ODBC Client Driver Help

Table of Contents

ODBC Client Driver Help

Help version 1.043

CONTENTS

Overview

The ODBC Client Driver provides a reliable way to connect ODBC data sources to OPC Client applications, including HMI, SCADA, Historian, MES, ERP, and countless custom applications. It provides single-point access to multiple ODBC data sources via ODBC. Through the power and speed of OPC, the server can offer ODBC data to any OPC/DDE enabled client.

ODBC Client Driver Features

- Multiple connection sessions to multiple ODBC data sources
- Read/Write to the SQL engine
- Automatic tag generation based on tables, queries, and stored procedures from the ODBC data source
- OPC DA
- Drag and drop editing
- CSV import and export for easy manipulation of tags
- Data Scaling
- System Tags for general error feedback to client applications
- Tags that can be added to the application at any time (even at Runtime)
- The ability to run as a System Service
- A built-in User Manager for complete control over individual user functionality and privileges

External Dependencies

This driver has external dependencies. It requires Microsoft Data Access Components (MDAC), which consist of several core components that provide various database technologies (including ODBC and its drivers).

Driver Setup

Supported ODBC Data Sources

ODBC-compliant data sources are supported. Examples include:

- Microsoft Access
- Microsoft SQL
- MySQL*
- Oracle

*If the project uses a MySQL data source, users must select **MySQL** from the Model drop-down list when creating a device. This will make the driver accommodate MySQL's data handling requirements. The PC that is running the OPC server must have the MyODBC driver installed, as well.

Communication Protocol

ODBC API

Maximum Channels and Devices

The maximum number of channels supported by this driver is 256. The maximum number of devices supported per channel is 4096.

Note: In this driver, the channel corresponds to the ODBC Driver and its particular ODBC data source.

Data Source Settings

The Data Source Settings tab is used to configure various data source settings.
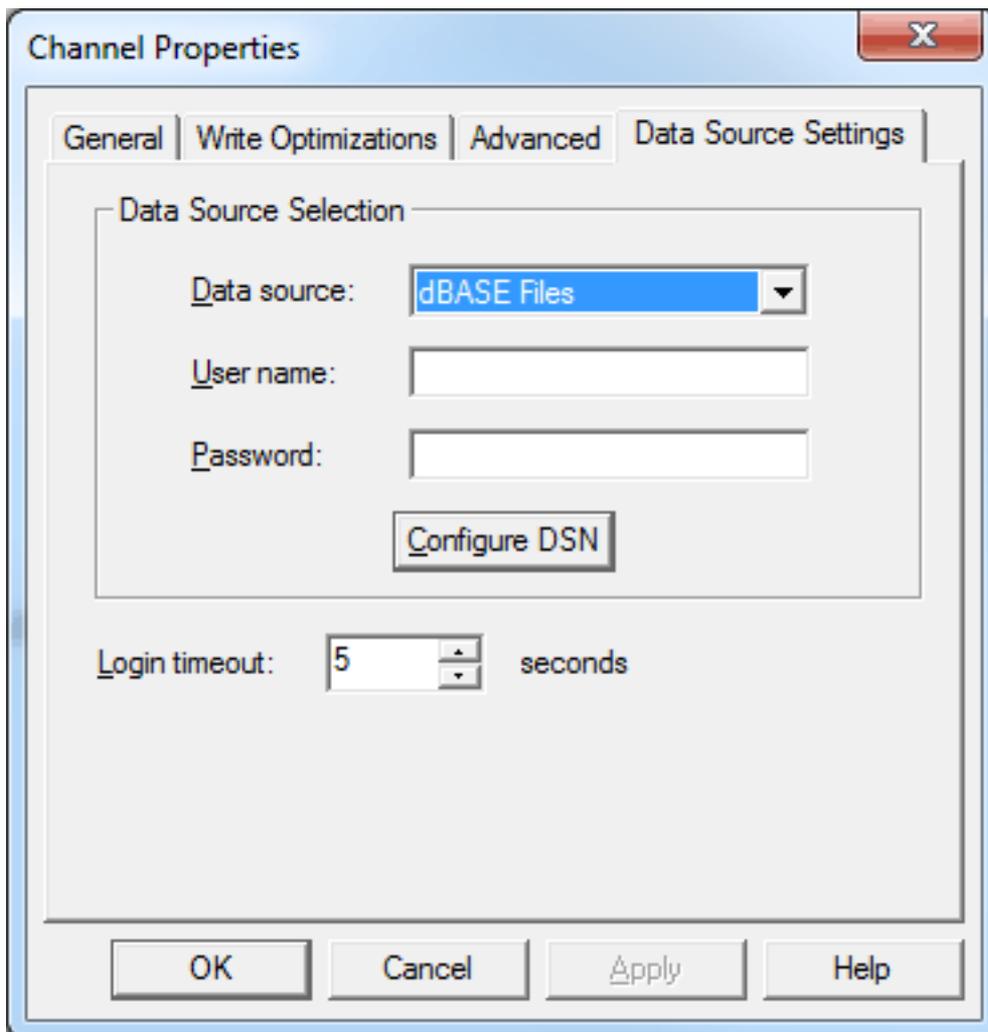
Descriptions of the parameters are as follows:

- **Data Source:** This parameter specifies the Data Source Name (DSN). A DSN must be created and configured for the data source before it will appear in the menu.
- **User Name:** This parameter specifies the user name. It is only needed if required by the data source (such as in SQL server authentication).
- **Password:** This parameter specifies the password. It is only needed if required by the data source (such as in SQL server authentication).
- **Configure DSN:** When clicked, this button will prompt the user to configure the DSN for access to a particular ODBC database.
- **Login Timeout:** This parameter specifies the number of seconds that the server will wait for a response before timing out when attempting to connect to the ODBC data source. The valid range is 1 to 99999 seconds. The default setting is 5 seconds.

Note: Some data sources can use Windows NT Authentication to gain access to the data source. If the data source is configured to use Windows NT Authentication, it will ignore the User Name and Password and will use the Network Login ID instead. For more information, refer to **Data Source Authentication**.

Configuring a DSN

The example below uses System DSNs, which are local to the computer but not user-dedicated. This means that they can be accessed by any user with privileges. User DSNs, which are also local to the computer, are only accessible by the current user. System DSNs must be used when the OPC server is configured to run as a System Service; otherwise, tags will neither be automatically generated nor be Good quality in projects in which they are already contained.

1. To start, click **Configure DSN** to invoke the **Microsoft® ODBC Data Source Administrator**.

2. Then, map an ODBC data source to a data source name.

   Note: Once a DSN has been created successfully, it will appear in the Data Source drop-down menu. For more information, refer to **Data Source Setup**.

Data Source Setup

For information on using the ODBC Client Driver to configure a data source in the OPC server, refer to the instructions below.

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 will launch 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: If using MySQL, the ODBC Driver must be installed on the PC that is running the OPC server.

1. To start, open the OPC server and create a new channel.

2. Next, right-click on the channel name and then select **Properties** | **Data Source Settings**.

3. Then, click **Configure DSN**.
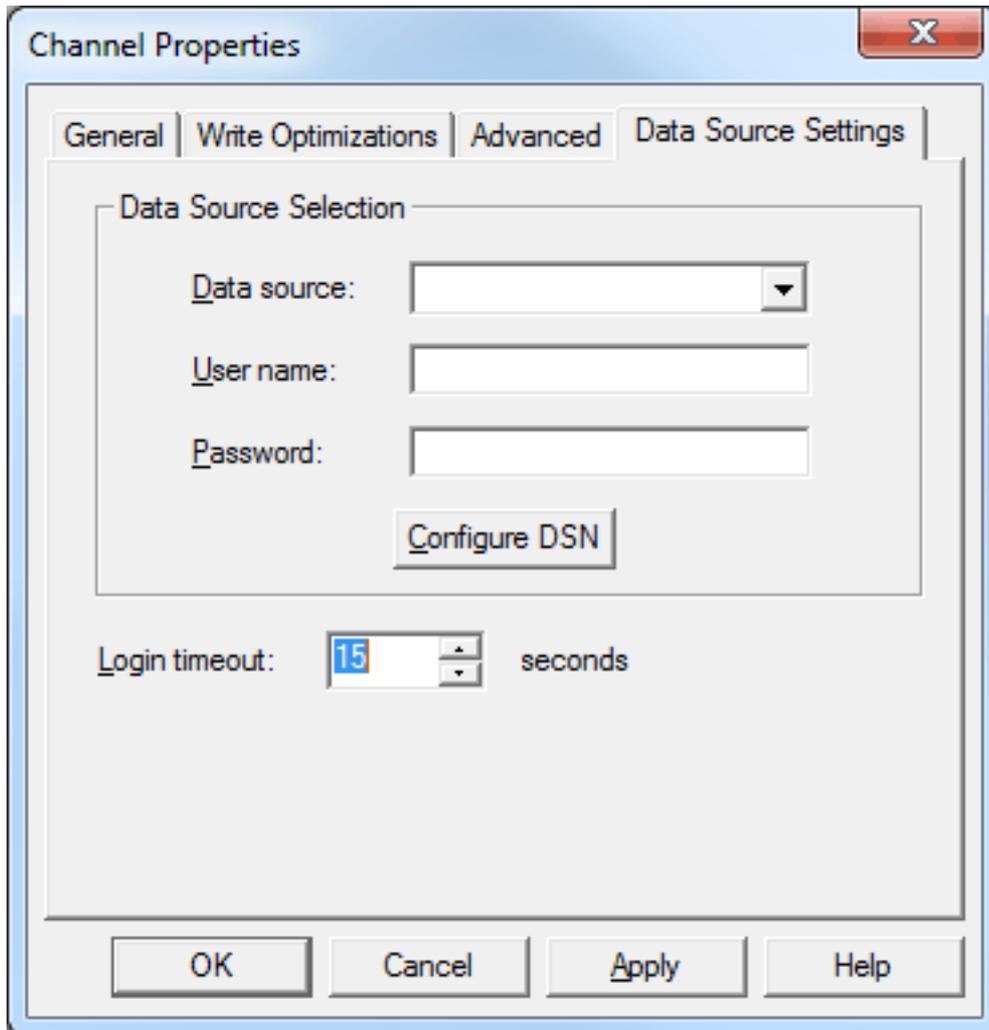
Note: This example uses System DSNs, which are local to the computer and can be accessed by any user with privileges. User DSNs, which are local to the computer but only accessible by the current user, can also be chosen.

