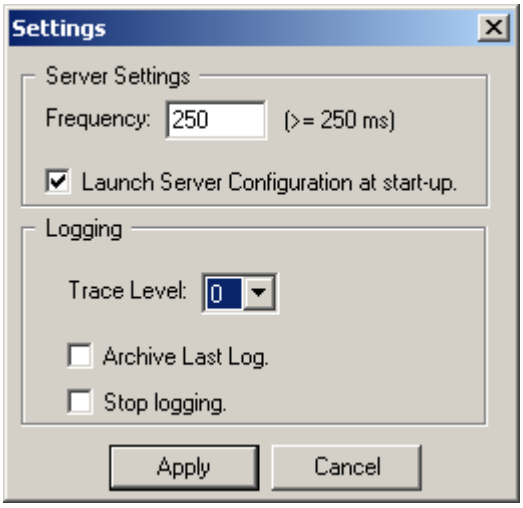


Figure 37: General Settings

Parameter	Description
Recovery cycle	The amount of time (in seconds) that the server uses to check for database connection status. If there is any connection problem, the server application tries to restore this connection (at least 30 s).
Log event file path	This is the full path of log event file used to trace database connections problems.
Trace level	Corresponds to the *LogLevel* parameter in the ini file “DBSrv_CfgFile.ini”. The new loglevel will be taken into consideration in runtime.
Archive last log	Corresponds to the *ArchiveLastLog* parameter in the ini file “DBSrv_CfgFile.ini”. Any modification for this parameter will be taken into consideration in runtime.

Stop logging	To stop logging.

Table 11: Server Settings

5.3.1. View Configuration Errors

You can view the last configuration/database problems. To do so, select the “View” menu ➔ “Error messages” menu item.

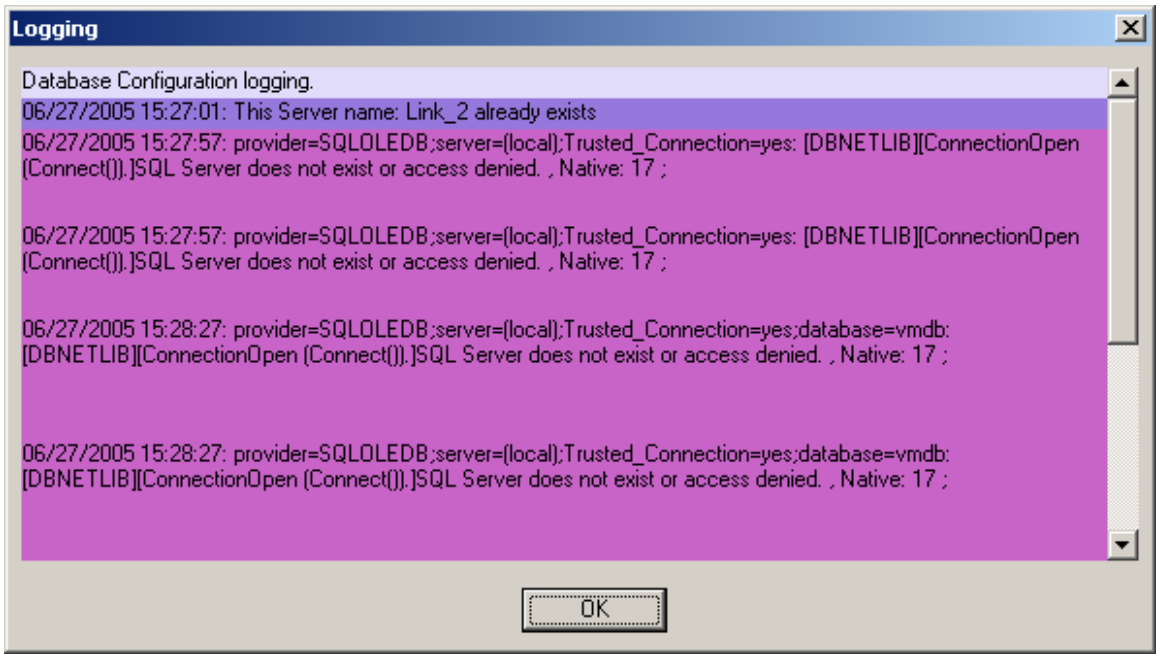


Figure 38: Logging Configuration Errors

DCOM CONFIGURATION

1. Configuring the Server for DCOM

To set up the server:

1. Logon with administrator privileges.

   Choose Run from the Windows Start menu and type **DCOMCNFG** on the computer where your target OPC Server is running, then click **OK** to run the program for configuring DCOM.

