DataLogger

Table of Contents

DataLogger

Help version 1.106

CONTENTS

Overview

Initial Setup Considerations

Setting up a DataLoggerConfiguration

System Tags

Error Descriptions

Condition Examples

All company and product names mentioned herein are the trademarks or registered trademarks of their respective owners.

Overview

DataLogger is an application that logs data from an OPC server to any ODBC-compliant database. DataLogger's tight integration with the OPC server provides substantial, unique benefits such as simple installation, high-efficiency performance, and easy tag browsing in the OPC browse space.

Feature Highlights

- Supports any ODBC-compliant database management system.
- Supports adding OPC data items through drag and drop.
- Has a user-friendly installation and configuration. If a Database Source is defined before DataLogger is launched, users can have an active logging configuration in less than ten steps.
- Has flexible triggering. Data logging can be enabled at the following times: always, at absolute times, or when an expression is true (such as when a tag's quality is bad). When enabled, logging can occur based on static / time interval, on log group item data change, on monitor item data change, and / or on start / stop condition transitions.
- Has improved, reliable information logging through the local store and forward file, which is used to bridge network and database outages or delays.
- Includes an OPC server Simulator Driver.
- Includes a two-hour demo for evaluation.

Runtime Performance Features

- Runs as a System Service.
- Can be easily scaled through its support of multiple concurrent logging processes (threads).
- Logs data directly from the local item list without reliance on external OPC servers.
- Supports both automatic table creation and the ability to append data to an existing table.
- Supports error recovery and can automatically reconnect if a DSN connection is lost.
- Supports an optional automatic configuration backup (in which the most recent copy of the configuration file is saved).
- Supports _System Tags that allow optional Runtime control from OPC client applications (such as enabling / disabling logging and monitoring logging status).

Initial Setup Considerations

The following topics should be reviewed before the first DataLogger Configuration is created.

System Requirements
External Dependencies
SQL Authentication
Windows Authentication
CSV Import / Export

System Requirements

Software Requirements

The following Microsoft Windows operating systems are supported:

- Windows 8
- Windows 7 Professional, Enterprise, and Ultimate
- Windows Server 2012
- Windows Server 2008 R2
- Windows Server 2008
- Windows Vista Business, Enterprise, and Ultimate
- Windows Server 2003 (Service Pack 2)
- Windows XP Professional (Service Pack 2)

Note: When installed on a 64-bit operating system, the application runs in a subsystem of Windows called WOW64 (Windows-on-Windows 64-bit). WOW64 is included on all 64-bit versions of Windows and is designed to make differences between the operating systems transparent to the user.

Hardware Requirements

The following hardware is required at a minimum:

- 2.0 GHz processor
- 1 GB installed RAM
- 180 MB available disk space
- Ethernet card

See Also:

External Dependencies

External Dependencies

This application has external dependencies. It requires that the ODBC driver for the Database Management System being used be installed on the PC that is running the OPC server. DataLogger supports the following ODBC drivers:

- SQL Native Client (necessary for SQL Server 2005)
- SQL Server ODBC Driver (compatible with pre-SQL Server 2005)
- MyODBC Driver 3.51 (for MySQL)
- Microsoft Access 4.0 ODBC Driver
- Linked Excel table support provided through the Microsoft Access 4.0 ODBC Driver

Notes:

1. Although DataLogger supplies TimeStamp values with a resolution to one thousandth of a second, certain databases are not capable of displaying a Date Format to the resolution of below one second.

2. Some databases do not support millisecond resolution. For more information on a specific database, refer to the product's vendor.

SQL Authentication

Click on the links below in order to jump to that section in the SQL Authentication setup.

Setting up SQL Authentication
Running as a System Service
Connecting Remotely as a System Service

Setting up SQL Authentication

The following instructions contain information on setting up an SQL authentication. This process usually only has to be done when the application is running as a System Service and is attempting to connect remotely to SQL server.

1. In the SQL manager, right-click on the SQL server icon and then open the SQL Server properties.
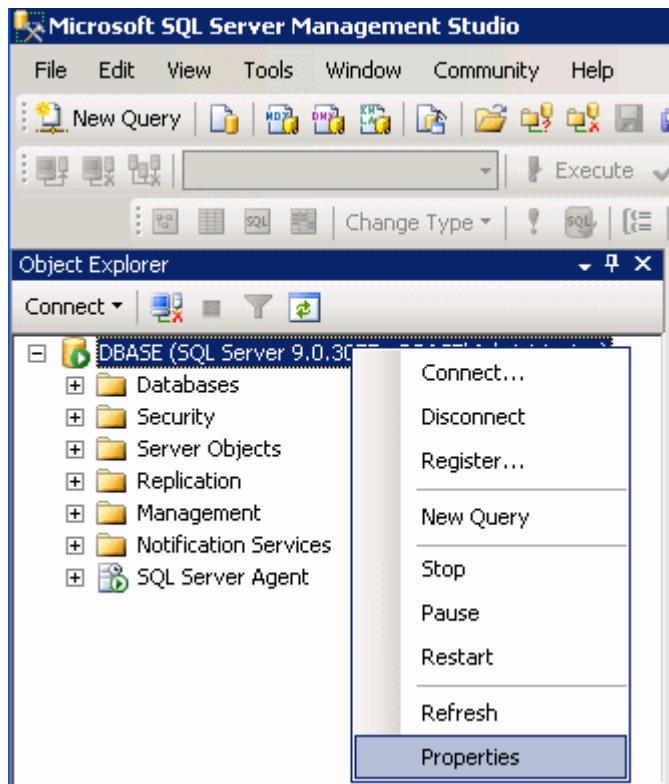
2. Select the **Security** page and choose the mixed authentication mode (**SQL Server and Windows Authentication mode** radio button).
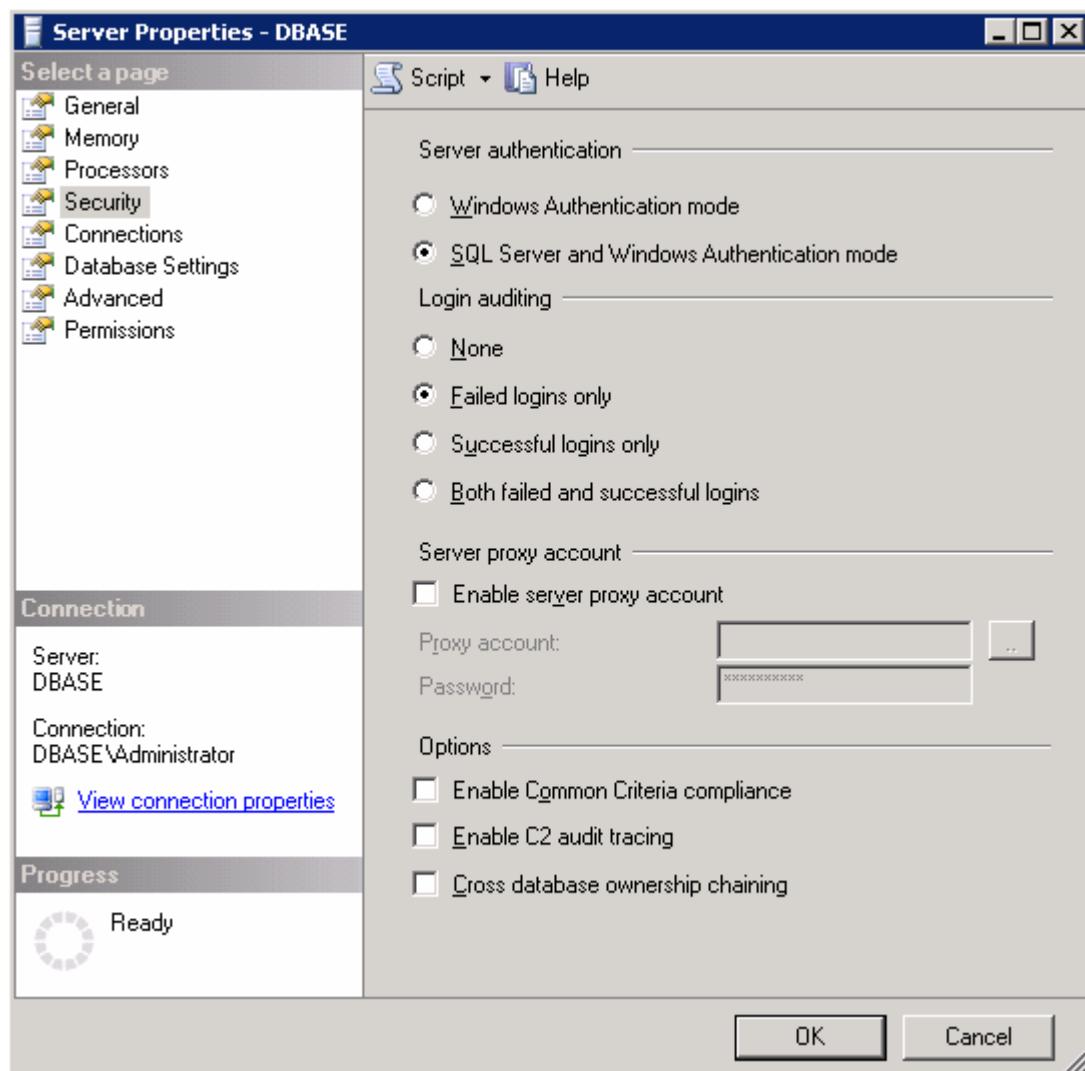
3. Proceed to the tree menu and right-click on the security folder. Click **Logins** | **New user**.

4. Create and define a user's privileges.
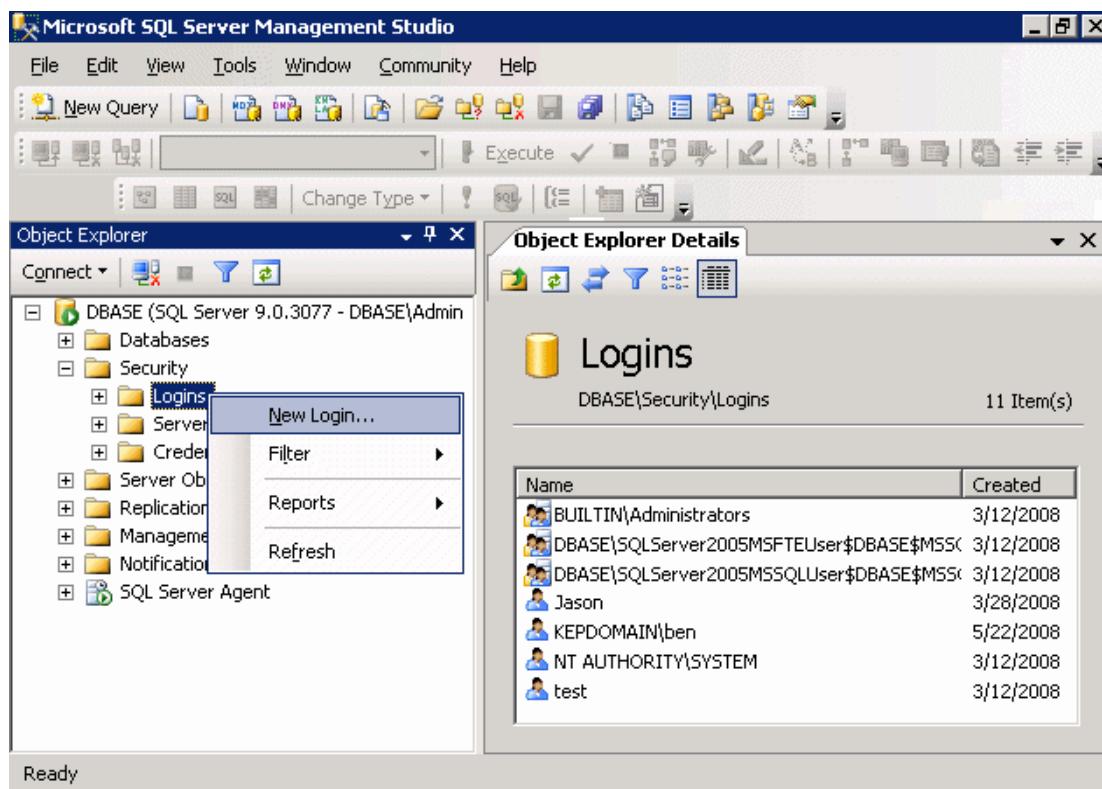
5. Under the **General** page, a user name and password must be defined.
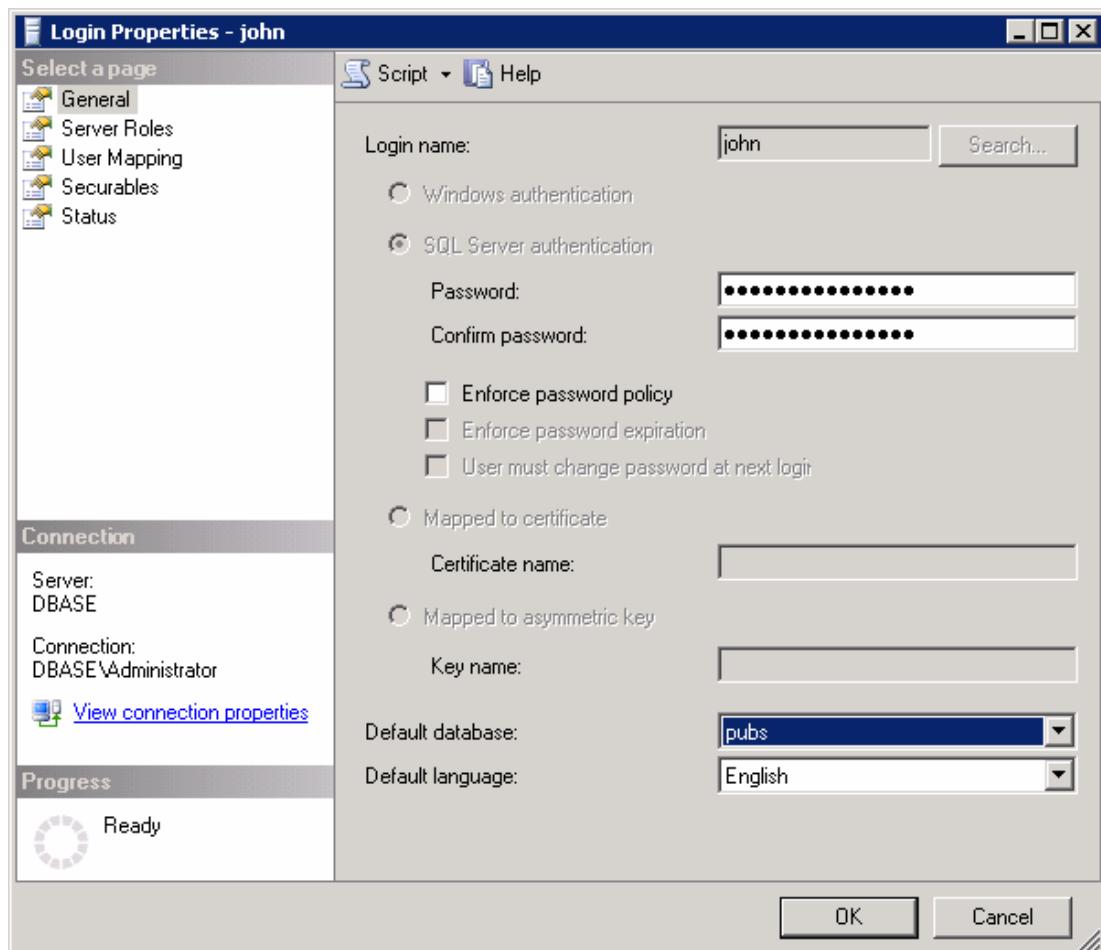
6. Next, click on the **User Mapping** tab and select the database to connect. Then select a role for the database that has been selected. In this example, **Public** is used.
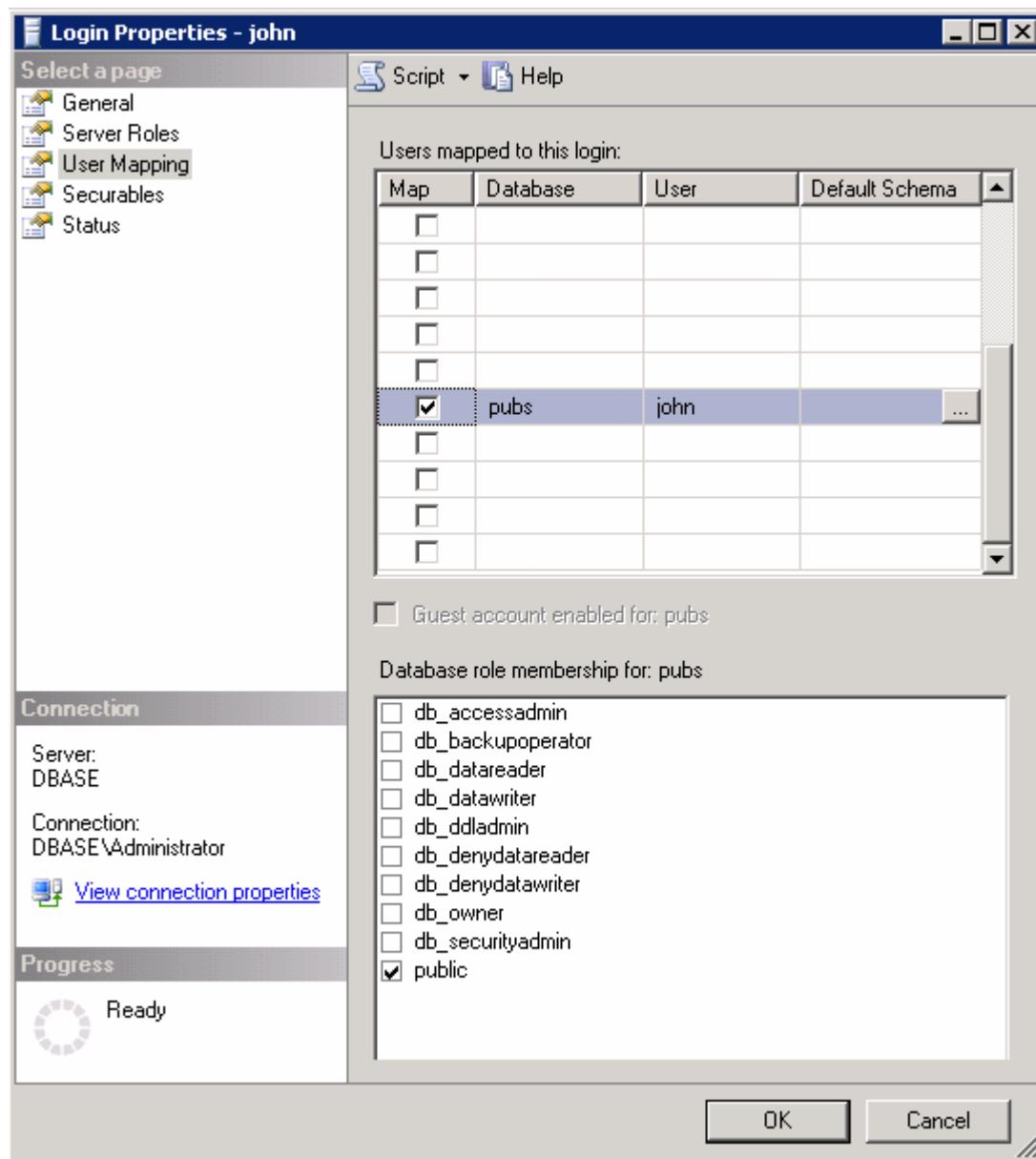
7. Next, right-click on the KEPServerEX **Administration** menu located in the System Tray. Then, select **Settings…** and click **Runtime Process**.

