Cyberlogic®

Data Logger Help

Database Logging for OPC Servers

Version 9

DATA LOGGER HELP

For Cyberlogic OPC Servers

Version 9

Document last revision date May 5, 2017

TABLE OF CONTENTS

INTRODUCTION

The Cyberlogic OPC Server provides OPC Data Access, Alarms & Events and XML Data Access functions. Its modular structure supports a variety of industrial devices and communication networks. As a result, the server maintains a set of common features, but has the flexibility to allow you to add optional features that you may need for specific applications.

Data Logger is one of these optional features. With it, you can store the values and other attributes of OPC data items in standard databases. The databases it supports include MS SQL Server, Oracle, MySQL and MS Access. The Data Logger feature is included with Cyberlogic's DHX OPC Enterprise Suite, MBX OPC Enterprise Suite, OPC Crosslink Enterprise Suite, OPC Datacenter Suite and OPC Datacenter Premier Suite.

Note This document includes only the information that is specific to the Data Logger feature. For information on the common features of the Cyberlogic OPC Server, refer to the Cyberlogic OPC Server Help.

Compatibility and Compliance

Data Logger is a feature of Cyberlogic's family of OPC products. The Cyberlogic OPC Server can provide data to any OPC client from any supplier, provided that the client software complies with OPC Foundation specifications.

Cyberlogic OPC products provide full compliance with the OPC Foundation specifications for:

- Data Access 3.0, 2.05a and 1.0a
- Alarms & Events 1.1
- XML Data Access 1.0
- Data Access Automation 2.02

These products are tested for compliance to the OPC specifications using the latest test software from the OPC Foundation. All Cyberlogic OPC products are certified for compliance by the OPC Foundation's Independent Testing Laboratory. In addition, they are tested annually for interoperability with other OPC products at the OPC Foundation's Interoperability Workshops.

WHAT SHOULD I DO NEXT?

The links below take you to the sections of this manual that tell you how to configure, use and troubleshoot the logger.

This document describes only the features specific to the logger. For information on the common features of the Cyberlogic OPC Server, refer to the Cyberlogic OPC Server Help.

Learn How Data Logger Works

If you are not familiar with the way that the logger stores data, you should begin by reading the Theory of Operation.

Read a Quick-Start Guide

First-time users of the Data Logger feature will want to read the Quick Start Guide for a step-by-step walk through a typical configuration session.

Get Detailed Information on the Configuration Editor

Experienced users who want specific information about how to configure the logger will find it in the Configuration Editor Reference section.

Verify That It Works or Troubleshoot a Problem

After you configure the server, you should verify that it operates as expected. Refer to the Validation and Troubleshooting section for assistance. In case of runtime problems, this section also provides problem-solving hints.

Print a Copy of This Document

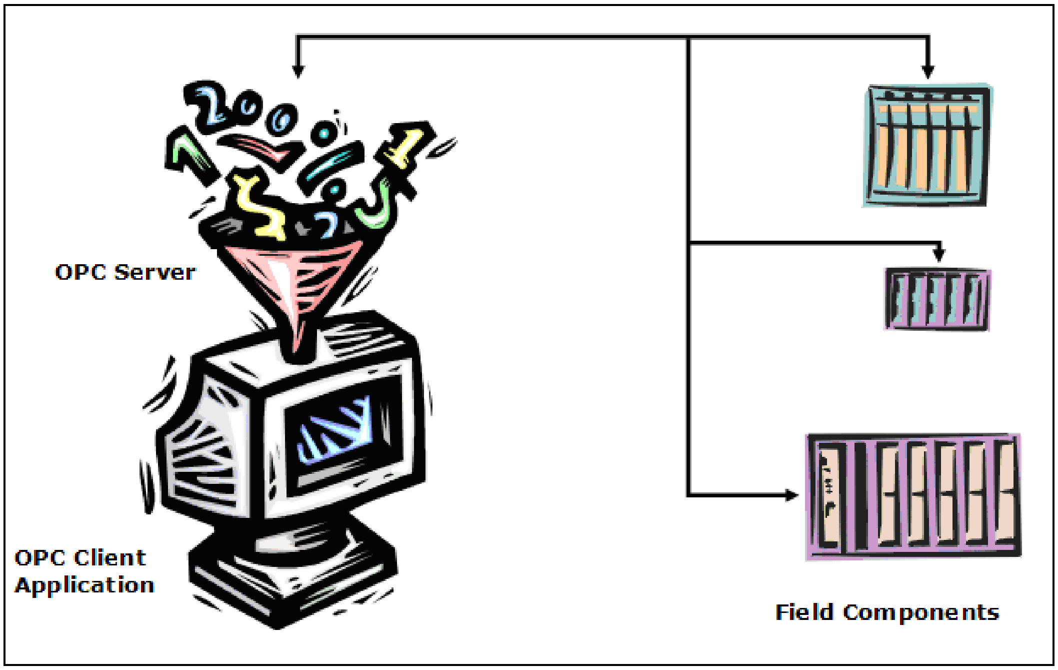
This document is also provided in PDF format. Use Adobe® Reader or another PDF reader program to view and print the PDF file.

Contact Technical Support

To obtain support information, open the Windows ***Start*** menu and go to ***Cyberlogic Suites***, and then select ***Product Information***.

Theory of Operation

This section explains the main features of the Cyberlogic OPC Server as they relate to the Data Logger feature. Refer to the Cyberlogic OPC Server Help for a full discussion of the common features of the Cyberlogic OPC Server. If you are new to OPC or the Cyberlogic OPC Server, we strongly recommend that you read the OPC Tutorial first. You will find it in the Help section of your product installation.



The basic function of an OPC server is shown in the figure above. The server reads data from field components and presents it, in a standard way, to OPC client applications. Typically, these field components are PLCs or similar devices.

OPC Logging

All OPC servers present their data to OPC clients in real time. But if you want to access the data later, you will have to store it in a database. Data Logger is an optional feature that provides that capability for Cyberlogic's OPC server. Once the data is logged, it is available to any application that can access that database. It need not be an OPC client application.

Advantages of Server-Based Logging

Some data loggers operate as OPC clients, and therefore require an OPC server to provide the data to be logged. That increases the complexity and cost of the system, and lowers its overall performance. The server already has the data, so it is simpler to have the server write it directly to the database.

Another advantage of server-based logging is that the server can expose the internal status and diagnostic information as OPC data items, which most client-based data loggers cannot do. When the server exposes these data items, any OPC client application can monitor and control the logging operation.