4. Next, click **Add**.
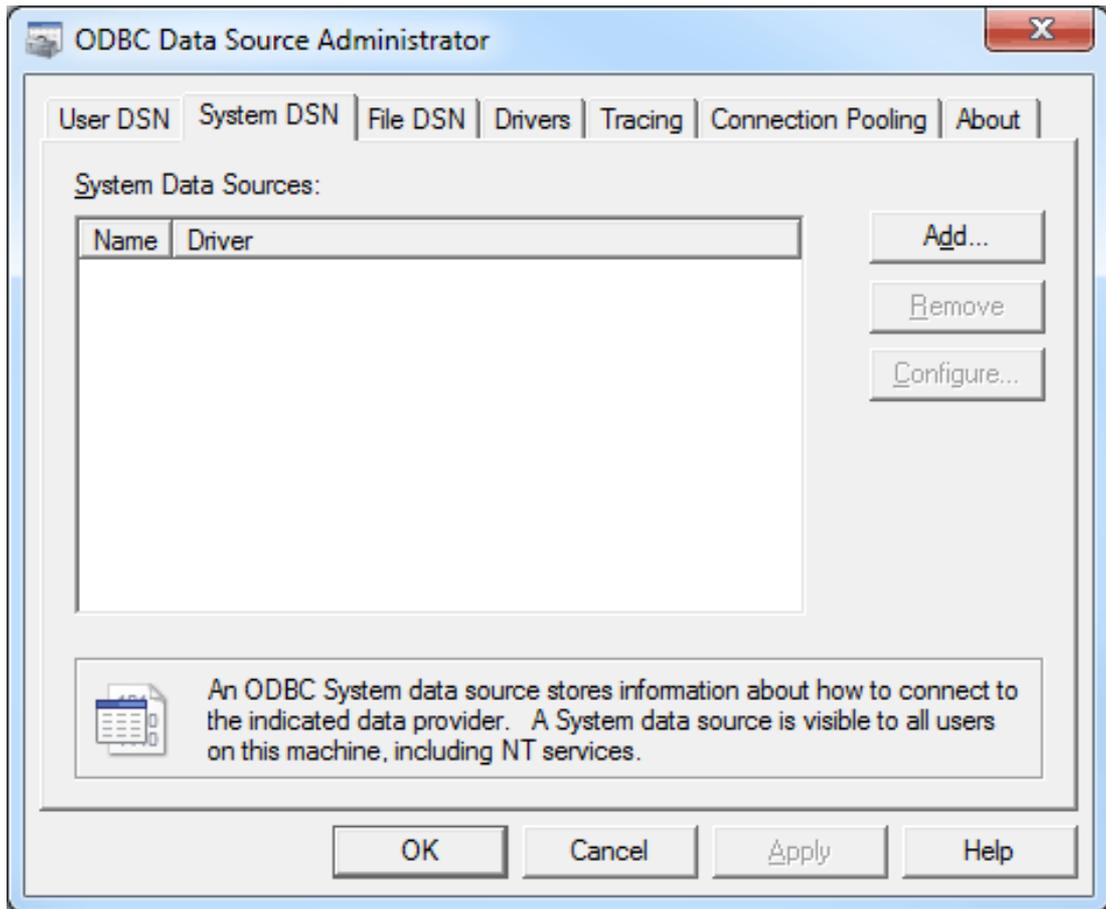
5. Then, select the desired data source. In this example, a SQL Server data source is used.
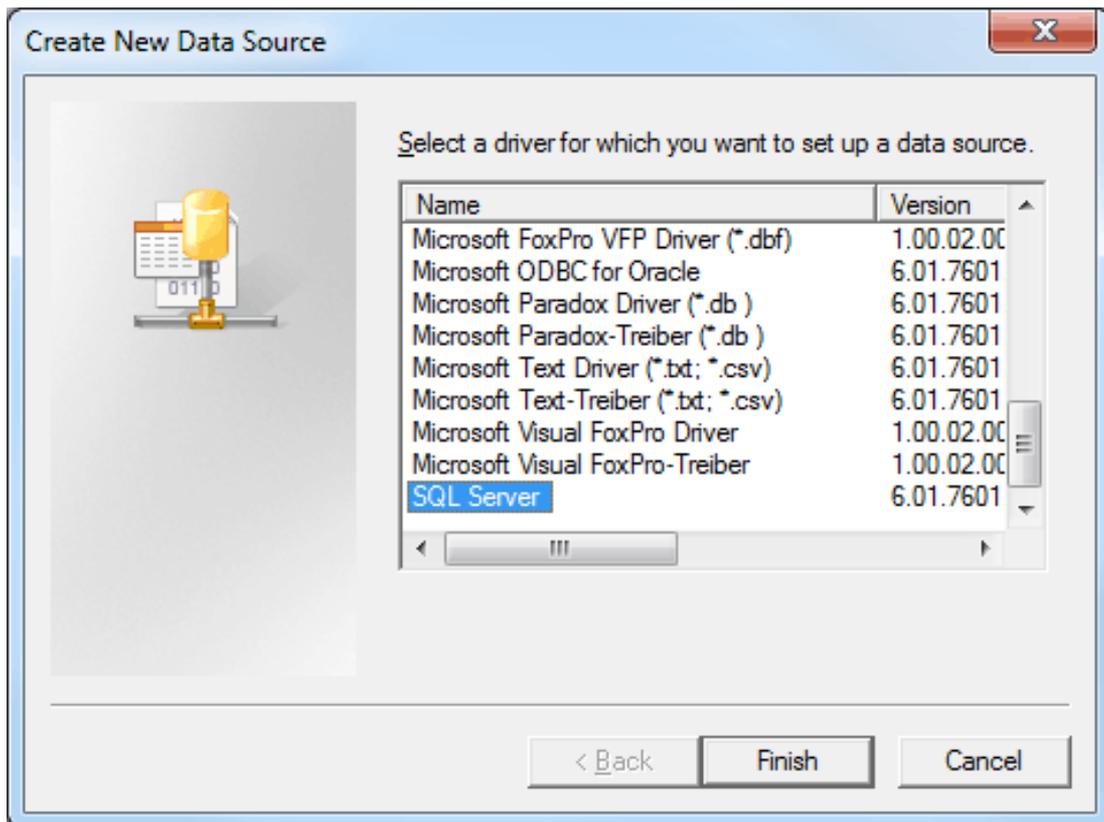
6. Name the ODBC data source, and select the SQL database to which it will connect. Then, click **Next**.
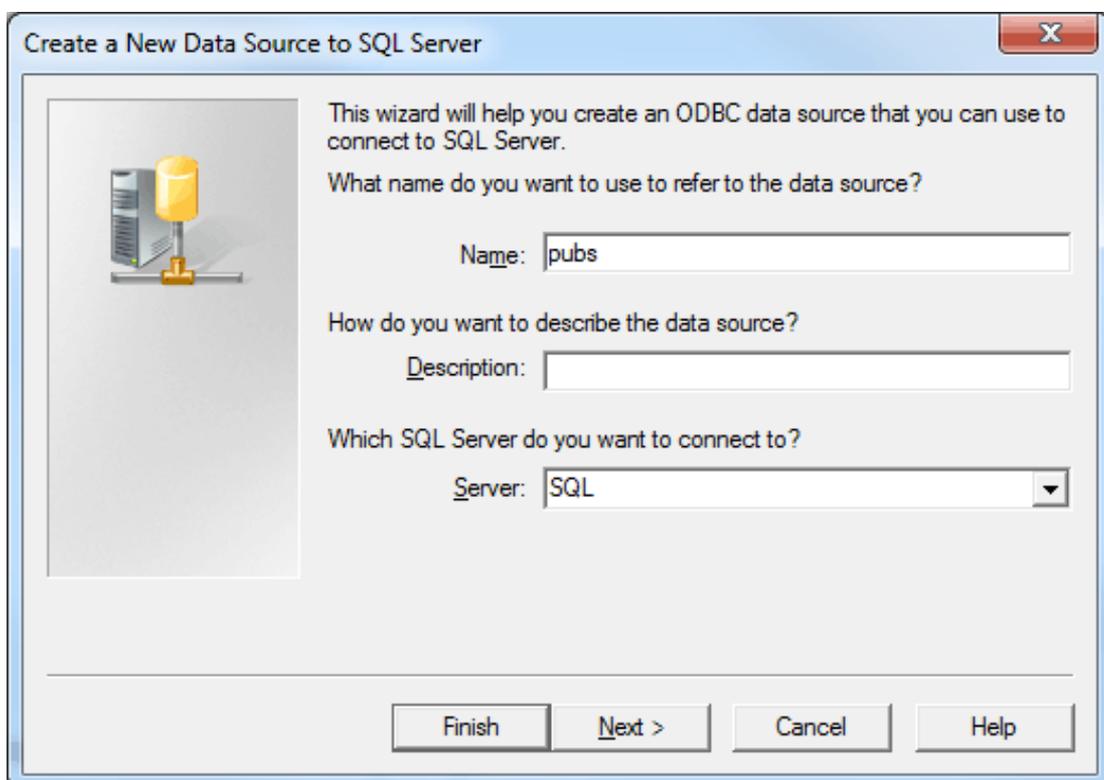
7. In **Microsoft SQL Server DSN Configuration**, the **Windows NT Authentication** option should be selected by default. If it is not, select it and then click **Next**.
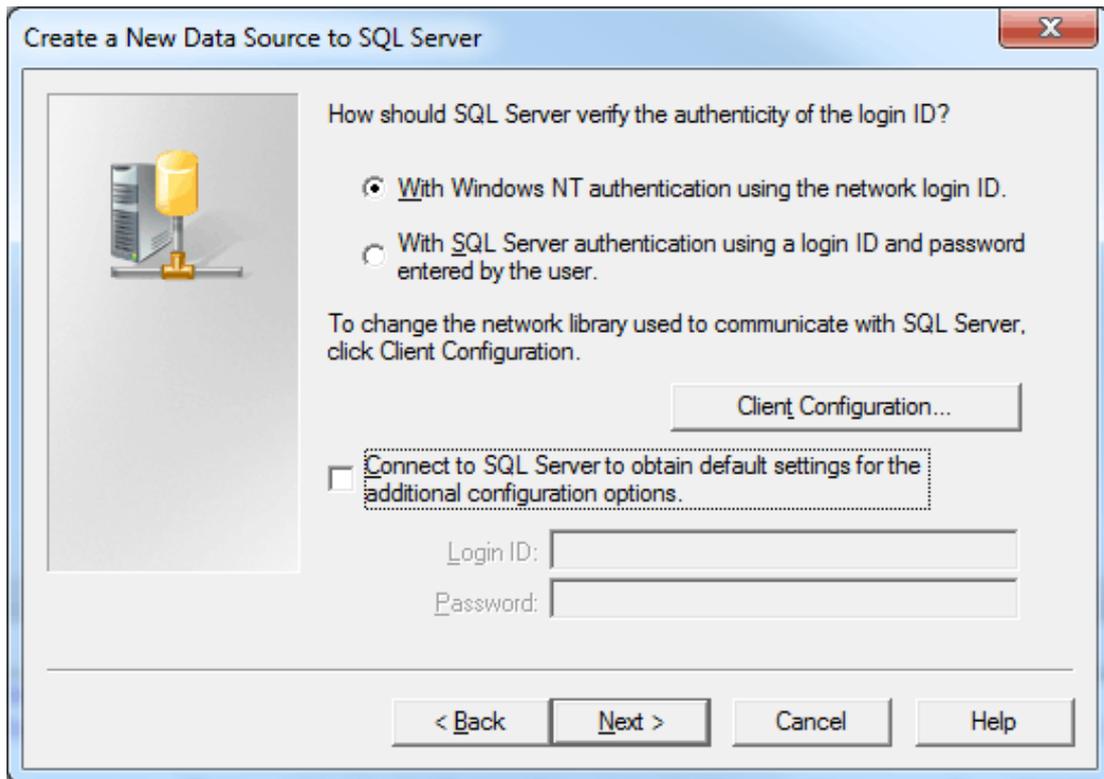
Note: If running the OPC server in System Service mode, refer to **Running as a System Service**.

8. Next, check **Change the default database to** and then select the desired database from the drop-down menu. Select the default database if the desired database is not available.
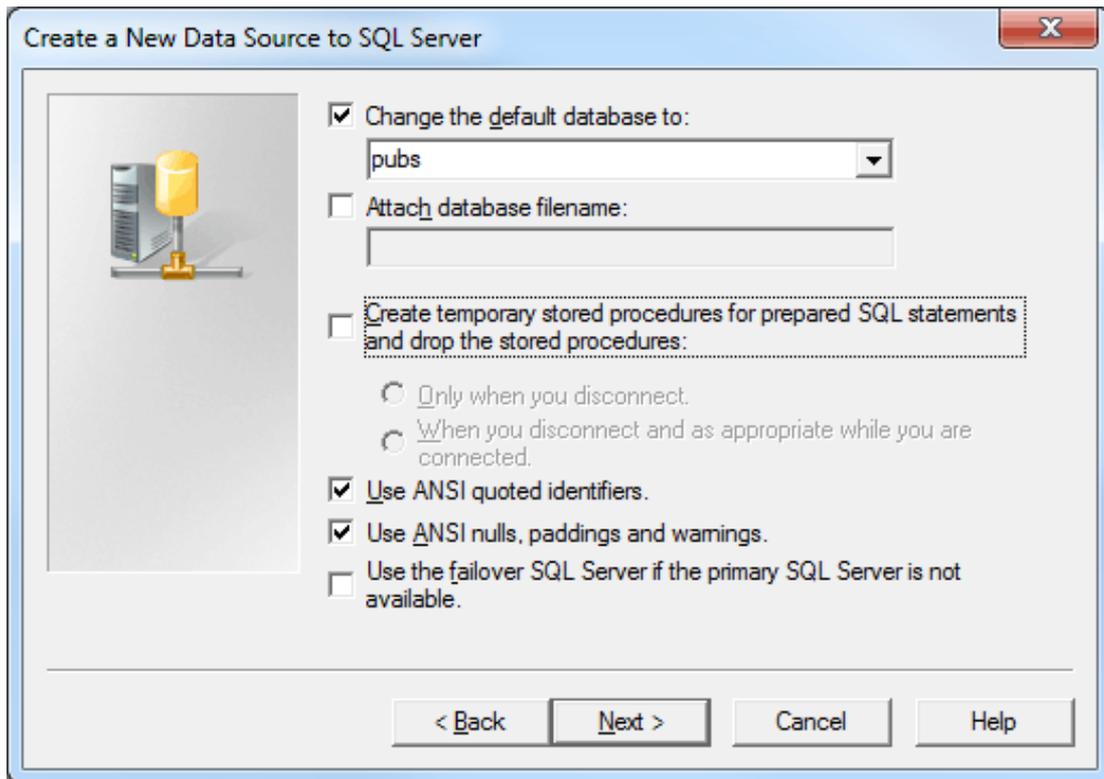
9. Continue through the wizard, keeping the remaining settings at their default values.

10. Upon completion, test the data source connection by clicking **Test Data Source**. If the connection is good, users will receive the message "Test Successful."

ODBC Microsoft SQL Server Setup

A new ODBC data source will be created with the following configuration:

Microsoft SQL Server ODBC Driver Version 06.01.7601

Data Source Name: pubs
Data Source Description:
Server: SQL
Database: (Default)
Language: (Default)
Translate Character Data: Yes
Log Long Running Queries: No
Log Driver Statistics: No
Use Regional Settings: No
Prepared Statements Option: Drop temporary procedures on
disconnect
Use Failover Server: No
Use ANSI Quoted Identifiers: Yes
Use ANSI Null, Paddings and Warnings: Yes
Data Encryption: No

Test Data Source...   OK   Cancel

Data Source Access Methods

Access Method Selection

Once a channel with a defined ODBC data source is created, a device must be created for each access method desired. Options include **Fixed Table**, **Dynamic Table**, **Query**, and **Stored Procedure**. This can only be selected during device creation.
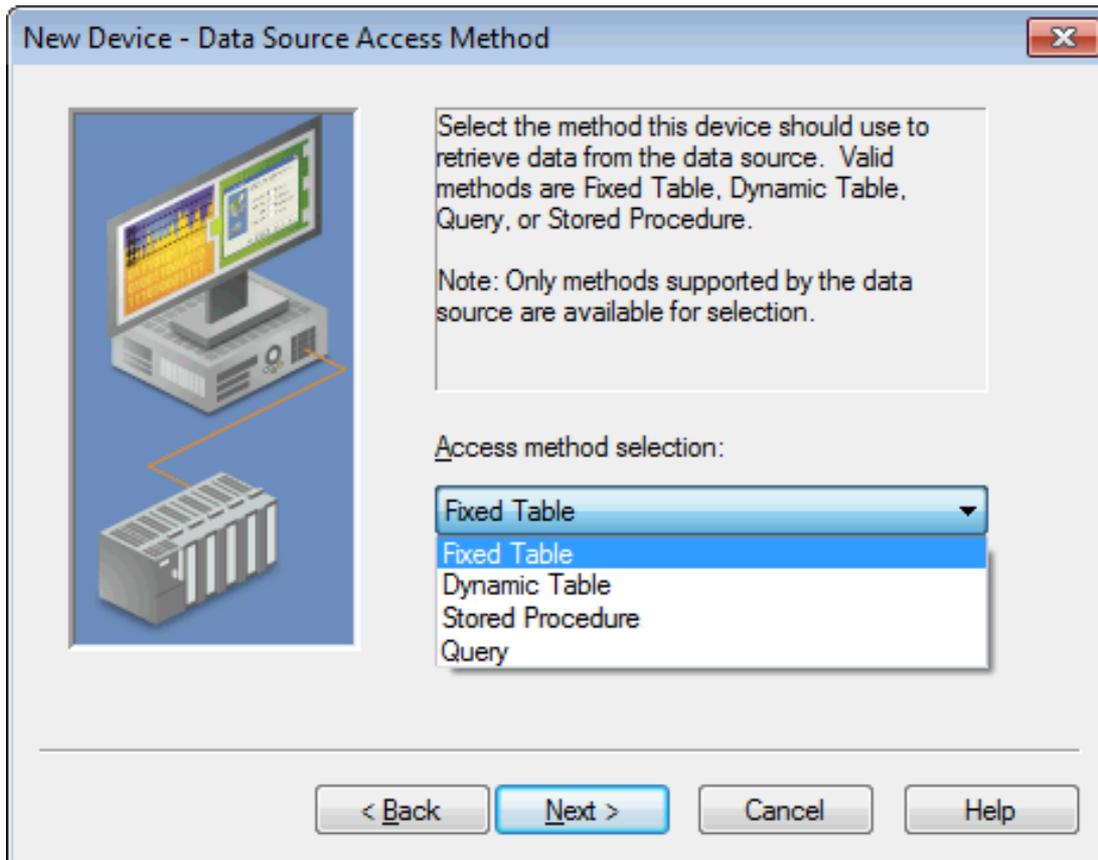
Fixed Table
Dynamic Table
Query
Stored Procedure

Notes:

1. The account configured in the DSN must have appropriate access permissions for these access methods to execute properly.