8. In **Selected Mode**, select **System Service**.

9. Click **OK**.
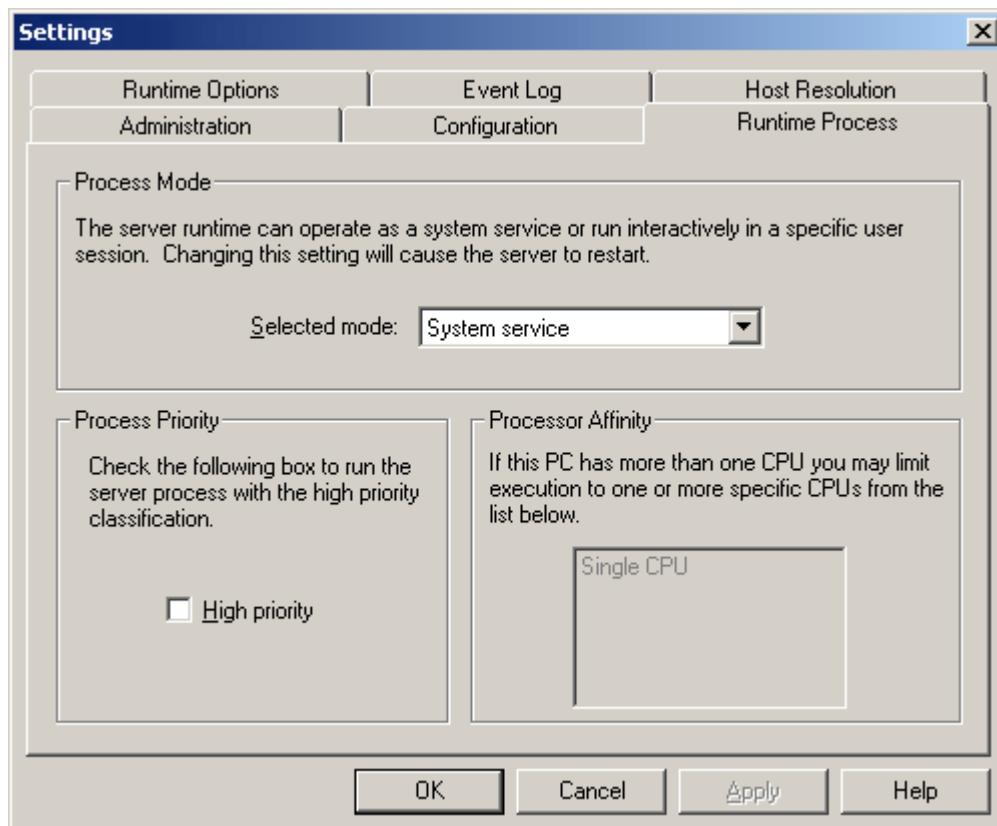
10. When the DSN is configured, a series of DSN setup dialogs appear. In **Create a New Data Source to SQL Server**, check **With SQL Server authentication using…** and **Connect to SQL Server to obtain…** and then enter the user's Login ID and password (that have been defined in the SQL Server).
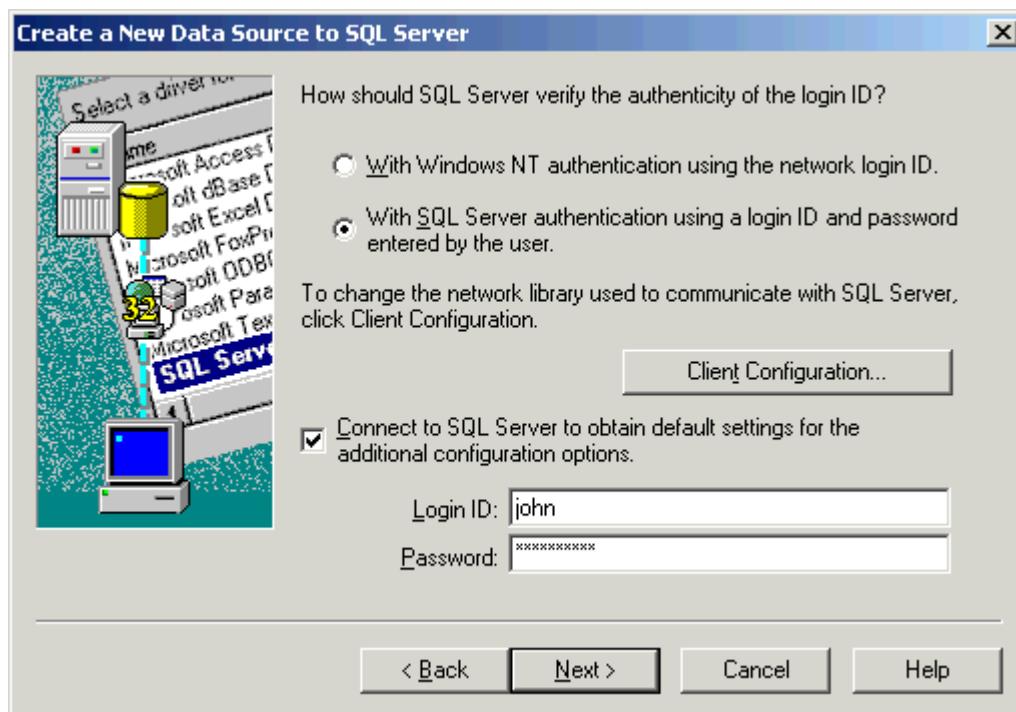
Running as a System Service

Normally, an OPC server that only supports stand alone program operation is forced to shut down when its host machine experiences a user login or logout. This server, however, can continue to supply OPC data across user login sessions by running as a System Service. The ability to run as a System Service is crucial for many applications where the server must provide data to OPC clients via DCOM. For these applications, the loss of a DCOM connection cannot be tolerated.

Note: For more information on running as a System Service, refer to the server's help documentation.

Connecting Remotely as a System Service

This ODBC communications application supports running as a service under supported Microsoft Windows operating systems. For operating system (OS) requirements, refer to the server's help documentation.

Windows Authentication

Windows Authentication allows the application to authenticate with the SQL server using Windows credentials. It requires that both the application and the SQL server be located on the same domain.

When the application is running in Interactive Mode, the Windows credentials of the user that launched the application are used during authentication. In most cases, this is the current logged-in user. As long as the user is part of the domain, and the SQL server is configured for Windows Authentication, it passes authentication.

When the application is running in System Service Mode, the NT AUTHORITY\SYSTEM account is used during authentication. This is a local account that fails Windows authentication. Users that require Windows Authentication in System Service Mode should refer to the instructions below.

1. To start, open the **Windows Service Configuration Manager**. Then, locate the Runtime service.

2. Right-click on the service and then select **Properties**.

3. Click **Log On**.
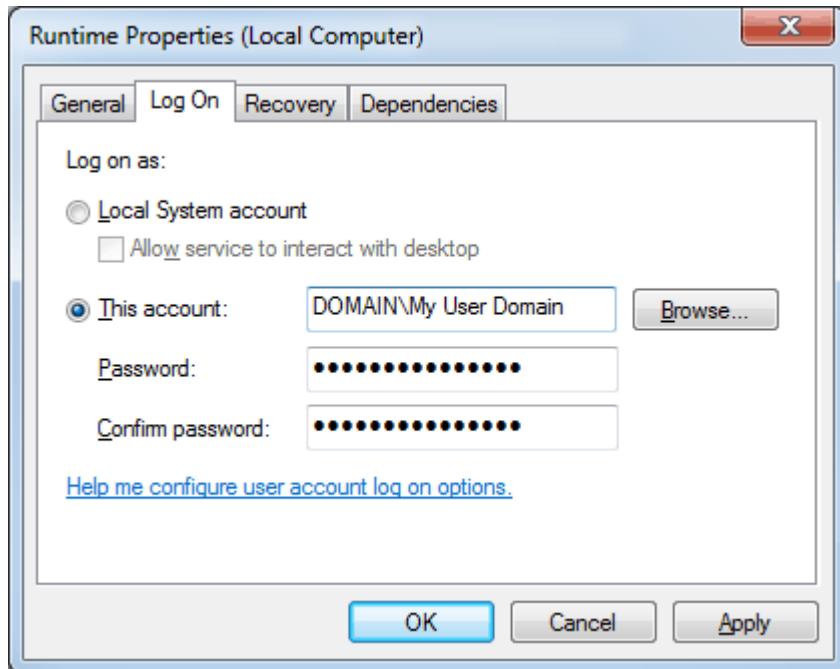
4. Next, select **This account**. Enter the domain name and password used for Windows Authentication.

5. Restart the service.

Note: The procedure described above may restrict the application's permissions if the domain account does not have administrative privileges on the system. If the account cannot be given administrative privileges, SQL Authentication should be used instead.

CSV Import / Export

The Data Logger Plug-In supports the import and export of server items in a Comma Separated Variable (CSV) file. Parameters that are not defined in the CSV file are assigned appropriate default values. To jump to a specific section, select a link from the list below.

Exporting a Data Logger Server Item List
Importing a CSV file into the Server
Using Other Characters as the Delimiter

Note: For information on specifying which character to use as the variable (comma or semicolon), refer to "Options - General" in the server help file.

Creating a Template

The easiest way to create an import CSV file is to create a template. For more information, refer to the instructions below.

1. To start, browse to a Log Group's **Data Map** tab. Then, add a server item to the **Server Item List**.

2. Next, click **Export CSV...** and save the CSV file.

3. Use this template in a spreadsheet application that supports CSV files, and then modify the file as desired.

Note: Users can save the CSV file to disk, and then re-import it into the same log group or a different log group.

Note: Microsoft Excel is an excellent tool for editing large groups of tags outside the server. Once a template CSV file has been exported, it can be loaded directly into Excel for editing.

Exporting a Data Logger Server Item List

Exporting a server item list generates a .CSV text file that contains a list of server items and their associated parameters. Column names must be exactly the same as those listed; however, columns may be in any order. Required columns are listed in bold print.

Column Names	Values
Item ID	The fully-qualified name of the server item. The maximum length of the field is 256 characters.
Numeric ID	A numeric alias or identifier for the server item. The valid range is 0 to 2147483647. The default setting is 0.
Data Type	The server item's data type. The content of this column is informational only; it is not considered when importing an item list.
Deadband Type	The type of deadband used to filter updates from the server item. Options include None, Absolute, and Percent.
Deadband	The deadband value. The value must be a positive, real number.
Range Low	The lower range of a percent deadband. The value may be any real number, but must be less than the high range value.
Range High	The higher range of a percent deadband. The value may be any real number, but must be greater than the low range value.

See Also:
Add Server Item / Modify Server Item

Importing a CSV File into the Server

Once the file has been edited, it can be re-imported into the server by clicking **File | Import CSV**.

Note: This option is only available when the Data Map tab is selected.

Using Other Characters as the Delimiter

When utilizing a CSV file that does not use a comma or semi-colon delimiter, do one of the following:

1. Save the project in XML. Then, perform mass configuration on the XML file instead of using CSV.

2. Perform a search-and-replace on the delimiter in the CSV file and then replace the delimiter with a comma or semicolon. The delimiter being used by the server (either comma or semi-colon) must be set to the replacement character.

Note: For more information, refer to "Options - General" in the server help file.

Setting up a DataLogger Configuration

A DataLogger configuration defines how data is extracted from an OPC server project and logged into a database. There is one DataLogger configuration for each OPC server project. Within a DataLogger configuration there are one or more log groups. A log group is a "data pipeline" between the OPC server project and a database table. A log group defines the following:

- A DSN connection to the database.
- The server items (such as OPC server tags) logged to the database.
- The format and name of the table within the database.
- The triggers that govern when data is logged (at a specific time, on data change, and so forth).

For more information on a specific DataLogger tab, select a link from the list below.

General Tab
Data Map Tab
Triggers Tab

Opening DataLogger and Adding a Log Group

For information on opening DataLogger, refer to the instructions below.

1. Start the OPC server and then open the OPC project.

2. Locate and select **DataLogger** in the treeview.

3. The first time that a DataLogger configuration is created for a server project, users must add a new log group. To do so, select the prompt. Alternatively, click the **New Log Group** icon.

   At this point, the left pane displays one log group with the default name "Untitled," and the parameters located in the General tab are enabled. For more information, refer to **General Tab**.

4. Once a log group has been defined, click the **Apply Changes** icon to save the log group. When a new log group is added (or when changes are made to an existing log group) the changes must be applied. If this is not done, the system prompts users to apply changes when exiting the configuration.

   Note: The image below shows three log groups in the configuration; the third one has been modified but not saved.

5. To make a copy of an existing log group, right-click on the log group and select **Copy**.

6. To remove (delete) a log group, right-click on the log group and select **Delete**.

General Tab

The General tab is used to specify general settings for a new log group. For more information on a specific section, select a link from the list below.

General Properties
Data Source Properties
Store and Forward
Batch Identifier
Memory Properties

Timestamp Properties

General Properties

Descriptions of the parameters are as follows:

- **Name:** This parameter specifies a name for the new log group. The maximum length is 256 characters. It cannot begin with an underscore, or include a period or double quote. The default name is "untitled".

- **Description:** This parameter specifies a longer, descriptive name. The maximum length is 4096 characters.

- **Update Rate:** This field specifies the update rate for data coming from the OPC server to the log group. The default setting is 100 milliseconds. The range for each unit is as follows:

  - **Milliseconds:** 10 to 4294967295.

  - **Seconds:** 1 to 4294967.

  - **Minutes:** 1 to 71582.

  - **Hours:** 1 to 1193.

  - **Days:** 1 to 49.

- **Enabled:** This option enables the DataLogger configuration. The default setting is checked.