Data Logger Features

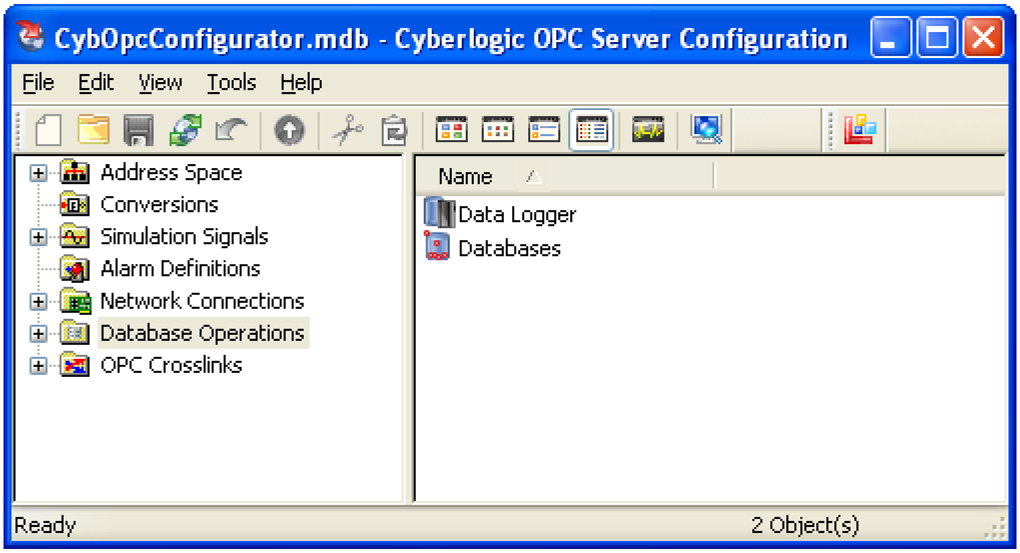
Cyberlogic's logger includes many advanced features, some of which are unique to Cyberlogic.

- Supports many popular databases: SQL Server, Oracle, MySQL, MS Access
- Concurrently logs to multiple databases, which may be of different types
- Supports local and remote database connections
- Logging tasks are organized in groups with common enable criteria and data acquisition settings
- Logging groups may be enabled dynamically by the value of a data item
- Data records may contain property values of a single item, multiple items, or be defined as parameters of a stored procedure
- A single write operation can insert a new record or update multiple existing records
- The target database table or stored procedure name, or the update criteria can be dynamically updated at runtime
- New records can be generated at a fixed interval, on data change, when triggered by a data item, or a combination of all of these methods
- Extensive set of standard trigger events that can be used to initiate logging
- Unlimited number of custom trigger events through the use of built-in Math & Logic engine
- The ability to purge old database records
- A database connection can be kept open, closed after an idle timeout, or closed after every write operation
- A database connection can be enabled dynamically at runtime
- Memory and disk cache to store data records if the database connection is slow or not available
- Memory cache size can be limited to prevent out of memory conditions
- Disk cache size may be limited by setting a minimum free disk space
- Status and diagnostic tags available at runtime

If you want a general description of how logging fits into the server configuration, continue with Main Server Features. For an example of how to configure the logger, skip to the Quick Start Guide.

Main Server Features

We will now look at the main features of the Cyberlogic OPC Server, as they relate to Data Logger.



The Cyberlogic OPC Server Configuration editor has seven main trees, representing the seven main areas that you will configure. These are:

Address Space Tree

This tree is required for most configurations. Here you will create and organize the data items, and define how they will be updated with new information from the field components. The values of these data items will then be available to client applications. Typically, you will also use data items to hold the values that the logger will store in the database. This is also where you will configure the trigger events that can be used to control the logging process. Refer to the Cyberlogic OPC Server Help for a full discussion of this tree. Refer to the Math & Logic Help for more information on configuring standard and custom trigger events.

Conversions Tree

The Conversions Tree is optional. In it, you define formulas that convert raw data values obtained from the field equipment into a form that is more useful to the client. For example, you can change a transducer's voltage value into a pressure value in pounds per square inch. Refer to the Cyberlogic OPC Server Help for a full discussion of this tree.

Simulation Signals Tree

This tree is also optional. If you want data items to use simulated values instead of real values, you can create various types of simulated data functions in this tree. Simulations are often useful for verification and troubleshooting. For example, you might want to simulate a data item that is being logged, so that you can verify that the database is updating properly. Refer to the Cyberlogic OPC Server Help for a full discussion of this tree.

Alarm Definitions Tree

You will use this optional tree to interface to OPC Alarms & Events clients. This tree allows you to define the desired alarm conditions and specify what information should be passed as they occur and clear. Refer to the Cyberlogic OPC Server Help for a full discussion of this tree.

Network Connections Tree

This tree is where you configure communication to OPC servers, PLCs and other data sources. You select the networks and interface devices you want to use, and configure each of the field components as nodes on those networks. Refer to the driver agent help files for more information.

Database Operations Tree

You configure Data Logger in this tree. In the Databases branch, you will specify the databases that will receive the data. In the Data Logger branch, you will define what data is to be logged, which database it should be stored in, and when the writes should occur.

OPC Crosslinks Tree

The OPC Crosslinks Tree lets you identify the data that you want to transfer among PLCs and OPC servers. You can also set up crosslinks to write the results of Math & Logic programs to other OPC servers or PLCs. Refer to the OPC Crosslink Help for a full discussion of this tree.