2. Users may receive a message that states the driver was unable to retrieve a data source component. It is important to verify in the ODBC administrator that the path to the data source is correct for the DSN. *For more information, refer to **Data Source Setup** and **ODBC Client Driver Specific Messages**.*

Fixed Table

Data can be retrieved from an ODBC data source table through the Fixed Table access method which allows access to each item in the ODBC data source table. After creating a device through the specified Fixed Table method, the driver will create tags in the server's tag database that correspond to each item in the ODBC data source table.

The basic flow of defining a fixed table is shown in the diagram below. Click on each step for instructions.

Table Selection

Once an access method has been selected, select a table from which to access data. All table names in the ODBC data source are available for selection. Once a table name is selected, the driver can be configured to automatically generate tags to represent the data in the table.
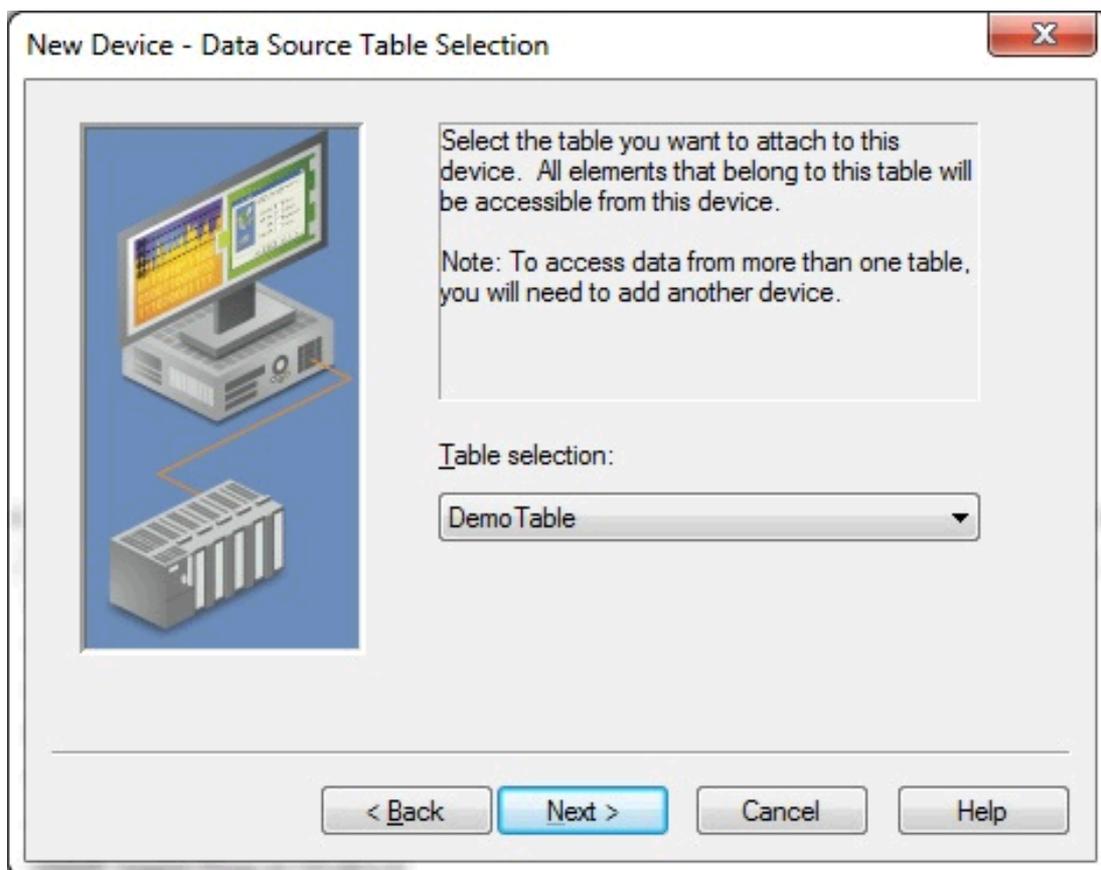
Auto Tag Database Generation Options

The Automatic Tag Generation (ATG) Options can be used to control how tags are automatically generated in the server's tag database. Tags that are automatically generated will differ depending on the type of access method selected.
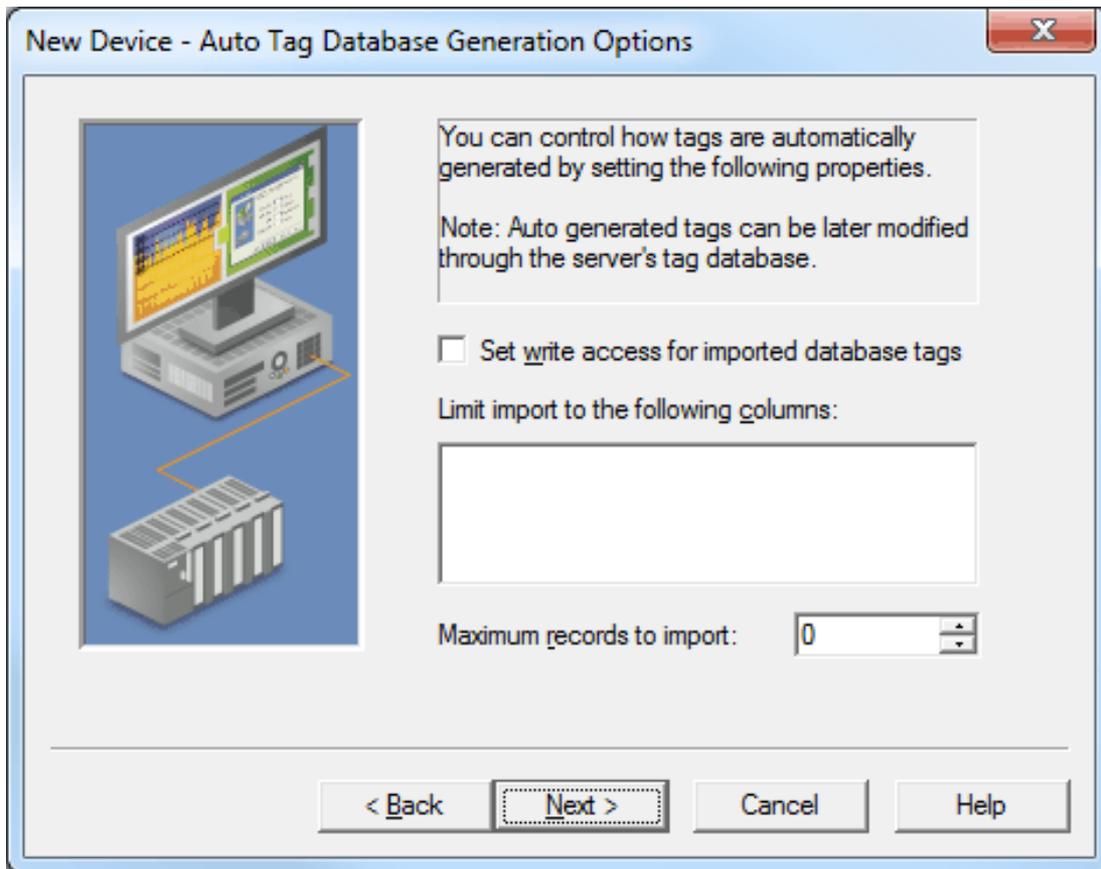
Descriptions of the parameters are as follows:

- **Set Write Access for Imported Database Tags:** This parameter specifies the default Write privileges when automatically generating tags. It only sets the tag's access privileges to Read/Write in the server: it does not affect the field's access privileges in the database.

    Note: Some data sources (such as Excel) do not support Writes.
- **Limit Import to the Following Columns:** This option specifies the data source's table columns that will be imported into the tag database. This option only applies to Fixed Table.
- **Maximum Records to Import:** This parameter specifies the maximum number of records to import. The default setting will automatically generate tags for all records in the table, but users can limit the number is desired. This parameter only applies to Fixed Table. The default setting is 0.

Timing Options

For better performance, the ODBC Client Driver can query the data source for updates at an interval that is independent of tag update rates.

Description of the parameter is as follows:

- **Query Interval:** This parameter specifies the frequency that the driver will query the data source for the latest data values. The valid range is 100 to 99999999 milliseconds. The default setting is 250 milliseconds.

Note: It is recommended that the client update rate for the tags generated for the table be at least **twice as fast as this query interval**.

Dynamic Tables

Data can be retrieved from an ODBC data source table through the Dynamic Table access method, which allows access to a single record set in an ODBC data source table via record tags. The record tags correspond to their respective columns in the data source table, and are dynamically controlled via internal control tags (which are accessible from a client application). This allows users to work with a particular record set in the table. The table below displays a list of the internal control tags that can be used to access the data.

Dynamic Table Control Tags	Access	Description
Move	Read/Write	Move a specified number of records forward (positive value) or backward (negative value) in a table.
MovePrev	Read/Write	Move to the previous record (on write complete).
MoveNext	Read/Write	Move to the next record (on write complete).
MoveLast	Read/Write	Move to the last record in the table (on write complete).
MoveFirst	Read/Write	Move to the first record in the table (on write complete).
CurrentRecord	Read Only	The index of the record that is currently being accessed.
RecordCount	Read Only	Number of records are in the table.

The basic flow of defining a dynamic table is shown in the diagram below. Click on each step for instructions.

Access Method → Table Selection → Tag Database → Timing Options

Table Selection

Once an access method has been selected, select a table from which to access data. All table names in the ODBC data source are available for selection. Once a table name is selected, the driver can be configured to automatically generate tags to represent the data in the table.

New Device - Data Source Table Selection

Select the table you want to attach to this device. All elements that belong to this table will be accessible from this device.

Note: To access data from more than one table, you will need to add another device.

Table selection:

DemoTable

< Back   Next >   Cancel   Help

Auto Tag Database Generation Options

The Automatic Tag Generation (ATG) Options can be used to control how tags are automatically generated in the server's tag database.


Descriptions of the parameters are as follows:

Set Write Access for Imported Database Tags: This parameter specifies the default Write privileges when automatically generating tags. It only sets the tag's access privileges to Read/Write in the server: it does not affect the field's access privileges in the database.

Notes:

1. For queries, this parameter only applies to SELECT statements. Tags generated from EXECUTE statements are always read only.

2. Some data sources (such as Excel) do not support Writes.

Timing Options

For better performance, the ODBC Client Driver can query the data source for updates at an interval that is independent of tag update rates.

Description of the parameter is as follows:

- **Query Interval:** This parameter specifies the frequency that the driver will query the data source for the latest data values. The valid range is 100 to 99999999 milliseconds. The default setting is 250 milliseconds.

Note: It is recommended that the client update rate for the tags generated for the table be at least **twice as fast as this query interval**.

Query

The ODBC Client Driver supports SELECT and EXECUTE queries in which a tag database can be created and defined according to specific needs. Users can retrieve tags from whole tables and specific records. Although the query access method can be used to execute stored procedures that return many result sets, the generated tags are from the first result set only. *For more information, see* **Choosing Query or Stored Procedure Access Method**.

The basic flow of defining a query is shown in the diagram below. Click on each step for instructions.

Access Method → Build Query → Trigger Definition → Tag Database → Timing Options

See Also:
Using the Query Access Method
Data Source Query Format
Modifying a Query

Building a Valid Query Statement
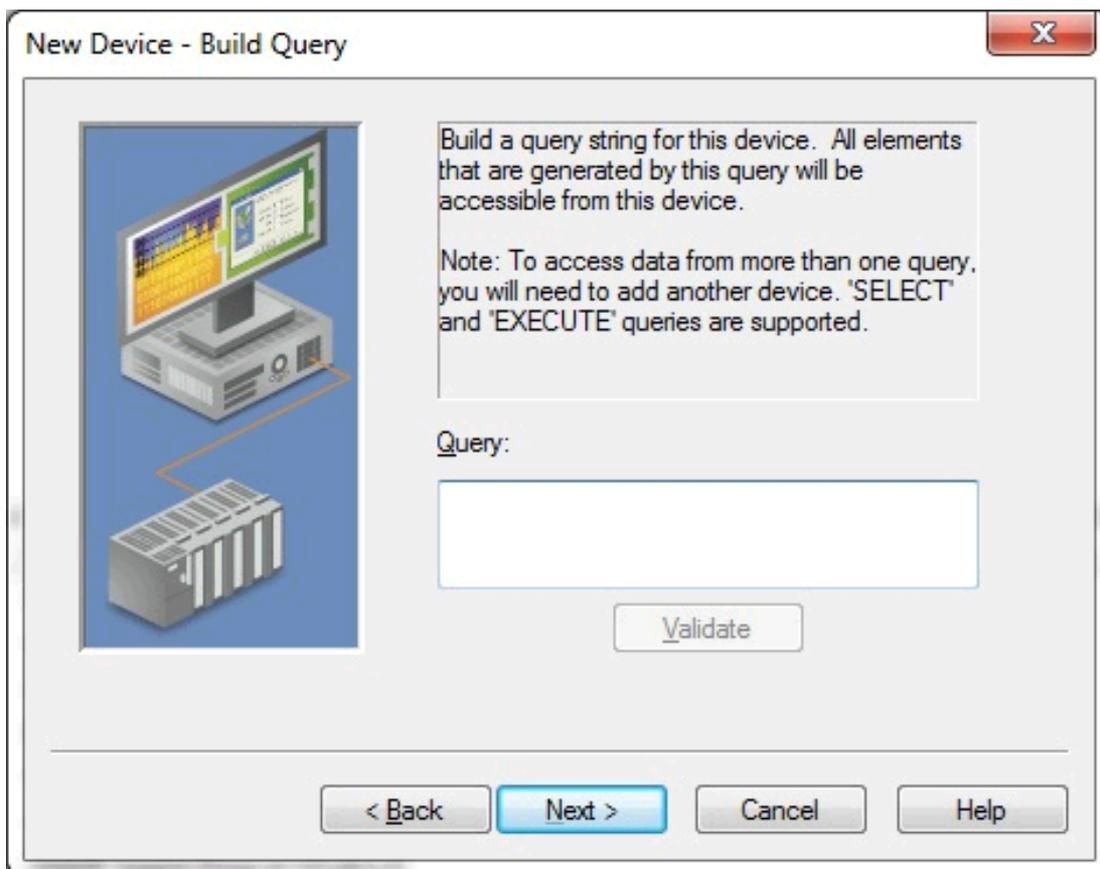
New Device - Build Query

Build a query string for this device. All elements that are generated by this query will be accessible from this device.

Note: To access data from more than one query, you will need to add another device. 'SELECT' and 'EXECUTE' queries are supported.