Data Source Properties

This section is used to configure a DSN connection to the database.
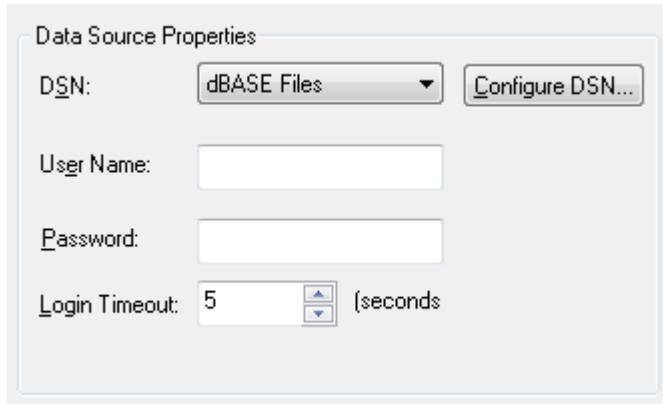
Descriptions of the parameters are as follows:

- **DSN:** This drop-down list specifies the Data Source Name (DSN). If the DSN is not listed, it must be configured. To do so, select **Configure DSN** and refer to **Setting up a DSN**. Once a DSN has been configured, it is displayed as a choice in the drop-down list.
- **User Name:** This parameter specifies a user name for the data source. It is optional unless the data source requires it.
- **Password:** This parameter specifies a password for the data source. It is optional unless the data source requires it.

  Note: Some data sources require that Windows NT Authentication be used in order to gain access. If the data source is configured to use Windows NT Authentication, it ignores the user name and password set in Data Source Properties and uses the network login ID instead. For more information, refer to **SQL Authentication**.
- **Login Timeout:** This parameter specifies how long the server waits for a response when attempting to connect to the DSN. At the end of that time, the connection attempt times out. The valid range is 1 to 99999 seconds. The default setting is 5 seconds.

Store and Forward

Enabling store and forward prevents data from being lost when connectivity to the ODBC data source is lost or when the database is unable to process requests fast enough. When these situations occur, the Log Group records the data that was captured during the connection loss to disk. Once connectivity is restored, the data on the disk is read, written to the ODBC source, and deleted.

Note: Clients can use the _Buffering Tag to monitor when the store file is being used. For more information, refer to **System Tags**.

Descriptions of the parameters are as follows:

- **Enabled:** When checked, this option enables the store and forward functionality. A store file is created at runtime with the path "<storage directory>\<log group name>.bin". When unchecked, the other Store and Forward settings are also disabled. The default setting is unchecked.

  Note: When store and forward functionality is enabled, the log group name must only consist of valid characters.
- **Storage directory:** This field specifies the directory in which a store file is created. The directory may be any valid absolute path that begins with "\\" or "<drive letter>:". The default setting is "<application profile directory>\DataLogger\".

  Note: Access to the store file depends on the permissions that are configured for the selected directory. The default directory allows read and write permissions to all users.

  Important: Utilizing a network drive as the store file location is risky. If the network goes down, any store files located on that network drive is inaccessible. As such, it is not recommended that a network drive be used as the storage directory.
- **Maximum storage size:** This parameter specifies the maximum file size for the store and forward file. New data is lost if the file size limit is reached. The valid range is 1 to 2047 MB. The default setting is 10 MB.

Batch Identifier

Descriptions of the parameters are as follows:

- **Batch ID Item:** This optional parameter specifies the ID of the server item to be used as a batch identifier.

  Note: If a Batch ID item is entered, the item is mapped to a database column by default. When a database transaction is triggered, the current value of the item is applied to the column for all rows associated with the transaction.
- **Data Type:** Identifies the data type of the value that will be produced by the server.

- **Update Rate:** This parameter specifies the update rate for the server item defined in the Batch ID Item field. The default setting is 1000 milliseconds.

Memory Properties

Memory Properties

Maximum row buffer size: 1000
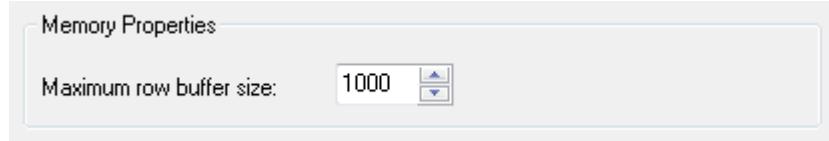
Description of the parameter is as follows:

- **Maximum row buffer size:** This parameter specifies the data change memory buffer allowed per item. The valid range is 1 to 99999. The default setting is 1000.

-

Map Numeric ID

Map Numeric ID to VARCHAR by default

Description of the parameter is as follows:

- **Map Numeric ID to VARCHAR by default** When checked, each item in the log group maps the numeric ID column to the VARCHAR(64) data type by default as opposed to the INTEGER data type. **Note:** Disabling clears all numeric IDs associated with the log group items.

Timestamp Properties

Timestamp Properties

☑ Use local time for timestamp inserts (uncheck for UTC)
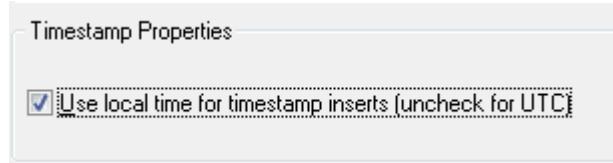
Description of the option is as follows:

- **Use local time for timestamp inserts (uncheck for UTC):** When checked, DataLogger uses the local time for timestamp values. When unchecked, the timestamp values are in Universal Time Coordinated (UTC).

Setting up a DSN

Before any DSN connection to a database can be used, it must be configured. After it has been configured, the DSN is displayed in the drop-down list for the **DSN** field.

Important: Users can configure a DSN using the Microsoft® ODBC Data Source Administrator launched from the operating system's Start menu. Because the server is a 32 bit application, it is necessary that the 32 bit version of the administrator be used when configuring a System DSN. A 64-bit operating system launches the 64-bit version of the administrator by default; therefore, users may need to browse to the SysWOW64 directory to access the 32-bit version.

Note: The ODBC drivers for the Database Management System being used must be installed on the PC that is running the OPC server. For a list of supported ODBC drivers, refer to **External Dependencies**.

1. On the **General Tab**, click **Configure DSN**.
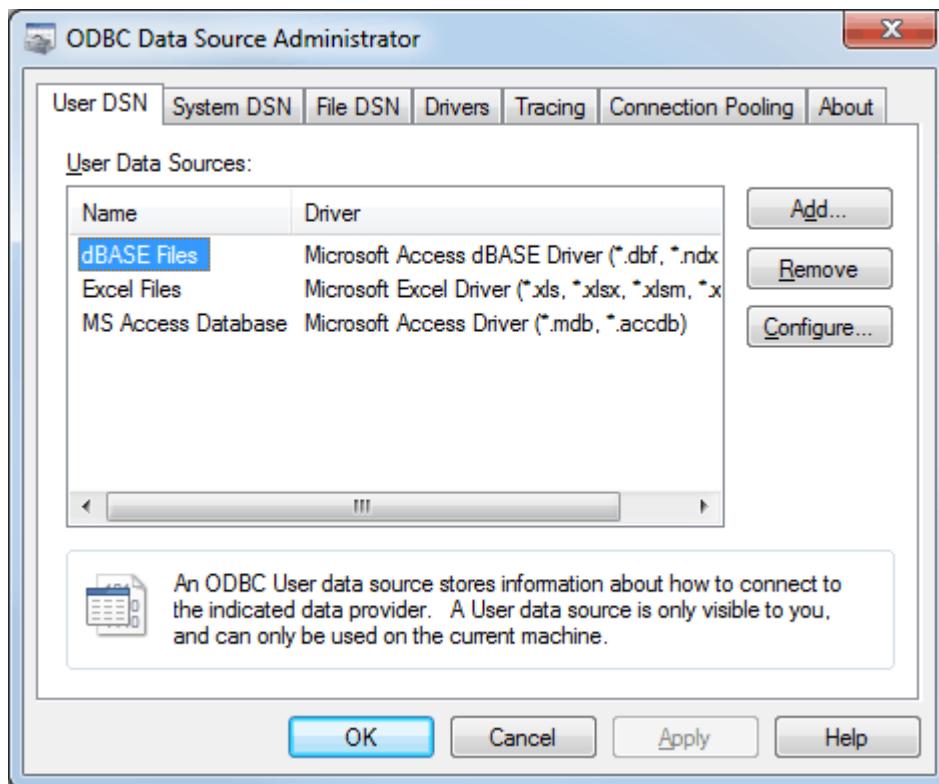
2. In **Microsoft® ODBC Data Source Administrator**, choose either the **User DSN** tab or **System DSN** tab.

   Note: For further help with the Microsoft® ODBC Data Source Administrator dialog, click **Help**.

3. Next, click **Add**.

4. In **Create New Data Source**, scroll down the list of drivers to locate the driver for the specific database.

5. Select the driver and then click **Finish**.

6. Continue through the wizard and complete the dialogs displayed. The dialogs and fields vary according to the database being connected.

7. The final dialog of the DSN Wizard may include a **Test** button for the Data Source. Whether or not a **Test** button is displayed depends upon the DSN being configured. If so, click the button to verify that the DSN setup is functional.

Data Map Tab

The Data Map tab is used to specify the Server List Items and how to log the data to tables. For more information on a specific section, select a link from the list below.

Server Item List
Table Selection
Table Format
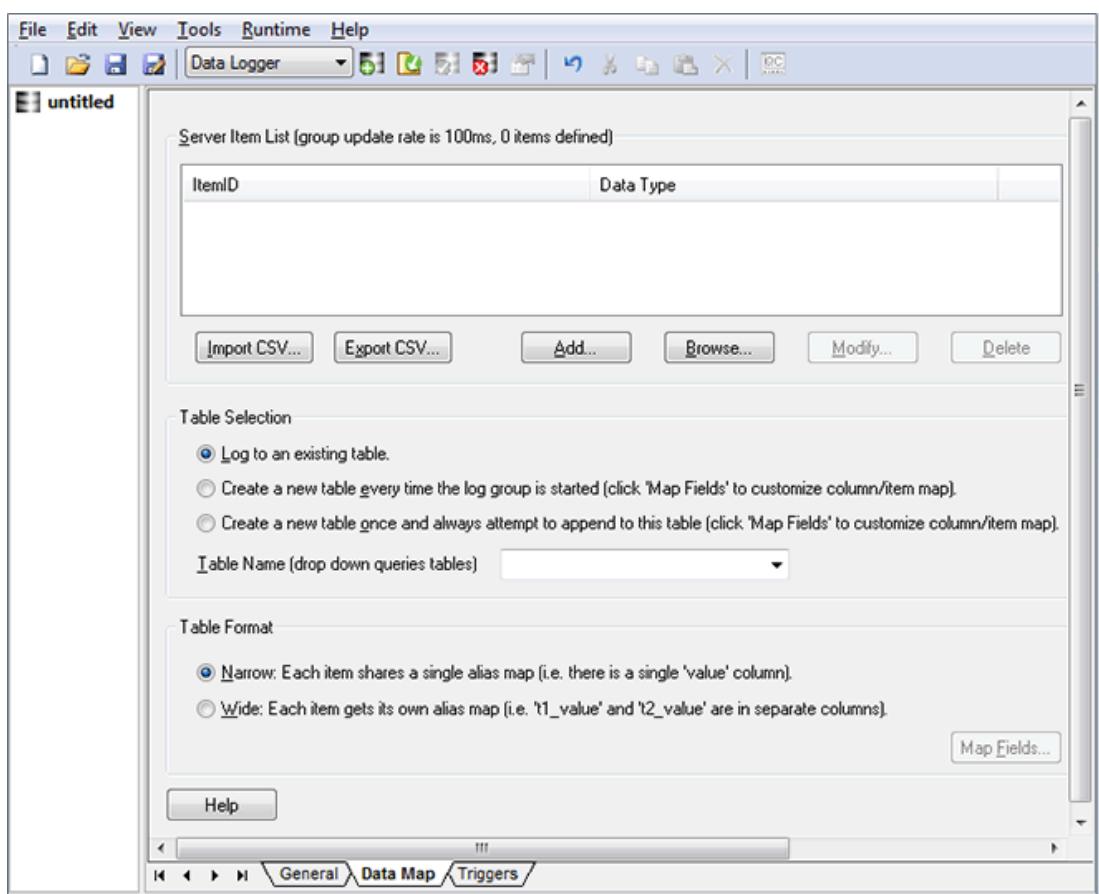
Server Item List
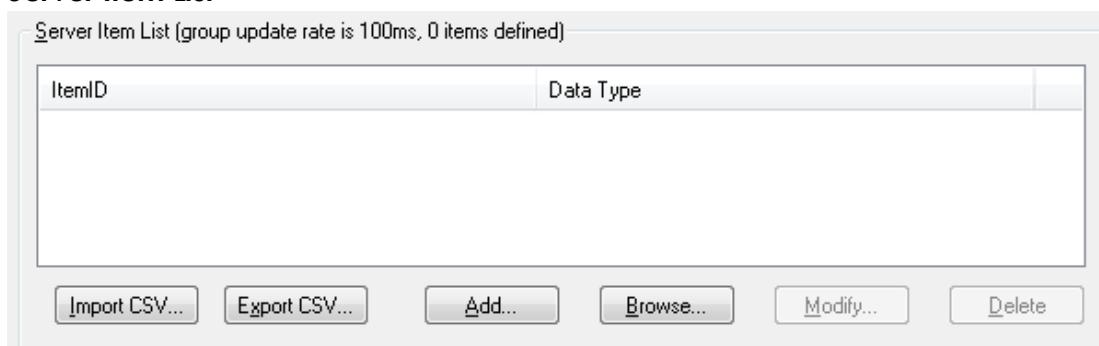
Descriptions of the parameters are as follows:

- **Import CSV:** When clicked, this button launches the Import from CSV dialog for locating and importing server items from a CSV file.
- **Export CSV:** When clicked, this button launches the Export from CSV dialog for exporting server items to a CSV file.
- **Add:** When clicked, this button launches the Add Server Item dialog for adding new server items. For more information, refer to **Add Server Item / Modify Server Item**.
- **Browse:** When clicked, this button launches the Tag Browser for locating tags to add as server items. For more information, refer to **Adding a Server Item Through Browse**.

- **Modify:** When clicked, this button launches the Modify Server Item dialog. This allows users to modify the server item that is selected in the Server Item List. For more information, refer to **Add Server Item / Modify Server Item**.
- **Delete:** When clicked, this button deletes the server item that is selected in the Server Item List.

Table Selection

Descriptions of the parameters are as follows:

- **Table Selection:** This parameter specifies both the table and how to log data to the table. Options include Log to an existing table, Create a new table every time the log group is started, and Create a new table once and always append to this table. The default setting is Log to an existing table. Descriptions of the options are as follows:
    - **Log to an existing table:** When selected, this option logs data to an existing table.
    - **Create a new table every time the log group is started:** When selected, this option generates a new table in the database each time the OPC server enters Runtime. Type a name for the table in the **Table Name** field. The first time this log group is enabled, the table is created. The next time the log group is enabled, another table is created with the name "tablename0". For example, a table named "Baseline" is created the first time the log group is enabled. The next time the log group is enabled, a table named "Baseline0" is created, then "Baseline1," "Baseline2," and so on.
    - **Create a new table once and always append to this table:** When selected, this option generates a new table in the database the first time that the log group is enabled. Data is logged to that same table each time the OPC server enters Runtime thereafter. Type a name for the table in the **Table Name** field. The first time this log group is enabled, the table is created. The next time the log group is enabled and every subsequent time, the data is appended to that same table.
    Note: To customize the mapping of server items to database columns, click **Map Fields**. For more information, refer to **Map Item Fields**.
- **Table Name:** This field either selects a table (through the drop-down list) or specifies a name for the new table to be created.
    Note: When using DataLogger to create tables, the maximum length of table names is 256 characters. In practice, however, the maximum depends on the database being used and ODBC driver's limits. Often, the maximum is 64 or 128 characters. Table names should only consist of letters and numbers, because non-alphanumeric characters can cause errors. If non-alphanumeric characters must be used in the table name, consult with the database and ODBC driver specifications for table naming restrictions.

Table Format
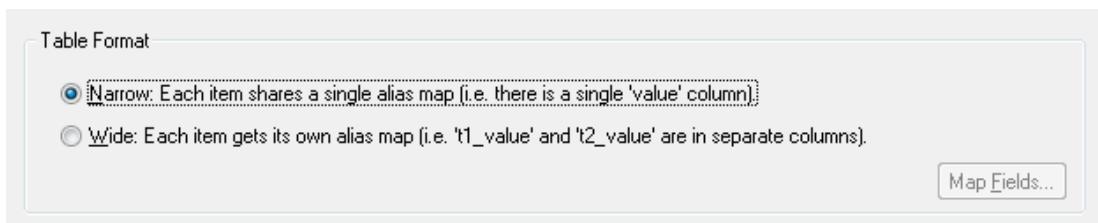
Description of the parameter is as follows:

- **Table Format:** This section specifies the format of the data table. Options include Narrow and Wide. The default setting is Narrow. Descriptions of the options are as follows:
    - **Narrow:** In this format, each item shares a single alias map. Each row includes 5 columns: Name, Numeric ID, Value, Quality, and Time.
    - **Wide:** In this format, each item gets its own map. Each row includes Numeric ID, Value, Time, and Quality for every server item.
    Note: For a more detailed description of the narrow and wide formats, refer to **Narrow vs. Wide Table Format**.