QUICK START GUIDE

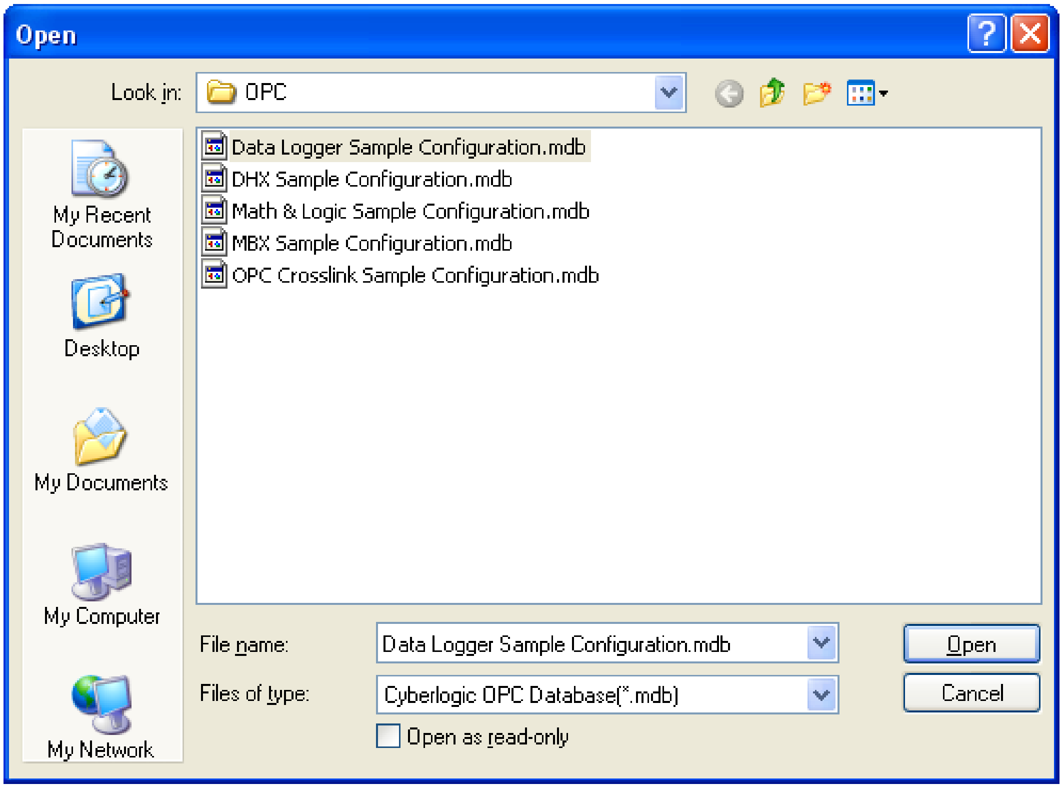
Before you can use the Cyberlogic OPC server, including Data Logger, you must configure it by using the OPC Server Configuration Editor. Data Logger users must configure the Network Connections and Database Operations trees, and will usually want to configure the Address Space tree. The remaining trees (Conversions, Simulation Signals, Alarm Definitions and OPC Crosslinks) are optional features used by some systems.

There are two ways you can quickly get familiar with Data Logger. First, you can look at the Sample Configuration Files, which you can use or modify for your application. You can also go to the Step-By-Step Example for a walk-through of a basic Data Logger configuration session.

Sample Configuration Files

The default installation of all Cyberlogic OPC Server, OPC Crosslink and OPC Datacenter Suites includes a set of sample configuration files. These samples will help you to understand how to configure the OPC server for your project.

To open a sample configuration file from the OPC Server Configuration Editor, open the ***File*** menu and then select ***Open Sample...*** .



Look in: OPC

Data Logger Sample Configuration.mdb
DHX Sample Configuration.mdb
Math & Logic Sample Configuration.mdb
MBX Sample Configuration.mdb
OPC Crosslink Sample Configuration.mdb

File name: Data Logger Sample Configuration.mdb

Files of type: Cyberlogic OPC Database(*.mdb)

□ Open as read-only

A browse window will open and you can select the configuration file you want. The available choices will depend on which OPC products you have installed.

The default location of the files is:

C:\Program Files\Common Files\Cyberlogic Shared\OPC.

Step-By-Step Example

This section shows you how to configure logging in the Database Operations tree. It does not cover any of the other parts of the configuration. For a step-by-step example of how to configure a complete OPC server, refer to the Cyberlogic OPC Server Help.

Use this example only as a guideline of how to configure the most common features. For detailed information on all of the Data Logger features, refer to the Configuration Editor Reference.

In this example, we will configure Data Logger to store production and downtime data in a Microsoft SQL Server database. It will create a new record every hour during the first shift of the day and add that record as a row in a table of the database.

Initial Preparation

Before you begin to configure Data Logger, you must create the desired database on the local computer or on another computer on the network. For this example, we assume you have already created an MS SQL Server database called *PlantDatabase* on the local machine. It contains a table called *FirstShiftRecords*, which is where you want to store the data. Consult your database documentation for information on how to create the database and table.

Within your OPC server, you must configure the data items you want to log. These data items may contain values taken directly from PLCs or other data sources, or they may be Math & Logic data items that contain calculated values. For this example, we assume you have created two registers: *Shift1Production* and *Shift1Downtime*. Consult the Cyberlogic OPC Server Help for information on how to create data items.

The procedure is divided into several sections:

- Create the Database
- Create the Trigger Data Item
- Create the Logging Group
- Configure the Data Record
- Configure the Disk Cache
- Save the Configuration and Update the Server

To begin, we will Create the Database.

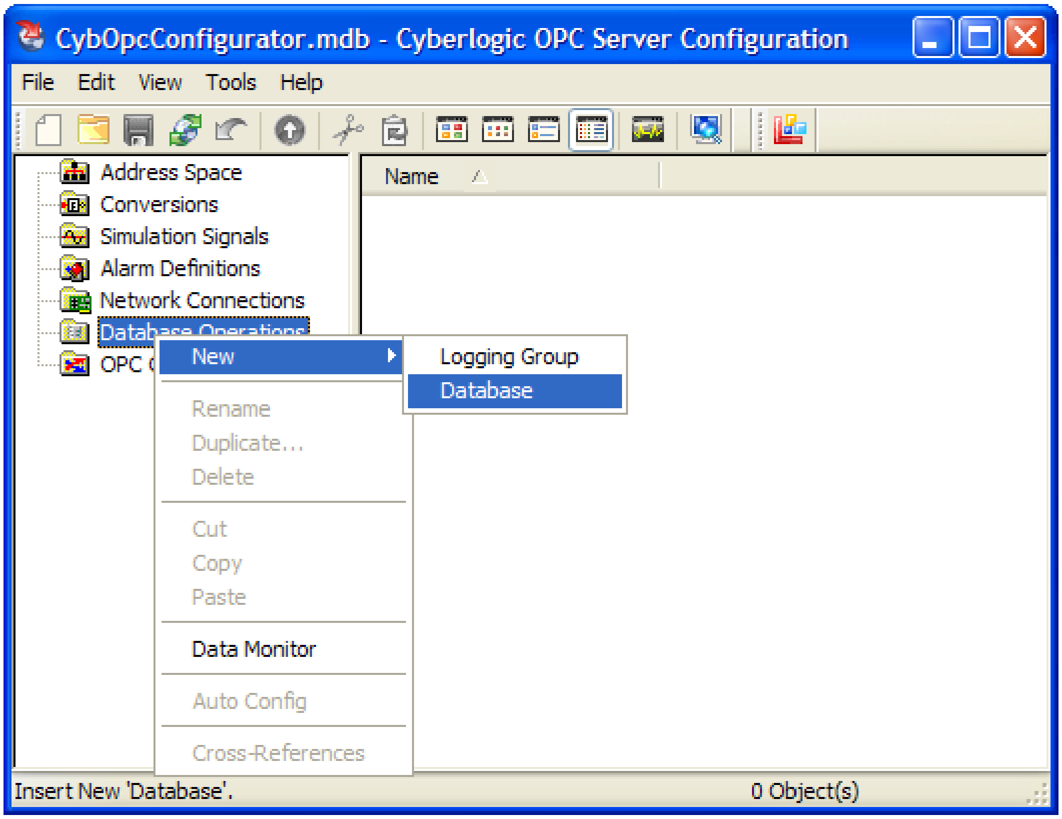
Create the Database

A database in the Database Operations tree specifies the computer and database that you will use to store the logged data. The software supports MS SQL Server, Oracle, MySQL and MS Access databases.