Query:

Validate

< Back | Next > | Cancel | Help

Descriptions of the parameters are as follows:

Query: This field is used to enter a valid query statement. SELECT and EXECUTE statements are supported.

Note: Supported EXECUTE statements include those with a valid stored procedure and an optional list of unnamed or named input parameters. The database and/or schema should not be used to qualify the stored procedure name. To reference a stored procedure in another database, create another ODBC data source to that database. Then configure a channel for that ODBC data source and a device to directly reference the stored procedure.

Validate: When pressed, this button determines whether the query statement is valid or invalid by executing the query.

Notes:

1. Tags will be generated if all properties are valid after the wizard **Finish** button has been pressed.

2. The **Validate** and **Next** buttons trigger validation of the query string. The validation process includes executing the query statement.

See Also:
Using the Query Access Method
Data Source Query Format
Modifying a Query

Miscellaneous Query Trigger Definition

The Miscellaneous property dialog allows setting a column that should act as the trigger to the client that the query execution completed and the tags have all been updated.
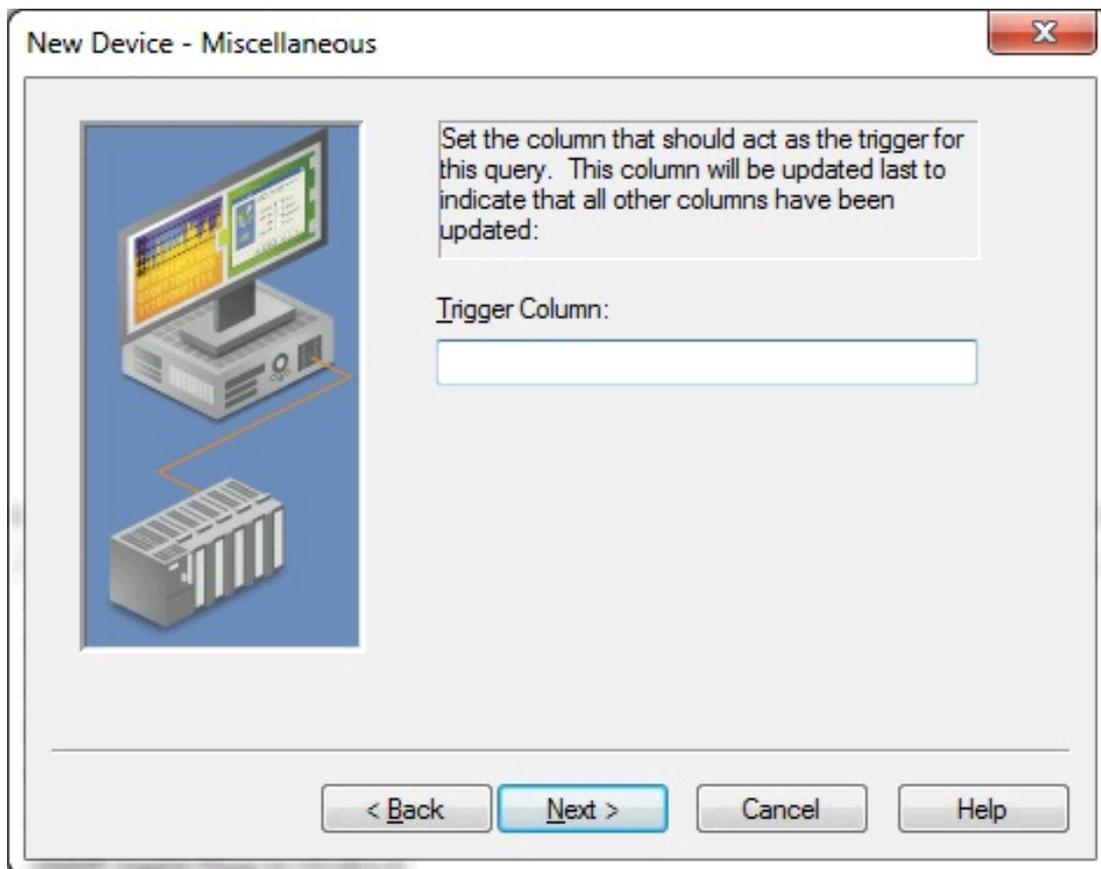
Trigger Column: This parameter specifies the field or column in the database that should be updated last. It is typically used as a trigger to indicate to the connected client that all other fields have been updated and can be processed.

For example, when the Query statement contains "SELECT job_id, job_name, job_desc WHERE job_name = 'manager'" and the Trigger column property is set to "job_id", the tags in the job_id tag group will be updated last.

See Also:
Using the Query Access Method
Data Source Query Format
Modifying a Query

Timing Options

For better performance, the ODBC Client Driver can query the data source for updates at an interval that is independent of tag update rates.
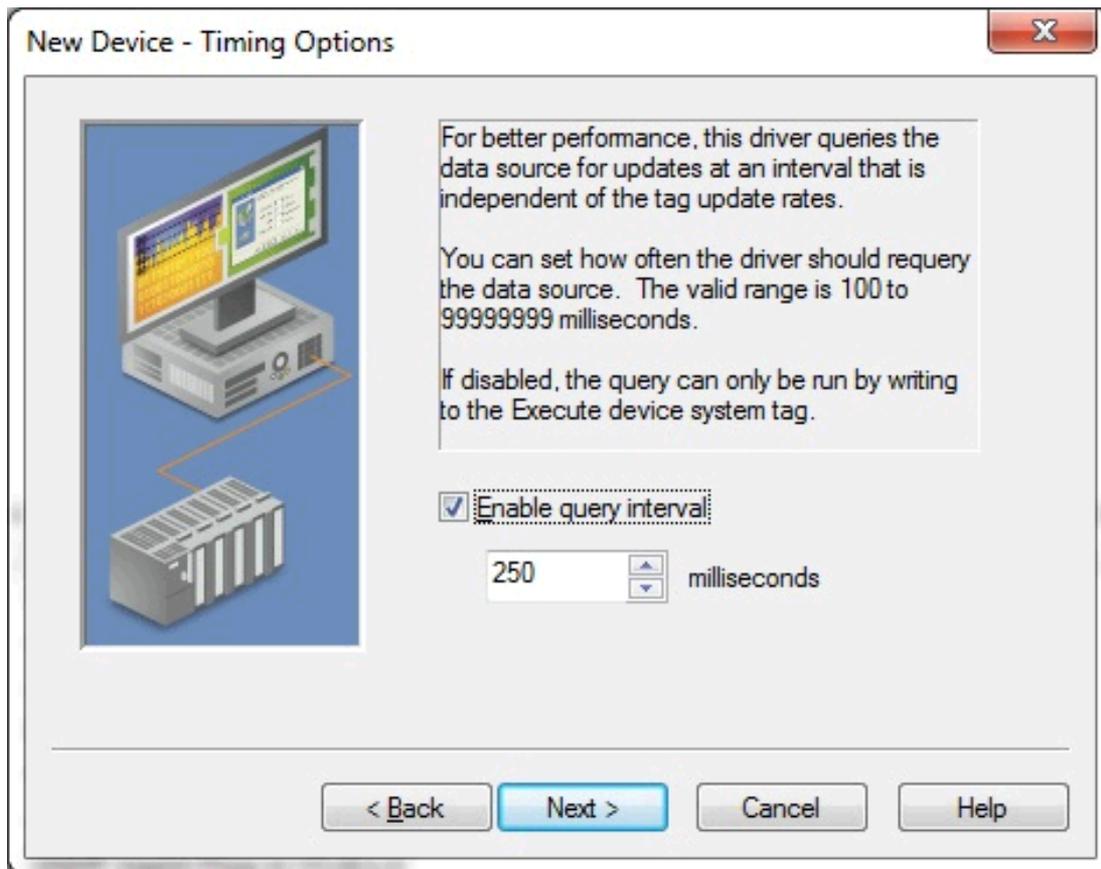
Descriptions of the parameters are as follows:

Enable query interval: When checked, this parameter enables the query interval. When unchecked, the query will not run unless a write to the Execute internal control tag occurs. The default setting is checked.

Query Interval: This parameter is available to configure if the Enable query interval parameter is checked. If available, this parameter specifies the minimum frequency at which the driver will query the data source for the latest data values. It is recommended that the client update rate for tags generated for the results of the query be at least twice as fast as this query interval. The query interval may need to be adjusted to ensure all the tags are requested by the client within that interval. The valid range is 100 to 99999999 milliseconds. The default setting is 250 milliseconds.

Note: When configured with a disabled or long query interval, synchronous requests from a client should be avoided as they will not complete until the query interval elapses or a write to the internal Execute tag occurs.

Auto Tag Database Generation Options

The Automatic Tag Generation (ATG) Options can be used to control how tags are automatically generated in the server's tag database.

Descriptions of the parameters are as follows:

Set Write Access for Imported Database Tags: This parameter specifies the default Write privileges when automatically generating tags. It only sets the tag's access privileges to Read/Write in the server: it does not affect the field's access privileges in the database.

Notes:

1. For queries, this parameter only applies to SELECT statements. Tags generated from EXECUTE statements are always read only.

2. Some data sources (such as Excel) do not support Writes.

Using the Query Access Method

For information on retrieving tags based on a query, refer to the instructions below.

1. In **Query**, type a SELECT or an EXECUTE query. Supported EXECUTE statements are limited to those with a valid stored procedure and an optional list of unnamed or named input parameters. The stored procedure name should not be qualified by a database and/or schema. To reference a stored procedure in another database, create another ODBC data source to that database. Then configure a channel for that ODBC data source and a device to directly reference the stored procedure.

   Note: Some ODBC data sources do not allow queries. *For more information, refer to the data source's help documentation.*

2. Pressing the **Validate**, **Update Tags**, **Apply**, or **OK** button executes the query statement to validate the statement. If no error message is received, the query is valid. An error message is posted to the event log if the query is invalid. If the query is invalid, make changes to the query statement or the data source connection and then re-attempt validation.

3. Pressing the **Update Tags**, **Apply**, or **OK** button executes the query, generates an internal Execute tag, and a tags database based on the results of the query. If a message is posted indicating that the query is

invalid, make changes to the query statement or the data source connection and then re-attempt validation and tag generation.

Important: For tags to generate, the query must result in at least one returned record. A null return will neither generate tags nor post error messages (since it is not considered to be an error).

Note: If a SELECT statement returns at least one record, an internal Execute tag is also generated to allow the client to trigger query execution. The internal Execute tag is always available for an EXECUTE statement even when no other tags result from the query.

Query Tags	Access	Description
Execute	Read/Write	Execute query statement (on write complete).

Data Source Query Format

For information on creating a query to retrieve specific data rows from a table, refer to the instructions below.

1. In **Query**, add a search condition to the query using the following format:

    SELECT [*column1, column2, ...*] FROM [*table*] WHERE [*search condition*];

2. The general format of a search condition is as follows:

    WHERE [*expression*] [*comparison operator*] [*expression*]

    Note: When using specific strings in the expression, single quotes may be needed. For example, "Select job_desc WHERE jobname = 'manager'".

A tag group is created for each column in the result set. A tag for each record in that column is created in the tag group with an address of the column name and record index.
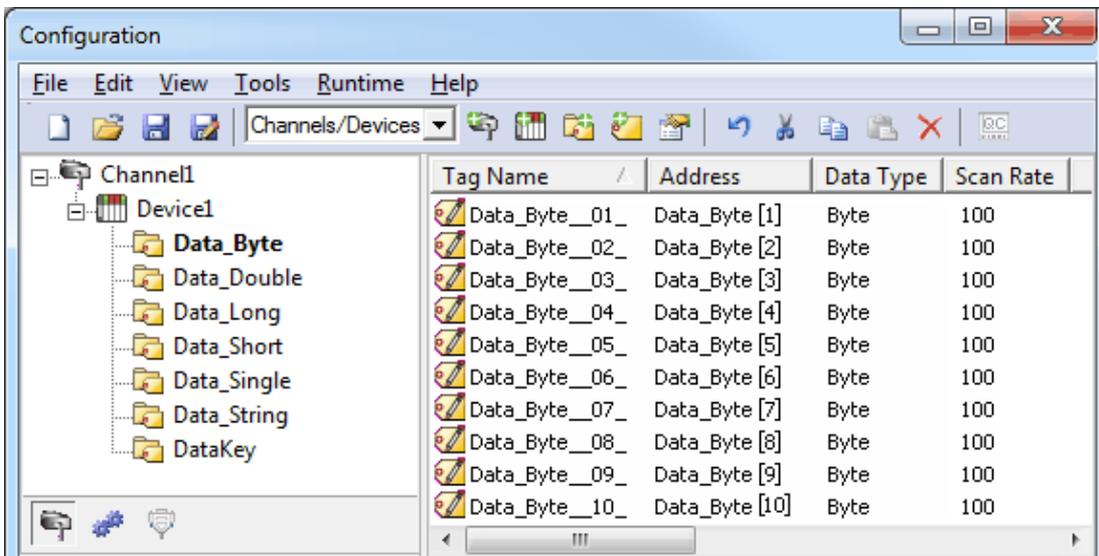
Modifying a Query Statement

Once a device has been defined with a query statement, changes to the statement can be made by opening the Device Properties and selecting the Query Settings tab.

Note: On the Query Settings Device Properties page, the **Validate**, **Update Tags**, **Apply**, and **OK** buttons all cause validation of the query string by executing the query statement.

Descriptions of the parameters are as follows:

Query: This field is used to enter a valid query statement. SELECT and EXECUTE statements are supported.

> **Note**: Supported EXECUTE statements are limited to those with a valid stored procedure and an optional list of unnamed or named input parameters. The stored procedure name should not be qualified by a database and/or schema. To reference a stored procedure in another database, create another ODBC data source to that database. Then configure a channel for that ODBC data source and a device to directly reference the stored procedure.

Validate: When pressed, this button determines whether the query statement is valid or invalid by executing the query.

Update Tags: When pressed, this button applies all changes to the device properties, executes the query, and refreshes the tag database. A tag group is created for each column in the result set. Within each group a tag is created for each row. Additionally, the internal Execute tag is created at the device level to allow the client to execute the query.

Set write access for imported database tags: This parameter specifies the default Write privileges when automatically generating tags. It only sets the tag's access privileges to Read/Write in the server: it does not affect the field's access privileges in the database. This parameter only applies to SELECT statements. Tags generated from EXECUTE statements are always Read-Only.

See Also:
Using the Query Access Method
Building a Query Statement
Data Source Query Format

Stored Procedure

The ODBC Client Driver supports access to stored procedures. *For more information, see **Choosing Query or Stored Procedure Access Method**.*

Guidelines for accessing stored procedures on the ODBC data source are as follows:

- The data source must support stored procedures.
- At least one custom stored procedure must be contained and tested by the data source before users can attempt to create and connect to the tag database via the ODBC Client Driver.
- The current account must have permission to access the stored procedure.
- The driver must have access to the data source via ODBC so that it can access and execute the stored procedure properly.