Adding a Server Item Through Browse

1. To start, click the **Browse** button.
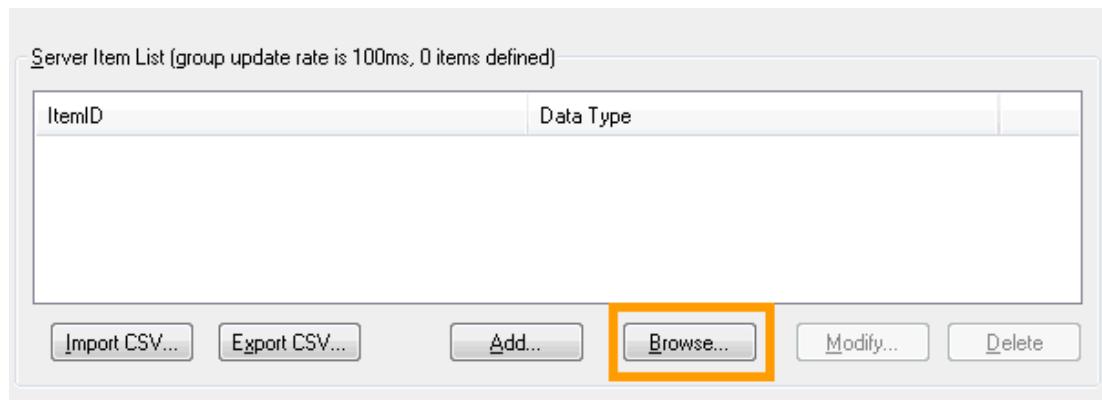
2. Navigate through the OPC server project to the server item(s) to be logged. To select multiple items, use CTRL-click or SHIFT-click. After the selections have been made, click **Apply**.
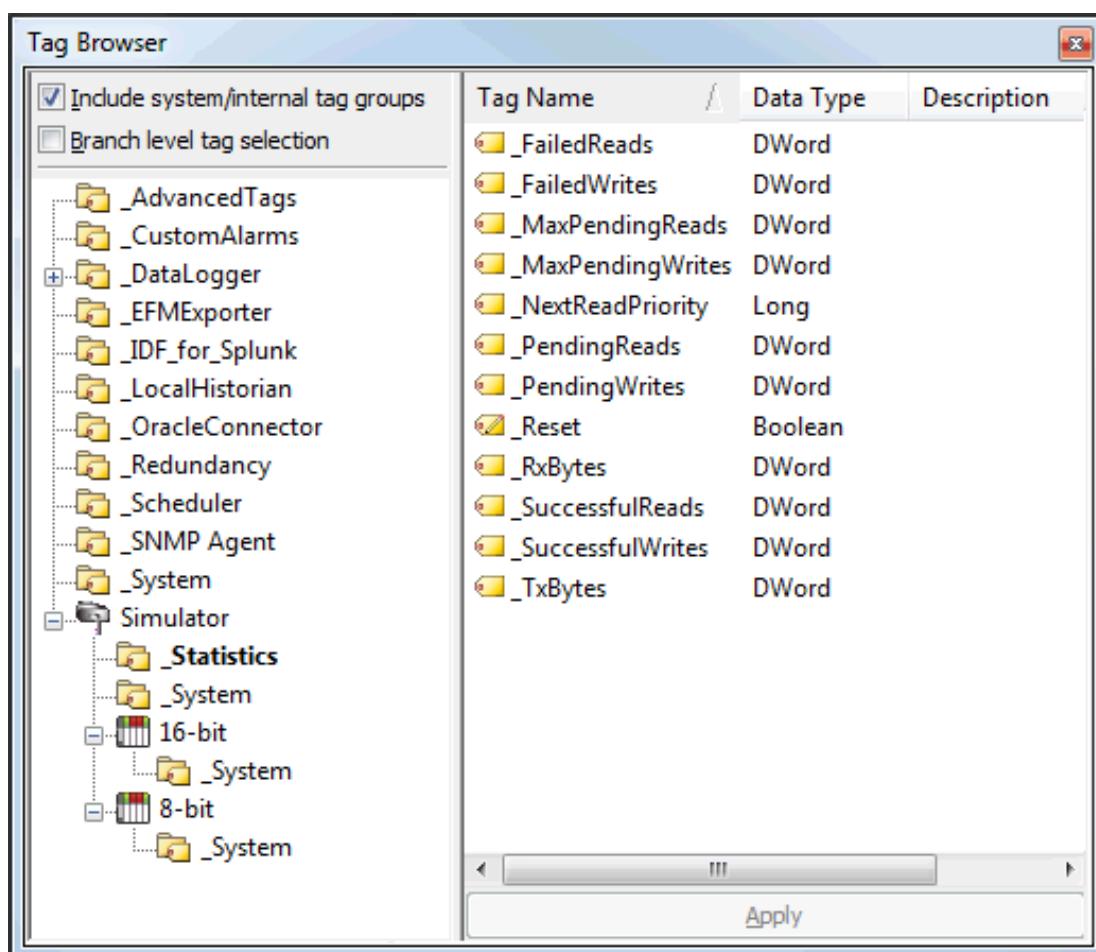
3. Repeat the previous steps to add more server items. As server items are chosen to log, they are added to the **Server Item List**.

4. To add a tag as a server item, click **Add**.

5. To modify a server item in the Server Item List, click **Modify**.

6. To import server items from a CSV file, click **Import CSV**.

7. Next, choose an option under **Table Selection**. Then, select whether each item shares or gets an alias in **Table Format**.
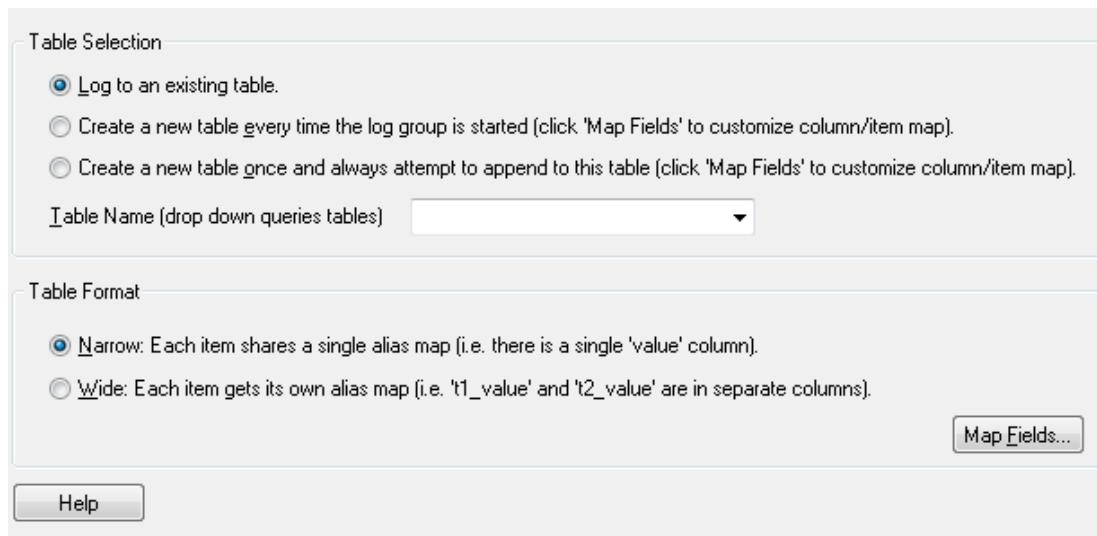
Note: Settings can be changed at any time; they take effect only when the **Apply Changes** button is pressed.

Add Server Item / Modify Server Item

This dialog is used define new items and to modify the item selected in the Server Item list.
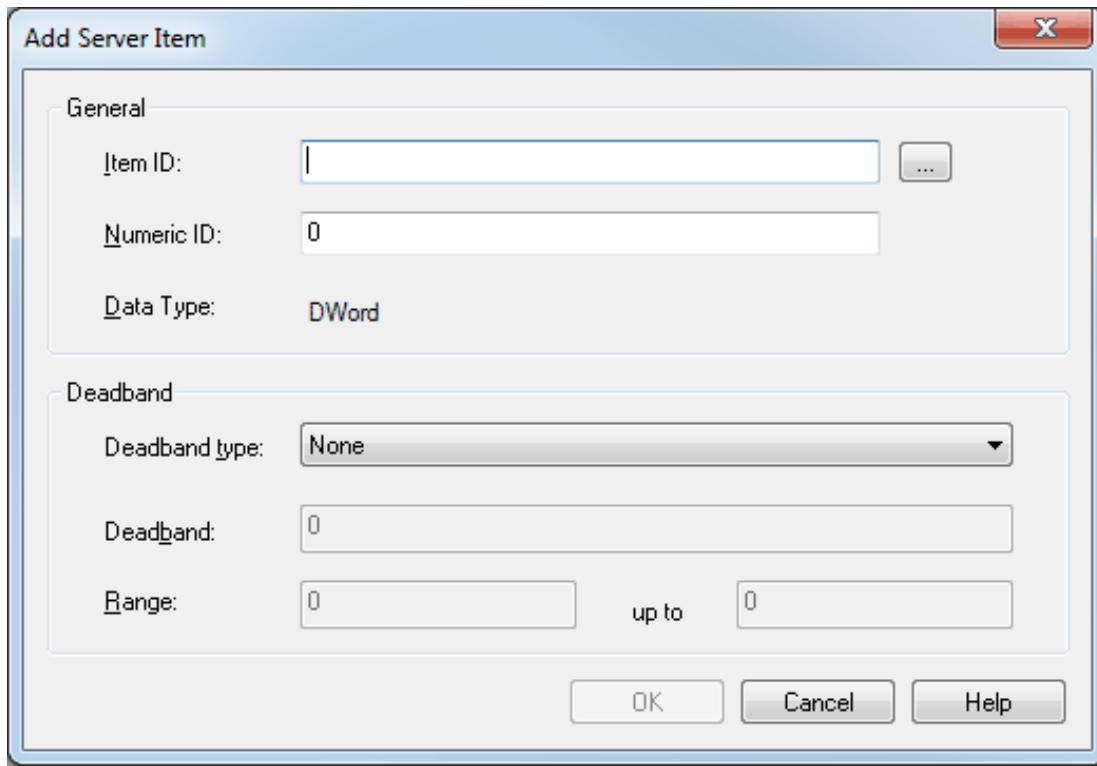
Descriptions of the parameters are as follows:

- **Item ID:** This parameter specifies the full name of the server item. Use the Browse button to search the OPC server project for an item. If an existing server item was chosen and **Modify** was selected, the Item ID field displays the ID of the server item being modified. The maximum length is 256 characters.
- **Numeric ID:** This parameter specifies a numeric alias or identifier for the server item. The valid range is 0 to 2147483647. The default setting is 0.
  Note: When the VARCHAR mapping is enabled, any string of length 64 is valid. Disabling limits the numeric value between 0 and 2147483647.
- **Data Type:** This parameter identifies the data type of the value that will be provided by the specified monitor item.
  Note: Data types available are based on the types supported by the plug-in(s) and driver(s) being logged (with some limitations).
- **Deadband Type:** This parameter specifies the item's deadband type. Options include None, Absolute, and Percent. The default setting is None. Descriptions of the options are as follows:
  - **None:** No deadband is used.
  - **Absolute:** If the difference between two consecutive item data updates is more than the amount specified in the Value field, the update is valid and can produce events. Otherwise, the update is ignored.
  - **Percent:** This option is similar to Absolute except that the Value field is a percentage of the specified range. For example, if the Value is 10 (10%) and the expected range is 0 to 10, the Absolute deadband would be ±1.

- **Deadband:** This parameter specifies the deadband value. Any positive number is supported. If the Deadband Type is set to None, this parameter is disabled. The default value is 0.
- **Range:** This parameter specifies the deadband's range. It is only enabled when Percent is the chosen deadband type. The default value is 0.

See Also:
Map Item Fields

Large Arrays

DataLogger supports logging arrays of the primitive data types.

Array information is converted to a tab-delimited character string for publication to the database table.

The **Map Item Fields** button allows the user to link the Server Item VALUE to the database field _VALUE. The database element _VALUE defaults to VARCHAR(64), but this size may not accommodate a large array. The type and size for _VALUE is dependent on the underlying database configured *(see the General tab)*.

Example:
With a MySQL DSN, a VARCHAR column length can be specified as a value from 0 to 255 before MySQL 5.0.3 and 0 to 65,535 in 5.0.3 and later versions. For MS SQL Server 2014 VARCHAR(N) can have a specified value for N between 1 and 8000, although the storage size is the actual length of the data plus 2 bytes. Specifying an invalid size for the _VALUE type can result in unrelated error messages returned from the database server or (silent) logging of incorrect values.

Notes:

1. Incorrect configuration of the DSN and/or item fields may result in failures that do not produce error messages. If data values appear "stale" (unchanged data logged to the database table even though the data changes are occurring), verify the column width is appropriate for the array size and the underlying database management system.

2. Arrays cannot be used with the Stop and Start trigger items when a value comparison is performed.

Map Item Fields

The Map Item Field dialog is used to map server items to columns in the database table. It shows the server items in the top half of the screen and the database table's columns in the bottom half. How it works depends on whether DataLogger is creating a new table in the database or if the table already exists. To invoke Map Item Fields, click **Data Map Tab** | **Map Fields**.

- If DataLogger is creating a new table, the item-to-column mappings is done automatically by DataLogger. Although users do not need to map the item fields, they can customize the columns. For more information, refer to **Modifying a Database Column**.
- If DataLogger is using a table that already exists, then the **Map Item Fields** dialog must be used to map the server items to database columns. For example, the screen below shows a configuration that has nine server items and one server item mapped to a database column. The server item Channel1.Line1.ConveyorSpeed_TIMESTAMP has been mapped to the database column Channel1_ Line1_ConveyorSpeed_TIMESTAMP.
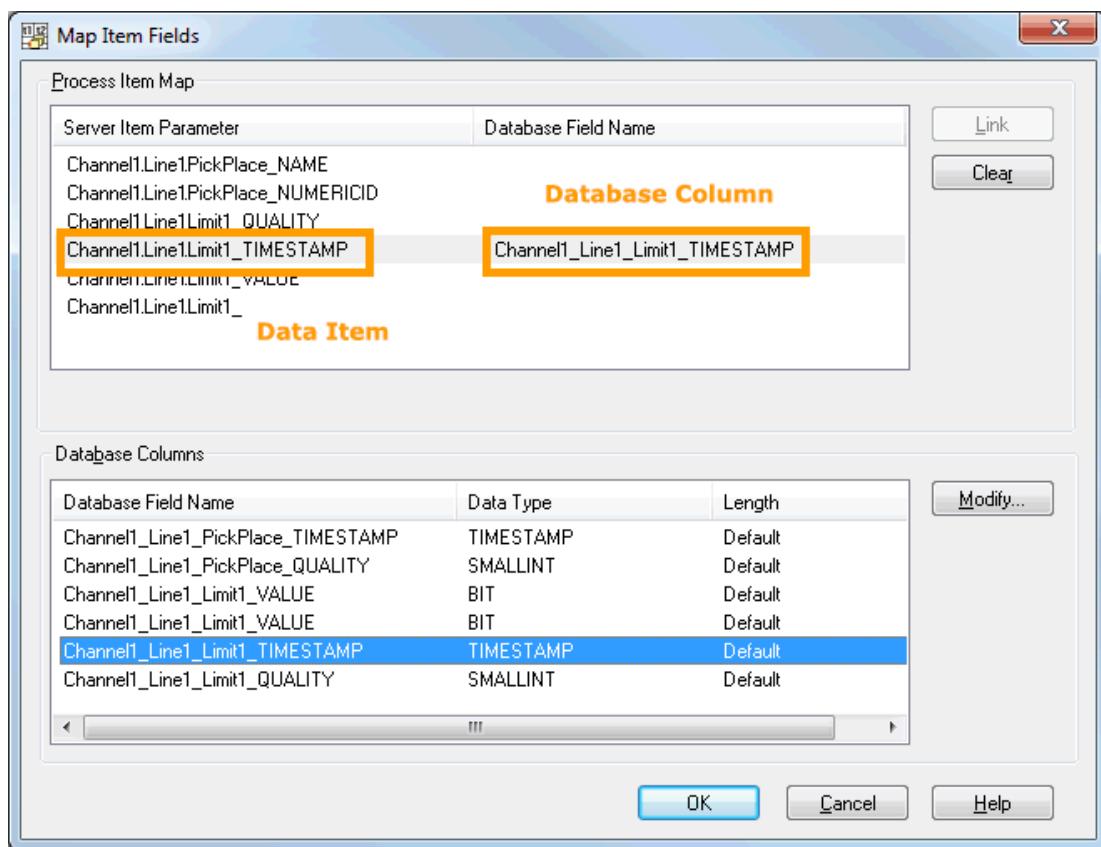
Mapping a Server Item