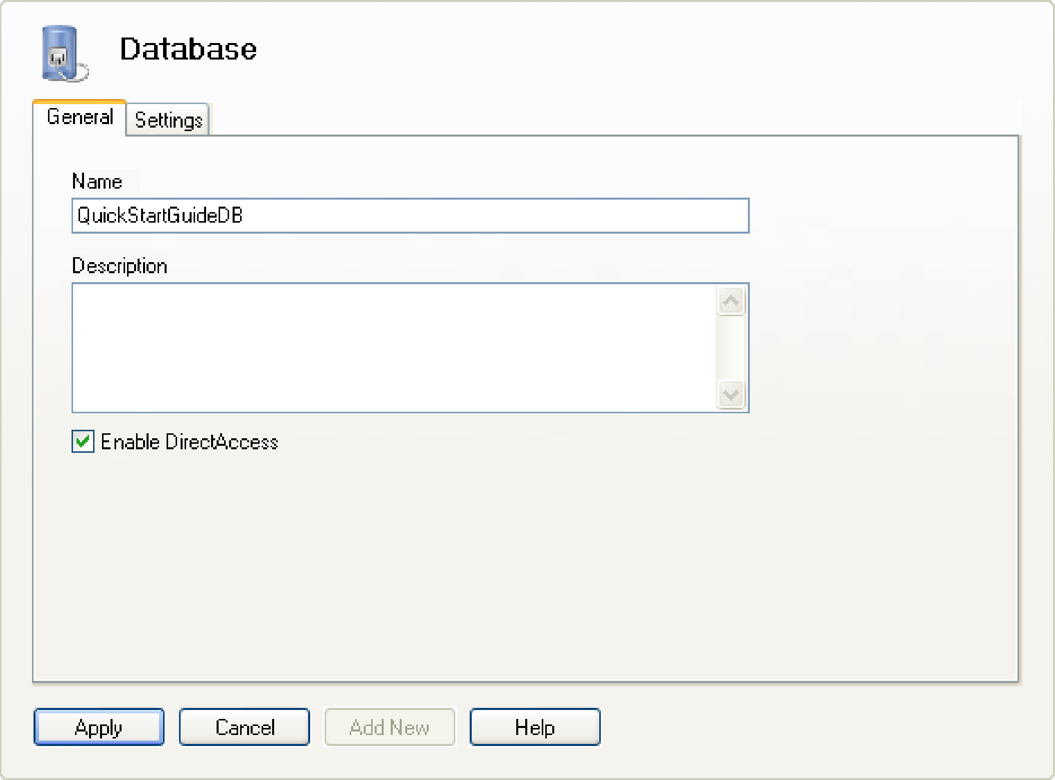
Note Before you begin to configure Data Logger, you must create the desired database on the local computer or on another computer on the network. Refer to Initial Preparation for more information about this requirement.

1. Open the Windows ***Start*** menu and go to ***Cyberlogic Suites***. From there, open the ***Configuration*** sub-menu and select the ***OPC Server*** menu item.

   The Cyberlogic OPC Server Configuration editor will open.



2. Right-click on the ***Database Operations*** branch and select ***New*** from the context menu, and then select ***Database***.

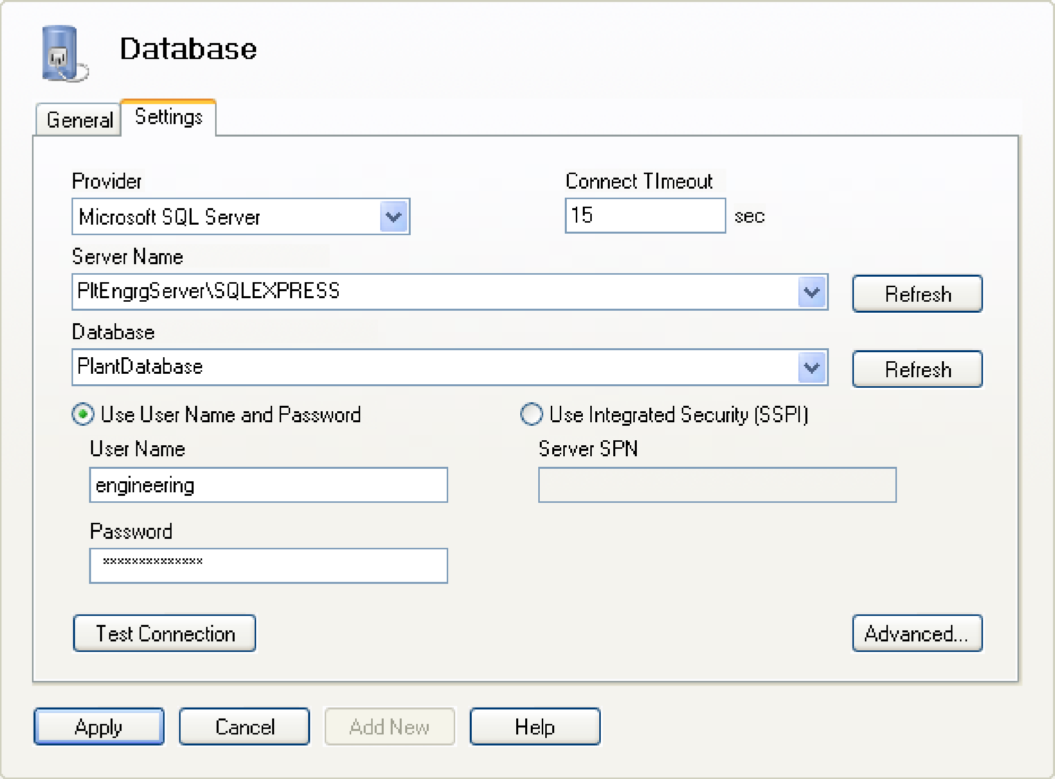


Name: QuickStartGuideDB

Description:

☑ Enable DirectAccess

3. Enter a name for the database in the ***Name*** field.



Provider: Microsoft SQL Server

Connect TImeout: 15 sec

Server Name: PltEngrgServer\SQLEXPRESS

Database: PlantDatabase

● Use User Name and Password ○ Use Integrated Security (SSPI)

User Name: engineering

Password: **************

Server SPN:

4. Select the ***Settings*** tab.

5. Under ***Provider***, select ***Microsoft SQL Server***.

6. Under ***Server Name***, select the computer and MS SQL Server instance you want Data Logger to use.

Note The list that the software returns when it searches for servers may show only the computer name. In that case, you must type in the name of the database server software instance.