   The DCOM configuration utility looks like this:

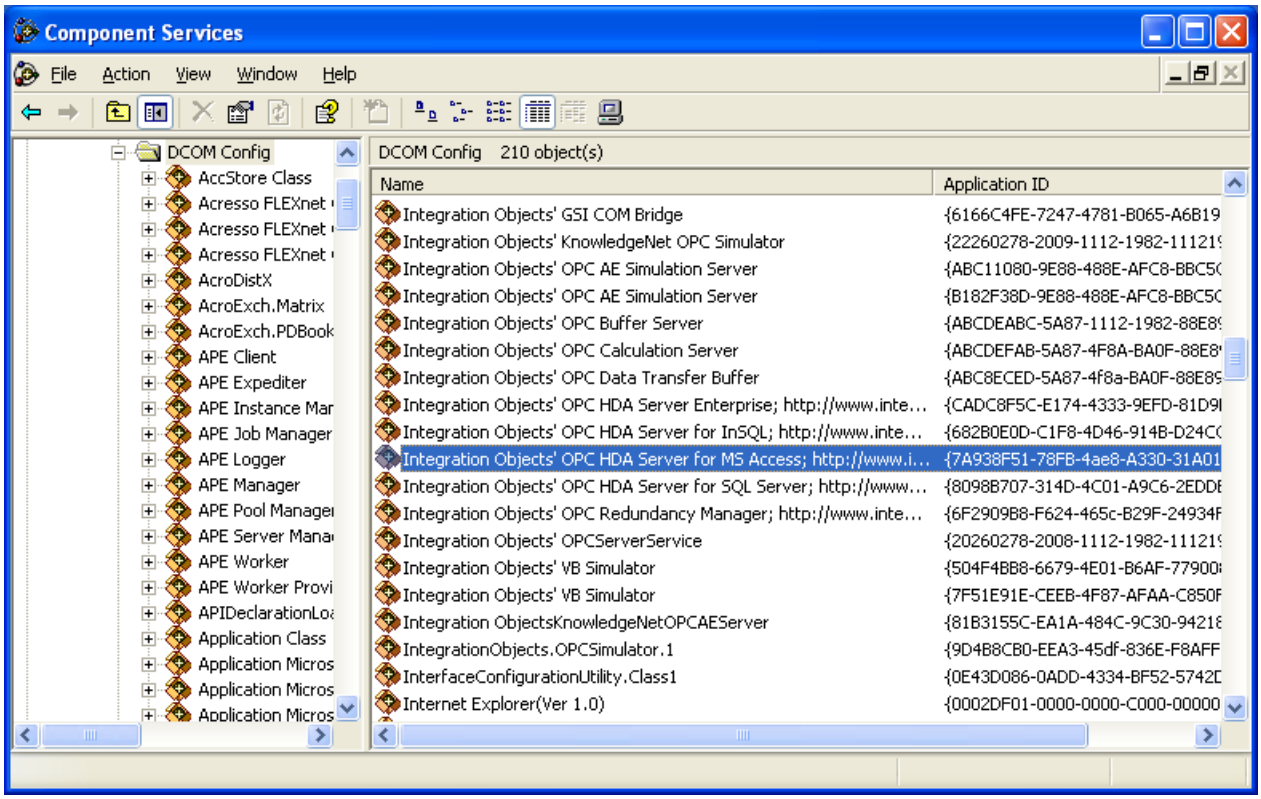


Figure 39: DCOM configuration utility

a. Ensure that the Enable Distributed COM on this computer is enabled.
b. Configure the Default Authentication Level to be Connect.
c. Configure the Default Impersonation Level to be Identity.

2. Configure the Default Properties tab.

The Default Properties tab should look like this:

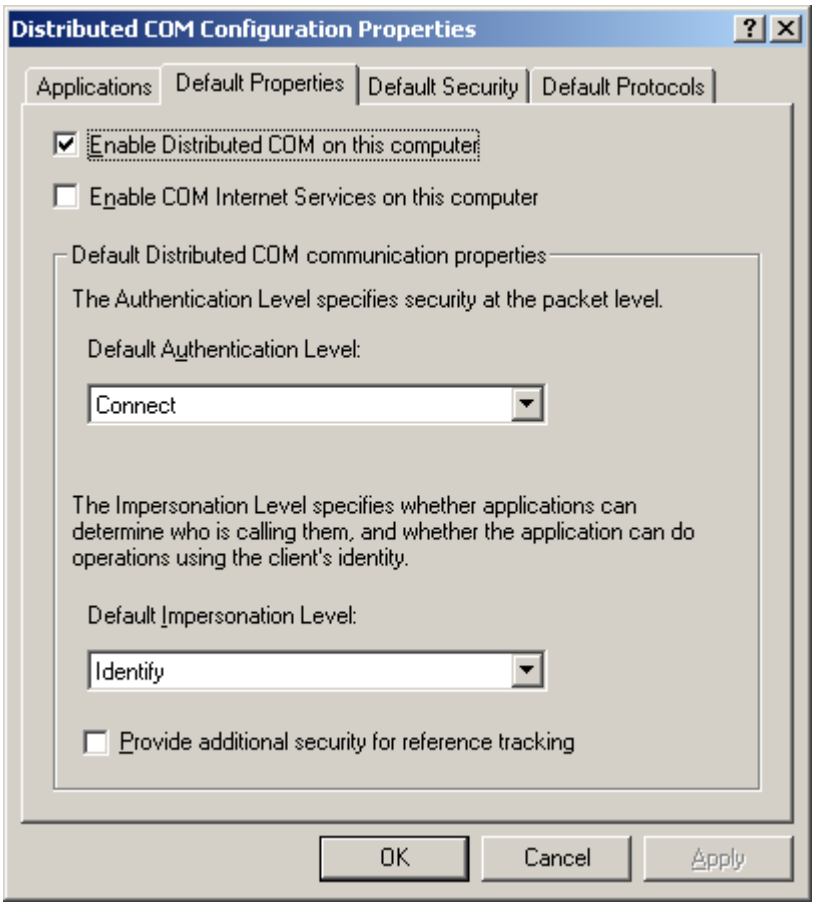


Figure 40: Default Properties tab

To configure DCOM settings for the OPC Server:

1. Click the **Application** tab in **DCOM Configuration** dialog and browse until you find the OPC Server.
2. Select the server and either double-click it or choose **Properties** to configure server-specific settings, as follows:
    a. On the **General** tab, we recommend that you leave the Authentication Level set to Default:

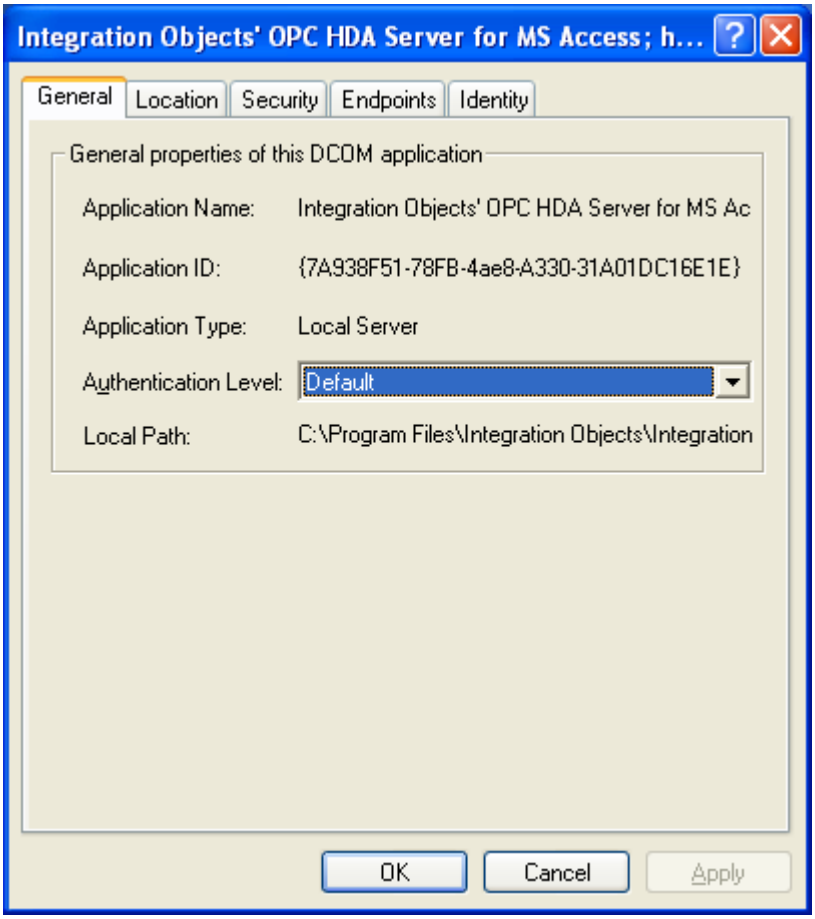


Figure 41: General Tab

b. On the **Location** Tab, ensure that **Run application on this computer** is the only check box that is checked:

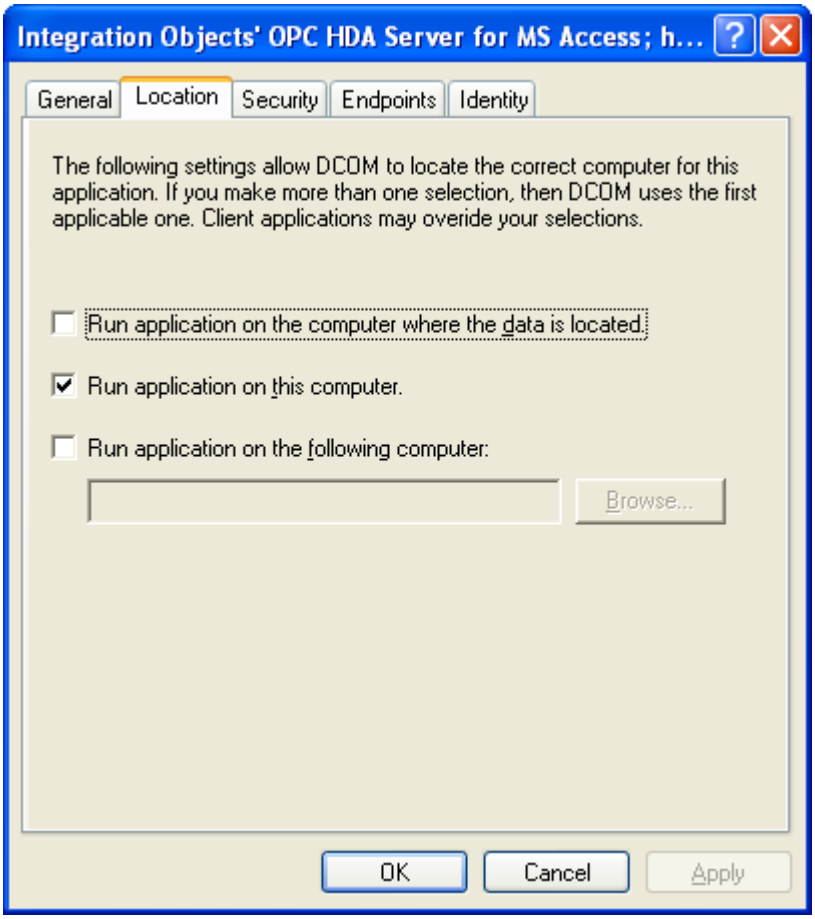


Figure 42: Location Tab

c. On the **Security** Tab, we recommend that you:

- Enable **Use default access permissions**, which means users/groups shown under Default Security tab in the DCOM configuration utility will have access to connect to this OPC Server.
- Enable **Use default launch permissions**.

The same rules apply for using custom launch permissions here as they do for custom access permissions. If you choose to use the custom permissions to override the defaults, specify which users/groups to which you want to grant permissions.

The dialog looks like this:

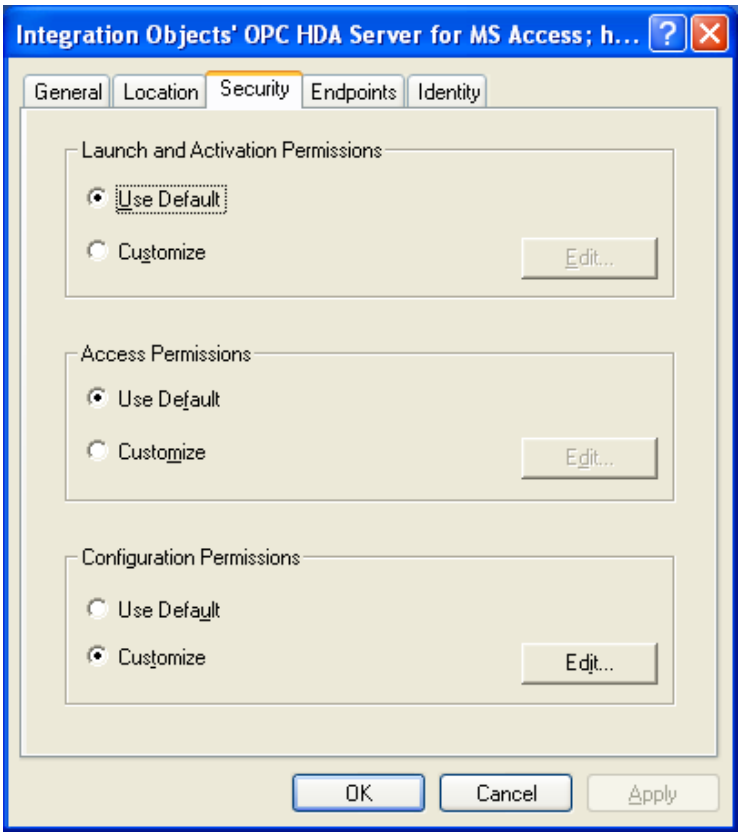


Figure 43: Security Tab

d. On the **Identity** Tab, specify under what user account you want the OPC Server to run. This is probably one of the most important settings for the OPC Server, depending on how you will be using your system.

You do not need to configure anything on the **Endpoints** tab.

2. Configuring the Client for DCOM

To set up the client:

1. Launch the DCOM Config Utility.
2. Configure the **Default Properties** tab as you did on the server side.
3. Click the **Default Security** tab:

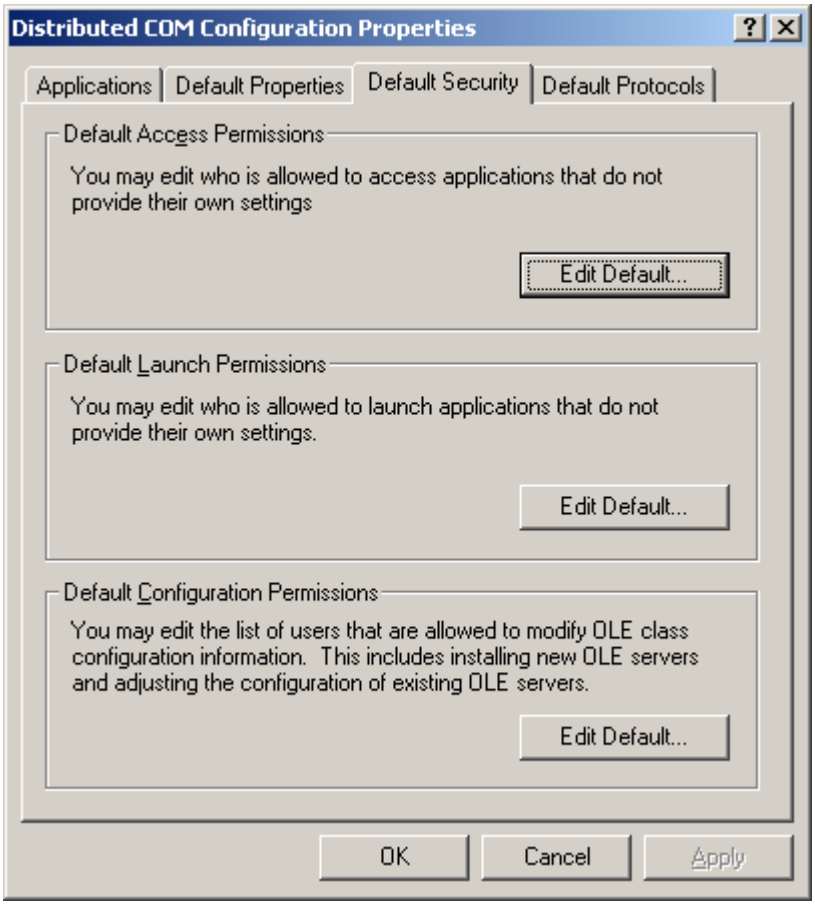


Figure 44: Default Security Tab

This is where you configure who has access to your OPC client from remote OPC servers. You only need to be concerned with the Default Access Permissions button on this tab.

4. Under Default Access Permissions:
   Click the **Edit Default** button and configure the users of remote OPC servers who you want to be able to make call-backs to this machine when your OPC client performs subscription-based read operations.

5. Configure the **Default Protocols** tab as you did on the server side.

Registering the OPC HDA Server for MS Access:

To register the OPC HDA Server on your client machine, you need to use a customized registry file. We shipped (*.reg) files for Windows NT 4.0, Windows XP professional and 2000 platforms. So, you just need to apply the appropriate file. Then, use the following steps to verify that the OPC server machine is properly delegated:

1. On the client machine, run the DCOM Config Utility.
2. Select your OPC server from the Applications tab and choose Properties.
3. On the General tab, be sure that there is an entry for Remote Computer and that the remote computer name is correct.
4. If the computer name is incorrect, select the Location tab.
5. Ensure the Run application on the following computer setting is checked.