1. To start, click on the **Server Item**.

2. Then, select the corresponding column.

3. Click **Link**. The database column name is displayed to the right of the server item.

Changing the Mapping

1. To start, click on the **Server Item**.

2. Then, click **Clear** in order to clear the field name.

3. Select the correct column.

4. Next, click **Link**. The database column name is displayed to the right of the server item.

Modifying a Database Column

When mapping server items to a new table, click **Modify** to change column parameters. For example, to change the default name for a column from "_NAME" to "_Line4v88," users would select the item and then click **Modify**. The **Modify Column Definition** dialog is used to change the automatically assigned default values for column name, SQL data type, and SQL length.
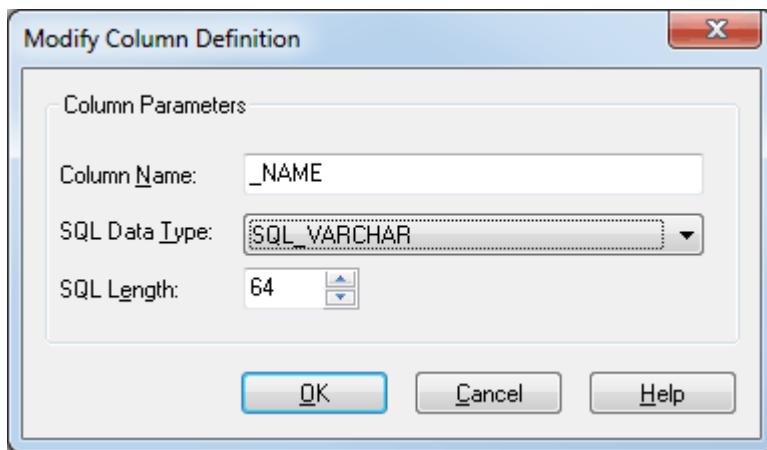
DataLogger enforces a 256 character limit on the Item ID Name, Server Item, and Database Field Name. In practice, however, the maximum length also depends on the limits of the database and ODBC driver being used. The maximum name length for a database column is often 64 or 128 characters. When data is being logged in Wide Format, the Server Item and Database Field Name values consists of the channel, device, and group names appended to the front of the Item ID name. This further limits the usable name length of the Item ID. For more information, refer to **Narrow vs. Wide Table Format**.

Note: Character limits are also affected by software versions and array size *(see also Large Arrays)*.

Narrow vs. Wide Table Format

DataLogger can map fields in either the **Narrow Format** or **Wide Format**. Users must specify the format to be used for each database table logged into. The setting is reached through the **Data Map Tab**.
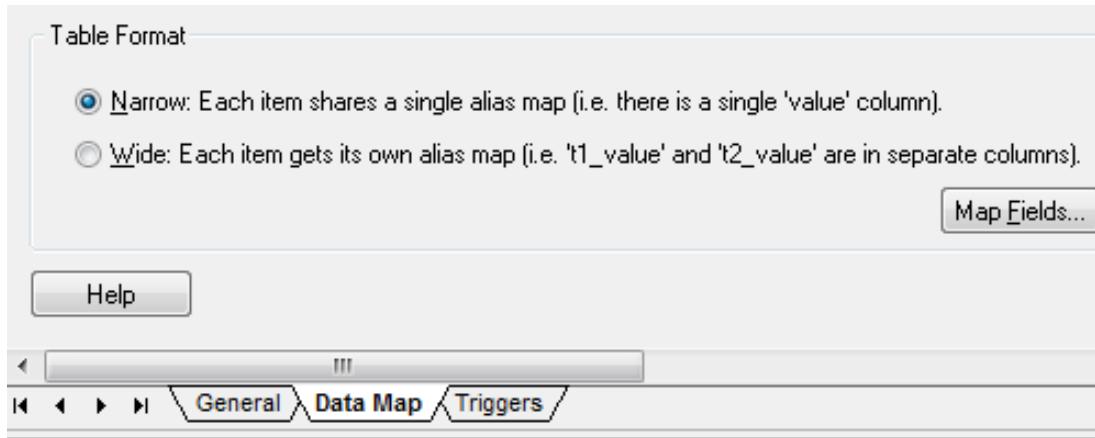
Narrow and Wide Format Examples
The following example shows the narrow and wide formats, with three server items that need to be logged.

- Ch_1.Dev_1.Temp
- Ch_1.Dev_1.Pos
- Ch_1.Dev_1.Speed

Narrow Format Examples
The two tables below show the three server items being logged using the Narrow Format. All three server items have been logged into five columns. The table below shows the narrow format with a static time interval (meaning every *x* seconds or minutes).*

Narrow Format with Data Logged on Static Interval

Name	Numeric ID	Value	Quality	Time
Ch_1.Dev_1.Temp	1000000	38	192	2007 02 16 13:44:26.832
Ch_1.Dev_1.Pos	1000001	22	192	2007 02 16 13:44:26.832
Ch_1.Dev_1.Speed	1000002	103	192	2007 02 16 13:44:26.832

In the table below, the same three server items are being logged using the narrow format; however, data is logged on data change. In this example, Ch_1.Dev_1.Temp has changed value, so it is the only row that is logged.

Narrow Format with Data Logged on Data Change*

Name	Numeric ID	Value	Quality	Time
Ch_1.Dev_1.Temp	1000000	38	192	2007 02 16 13:38:02.142

*There was a temperature change (item 1). The other two items did not change.

Wide Format Example

The next example demonstrates Wide Format. When wide format is selected, four columns are logged for each server item. The four columns are: _VALUE, _NUMERICID, _TIMESTAMP and _QUALITY.

In this example, there are three server items: Ch_1.Dev_1.Temp, Ch_1.Dev_1.Pos, and Ch_1.Dev_1.Speed. In wide format, this would result in up to twelve columns. Columns can be unmapped in the **Map Item Fields** dialog so that only necessary columns are logged. In the examples below, the _NUMERICID column has been unmapped for each server item.

Ch_1.Dev_1.Temp_VALUE
Ch_1.Dev_1.Temp_TIMESTAMP
Ch_1.Dev_1.Temp_QUALITY
Ch_1.Dev_1.Pos_VALUE
Ch_1.Dev_1.Pos_TIMESTAMP
Ch_1.Dev_1.Pos_QUALITY
Ch_1.Dev_1.Speed_VALUE
Ch_1.Dev_1.Speed_TIMESTAMP
Ch_1.Dev_1.Speed_QUALITY

Wide Format

Ch_1.Dev_1.Temp_ VALUE	Ch_1.Dev_1.Temp_ TIMESTAMP	Ch.1.Dev_1.Temp_ QUALITY	Ch_1.Dev_ 1.Pos
38	2007 02 16 13:44:26.832	192	22

Notes:

1. Only a portion is shown due to the width of the sample table. With the wide format, all fields are logged each time data is logged. In this example, nine columns would be logged every time DataLogger inserted data into the database table. If data was being logged on a static interval, nine columns would be logged every x milliseconds. If data was being logged on data change, nine

columns would be logged every time any of the fields had a data change.

2. For more information on the options available to log data on static intervals, data change, or on transition from a start condition to a stop condition, refer to **Add Trigger: Logging Conditions**, **Add Trigger: Based on Condition (Expression)**, and **Add Trigger: Based on Time (Absolute)**.

3. Log groups are limited to 300 columns using wide table format with a MySQL database.

Triggers Tab

The Triggers tab is used to define one or more triggers for a log group. When it is first displayed, the Logging Triggers list displays one default trigger. Each trigger is defined by two major parameters: when and how.

1. When is the trigger true?

- **Always Triggered:** The trigger is always true as long as the OPC server is in Runtime. For example, the trigger does not have a false state, except for when the OPC server is inactive.
- **Based on Time (Absolute):** The trigger is true only for certain days and hours. For example, from Monday to Friday, 8 AM to 5 PM.
- **Based on Condition (Expression):** The trigger is true when conditions in an expression are true. For example, a trigger can be defined to be true when the value of tag XYZ is greater than 100.

2. When the trigger is true, how is data logged?
While a trigger is true, data is logged (or inserted into a database table) in one or more of the following ways.

- **Log on Static Interval:** Data is logged at regular intervals. For example, every 500 milliseconds.
- **Log on Data Change of any value within the log group:** Data is logged only when the value of the server item in the log group changes. For example, if the log group has four server items and the trigger is set to log on data change, data is only logged when there is a change in at least one of those four server items. If the configured table format is narrow, then only the item that changed is logged. If the configured table format is wide, then the current value of all items in the log group is logged when any of the items change.
- **Log entries for all items when the Monitored Item's value changes:** Data is logged for all items in the log group whenever the monitored item's value changes.
- For the Based on Time (Absolute) and Based on Condition (Expression) trigger types, data can also be logged when the trigger transitions from true to false or false to true in one or both of the following ways:

    - **Log all items in log group once when start condition is met:** Data is logged for all items in the log group whenever the trigger goes from false to true.
    - **Log all items in log group once when stop condition is met:** Data is logged for all items in the log group whenever the trigger goes from true to false.

Notes:

1. The format of the data being logged depends on whether the narrow or wide format is chosen for the log group. For more information, refer to **Narrow vs. Wide Table Format**.

2. Arrays cannot be used with the Stop and Start trigger items when a value comparison is performed.

See Also:
Condition Examples
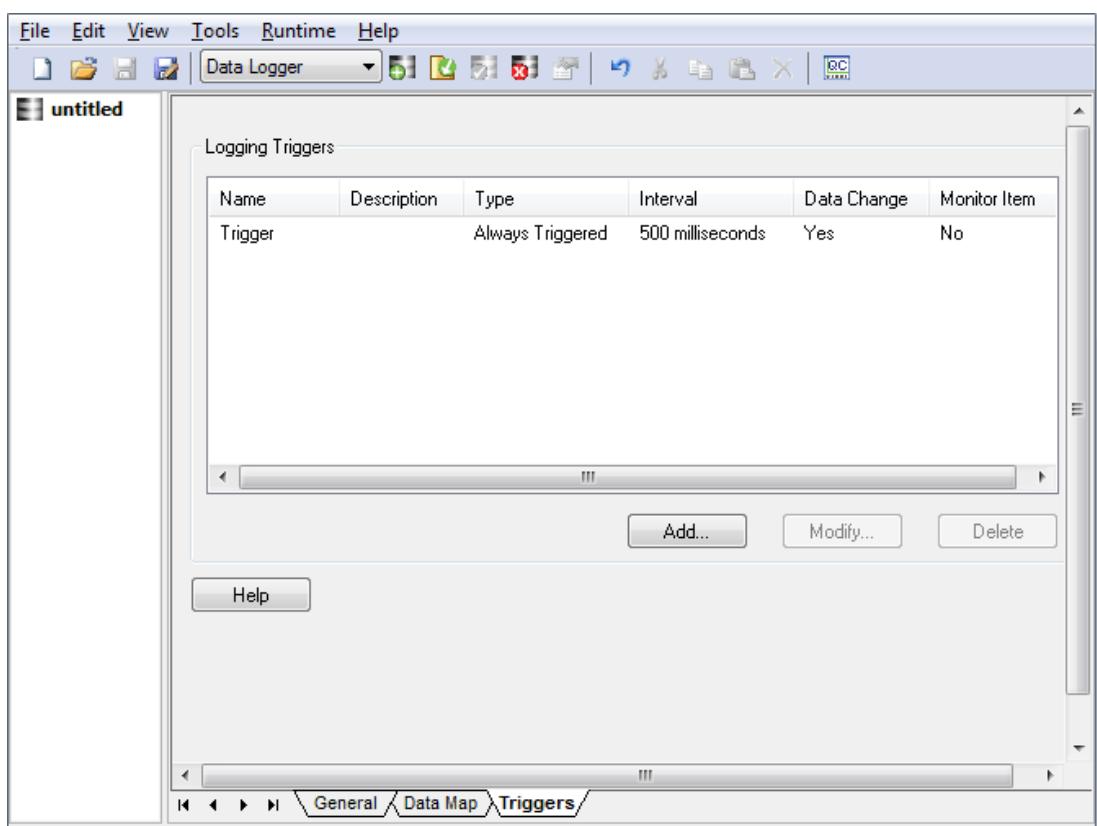
Descriptions of the buttons are as follows:

- **Add:** When clicked, this button launches the Add Trigger: General dialog for adding a trigger.
- **Modify:** When clicked, this button launches the Add Trigger: General dialog for modifying the trigger selected in the Logging Triggers list.
  Note: The Add and Modify dialogs are the first in a series of dialogs that lead users through the trigger-definition process. For more information, refer to **Add / Modify Trigger: General**.
- **Delete:** When clicked, this button removes the trigger selected in the Logging Triggers list.

Note: For more information on each trigger dialog, click **Help**.

Add / Modify Trigger: General
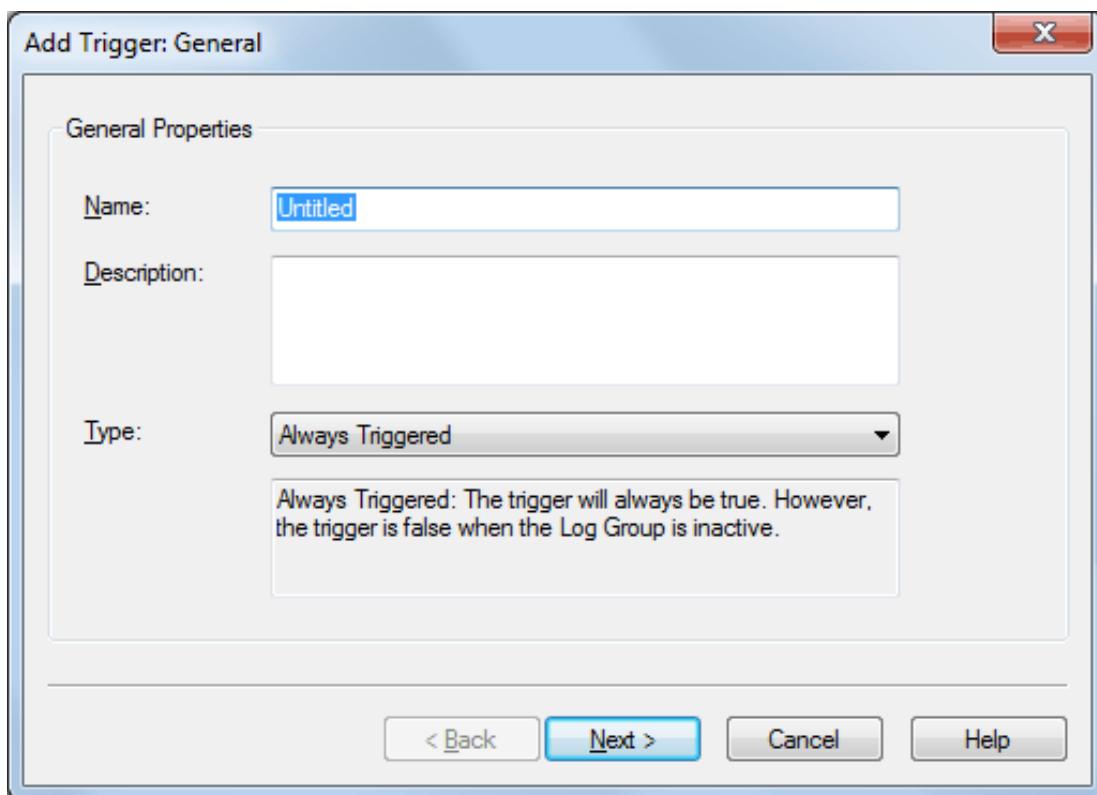
Descriptions of the parameters are as follows:

- **Name:** This parameter specifies the trigger name. It cannot include a period or double quote, or begin with an underscore. The maximum length is 256 characters.
- **Description:** This parameter specifies a description of the trigger. The maximum length is 4096 characters.
- **Type:** This parameter specifies the type of trigger. Options include Always Triggered, Based on Time (Absolute), and Based on Condition (Expression). The default setting is Always Triggered. Descriptions of the options are as follows:

  - **Always Triggered:** The trigger is always be true; that is, it has no false state except for when the Log Group is inactive.
  - **Based on Time (Absolute):** The trigger is true during an absolute time of day only. When selected, the **Add Trigger: Based on Time (Absolute)** dialog is displayed once **Next** is clicked.
  - **Based on Condition (Expression):** The trigger is true or false based on a conditional expression. For example, it is true when the value of tag "x" is greater than 100; otherwise, it is false. When selected, the **Add Trigger: Based on Condition (Expression)** dialog is displayed once **Next** is clicked.