7. Under ***Database***, select the desired database within that server.

8. Select the appropriate security choice. If you choose ***Use User Name and Password***, enter a ***User Name*** and ***Password***.

9. Click ***Test Connection*** to verify that Data Logger can connect to the database.

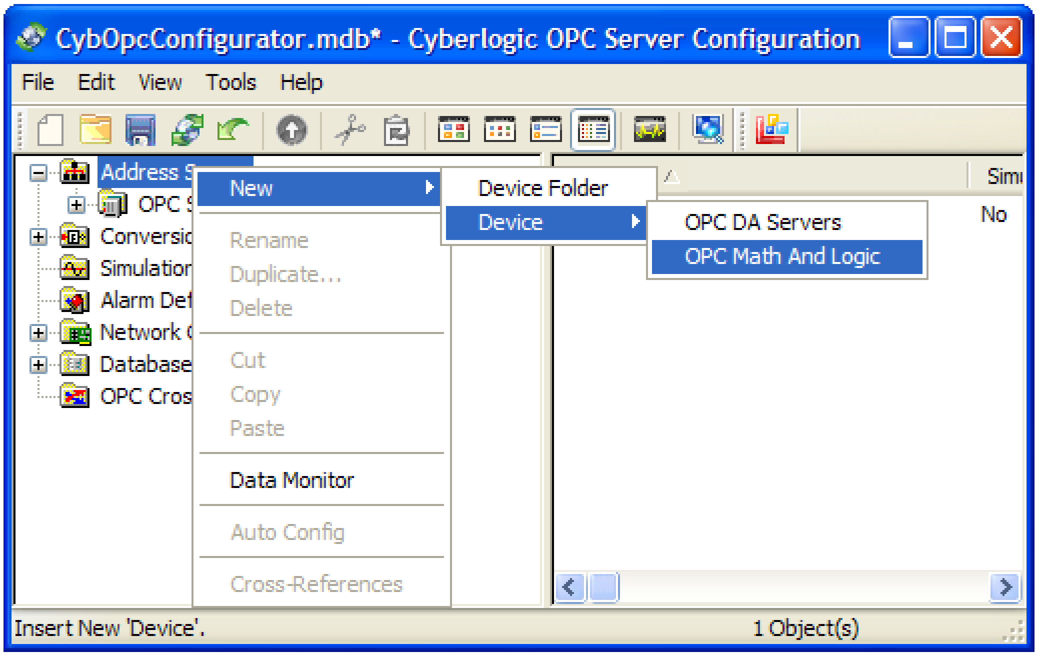
10. Click ***Apply***. You can now use *QuickStartGuideDB* within your configuration.

11. Go to Create the Trigger Data Item to continue.

Create the Trigger Data Item

There are many ways to trigger a logging group to save a record. In this example, we will use a Math & Logic trigger item.

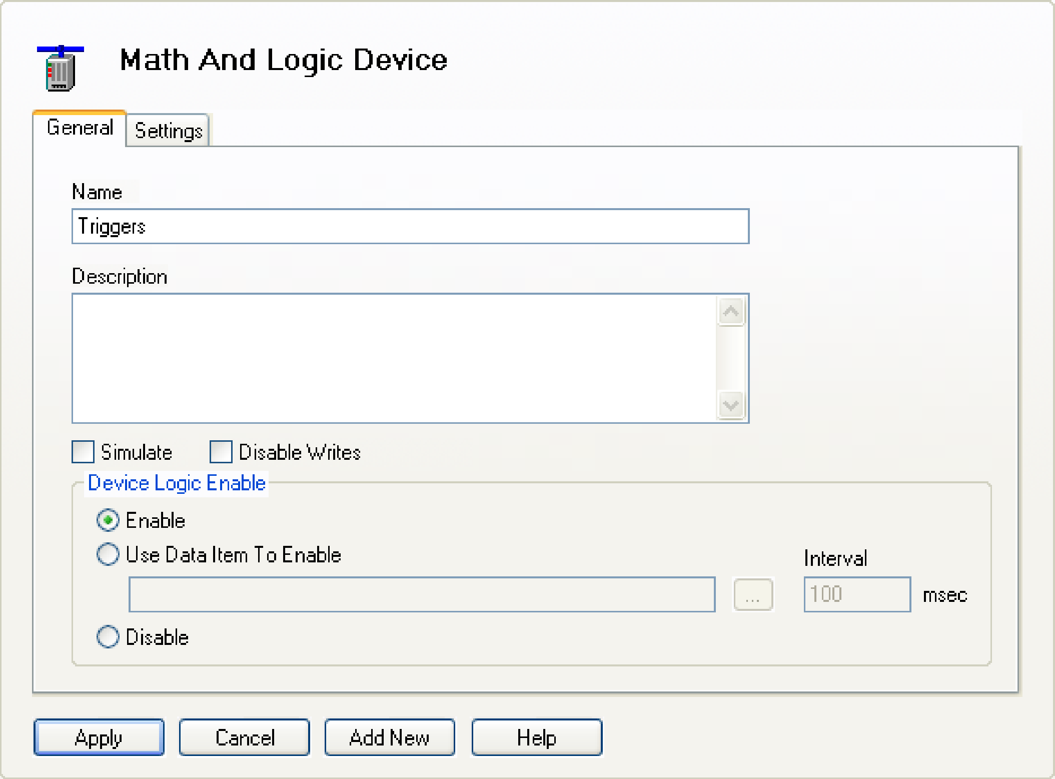
Note The full Math & Logic feature is part of the Premier and Enterprise suites. However, the standard OPC Datacenter Suite includes the ability to use preprogrammed Math & Logic data items as trigger and switch signals. This is the type we will use here. If you have a Premier or Enterprise suite, you can also use Math & Logic to create your own custom-programmed data items.



Insert New 'Device'.

1 Object(s)

1. Right-click on the ***Address Space*** root and select ***New*** from the context menu, then select ***Device*** and then ***OPC Math & Logic***.