Note: The ODBC Client Driver uses MyODBC to communicate with MySQL. At the time of release, MyODBC did not support stored procedures; as such, the ODBC Client Driver's support of MySQL does not include stored procedures.

Stored Procedure Selection
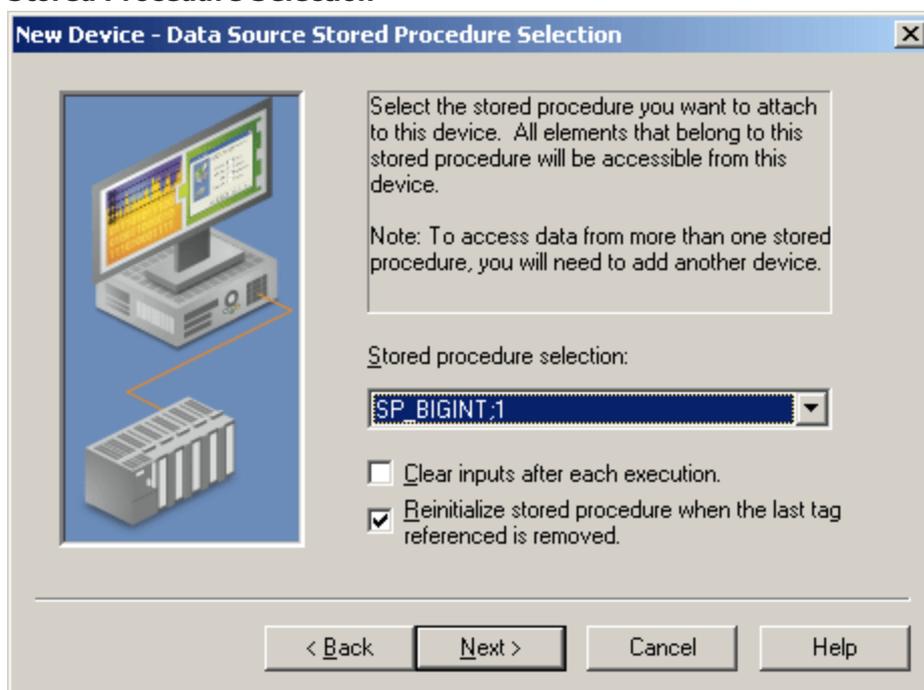
Descriptions of the parameters are as follows:

- **Stored procedure selection:** This parameter specifies the stored procedure that executes when a user writes to the internal Execute tag.
- **Clear inputs after each execution:** When checked, the OPC server clears the input values after the stored procedure is executed, returning them to the default values for the data types. The default setting is unchecked.
- **Reinitialize stored procedure when the last tag referenced is removed:** When checked, the OPC server will reset the input values after the last active client is disconnected from the server. When unchecked, the input parameters will retain their last known values when the client is reconnected. The default setting is checked.

See Also:
Choosing Query or Stored Procedure Access Method
Creating the Tag Database
Selecting a Different Stored Procedure

Creating the Tag Database

Once the stored procedure has been selected, users can continue through and finish the wizard. The tag database of 'input', 'output', and 'return value' tags are created based on the selected stored procedure. An internal Execute tag is created in the tag database to allow the client to control when to execute the procedure. After writing appropriate values to the input tags (if applicable), users can write to the Execute tag to execute the stored procedure.

Stored Procedure Tags	Access	Description
Execute	Read/Write	Execute stored procedure (on write complete).

The example below displays a tag database that was created from a successful stored procedure.
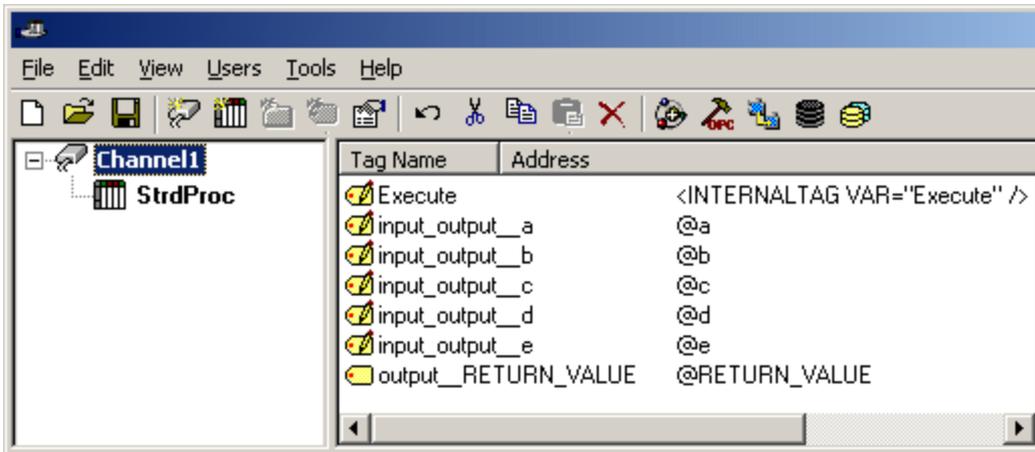
Selecting a Different Stored Procedure

Once a device has been defined with a stored procedure, a new stored procedure may be selected by opening the **Device Properties** page and selecting the **Stored Procedure** tab. The list of available stored procedures is generated based on the configured ODBC data source.

Note: Pressing the **Update Tags**, **Apply**, or **OK** button generates an internal Execute tag and input, output, and return value tags based on the definition of the stored procedure.
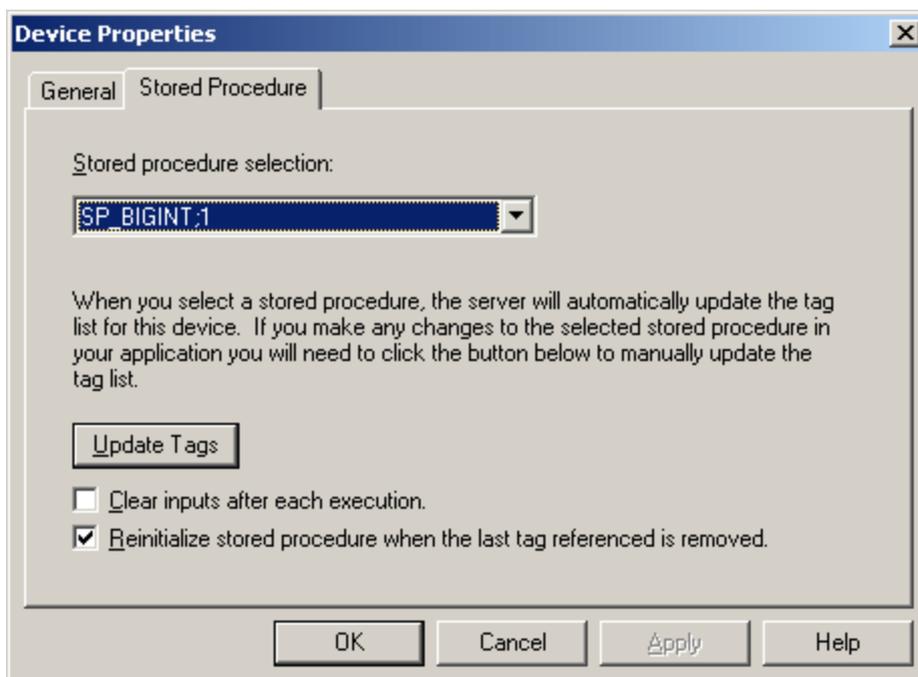
Descriptions of the parameters are as follows:

Stored procedure selection: This parameter specifies the stored procedure that will execute when a user writes to the Execute internal control tag.

Update Tags: When pressed, this button refreshes the tag database with an internal Execute tag and input, output, and return value tags based on the definition of the stored procedure.

Clear inputs after each execution: When checked, the OPC server will clear the input tag values after the stored procedure is executed, thus returning them to the appropriate default value for its data type. The default setting is unchecked.

Reinitialize stored procedure when the last tag referenced is removed: When checked, the OPC server will reset the input values after the last active client is disconnected from the server. When unchecked, the input parameters will retain their last known values when the client is reconnected. The default setting is checked.

See Also:
Choosing Query or Stored Procedure Access Method

Choosing Query or Stored Procedure Access for Stored Procedure Execution

Only the Query Access Method allows directly executing a SELECT statement. However, both the Query Access Method and the Stored Procedure Access Method allow executing stored procedures.

Use the Stored Procedure Access Method to execute a stored procedure to:

- Retrieve a return value and/or output parameters
- Allow the client to vary the input parameters
- Only execute the stored procedure when the client writes to the Execute tag (the Query Access Method executes on validation, Auto Tag Generation, and when the client writes to the Execute tag)

Use the Query Access Method to execute a stored procedure to:

- Retrieve data from the first result set
- Execute with the default parameter values

See Also:
Query Access Method
Stored Procedure Access Method

Running as a System Service

Unlike OPC servers that only support Stand Alone program operation, this server can continue to supply OPC data across user login or logout sessions by running as a System Service. This is crucial for applications in which the server must provide data to OPC clients via DCOM. For these applications, the loss of a DCOM connection cannot be tolerated.

Connecting Remotely in System Service Mode

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

Data Source Authentication

Windows NT Authentication

Some data sources may be configured to use authentication for security. The SQL Server can authenticate with Windows NT authentication by default using the Network Login ID. This makes the SQL server login transparent, and eliminates the need for a user name and password in the channel's Data Source Settings tab.

SQL Server Authentication

The SQL server also has its own authentication method. To connect the OPC server to the SQL server using SQL authentication, users must set the authentication method in the SQL server to allow SQL authentication. Users must also specify the Login ID and Password. After these parameters are defined, the user name and password may then be entered in the OPC server in the channel's Data Source Settings tab. For more information, refer to **Setting up SQL Authentication**.

See Also: **Running as a System Service**
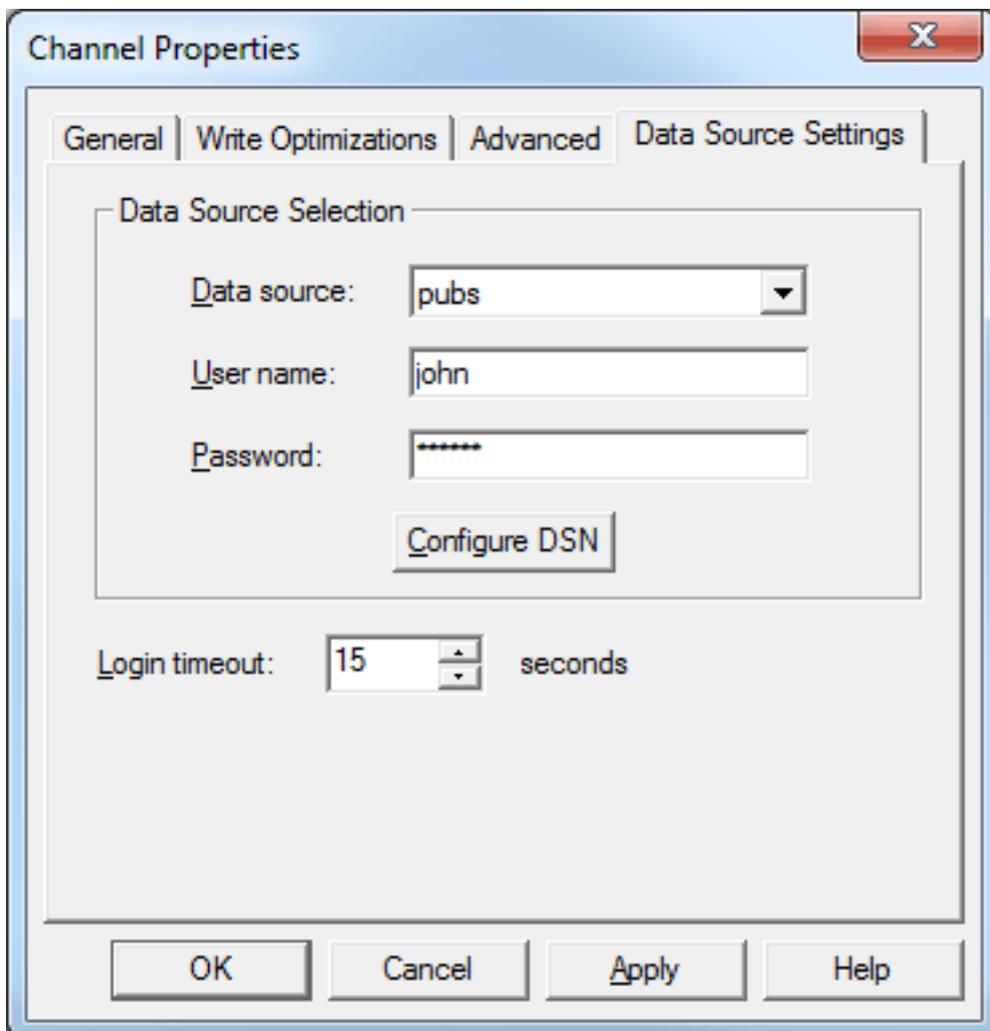
Setting up SQL Authentication

For information on setting up SQL authentication, refer to the instructions below. These steps usually only need to be completed when running as a System Service and attempting to connect remotely to an SQL server.