Modifying General Properties

To invoke the **Modify Trigger - General** dialog, select a trigger and then click **Modify**. The fields in the dialog are the same as those displayed when adding a new trigger, except that the field values reflect the existing trigger information. For individual field definitions, see the descriptions above.
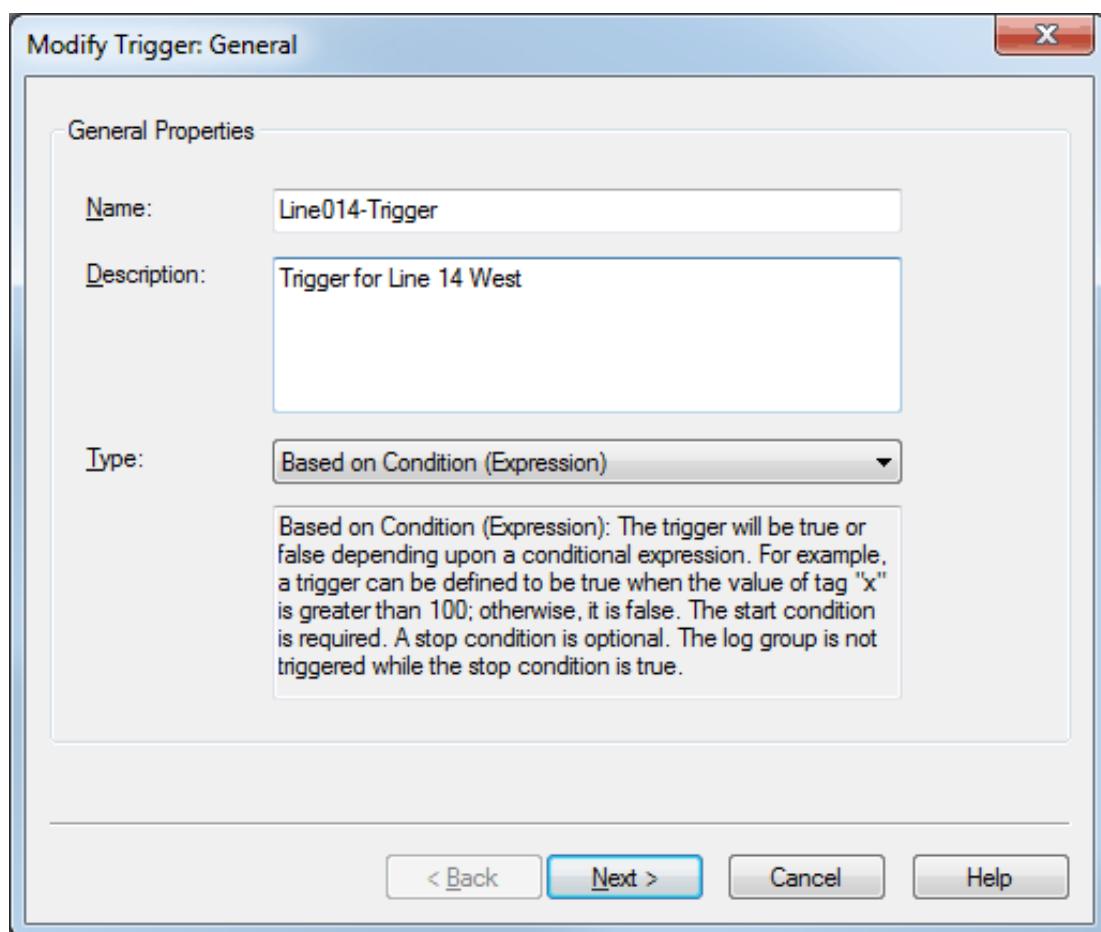
Add Trigger: Based on Time (Absolute)

For information on adding an Absolute trigger, refer to the instructions below.
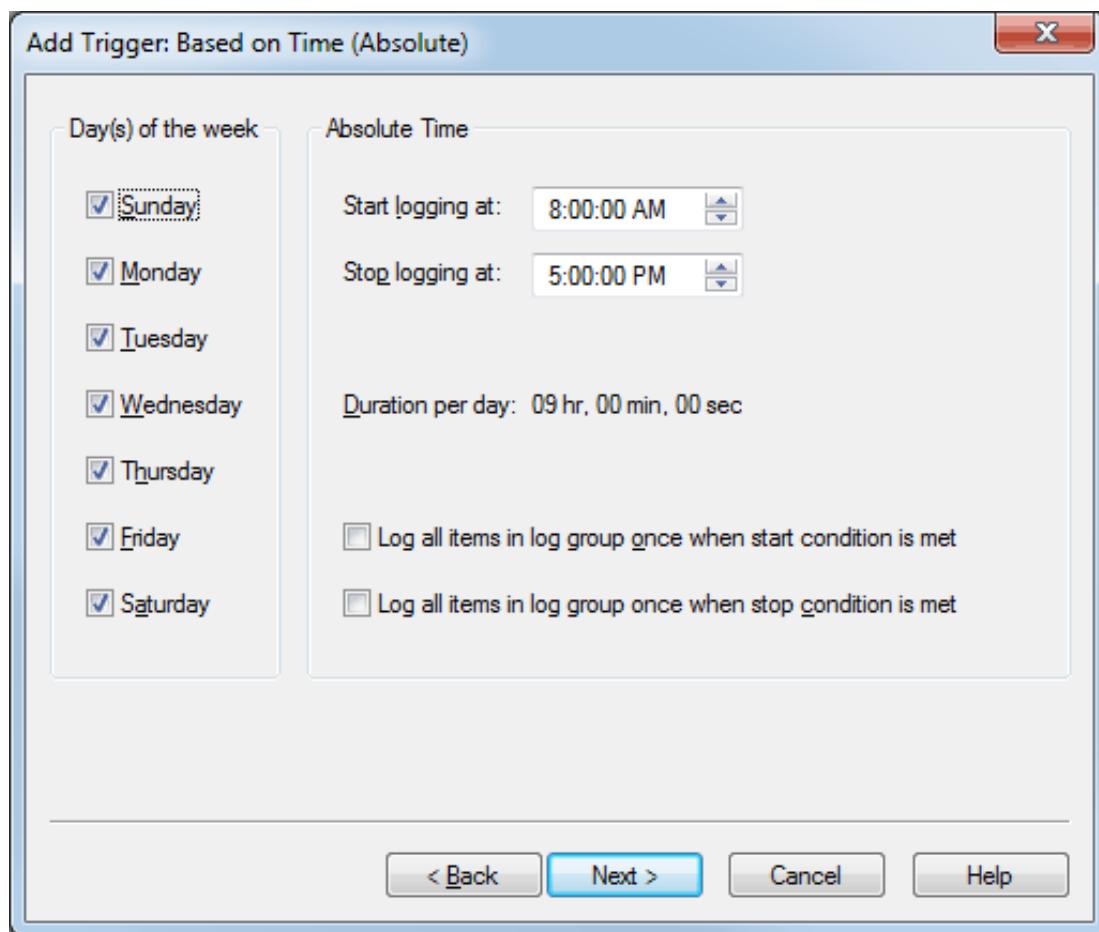
1. To start, select one or more days in the **Day(s) of the Week** column.

2. Click on the **Start logging at** field and enter the start time (in hours, minutes or seconds). Then, enter **AM** or **PM**.

3. Click on the **Stop logging at** field and set the stop time (in hours, minutes or seconds).

4. The total duration per day is displayed in the **Duration per day** field.

5. Optionally configure the logging behavior when the start and / or stop times are met.

Notes:

1. The **Start logging at** and **Stop logging at** fields control the logging time for the days selected in the Days column. In the example shown above, the trigger would be true from Sunday through Saturday, from 8:00:00 AM to 5:00:00 PM.

2. Overlapping midnight (meaning, overlapping from one day to the next) requires that two triggers be created because this dialog defines the duration per day. For example, in order to start logging at 9 PM on Friday and stop at 3 AM on Saturday, then one trigger would be created with Friday checked and **Start logging at** set to 9:00:00 PM and **Stop logging at** set to 11:59:59 PM. Another trigger would have to be created with Saturday checked and **Start logging at** set to 12:00:00 AM and **Stop logging at** set to 3:00:00 AM.

Add Trigger: Based on Condition (Expression)

Select a link from the list below to jump to a specific Trigger Start / Stop topic.

Overview

Start Condition

Stop Condition

See Also:

Condition Examples

Overview of Start and Stop Conditions

The Event Expression dialog is used to define start and stop conditions for expression triggers. Note the following:

- An event expression trigger can only include one start condition and one stop condition. To have the log group be triggered by multiple expressions, users must create multiple event expression triggers.
- At least one start condition is required for an event expression. A stop condition is optional.
- The log group is triggered when the start condition becomes true. The log group remains triggered regardless of subsequent changes to the start expression state.
- If there is a stop condition, the log group becomes not-triggered when the stop condition is true. If there is no stop condition, the log group remains triggered until the OPC server Runtime stops.
- If the start and stop conditions are both true, then the log group is not triggered. While the start condition is true, the stop condition controls whether or not the log group is triggered.
- Log groups are not triggered while the stop condition is true.
- Arrays cannot be used for the start and stop trigger items when a value comparison is to be performed.

Start Condition
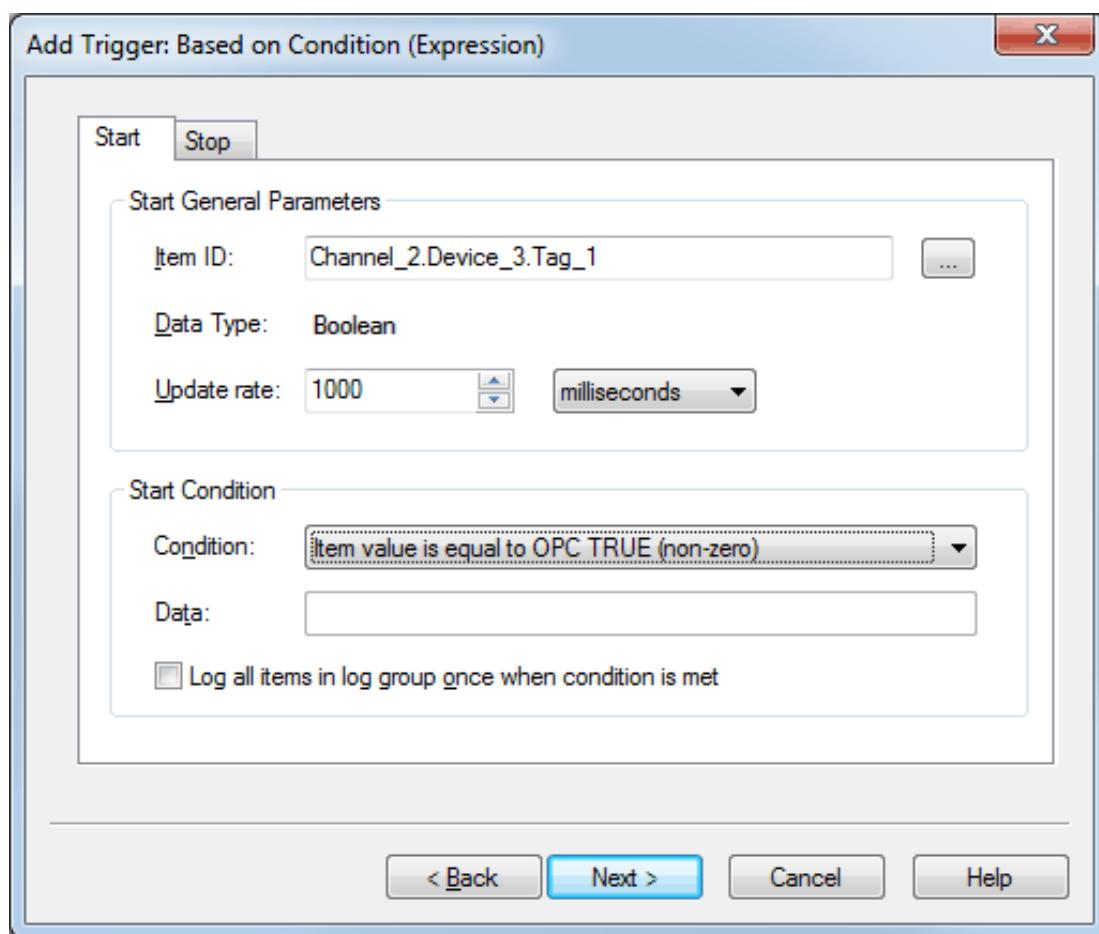
Descriptions of the parameters are as follows:

- **Item ID:** This parameter specifies the ID of the server item that controls the condition. The server item can be one that has been selected to be logged, or it can be an item that is not being logged. To search for the server item, click the Browse button to locate and select an Item ID.
- **Data Type:** This parameter identifies the data type of the value that will be provided by the server item.
- **Update Rate:** This parameter specifies a different update rate for the server item defined in the **Item ID** field. The log group's global update rate is set on the **General Tab**. The default setting is 1000 milliseconds.

  Note: The value in this field can be different from the value in **Update Rate** on the **Stop Condition** tab.
- **Start Condition:** This parameter specifies the Condition. If applicable, enter the conditional value in the **Data** field. For more information, refer to "Examples" below.
- **Log all items in log group once when condition is met:** When checked, data is logged once when the configured condition becomes true. For example, if the condition is "Item data set to bad quality," all entries in the log group log once when the item quality becomes bad quality. The default setting is unchecked.

Examples

1. If **Condition** is set to "Item value is equal to OPC TRUE (-1)," the condition is true if the server item in the **Item ID** field is true. For this condition, leave the **Data** field blank as it is not necessary. Note that

this assumes that the value in **Data Type** is Boolean; if any other data type is entered, then the server item value must be -1 in order to make this condition true.

2. If **Condition** is set to "Item data set to bad quality," the condition is true if the server item in the **Item ID** field is quality=bad. For this condition, leave the **Data** field blank as it is not necessary.

3. If **Condition** is set to "Item data set not equal to a specific value" and the **Data** field value is 144, then the condition is true if the server item in the Item ID field is not equal to 144 (it could be higher or lower).

4. If **Condition** is set to "Item data set less than a specific value" and the **Data** field value is 144, then the condition is true if the server item in the Item ID field is less than 144.

5. If **Condition** is set to "Item data has not changed over a certain time period (ms)" and the **Data** field value is 15000 (15,000 milliseconds), then the condition is true if the tag in the **Item ID** field has not changed value for 15 seconds.