Name: Triggers

Description:

□ Simulate □ Disable Writes

Device Logic Enable

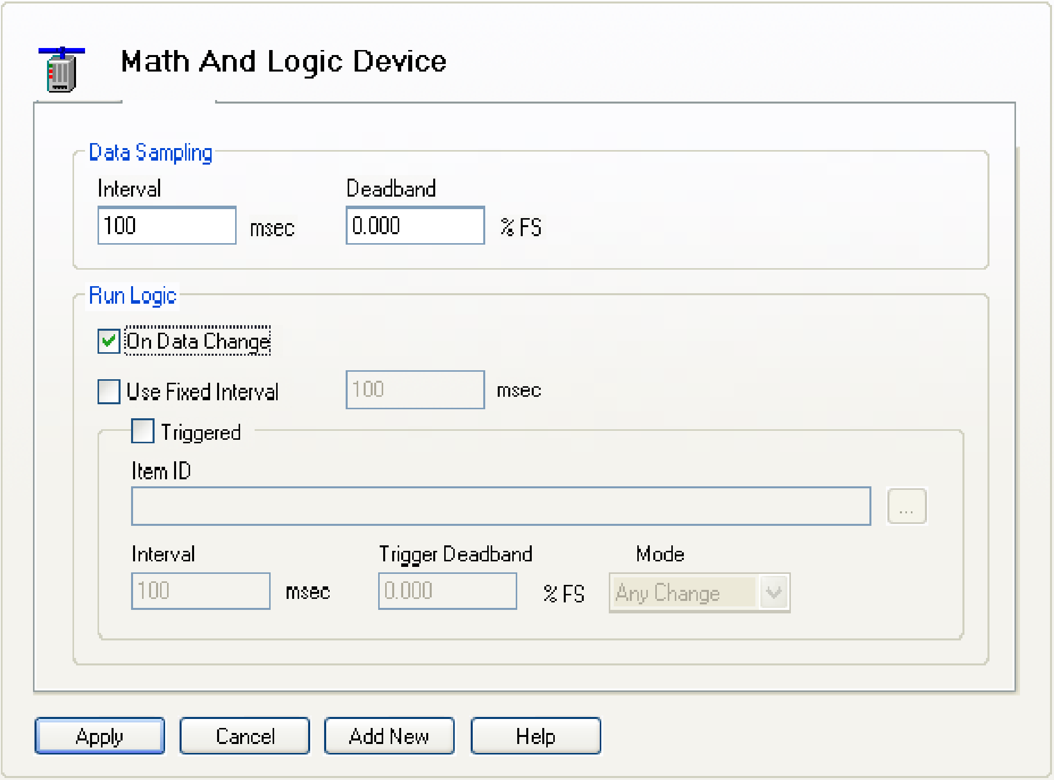
● Enable

○ Use Data Item To Enable

Interval 100 msec

○ Disable

2. Select the device's ***General*** tab, and enter ***Triggers*** in the ***Name*** field.



Math And Logic Device

Data Sampling

Interval 100 msec

Deadband 0.000 % FS

Run Logic

☑ On Data Change

☐ Use Fixed Interval 100 msec

☐ Triggered

Item ID

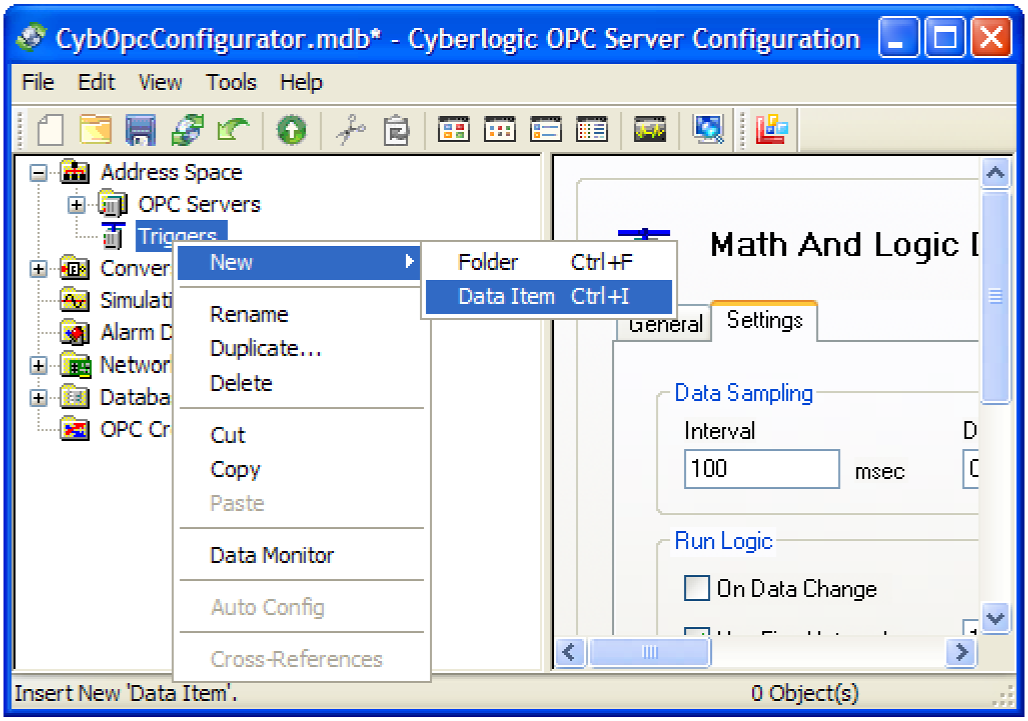
Interval 100 msec

Trigger Deadband 0.000 % FS

Mode Any Change

3. On the ***Settings*** tab, verify that only the ***On Data Change*** box is checked.

4. Click ***Apply***.

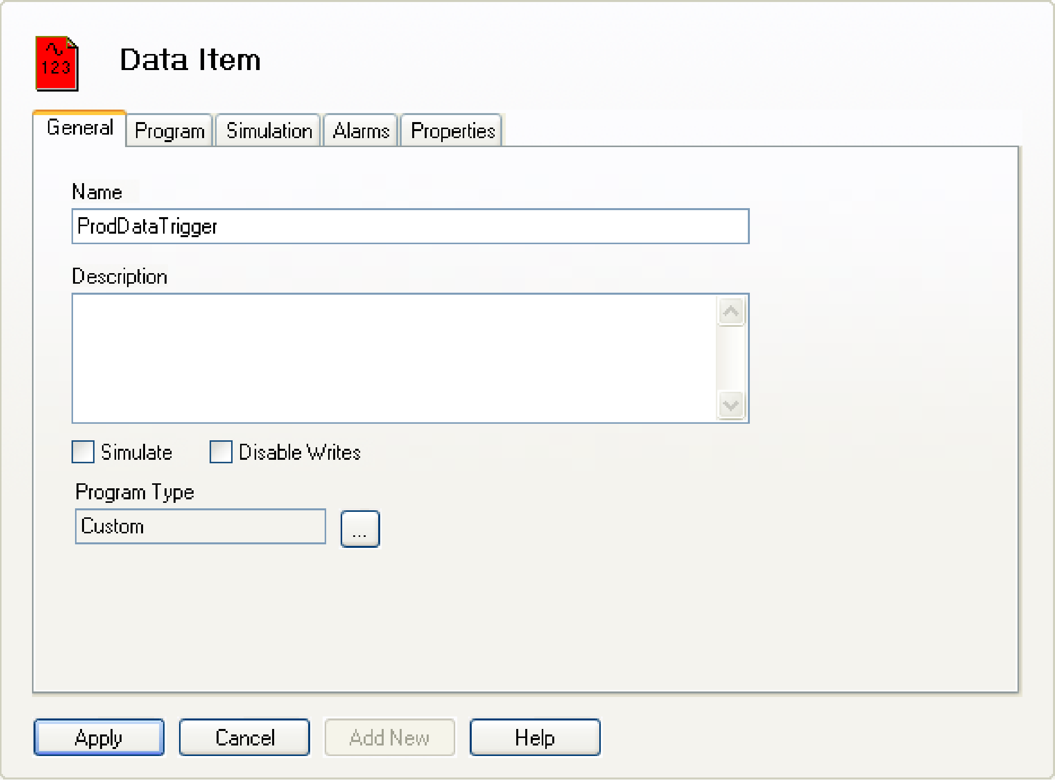