In the Dialog box beneath this selection, type in the correct computer name for your OPC Server (see the figure below).

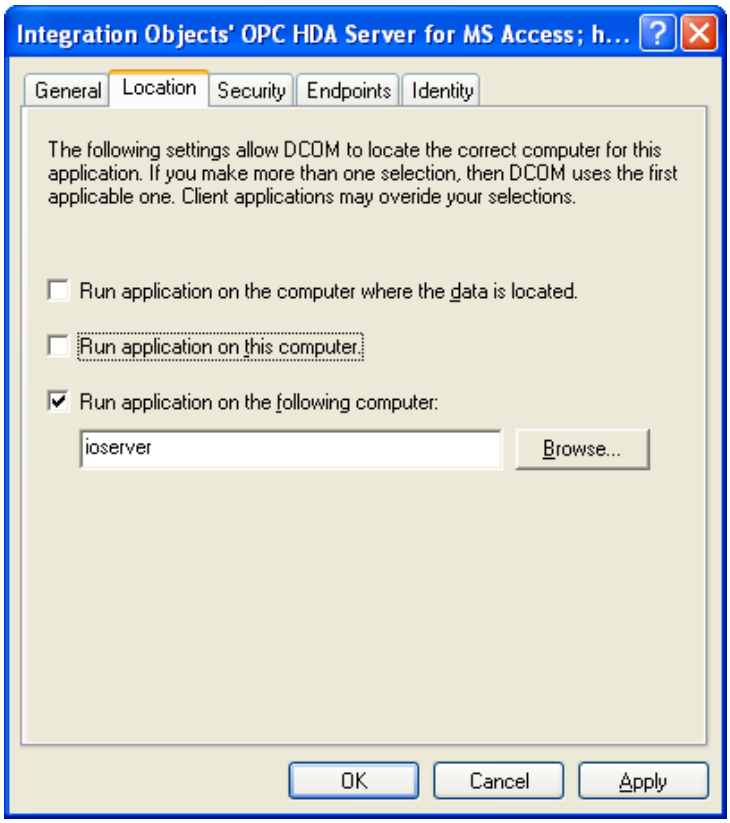


Figure 45: Location Tab

You can also use the following steps to verify the remote computer name by using the Windows Registry:

1. Run regedit.exe.
2. The remote server name is specified in the following registry key:

HKEY_CLASSES_ROOT\AppID\{The CLSID of the OPC server}\RemoteServerName

TROUBLESHOOTING

1. Logging

The OPC Server for MS Access produces a log file named “DBSrv_LogEvent.LOG” that records errors and debugging information for the server. As this server is based on Integration Objects’ OPC server DLL (DXServerDll.dll), you can view all logged DLL events in a separate file “SrvToolkit_LogEvent.LOG” for easy diagnostics. If difficulties occur with the server, the log files can be extremely valuable for troubleshooting. Under normal operations, the server logs very little information.
These log files are generated at start-up under the setup folder where the IODBServer.exe is located.

This OPC server incorporates a configuration file “DBSrv_CfgFile.ini” which includes several logging parameters, a timing parameter (used to improve the performance of the server) and some database recovery parameters. All these parameters have default settings and can be changed at start-up by editing the configuration file or using the server GUI.

To change this file:

1. Open DBSrv_CfgFile.ini in a text editor.
2. Edit any of the parameters listed in the following tables:

Or you can update these parameters through the GUI of the server.

Log Setting	Description	Default Value
LogFileMaxSize	The maximum log file size, in bytes. Once this size is reached during run-time, the log file is overwritten.	1048576*2 ~ 2 Mo (MegaByte)
LogLevel	The log level. Possible Values are: 0: Only fatal error messages are logged. 1: All critical error messages are logged. 2: All errors are logged.	0

	3: All warnings are logged. 4: All information is logged. 5: For Debug information. The higher the log level, the more information is recorded. We recommend using level 0 for a better performance of the server.	
ArchiveLastLog	TRUE: Old file is copied to an intermediate file with incremental extension, before being overwritten. FALSE: Any pre-existing log file is erased and overwritten at start-up.	FALSE
LoadConfigOnStartUp	TRUE: The server configuration will be loaded at OPC server startup. FALSE: The server starts without loading configuration file (no tags in the server address space).	TRUE
ConfigFilePath	The full path of the server configuration file for the database.	