1. To start, open the **SQL Server Management Studio**. Then, right-click on the SQL server icon in the tree hierarchy and select **Properties**.
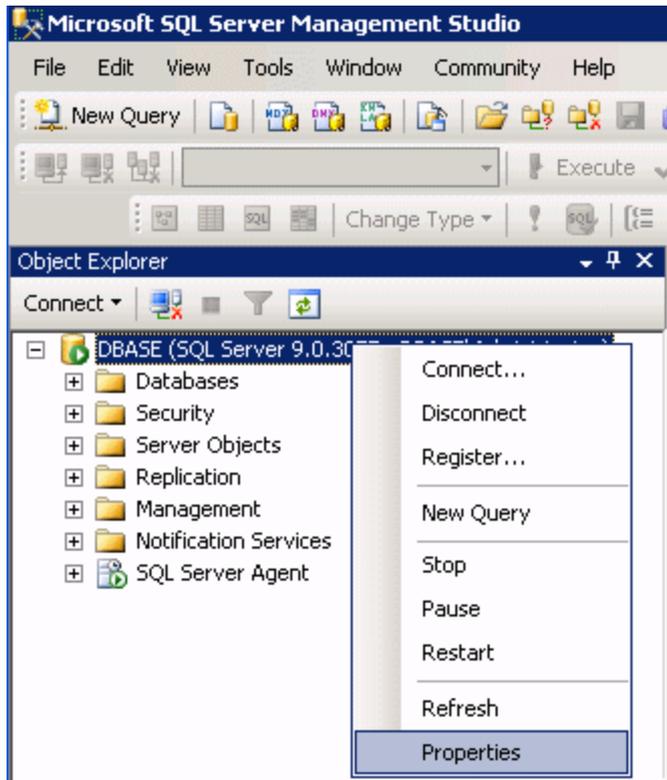
2. Open the **Security** page, and then choose **SQL Server and Windows Authentication Mode**. Then, click **OK**.
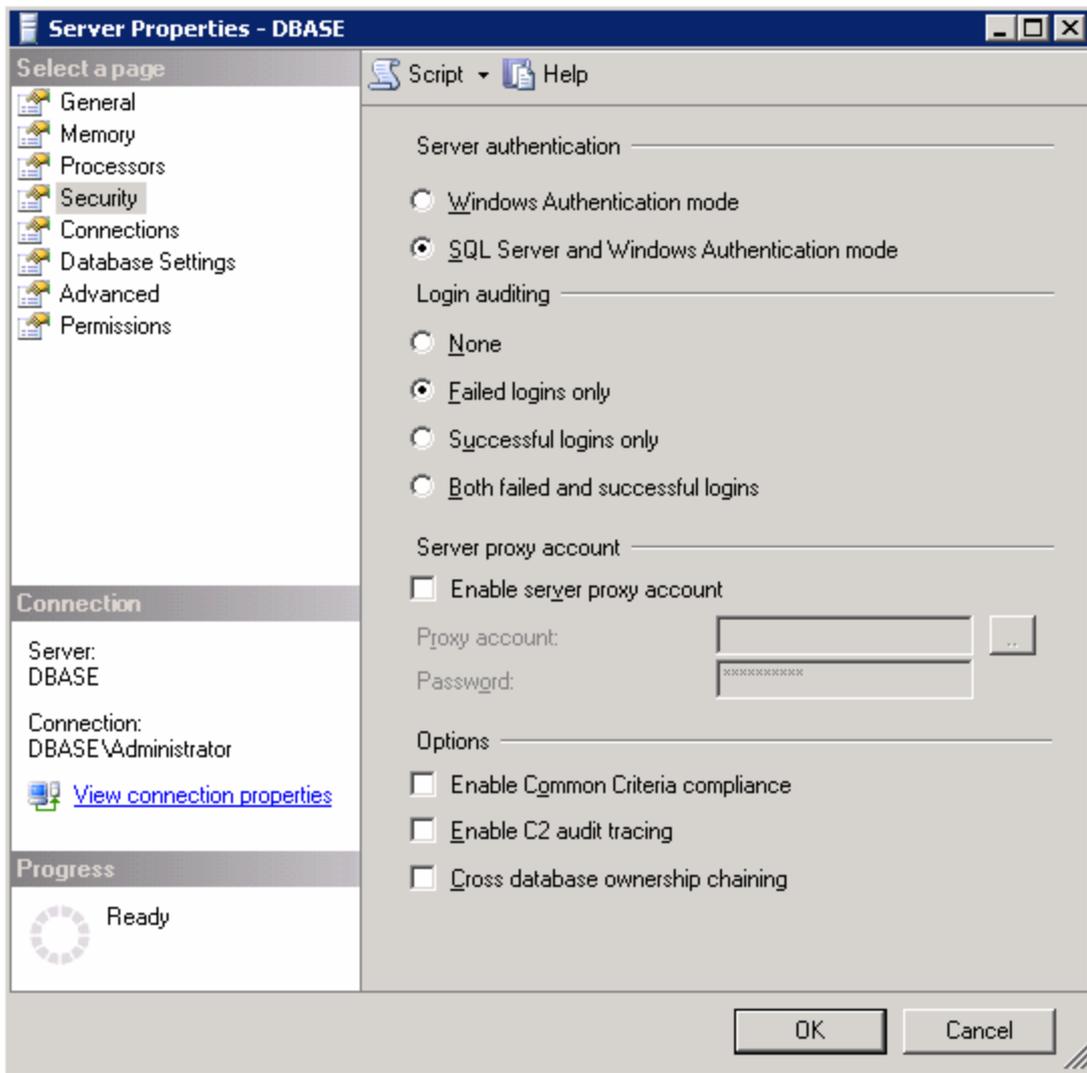
3. In the tree hierarchy, expand **Security**. Then, right-click on **Logins** and select **New Login**.
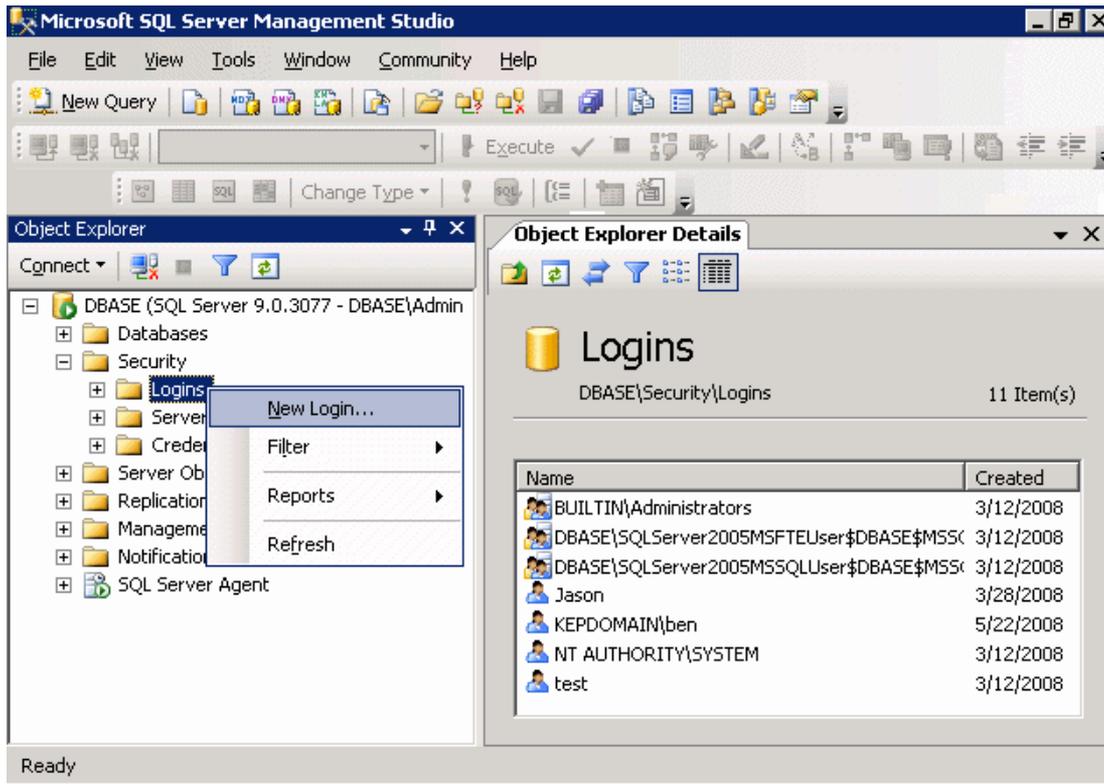
4. Create a new user and define its privileges. Open the **General** page, and then define a user name and password.
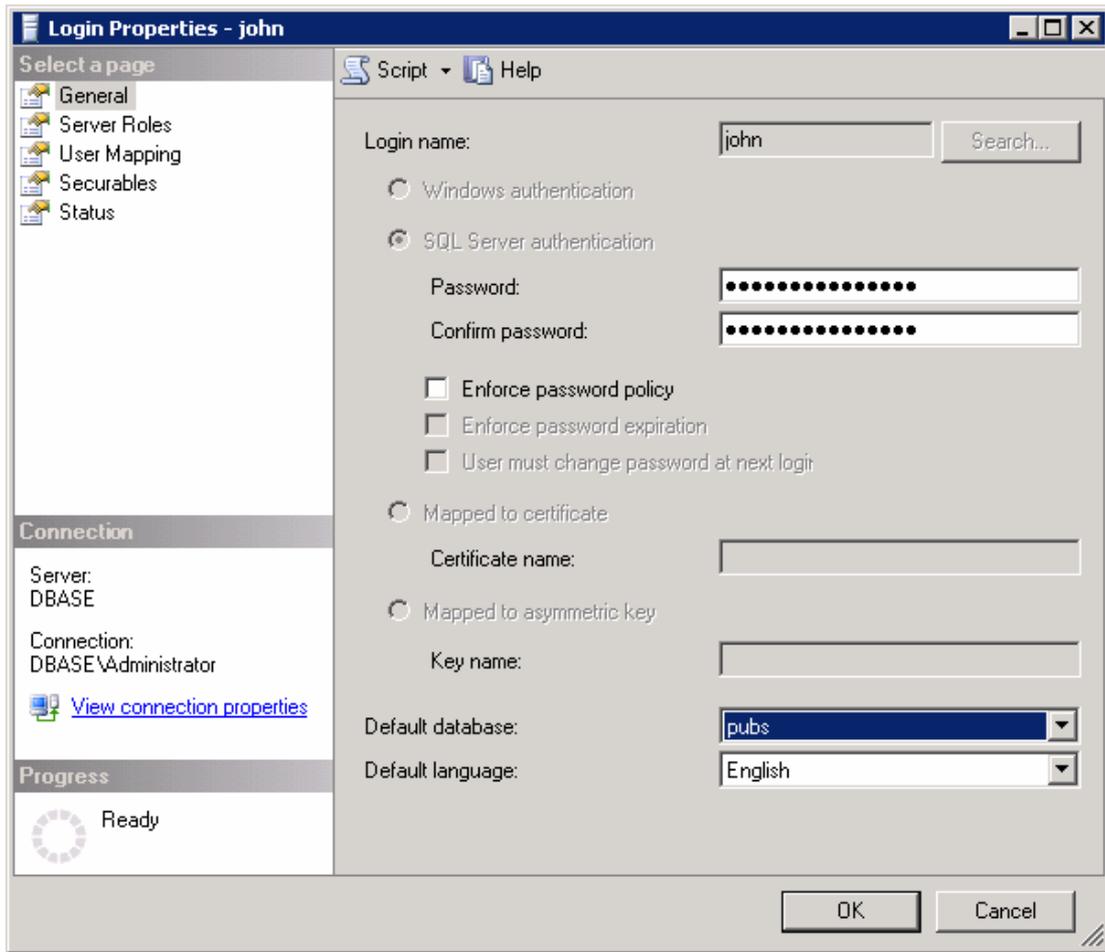
5. Next, open **User Mapping** and select the database to which the user will connect. Then, select a role for the database that has been selected. In this example, **Public** is used.
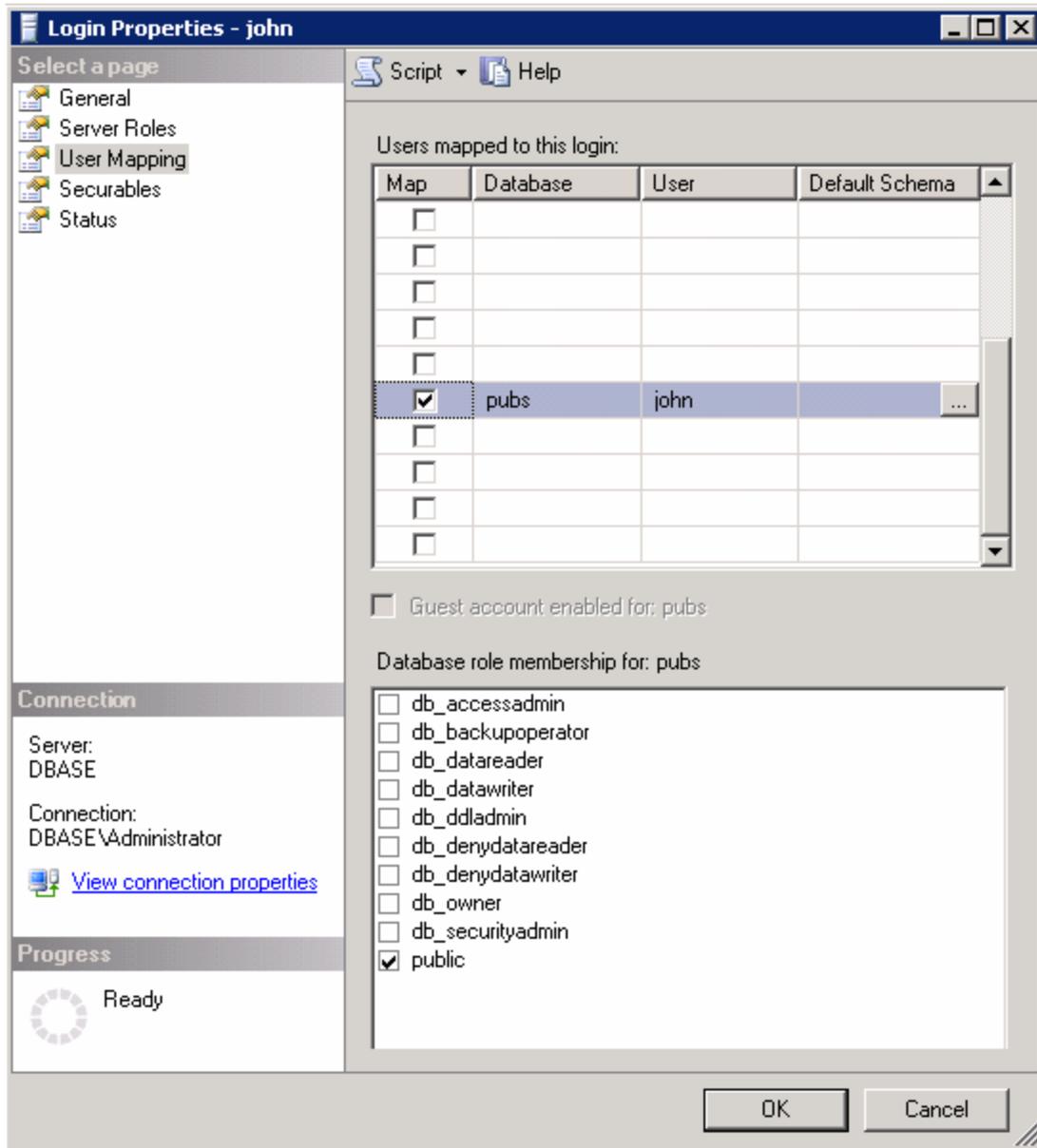
Note: The data source name (DSN) should have been configured before the OPC server was put in System Service Mode. If this has not yet been done, proceed to **ODBC Data Source Setup** before continuing with the instructions below.