CybOpcConfigurator.mdb* - Cyberlogic OPC Server Configuration

Insert New 'Data Item'.

0 Object(s)

5. Right-click on the ***Triggers*** device you just created, and select ***New*** from the context menu, and then select ***Data Item***.



Data Item

Name: ProdDataTrigger

Description:

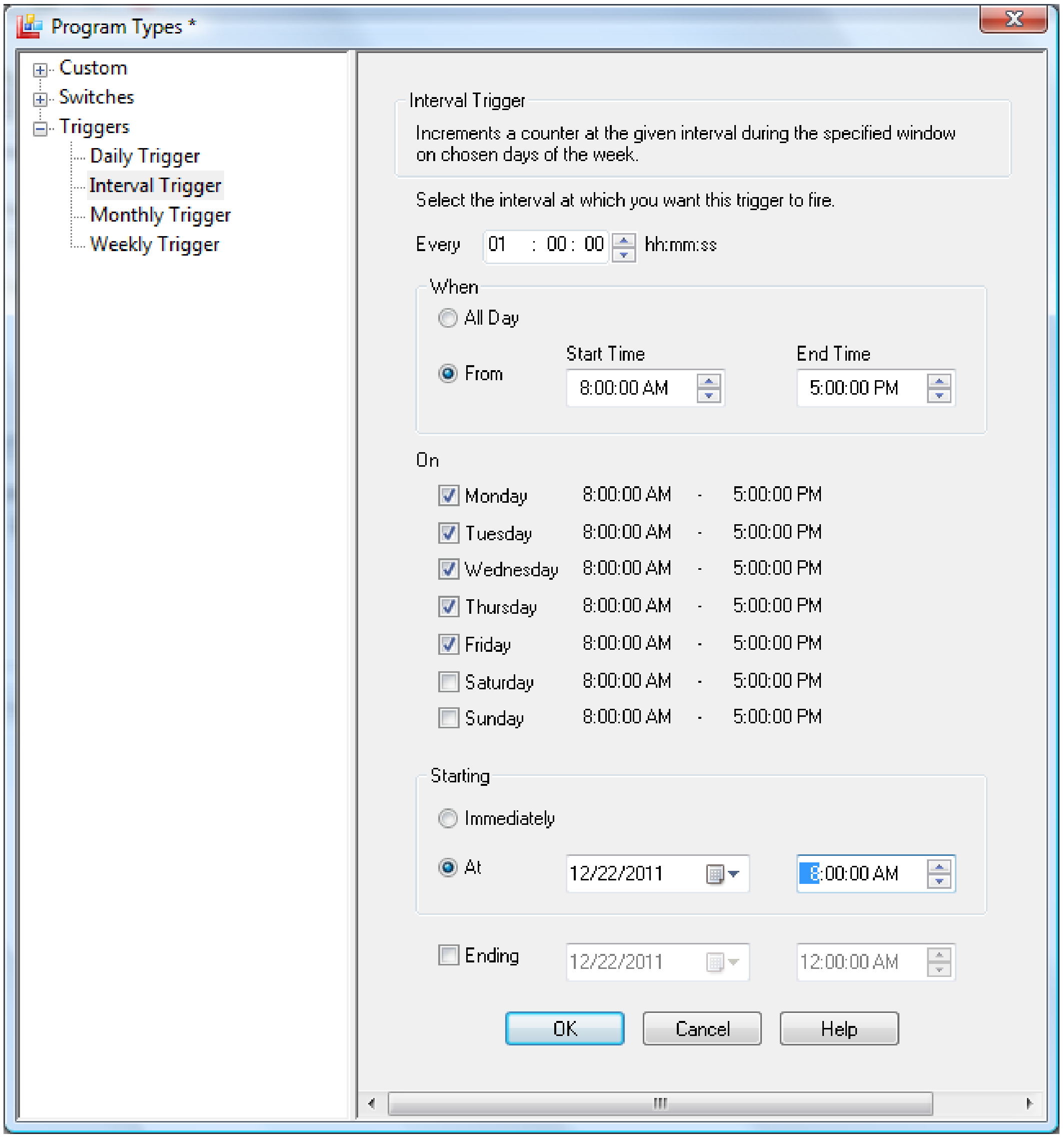
□ Simulate □ Disable Writes

Program Type: Custom

6. Go to the ***General*** tab and enter ***ProdDataTrigger*** in the ***Name*** field.

7. Click the browse button beside the ***Program Type*** field.

   The Select Program Type window will open.



Interval Trigger

Increments a counter at the given interval during the specified window on chosen days of the week.

Select the interval at which you want this trigger to fire.