Table 12: Log settings

PerfParameters	Description	Default Value
SERVER_RATE	This parameter is the frequency at which the server handles the asynchronous reads/updates.	250 ms (milliseconds)
ClientRequestTimeout	Recovery time before initiating restore procedure of the OPC Server. If set to 0, the automatic restore feature will be disabled	0 s (Seconds)

Table 13: Performance parameters

DBRecovery	Description	Default Value
sCycle	This parameter is the frequency at which the server checks for the database connection state.	30 s (seconds)
DBLogFile	The full path of the log file for tracing database	

	problems.	

Table 14: DBRecovery

ServerSetting	Description	Default Value
Delimiter	OPC Item IDs delimiter	/

Table 15: Server settings

3. Save the file for the settings to take effect.

Sample Configuration File

```
[LogSetting]
LogFileMaxSize=2097152
LogLevel=0
ArchiveLastLog=FALSE
LoadConfigOnStartUp=TRUE
ConfigFilePath=C:\Program Files\Integration Objects\Integration Objects' OPC
HDA Server for MS Access\Config.xml
[PerfParameters]
Server_Rate=250
ClientRequestTimeout=0
[DBRecovery]
sCycle=30
DBLogFile=C:\Program Files\Integration Objects\Integration Objects' OPC HDA
Server for MS Access\DBRecovery_LogEvent.LOG
[ServerSetting]
Delimiter=/
```

Figure 46: Configuration File

You may change configuration parameters in the toolkit configuration file "SrvToolkit_CfgFile.ini". You may increase or decrease trace level.

Database logging

All database connection problems are recorded in a configuration file that you can customize its full path (by default DBRecovery_LogEvent.LOG under the setup folder).

Service logging

Service event information are logged in the Application logs of the Event Viewer (*Source* = “OPCHDAServer.MSA”).

2. DCOM

This section addresses some DCOM related problems while using OPC servers:

Problem 1: You have an “Access denied” error on the client machine. The client and server are running on standalone machines (meaning not on the same domain).

Let’s assume that the OPC client is running on machine A and the OPC server on machine B.
When the OPC client and server are on different computers, you have to give each computer access to the other by giving access permissions. The permission issue is crucial for proper DCOM configurations.
Here the server is running on a standalone machine. So the ONLY user accounts it will trust are those it finds in its own "local" security database. Here is how this can get you into trouble when setting up an OPC client to server connection.
To allow a remote client to access the DCOM server, the DCOM utility uses the Windows Security database. For this reason, you cannot give access to a user account which does not show up in this database.

Here is the issue:

1. You can add Machine B into the same domain as Machine A (or in a trusted domain), which is the safest way to correctly set up the communication between the OPC client and the OPC server.
2. You need to create the EXACT SAME user account name AND password on BOTH machines (for example User1 (login), PWD1 (password)). Once you have that set up, when Machine A comes calling on Machine B with an OPC request and identifies himself as User1 with PWD1 password, Machine B will look in its database, see the same account name, the same password, and same "come on in request from Machine A". When Machine B goes to return its data from the OPC server to the OPC client on machine A, the OPC server will call Machine A as User1 with a password -- Machine A will look in its database, see that it has that account, and accept the call. This

workaround should resolve the communication problem between the OPC client and server.

Problem 2: You have been running your OPC client on a Windows XP machine. When upgrading the machine to XP Service Pack 2, the OPC client becomes unable to connect to the OPC server.

This is a common problem when using OPC via DCOM with Microsoft Windows XP Service Pack 2.
In fact, when Service Pack 2 is installed with its default configuration settings, OPC communication via DCOM will cease to work.

To resolve this issue, you have to reconfigure your settings for:

1. The Windows XP firewall.
2. DCOM.

Click on this link Using OPC via DCOM with XP SP2.pdf to download the OPC Foundation document that describes all steps to apply new settings.

Problem 3: Launching the OPC Server without needing to log to the System

To launch the OPC Server without needing to log to the System, proceed to the following steps:

1- Stop the OPC Server and verify that the windows service is stopped.

2- Open Windows DCOM Configuration using the command DCOMCnfg

3- Locate the DCOM component Integration Objects' OPC HAD Server for SQL MS Access.