See Also:
Start Condition Examples
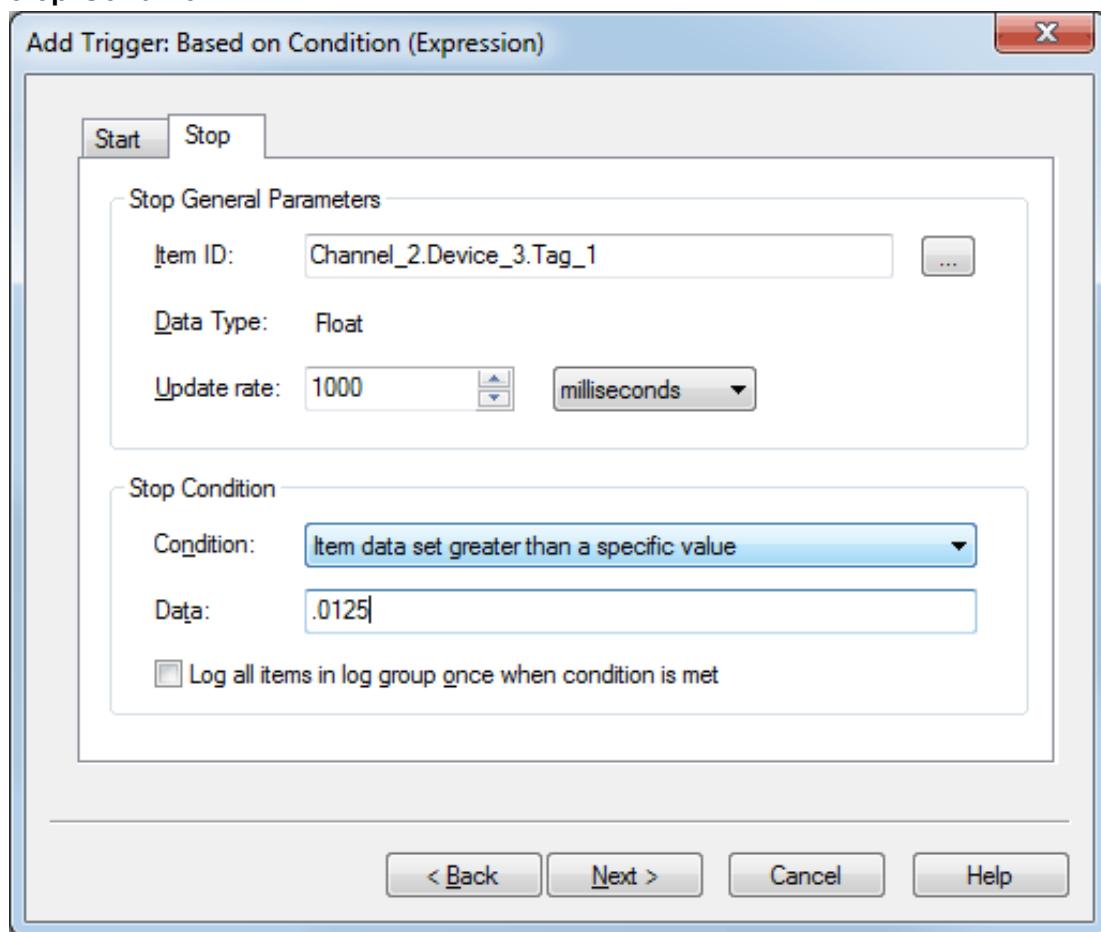
Stop Condition

Descriptions of the parameters are as follows:

- **Item ID:** This parameter specifies the ID of the server item that controls the condition. The server item can be one that has been selected to be logged, or it can be an item that is not being logged. To

search for the server item, click the Browse button to browse and select an Item ID.

- **Data Type:** This parameter identifies the data type of the value that will be provided by the server item.
- **Update Rate:** This parameter specifies a different update rate for the server item defined in the **Item ID** field. The log group's global update rate is set on the **General Tab**. The default setting is 1000.
  Note: The value in this field can be different from the value in **Update Rate** on the **Start Condition** tab.
- **Stop Condition:** This parameter specifies the **Condition**. If applicable, enter the conditional value in the **Data** field. For more information, refer to "Examples" below.
- **Log all items in log group once when condition is met:** When checked, data is logged once when the configured condition becomes true. For example, if the condition is "Item data set greater than a specific value," all entries in the log group log once when the item data becomes greater than the configured specific value. The default setting is unchecked.

Examples

1. If **Condition** is set to "Item data set less than a specific value" and the **Data** field value is 144, the condition is rue if the server item in the **Item ID** field is less than 144.

2. If **Condition** is set to "Item data has not changed over a certain time period (ms)" and the **Data** field value is 15000 (15,000 milliseconds), then the condition is true if the tag in the **Item ID** field has not changed value for 15 seconds.

3. If **Condition** is set to "Item data set to bad quality," the condition is true if the server item in the **Item ID** field is quality=bad. For this condition, leave the **Data** field blank as it is not necessary.

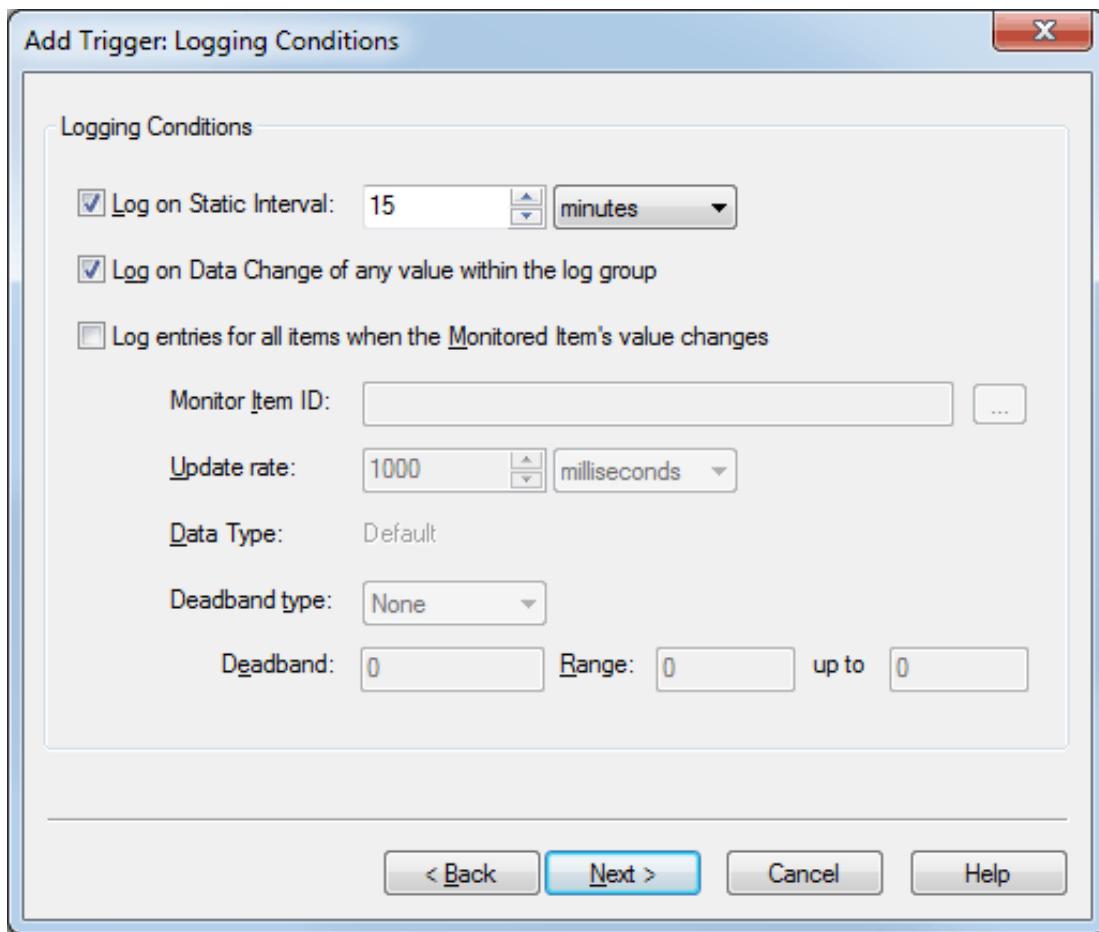## See Also:
Stop Condition Examples

Add Trigger: Logging Conditions

Descriptions of the parameters are as follows:

- **Log on Static Interval:** When checked, data is logged on a static interval (time-based). The valid range is 10 milliseconds to 49 days. The default setting is checked. The default value is 500 milliseconds. For more information, refer to **Update Rate**.
- **Log on Data Change of any value within the log group:** When checked, data is only logged when the value of the server item in the log group changes. For example, if the log group has four server items and the trigger is set to log on data change, then data is only logged when there is a change in at least one of those four server items. If the configured table format is narrow, then only the item that changed is logged. If the configured table format is wide, then the current value of all items in the log group is logged when any of the items change. The default setting is checked.
- **Log entries for all items when the Monitored Item's value changes:** When checked, data for all items in the log group are logged whenever the monitored item's value changes. The default setting is unchecked.
    - **Monitor Item ID:** This parameter specifies the full name of the server item. To locate an item in the OPC server project, click **Browse**. Upon selection, the Monitor Item ID parameter is populated with the ID of the selected server item. The maximum length is 256 characters.
    - **Update rate:** This parameter specifies the update rate for data coming from the OPC server for the selected item. The default setting is 1000 milliseconds. The range for each unit is:
        - **Milliseconds:** 10 to 4294967295.
        - **Seconds:** 1 to 4294967.
        - **Minutes:** 1 to 71582.

- **Hours:** 1 to 1193.
- **Days:** 1 to 49.
- **Data Type:** This parameter identifies the data type of the value that will be provided by the specified monitor item.
- **Deadband type:** This parameter specifies the item's deadband type. Options include None, Absolute, and Percent. The default setting is None. Descriptions of the options are as follows:
  - **None:** No deadband is used.
  - **Absolute:** The absolute change in data required to be considered a data change.
  - **Percent:** This option is similar to Absolute except that the Value field is a percentage of the specified range. For example, if the Value is 10 (10%) and the expected range is 0 to 10, the Absolute deadband would be ±1.
- **Deadband:** This parameter specifies the deadband value. Any positive number is supported. If the Deadband Type is set to None, this parameter is disabled. The default value is 0.
- **Range:** These parameters specify the low and high range of the server item for use in applying the percent deadband. The range is only enabled and required when the Deadband Type is Percent. The default setting for both the low and high value is 0; however, the high range value must be greater than the low range value.

Update Rate

The Update Rate affects Static Interval, Data Change, and Trigger Start To Stop selections in two ways:

- The Update Rate should be set to a value less than the Static Interval value, so that the log group's server items are updated more frequently than the static interval.
- Since the update rate determines when the server item changes are processed by DataLogger, the lower the update rate, the more frequently data changes are received and processed by DataLogger.

Timestamp Values

DataLogger maintains two timestamp values for each OPC server item in a log group:

- One timestamp is updated each time the OPC server scans the controller address, regardless of whether the value has changed since the last scan. As such, this timestamp reflects the "freshness" of the value. This is the timestamp value logged to the database if the insertion was triggered by a static-interval trigger (that is, when **Log on Static Interval** is checked).
- The other timestamp is updated each time DataLogger has detected a data-value change. If the value of the server item has not changed, the timestamp is not updated. This is the timestamp value logged to the database if the insertion was triggered by a change in the data value (that is, when **Log on Data Change** is checked).

Add Trigger: Summary

Review the details and click **Finish**. Alternatively, click **Back** to return to previous dialogs and make changes.

Add Trigger: Summary

Summary

```
Log on Data Change: Yes
Start Snapshot: No
Stop Snapshot: No

Start Time: 08:00:00
Stop Time: 17:00:00

Day(s) of week:
Sunday
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
```

< Back   Finish   Cancel   Help
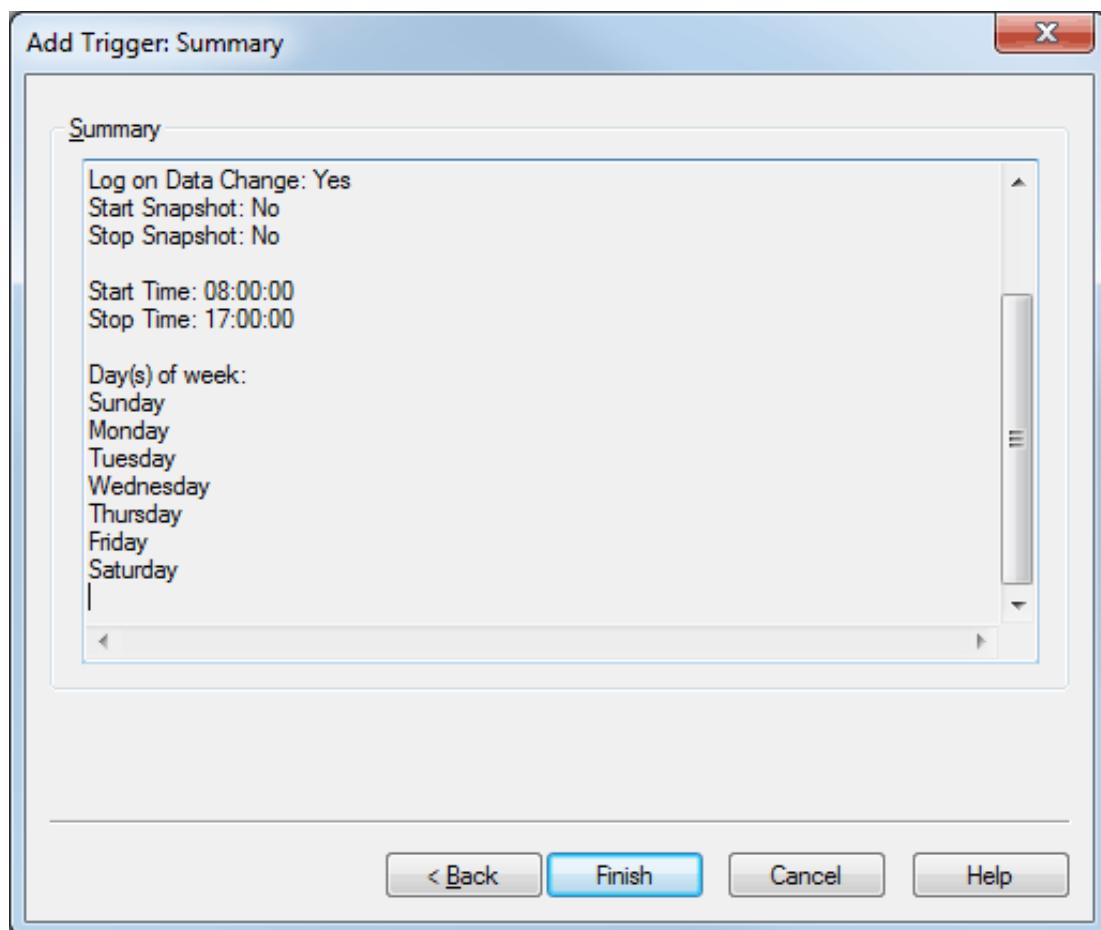
System Tags

_DataLogger (Root)

Tag	Access	Description
_TriggeredGroupCount	Read Only	The number of log groups that are currently triggered. A log group is triggered when at least one of its triggers is true.
_NonTriggeredGroupCount	Read Only	The number of log groups that are not currently triggered.
_EnabledGroupCount	Read Only	The number of log groups that are currently enabled, whether they were enabled at Runtime or via the _Enabled System Tag.
_DisabledGroupCount	Read Only	The number of log groups that are not currently enabled.

_DataLogger.<*log group name*>

Tag	Access	Description
_Buffering	Read Only	When set True, the log group's store file currently contains data.*
_Description	Read Only	Configured description of the log group.
_Enabled	Read / Write	The log group is evaluating server items and processing triggers. When a log group is enabled, it may or may not be triggered.
_Error	Read Only	The log group is in an error state (such as if it failed to connect to the database).
_LogDataBit	Read / Write	When set True, all server items are logged. The _LogDataBit tag is reset to False automatically.
_NoError	Read Only	The log group is not in an error state.
_SessionInsertionCountHi	Read Only	The upper 32 bits of the insertion count. Insertion count is the number of data insertions since the log group was last enabled.
_SessionInsertionCountLo	Read Only	The lower 32 bits of the insertion count.
_TimeToOpenRecordsetMS	Read Only	The time it took to initially open a connection to the database in milliseconds.
_Triggered	Read Only	The log group is triggered, that is, at least one of the log group's triggers is true. When a log group is triggered, the triggers that are true insert data on static interval, data change, and / or on transition from a start condition to a stop condition, as determined by the logging settings in **Add Trigger: Logging Conditions**, **Add Trigger: Based on Condition (Expression)**, or **Add Trigger: Based on Time (Absolute)**.

*This tag may be True if the database is disconnected and the log group is currently logging directly to the store file. It may also be True if the database has recently been reconnected and the log group is still in the process of forwarding its contents to the database.

Error Descriptions

The following error / warning messages may be generated in the Event Log, which is displayed in the lower pane of the OPC server. The messages below are listed in alphabetical order. Click on a message below to view a description of the error message.

Error importing CSV log item record <number>: item ID <tag name> has an unsupported data type.
Error importing CSV log item record <number>: must enter a valid item ID.
Error importing CSV log item record <number>: item ID <tag name> does not reference a valid tag.
Failed to create table for unknown reason.
Failed to create table name <table> for log group <log group> on DSN <DSN>.
Failed to create the data table for log group <log group>. (Memory Exception).
Failed to create the data table for log group <log group>. (Reason: <reason>).
Failed to open store and forward file <filename>: <reason>.
Failed to register batch ID <batch ID> for log group <log group>.
Failed to register log item for log group <log group> (Log item: <log item>).
Failed to register trigger item for log group <log group>. (Trigger: <trigger>).
Failure reading from store and forward file <filename>: <reason>.
Failure writing to store and forward file <filename>: <reason>.
Invalid store and forward file <filename>.
Log groups using MySQL DSN and the 'wide' table format are limited to <count> items. The DataLogger runtime rejects logging activity for this group until the item count is reduced.
Log group <log group> failed to create table <table name> on DSN <DSN>.
Log group <log group> failed to validate table <table name>. (Reason: <reason>).
Log group <log group> has requested creation of at least one TIMESTAMP column for table <table>. This has been adjusted to the MS-SQL required DATETIME.
Saving store and forward file <old filename> as <new filename>.
Store and Forward file <filename> has reached maximum size of <file size> MB.
Store and Forward file <filename> is incompatible with current log item list.
Table name <table name> exists for log group <log group> on DSN <DSN>, which is in "create table once and append" mode.
The log items in log group <log group> need to be mapped to the column in table before logging occurs. Please setup this mapping using the Data Map tab.
The query failed because the data source is not appendable. Please double check the user permissions.
There are no log items to log for log group <log group>.
Unable to connect to data source <source>.
Unable to connect to data source <source> (Memory Exception).
Unable to connect to data source <source> (Reason: <reason>).
Unable to open recordset on log group <log group>. (Memory Exception)
Unable to open recordset on log group <log group>. (Reason: <reason>).
Unable to query recordset on Log Group <log group> (Memory Exception)

Unable to query recordset on Log Group <log group>. (Reason: <reason>).
Unable to validate table <table> (Memory Exception).
Unable to validate table <table> (Reason: <reason>).
Unknown error while executing query in table <table> for log group <log group>.