6. In the OPC server, select **Tools** | **Options**. Then, open the **Runtime Process** tab.

7. In the **Selected Mode** drop-down menu, select **System Service**. Then, click **OK**.
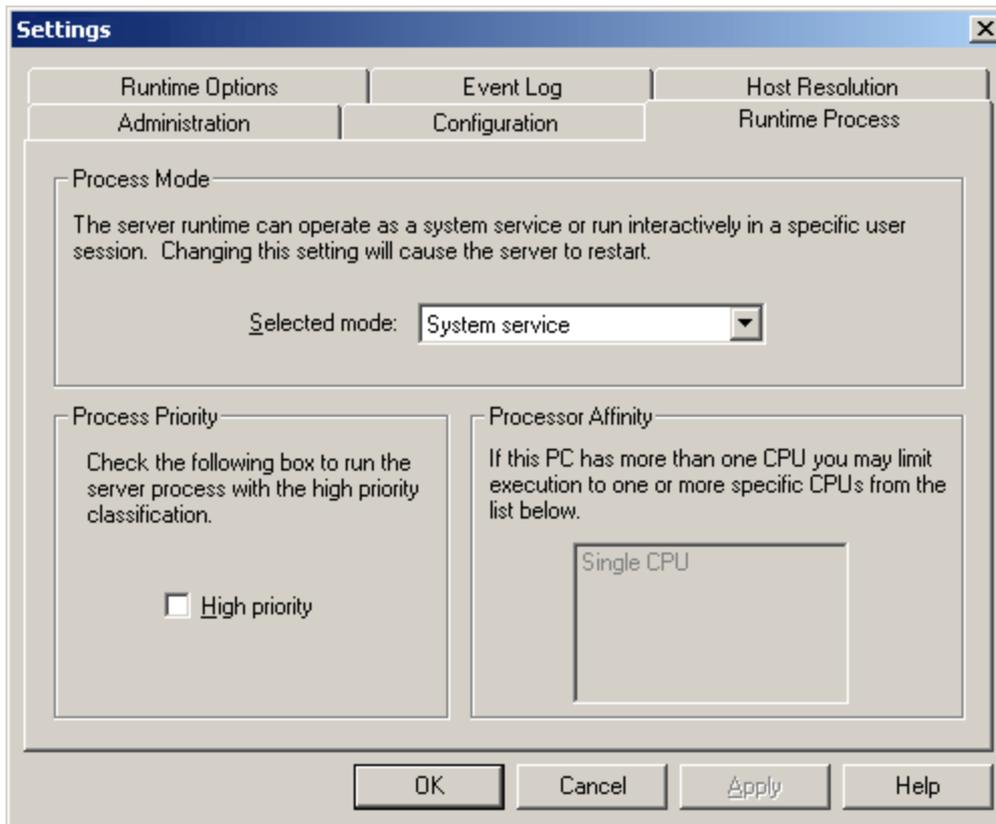
8. Next, right-click on the channel and select **Properties**. Open the **Data Source Setting** tab, and then click **Configure DSN**.

Channel Properties

General | Write Optimizations | Advanced | Data Source Settings

Data Source Selection

Data source: dBASE Files

User name:

Password:

Configure DSN

Login timeout: 5 seconds

OK   Cancel   Apply   Help
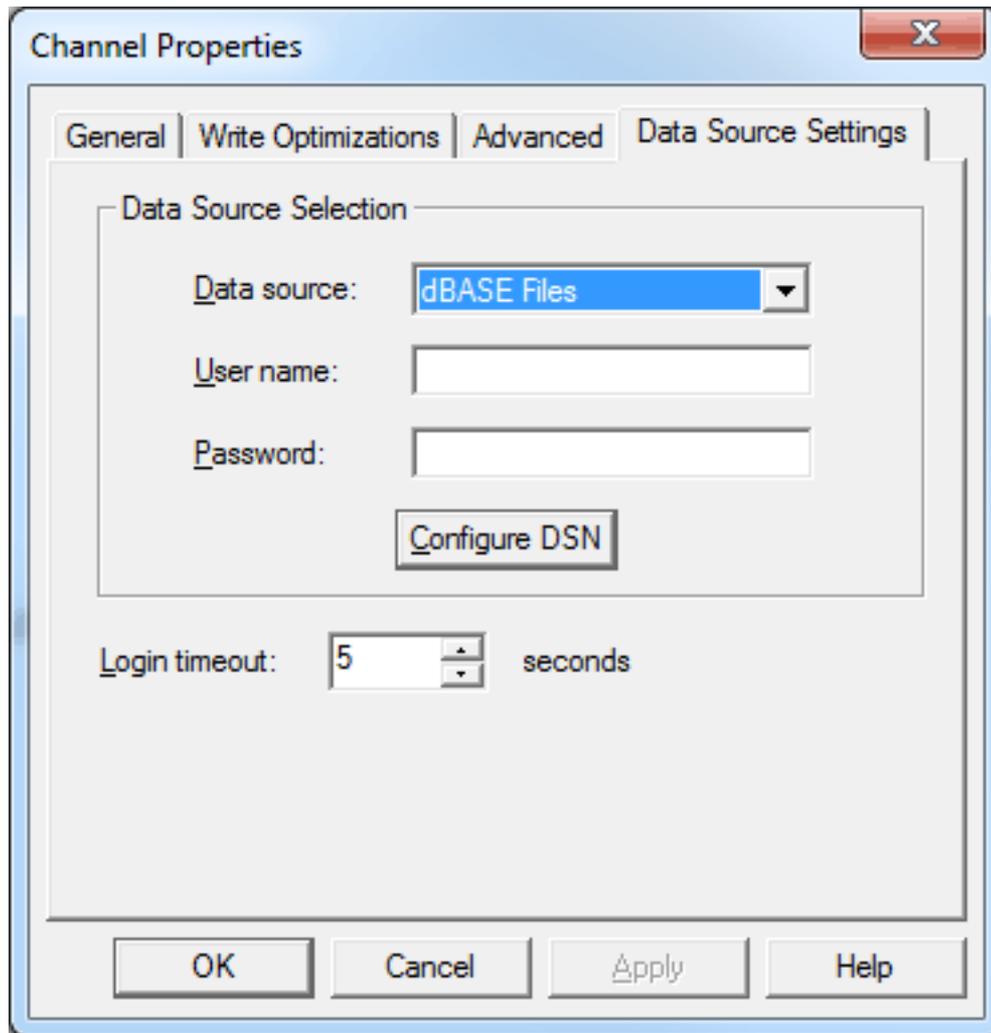
9. Select the newly created DSN, and then click **Configure**.
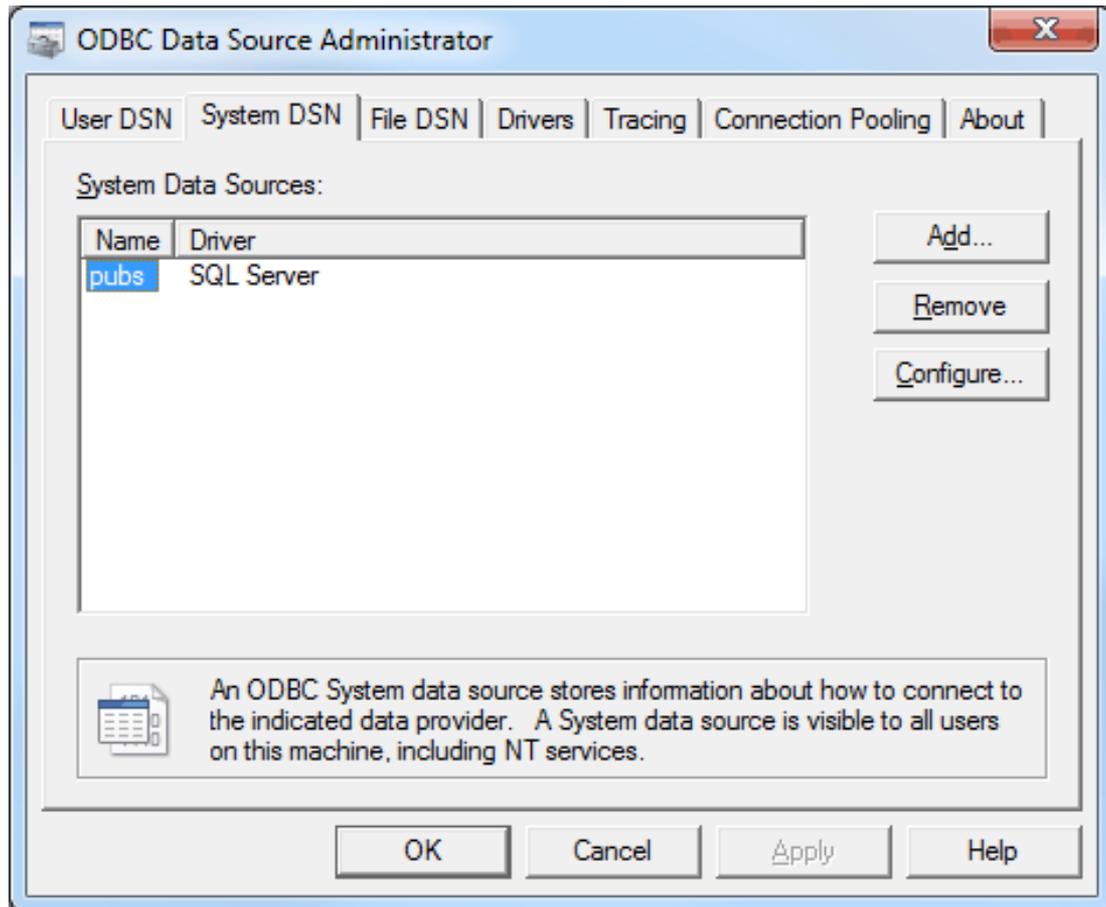
10. Confirm the data source name and the server name. Then, click **Next**.

11. Next, select the **SQL Server Authentication** radio button.

12. Check the box next to **Connect to SQL Server to obtain default settings...** and then enter the user name and password for the user that was defined in the SQL Server.
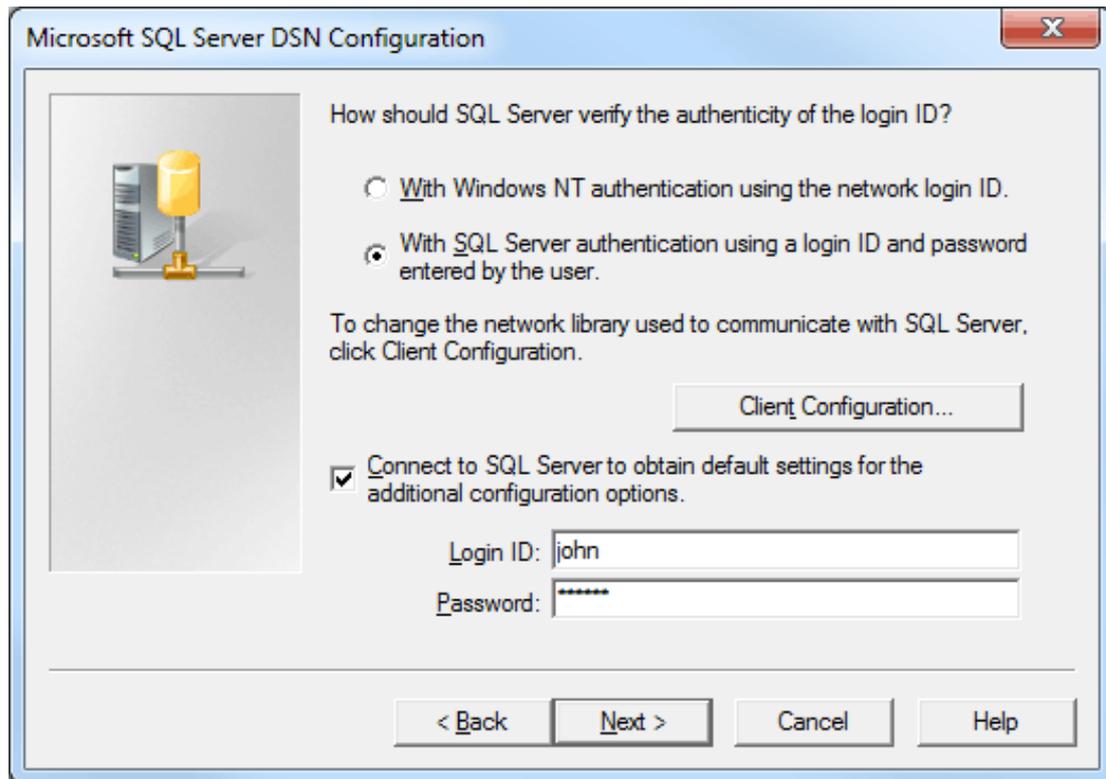
13. Continue through the wizard, keeping the default settings.