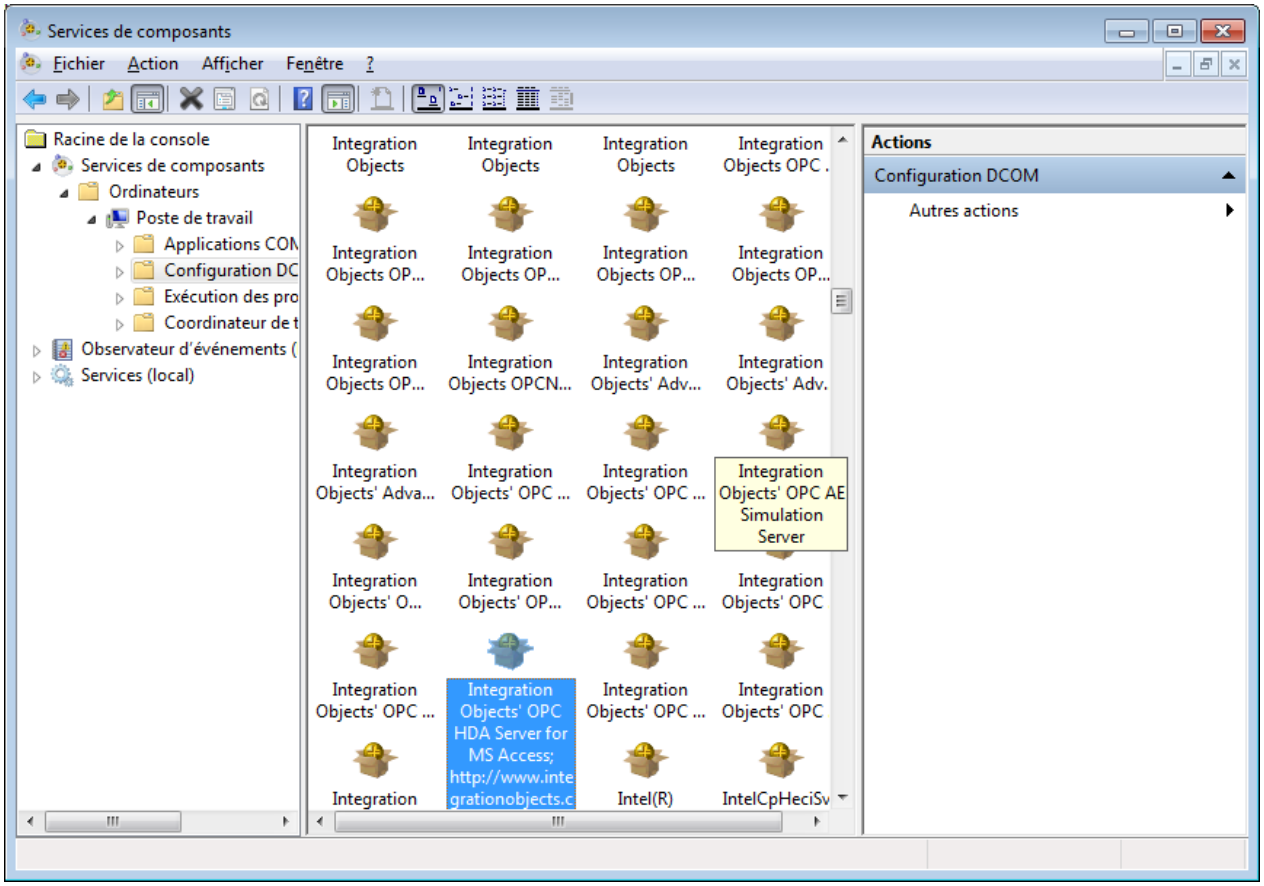


Figure 47: DCOM Settings

4- Right click then select properties from the displayed menu:

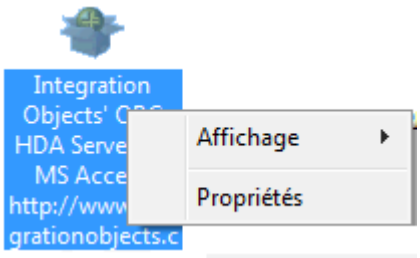


Figure 48: Properties

5- Select Identity Tab

6- Choose This user

TIPS

OPC

1. Truncated Data

In update methods like insert and update, data may be truncated if you choose a variant type different to the canonical one.
Example: Insert a value with data type = VT_UI2 knowing that the canonical data type is VT_I4.
Original value: 223355 ➔ 26747

2. Server Specific Errors

Integration Objects defines the following error codes for this OPC server:

Error Code	Description
IO_E_DB_STATE	Database problem has occurred: may be an exception or a broken connection.

For additional information on this guide, questions or problems to report, please contact:

Offices
- Houston, USA: +1 713 609 9208
- Genoa, Italy: +39 34 75 83 93 47
- Tunis, Tunisia: +216 71 195 360

Email
- Support Services: customerservice@integrationobjects.com
- Sales: sales@integrationobjects.com

To find out how you can benefit from other Integration Objects products and custom-designed solutions, please visit us on the Internet:

Online
- www.integrationobjects.com