Error importing CSV log item record <number>: item ID <tag name> has an unsupported data type.

Error Type:
Error

Possible Cause:
The log item references a tag with an unsupported data type.

Solution:

1. Update the log item with a reference to a tag with a valid data type.

2. Set the data type of the referenced tag to a valid data type.

Error importing CSV log item record <number>: item ID <tag name> does not reference a valid tag.

Error Type:
Error

Possible Cause:
The log item references a tag that does not exist in the project.

Solution:

1. Add the tag back into the project.

2. Reference the correct tag.

Error importing CSV log item record <number>: must enter a valid item ID.

Error Type:
Error

Possible Cause:
The log item cannot reference an empty string.

Solution:

1. Remove the invalid log item.

2. Update the log item with a reference to a valid tag.

Failed to create table for unknown reason.

Error Type:

Serious / Runtime

Possible Cause:

An unknown failure occurred while generating the table. This is generally due to an ODBC driver error.

Solution:

Verify the DSN settings and consult the ODBC driver documentation and / or create a table manually.

See Also:

Defining the Database Table

Setting up a DSN

Failed to create the data table for log group <log group>. (Reason: <reason>).

Error Type:

Serious / Runtime

Possible Cause:

1. DataLogger was unable to create a table because of the reason provided by the ODBC driver; such as, "Syntax error in field definition."

2. Microsoft Access databases are being used while the DataLogger log group is configured to create a new table on start and the Database field name with data type is set to VARCHAR(64). Microsoft Access attempts to transform the data type to WVARCHAR(128), but does not allow the new table to be created after the DataLogger column mapping is updated to use WVARCHAR(128).

3. The data includes an array that is too large or improperly defined.

Solution:

1. Verify the DSN settings, consult the ODBC driver documentation, and / or create a table manually.

2. Set the DataLogger log group to log to an existing table instead of attempting to log to a new table on start. Alternatively, if the database fields map to a VARCHAR data type with a default length of 64, increase the length to 128. In order for this change to work, users must either delete existing tables in the database that use the same table naming convention or create a new table name before enabling the updated log group. To prevent this issue entirely, use Microsoft SQL or MySQL instead of Microsoft Access.

3. Address the data type and character limit issues *(see Large Arrays)*.

See Also:

Defining the Database Table

Setting up a DSN

Setting up a DataLogger Configuration

Failed to create table name <table> for log group <log group> on DSN <DSN>. (Reason: <reason>).

Error Type:
Warning / Runtime

Possible Cause:
DataLogger was unable to create a table for the reason given.

Solution:
Verify the DSN settings and consult the ODBC driver documentation and / or select create a table manually.

See Also:
Defining the Database Table
Setting up a DSN

Failed to create the data table for log group <log group>. (Memory Exception)

Error Type:
Serious / Runtime

Possible Cause:
DataLogger was unable to create a table because of a memory exception.

Solution:
Verify the DSN settings and consult the ODBC driver documentation and / or create a table manually.

See Also:
Defining the Database Table
Setting up a DSN

Failed to open store and forward file <filename>: <reason>.

Error Type:
Warning

Possible Cause:
DataLogger was unable to open the store file. This may be due to the following reasons:

1.   The file has already been opened by another process.

2.   The configured directory is invalid or Read Only.

Solution:

1.   Verify that the Store and Forward file is not already open by another process.

2.   Verify that the configured directory is valid and contains proper permissions.

Failed to register batch ID <batch ID> for log group <log group>.

Error Type:
Warning / Runtime

Possible Cause:
The Batch Identifier (ID) does not exist in the server configuration or has an invalid address format.

Solution:
On the **General tab**, verify the Batch Identifier item exists in the server configuration.

Failed to register log item for log group <log group>. (Log item: <log item>)

Error Type:
Warning / Runtime

Possible Cause:
The specified item does not exist in the server configuration or has an invalid address format.

Solution:
Check the server item and verify that it is a valid static address.

Failed to register trigger item for log group <log group>. (Trigger: <trigger>)

Error Type:
Warning / Runtime

Possible Cause:

1. A start or stop item of a defined trigger expression does not exist in the server configuration or has an invalid address format.

2. The defined log on data change monitor item does not exist in the server configuration or has an invalid address format.

Solution:

1. Use the **Triggers tab** to verify the start and stop items for each trigger configuration.

2. Verify that each trigger configuration's log on data change monitor item address exists in the server configuration.

Failure reading from store and forward file <filename>: <reason>.

Error Type:
Warning

Possible Cause:
DataLogger cannot read from the store file. If located remotely, access to the file may have been interrupted.

Solution:
Verify that the remote file's location is accessible.

Failure writing to store and forward file <filename>: <reason>.

Error Type:
Warning

Possible Cause:
DataLogger cannot write to the store file. This may be due to one of the following reasons:

1. The hard disk does not have enough space remaining for data to be written to the file.

2. If located remotely, access to the file may have been interrupted.

Solution:

1. Increase the available space or verify that the hard disk has adequate space remaining.

2. Verify that the remote file's location is accessible.

Invalid store and forward file <filename>.

Error Type:
Warning

Possible Cause:
The "<log group name>.bin" file located in the configured Store and Forward directory is not a valid store file.

Solution:
Move or delete the existing file.

Log group using MySQL DSN and the 'wide' table format are limited to <count> items. The DataLogger runtime rejects logging activity for this group until the item count is reduced.

Error Type:
Warning / Runtime

Possible Cause:
Logging activity is rejected until this count is reduced – The MySQL ODBC driver.

Solution:
Use a different DBMS or use multiple log groups (logging to multiple tables) to 'break up' the total number of items to be logged.

See Also:
Narrow vs. Wide Table Format

Log group <log group> failed to validate table <table name>. Reason: <reason>.

Error Type:
Serious / Runtime

DataLogger rejected the table format because it does not match the configured data map. There are two possible causes:

Possible Cause #1:
On the Data Map tab for this log group, "Log to an existing table" was selected as the Table Selection option. The table was identified in the Table Name field. This error message indicates that the table does not exist.

Solution #1:
Create the table using the database management system.

Possible Cause #2:
On the Data Map tab for this log group, "Create a new table once and always attempt to append" was selected as the Table Selection option, but the table's columns do not match the configured data mapping.

Solution #2:

1. Delete the existing table, then let DataLogger create a new table once and append to it thereafter.

2. On the Data Map tab, click **Map Fields** and delete the server items for which there are no corresponding database columns.

See Also:
Map Item Fields
Data Map Tab
Defining the Database Table

Log group <log group> has requested creation of at least one TIMESTAMP column for table <table>. This has been adjusted to the MS-SQL required DATETIME.

Error Type:
Warning / Runtime

Possible Cause:
The DSN type is Microsoft SQL, which does not support this use of the TIMESTAMP type.

Solution:
DataLogger runtime converts TIMESTAMP columns to DATETIME columns when creating tables. Map only DATETIME columns if using Microsoft SQL Server.

Saving store and forward file <old filename> as <new filename>.

Error Type:
Informational

Possible Cause:

The "<log group name>.bin" file that is located in the configured storage directory is either an invalid store file or was created with a different log item or table format configuration.

Solution:

Accept the new filename or remove the existing file and try again.

Store and forward file <filename> has reached maximum size of <file size> MB.

Error Type:

Warning

Possible Cause:

DataLogger cannot add more data to the store file because doing so would exceed the maximum file size.

Solution:

1. Disable the log group and then move or rename the store file. Once complete, re-enable the log group.

2. Increase the value of the Maximum Storage Size property and then apply the changes.

Store and forward file <filename> is incompatible with current log item list.

Error Type:

Warning

Possible Cause:

The log group's configuration does not match the configuration under which the store file was populated. Either the log item list or the table format has changed.

Solution:

The old file is archived in an "_archive" sub-directory of the configured storage directory. To use the file, rename it to "<log group name>.bin" and then move it into the configured storage directory. Once complete, revert the project to the DataLogger configuration with which the file was created.

Table name <table> exists for log group <log group> on DSN <DSN>, which is in "create table once and append" mode.

Error Type:

Warning / Runtime

Possible Cause:

The table name specified already exists for this log group on this DSN.

Solution:

Write to a new or different table, rename the existing table, or append to the table.

The log items in log group <log group> need to be mapped to the columns in table <table name> before logging occurs. Please set up this mapping using the Data Map tab.

Error Type:
Warning / Runtime

Possible Cause:
DataLogger detected an empty data map.

Solution:
Check the log group's Data Map tab settings and (either select one of the automatic table-creation options or map the server items).

See Also:
Data Map Tab
Map Item Fields

The query failed because the data source is not appendable. Please double check the user permissions.

Error Type:
Serious / Runtime

Possible Cause:
The database table is not appendable.

Solution:
Often the user must explicitly set an 'append' permission on the table using the database software.

There are no log items to log for log group <log group>.

Error Type:
Warning / Runtime

Possible Cause:
DataLogger has detected a log group with no log items.

Solution:
Open the log group and verify there are items selected to log.
Use the **Map Item Fields** to add items to the log group.

Unable to connect to data source <source>.

Error Type:
Serious / Runtime

Possible Cause:
DataLogger is unable to connect to the data source and is unable to determine the cause of the connection failure.

Solution:
Verify the DSN settings and consult the ODBC driver documentation.

See Also:
Setting up a DSN

Unable to connect to data source <source>. (Memory Exception)

Error Type:
Serious / Runtime

Possible Cause:
The ODBC driver was unable to make the connection because of a memory exception.

Solution:
Use a different DSN with a different ODBC source.

See Also:
Setting up a DSN

Unable to connect to data source <source>. (Reason: <reason>).

Error Type:
Serious / Runtime

Possible Cause:
DataLogger was unable to connect to the data source because of the reason provided by the ODBC driver.

Solution:
If the database is on a remote computer, verify the user permissions are correct and that the network connection is active.

Unable to open recordset on log group <log group>. (Memory Exception)

Error Type:
Serious / Runtime

Possible Cause:
DataLogger was unable to create the mechanism that would allow it to perform an insertion because of a memory exception.

Solution:
If the database is on a remote computer, verify the user permissions are correct and that the network connection is active.

Unable to open recordset on log group <log group>. (Reason: <reason>).

Error Type:

Serious / Runtime

Possible Cause:
DataLogger was unable to create the mechanism that would allow it to perform an insertion because of the reason provided by the ODBC driver.

Solution:
If the database is on a remote computer, verify the user permissions are correct and that the network connection is active.

Unable to query recordset on log group <log group>. (Memory Exception)

Error Type:
Serious / Runtime

Possible Cause:
DataLogger was unable to perform the insertion because of a memory exception.

Solution:
Use a different DSN with a different ODBC source.

See Also:
Setting up a DSN

Unable to query recordset on log group <log group>. (Reason: <reason>).

Error Type:
Serious / Runtime

Possible Cause:
DataLogger is unable to perform an insertion because of the reason provided by the ODBC driver.
The data includes an array that is too large or improperly defined.

Solution:
Use a different DSN with a different ODBC source.
Address the data type and character limit issues *(see Large Arrays)*.

See Also:
Setting up a DSN

Unable to validate table <table>. (Memory Exception)

Error Type:
Serious / Runtime

Possible Cause:
DataLogger was unable to validate the table format because the ODBC driver encountered a memory exception while opening the table.

Solution:

If the database is on a remote computer, verify the user permissions are correct and that the network connection is active.

Unable to validate table <table>. (Reason: <reason>).

Error Type:
Serious / Runtime

Possible Cause:
DataLogger was unable to validate the table format because the ODBC driver failed to open the table (and provided an error string).

Solution:
If the database is on a remote computer, verify the user permissions are correct and that the network connection is active.

Unknown error while executing query in table <table> for log group <log group>.

Error Type:
Serious / Runtime

Possible Cause:
DataLogger encountered an unknown error while executing an insertion on the opened recordset.

Solution:
If the database is on a remote computer, verify the user permissions are correct and that the network connection is active.

Condition Examples

For examples of a specific condition, select a link from the list below.

Start Condition Examples
Stop Condition Examples
Start / Stop Combined Examples

Start Condition Examples

Example One

If Condition is set to "Item value is equal to OPC TRUE (-1)," the condition is true if the server item in the Item ID field becomes true. For this condition, leave the Data field blank because it is not necessary.

Note: This assumes that the value in Data Type is Boolean. If any other data type is entered, the server item must be -1 in order to make the condition true.

Start Condition	Stop Condition	Log Group Trigger State
False	False	False
True	False	True

Example Two

If Condition is set to "Item data set to bad quality," the condition is true if the server item in the Item ID field becomes Quality=Bad. For this condition, leave the Data field blank because it is not necessary.

Start Condition	Stop Condition	Log Group Trigger State
False	False	False
True	False	True

Example Three

If Condition is set to "Item data set not equal to a specific value" and the Data field value is 144, then the condition is true if the server item in the Item ID field is not equal to 144. The value may be higher or lower.

Start Condition	Stop Condition	Log Group Trigger State
False	False	False
True	False	True

Example Four

If Condition is set to "Item data set less than a specific value" and the Data field value is 144, then the condition is true if the server item in the Item ID field is less than 144.

Start Condition	Stop Condition	Log Group Trigger State
False	False	False
True	False	True

Example Five

If Condition is set to "Item data has not changed over a certain time period (ms)" and the Data field value is 15000 (15,000 milliseconds), then the condition is true if the tag in the Item ID field has not changed value for 15 seconds.

Start Condition	Stop Condition	Log Group Trigger State
False	False	False
True	False	True

Stop Condition Examples

Example One

If Condition is set to "Item data set less than a specific value" and the Data field value is 144, then the condition is true if the server item in the Item ID field is less than 144.

Start Condition	Stop Condition	Log Group Trigger State
True	True	True
True	False	False

Example Two

If Condition is set to "Item data has not changed over a certain time period (ms)" and the Data field value is 15000 (15,000 milliseconds), the condition is true if the tag in the Item ID field has not changed value for 15 seconds.

Start Condition	Stop Condition	Log Group Trigger State
True	True	True
True	False	False

Example Three

If Condition is set to "Item data set to bad quality," the condition is true if the server item in the Item ID field is Quality=Bad. For this condition, leave the Data field blank because it is not necessary.

Start Condition	Stop Condition	Log Group Trigger State
True	True	True
True	False	False

Start / Stop Combined Examples

Example One

The Start Condition is set to log on "Item value is equal to OPC TRUE (-1)" and there is no Stop Condition. The tag for the Start Condition toggles between True and False.

Trigger Tag Value	Start Condition	Stop Condition	Log Group Trigger State
0	False	False	False
-1	True	False	True
0	False	False	True
-1	True	False	True

Example Two

The Start Condition is set to log on "Item value is equal to OPC TRUE (-1)" and the Stop Condition is set to "Item value is equal to OPC FALSE (0)". The Start and Stop triggers use the same tag.

Trigger Tag Value	Start Condition	Stop Condition	Log Group Trigger State
0	False	False	False
-1	True	False	True
0	False	True	False
-1	True	False	True

Example Three

The Start Condition is set to log on "Item value is equal to OPC TRUE (-1)" using Tag A and the Stop Condition is set to "Item value is equal to OPC TRUE (-1)" using Tag B. Tag A rarely changes but Tag B toggles between True and False.

Trigger Tag Value (Tag A / Tag B)	Start Condition	Stop Condition	Log Group Trigger State
0,0	False	False	False
-1,0	True	False	True
-1,-1	True	True	False
-1,0	True	False	True
-1,-1	True	True	False
-1,0	True	False	True
-1,-1	True	True	False
0,-1	False	True	False
0,0	False	False	False

Note:
Arrays cannot be used with the Stop and Start trigger items when a value comparison is performed.

Index

A

C

D

E

F

G

I

L

M

N

O

S