14. Upon completion, test the data source connection by clicking **Test Data Source**. If the connection is good, users will receive the message "Test Successful".
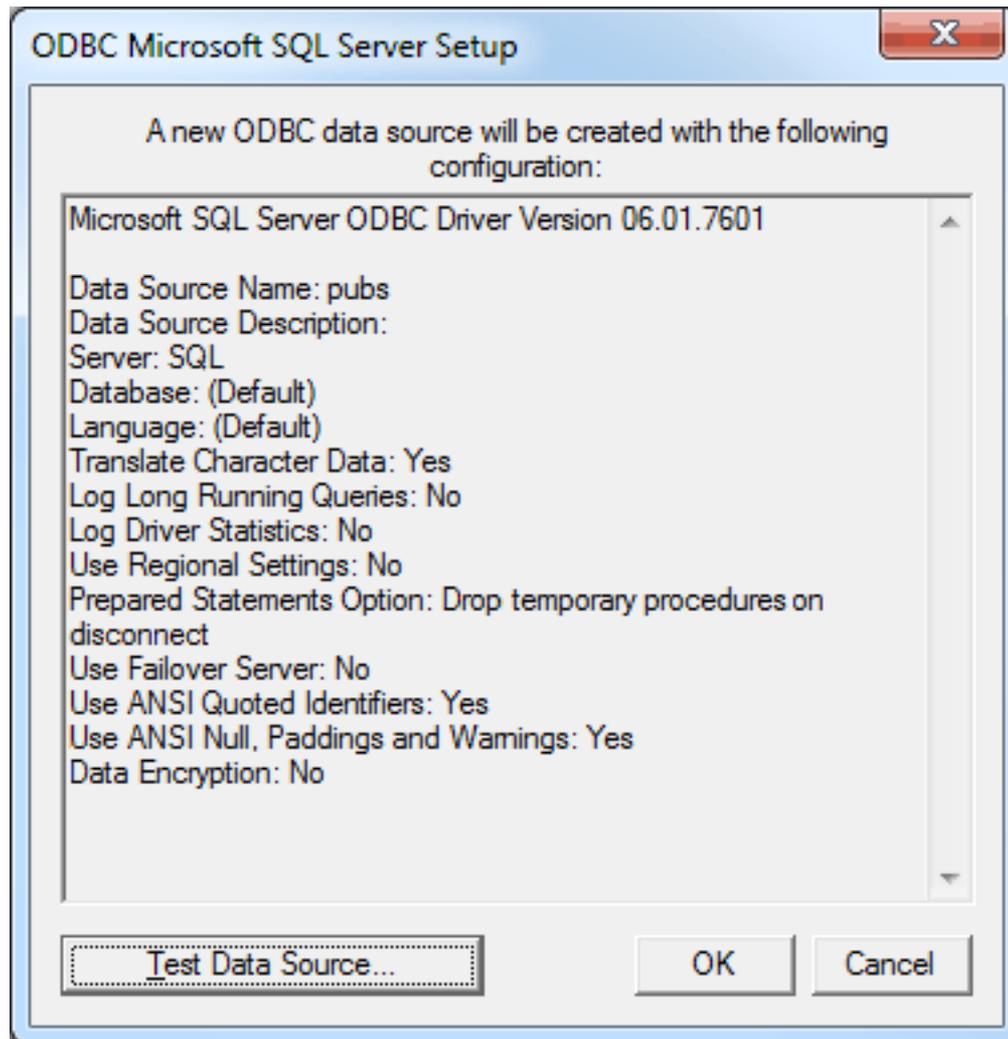
Data Types Description

When creating tags, this driver converts the Transact_SQL data type to the following tag data types:

Transact-SQL Data Type	Tag Data Type	Tag Data Type Description
Bit	Boolean	Single bit
Tinyint	Byte	Unsigned - bit value bit 0 is the low bit bit 7 is the high bit
Int Smallint	Long	Signed 32-bit value bit 0 is the low bit bit 30 is the high bit bit 31 is the sign bit
Real	Float	32-bit floating point value.
Float	Double	64-bit floating point value
Bigint Char Date Datetime2 Datetimeoffset Decimal Money Nchar Numeric Nvarchar Smallmoney Time Varchar	String	Null-terminated character array. This driver limits string lengths to 256 characters.
Datetime Smalldatetime	Date	64-bit floating-point value representing the number of milliseconds since December 31, 1899. Tag value string literal format: YYYY-MM-DDThh:mm:ss.000

Note: The default data type is String.

Address Descriptions

Tag validation will only occur when a tag is requested by a client application. If a link to the ODBC data source cannot be made by the driver, the tag's data quality will be set to Bad. This will also occur if the address that the tag references cannot be resolved by the driver; such as, if the address cannot be mapped to the data set in the ODBC data source. When this occurs, an appropriate error message will be placed in the OPC server's Event Log.

See Also: Error Descriptions

Error Descriptions

The following categories of messages may be generated. Click on a link for a list of related messages.

Address Validation
Device Status Messages
ODBC Client Driver Messages

Address Validation

The following messages may be generated. Click on the link for a description of the message.

Missing address.
Device address <address> contains a syntax error.
Data type <type> is not valid for device address <address>.
Device address <address> is read only.

Missing address.

Error Type:
Warning

Possible Cause:
A tag address that has been specified statically has no length.

Solution:
Re-enter the address in the client application.

Device address <address> contains a syntax error.

Error Type:
Warning

Possible Cause:
A tag address that has been specified statically contains one or more invalid characters.

Solution:

1. If the address is dynamic, re-enter it in the client application.

2. If the address is static, re-enter it in the server application.

Data Type <type> is not valid for device address <address>.

Error Type:
Warning

Possible Cause:
A tag address that has been specified statically has been assigned an invalid data type.

Solution:
Modify the requested data type in the client or server application.

Device address <address> is read only.

Error Type:
Warning

Possible Cause:
A client has requested an access mode that is not compatible with what the data source supports for that tag address.

Solution:
Change the access mode in the client application, under device properties or under the actual tag in the server's tag data base.

Note:
Although some data sources do not support write access, others may support write for certain items.

Device Status Messages

The following message may be generated. Click on the link for a description of the message.

Write request rejected on read only item reference <channel name><device name><address>.

Write request rejected on read only item reference <channel name><device name><address>.

Error Type:
Serious

Possible Cause:

1. The item is read only; it cannot be written.

2. The Auto Tag Generation property to **set write access for imported database tags** is unchecked and the tag was created as read only.

Solution:

1. Determine if the item should be able to be written and change the tag access rights.

2. Enable the **Set write access for imported database tags** property and regenerate the tags.

ODBC Client Driver Specific Messages

The following messages may be generated. Click on a link for a description of the message.

Messages presented in a pop-up dialog:
Query validation failed due to the following: (Reason: ODBC Specific Error).
Unable to retrieve table names (Reason: ODBC Specific Error).
Unable to retrieve table column names (Reason: ODBC Specific Error).
Unable to retrieve table row count (Reason: ODBC Specific Error).
Unable to retrieve stored procedure names (Reason: ODBC Specific Error).
Unable to retrieve supported access methods (Reason: ODBC Specific Error).

Messages presented in the Event Log:
Call to execute stored procedure <stored procedure name> on device <device name> failed (Reason: ODBC Specific Error).
Data source <data source name> is configured as read only (all writes will be rejected). Please refer to the database configuration.
Unable to connect to data source on channel <channel name> (Reason: ODBC Specific Error).
Unable to open recordset on device <device name> (Reason: ODBC Specific Error).
Unable to query recordset on device <device name> (Reason: ODBC Specific Error).
Unable to retrieve database value for tag <tag address> on device <device name> (Reason: ODBC Specific Error).
Unable to write database value <value> for tag <tag address> on device <device name> (Reason: ODBC Specific Error).
Unable to read internal tag <tag address> on device <device name> (Reason: ODBC Specific Error).
Unable to write to internal tag <tag address> on device <device name> (Reason: ODBC Specific Error).
Unable to read parameter tag <tag address>. (Reason: ODBC Specific Error).
Unable to write to parameter tag <tag address> (Reason: ODBC Specific Error).
Unable to execute stored procedure <stored procedure name> on device <device name>. (Reason: ODBC Specific Error).
Unable to write to item <tag address> (Reason: ODBC Specific Error).

Call to execute stored procedure <stored procedure name> on device <device name> failed. (Reason: ODBC Specific Error).

Error Type:

Serious

Possible Cause:

1. The ODBC data source is not available.

2. The ODBC data source is not accessible based on user name and password.

3. The input parameters are not valid.

Solution:

1. Verify that the ODBC data source is configured properly and that it is accessible by the user or from a remote machine.

2. Verify that input parameters are set properly before executing the stored procedure.

Note:
The ODBC Specific Error returned by the ODBC data source driver contains specific information on the cause of the problem.

Data source <data source name> is configured as Read Only (all writes will be rejected).

Error Type:
Serious

Possible Cause:
The ODBC data source is configured such that writes are not allowed.

Solution:
If writes are required, ensure that the data source is configured to allow write access.

Query validation failed due to the following: (Reason: ODBC Specific Error).

Error Type:
Serious

Possible Cause:

1. The ODBC data source is not available.

2. The ODBC data source cannot satisfy the requirements of the query statement.

Solution:

1. Verify that the ODBC data source is configured properly and that it is accessible by the user or from a remote machine.

2. Verify that the query statement is valid for the ODBC data source.

Note:
The ODBC Specific Error returned by the ODBC data source driver contains specific information on the cause of the problem.

Unable to retrieve table names. (Reason: ODBC Specific Error).

Error Type:
Serious

Possible Cause:

1. The ODBC data source is not available.

2. The ODBC data source does not contain any tables.

Solution:

1. Verify that the ODBC data source is configured properly and that it is accessible by the user or from a remote machine.

2. Verify that the ODBC data source contains at least one table.

Note:
The ODBC Specific Error returned by the ODBC data source driver contains specific information on the cause of the problem.

Unable to retrieve table column names. (Reason: ODBC Specific Error).

Error Type:
Serious

Possible Cause:

1. The ODBC data source is not available.

2. The ODBC data source table does not contain any columns.

Solution:

1. Verify that the ODBC data source is configured properly and that it is accessible by the user or from a remote machine.

2. Verify that the ODBC data source table contains at least one column.

Note:
The ODBC Specific Error returned by the ODBC data source driver contains specific information on the cause of the problem.

Unable to retrieve table row count. (Reason: ODBC Specific Error).

Error Type:
Serious

Possible Cause:

1. The ODBC data source is not available.

2. The ODBC data source table does not contain any records.

Solution:

1. Verify that the ODBC data source is configured properly and that it is accessible by the user or from a remote machine.

2. Verify that the ODBC data source table contains at least one record.

Note:
The ODBC Specific Error returned by the ODBC data source driver contains specific information on the cause of the problem.

Unable to retrieve stored procedure names. (Reason: ODBC Specific Error).

Error Type:
Serious

Possible Cause:

1. The ODBC data source is not available.

2. The ODBC data source table does not contain any stored procedures.

Solution:

1. Verify that the ODBC data source is configured properly and that it is accessible by the user or from a

remote machine.

2. Verify that the ODBC data source contains at least one stored procedure.

Note:
The ODBC Specific Error returned by the ODBC data source driver contains specific information on the cause of the problem.

Unable to retrieve supported access methods. (Reason: ODBC Specific Error).

Error Type:
Warning

Possible Cause:
The ODBC data source is not available.

Solution:
Verify that the ODBC data source is configured properly and that it is accessible by the user or from a remote machine.

Note:
The ODBC Specific Error returned by the ODBC data source driver contains specific information on the cause of the problem.

Unable to connect to data source on channel <channel name>. (Reason: ODBC Specific Error).

Error Type:
Serious

Possible Cause:

1. The ODBC data source is not available.

2. The ODBC data source is not accessible based on user name and password.

Solution:

1. Verify that the ODBC data source is configured properly and that it is accessible by the user or from a remote machine.

2. Verify that the authentication settings are properly configured in the data source (if applicable).

Note:
The ODBC Specific Error returned by the ODBC data source driver contains specific information on the cause of the problem.

See Also:
Data Source Authentication

Unable to open recordset on device <device name>. (Reason: ODBC Specific Error).

Error Type:
Serious

Possible Cause:

1. The ODBC data source is not available.

2. The ODBC data source is not accessible based on user name and password.

3. The table or query specified by the device no longer exists.

Solution:

1. Verify that the ODBC data source is configured properly and that it is accessible by the user or from a remote machine.

2. Verify that the table or query is supported by the data source.

Note:
The ODBC Specific Error returned by the ODBC data source driver contains specific information on the cause of the problem.

Unable to query recordset on device <device name>. (Reason: ODBC Specific Error).

Error Type:
Serious

Possible Cause:
The ODBC data source is no longer available.

Solution:
Verify that the ODBC data source is accessible by the user or from a remote machine.

Note:
The ODBC Specific Error returned by the ODBC data source driver contains specific information on the cause of the problem.

Unable to retrieve database value for tag <tag address> on device <device name>.

Error Type:
Serious

Possible Cause:
1. The ODBC data source is no longer available.

2. The tag represents an address that cannot be resolved into a value available from the data source.

Solution:
1. Verify that the ODBC data source is accessible by the user or from a remote machine.

2. Verify that the tag represents an address that is supported by the data source and/or access method.

Unable to write database value <value> for tag <tag address> on device <device name>.

Error Type:
Serious

Possible Cause:
1. The ODBC data source is no longer available.

2. The tag represents an address that cannot be resolved into a value available from the data source.

3. The ODBC data source does not allow write access to this value.

Solution:
1. Verify the ODBC data source is accessible by the user or from a remote machine.

2. Verify that the tag represents an address that is supported by the data source and/or access method.

3. Verify that the value is write-able in the ODBC data source configuration.

Unable to read internal tag <tag address> on device <device name> (Reason: ODBC Specific Error).

Error Type:
Serious

Possible Cause:
The internal tag depends on a successful response from the ODBC data source which is no longer available.

Solution:
Verify that the ODBC data source is still accessible by the user or from a remote machine.

Note:
The ODBC Specific Error returned by the ODBC data source driver contains specific information on the cause of the problem.

Unable to write to internal tag <tag address> on device <device name> (Reason: ODBC Specific Error).

Error Type:
Serious

Possible Cause:
The internal tag depends on a successful response from the ODBC data source which is no longer available.

Solution:
Verify the ODBC data source is still accessible by the user or from a remote machine.

Note:
The ODBC Specific Error returned by the ODBC data source driver contains specific information on the cause of the problem.

Unable to read parameter tag <tag address>. (Reason: ODBC Specific Error).

Error Type:
Serious

Possible Cause:
The parameter tag for a stored procedure is no longer available or requires a data type not supported by the driver.

Solution:
Verify that the parameter tag for the stored procedure is valid and the data type is supported by the driver.

Note:
The ODBC Specific Error returned by the ODBC data source driver contains specific information on the cause of the problem.

Unable to write to parameter tag <tag address> (Reason: ODBC Specific Error).

Error Type:
Serious

Possible Cause:
The parameter tag for a stored procedure is no longer available or requires a data type not supported by the driver.

Solution:
Verify that the parameter tag for the stored procedure is valid and the data type is supported by the driver.

Note:
The ODBC Specific Error returned by the ODBC data source driver contains specific information on the cause of the problem.

Unable to execute stored procedure <stored procedure name> on device <device name>. (Reason: ODBC Specific Error).

Error Type:
Serious

Possible Cause:
An operation that is required to be performed prior to calling the stored procedure has failed. Examples might include:

- Failed to bind parameter <parameter>.
- Procedure requires unsupported parameter type(s) <parameter type>.
- Failed to bind column <column>.
- Fetch result data failed.
- A connection to the data source failed.

Solution:
Verify that the ODBC data source is configured properly for this stored procedure. Respond to the specific reasons provided.

Note:
The ODBC Specific Error returned by the ODBC data source driver contains specific information on the cause of the problem.

Unable to write to item <tag address> (Reason: ODBC Specific Error).

Error Type:
Serious

Possible Cause:

1. The tag is no longer available or requires a data type not supported by the driver.

2. The OPC Server is unable to communicate with the data source.

3. The database is read only.

4. If the data source is on another PC (other than the OPC server), communications to that PC might be lost.

Solution:

1. Verify that the tag is valid and the data type is supported by the driver.

2. Verify that the data source configuration is correct.

3. Verify that the data source has write permissions.

4. Check physical connections between the OPC server PC and the data source PC.

Note:
The ODBC Specific Error returned by the ODBC data source driver contains specific information on the cause of the problem.

See Also:
Data Source Setup

Index