Every 01 : 00 : 00 hh:mm:ss

When

○ All Day

● From

Start Time 8:00:00 AM

End Time 5:00:00 PM

On

☑ Monday	8:00:00 AM	-	5:00:00 PM
☑ Tuesday	8:00:00 AM	-	5:00:00 PM
☑ Wednesday	8:00:00 AM	-	5:00:00 PM
☑ Thursday	8:00:00 AM	-	5:00:00 PM
☑ Friday	8:00:00 AM	-	5:00:00 PM
□ Saturday	8:00:00 AM	-	5:00:00 PM
□ Sunday	8:00:00 AM	-	5:00:00 PM

Starting

○ Immediately

● At 12/22/2011 8:00:00 AM

□ Ending 12/22/2011 12:00:00 AM

8. Open the ***Triggers*** branch and select ***Interval Trigger***.

9. Set ***Every*** to ***01:00:00*** to trigger every 1 hour.

10. Select ***From*** in the When group.

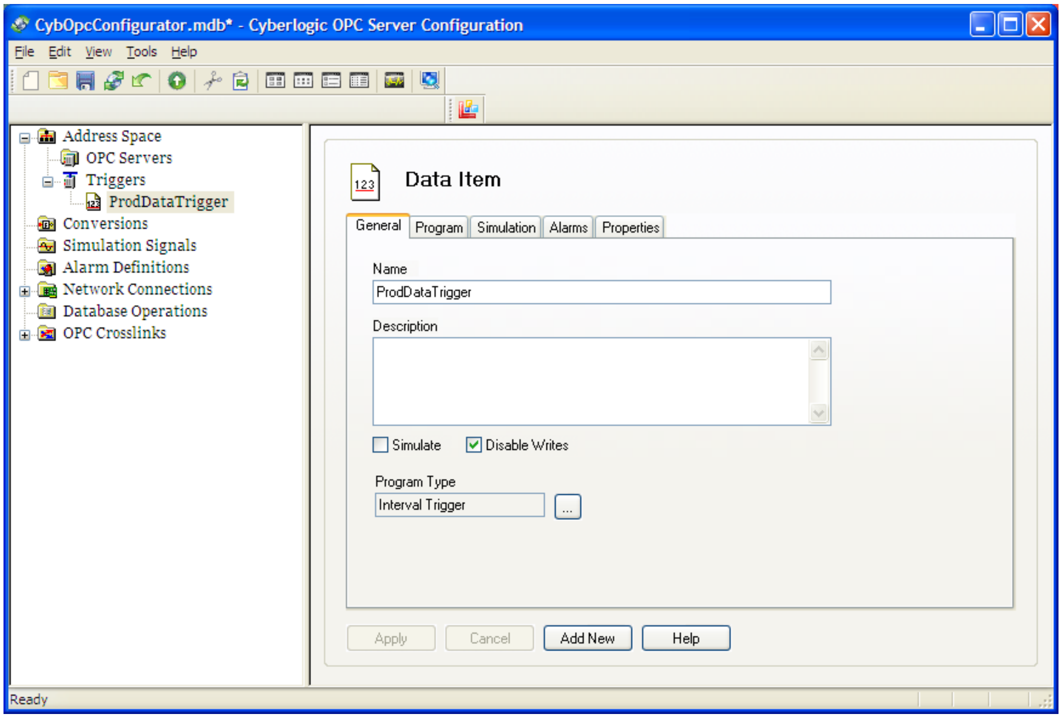
11. Set the ***Start Time*** to ***8:00:00 AM***.

12. Set the ***End Time*** to ***5:00:00 PM***.

13. Check the day boxes for ***Monday*** through ***Friday***.

14. Select ***At*** in the Starting group and set the date to today's date and set the time to ***8:00:00 AM***.

15. Click ***OK***.



Name: ProdDataTrigger

Description:

□ Simulate ☑ Disable Writes

Program Type: Interval Trigger

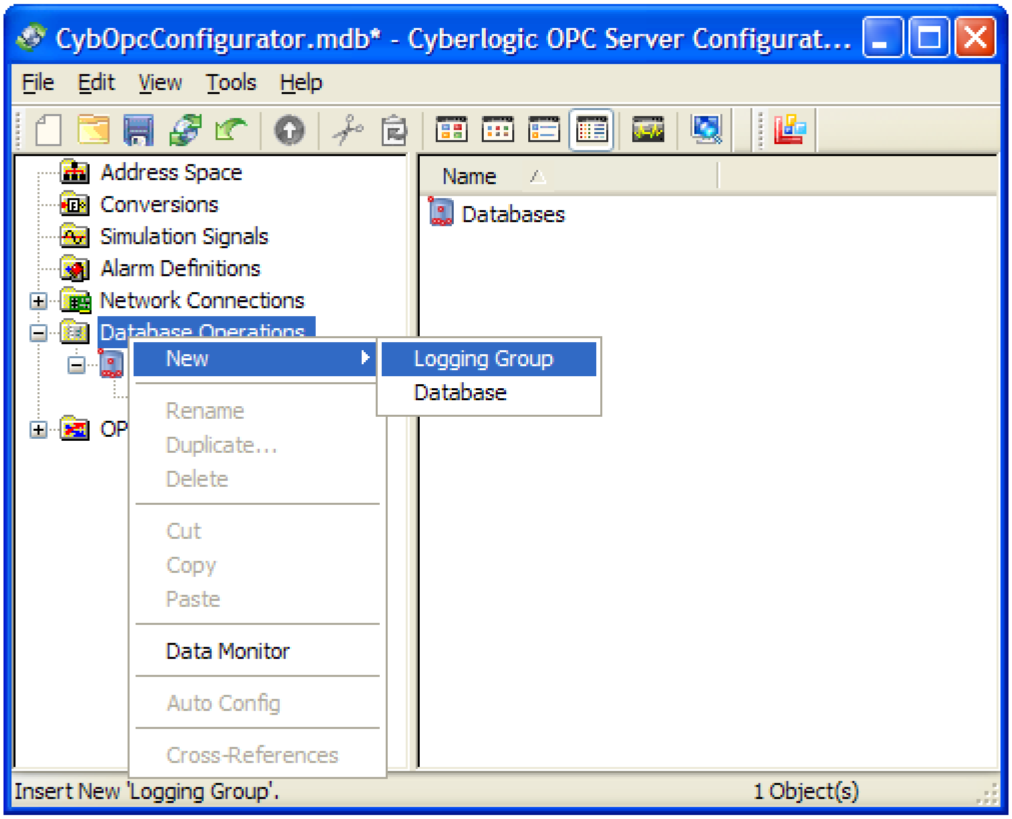
The Interval Trigger program, with the parameters you selected, will be applied to the data item. This will let you use this data item to trigger the logging operations every hour on the hour, between 8:00 a.m. and 5:00 p.m. on weekdays.

16. Click ***Apply***.

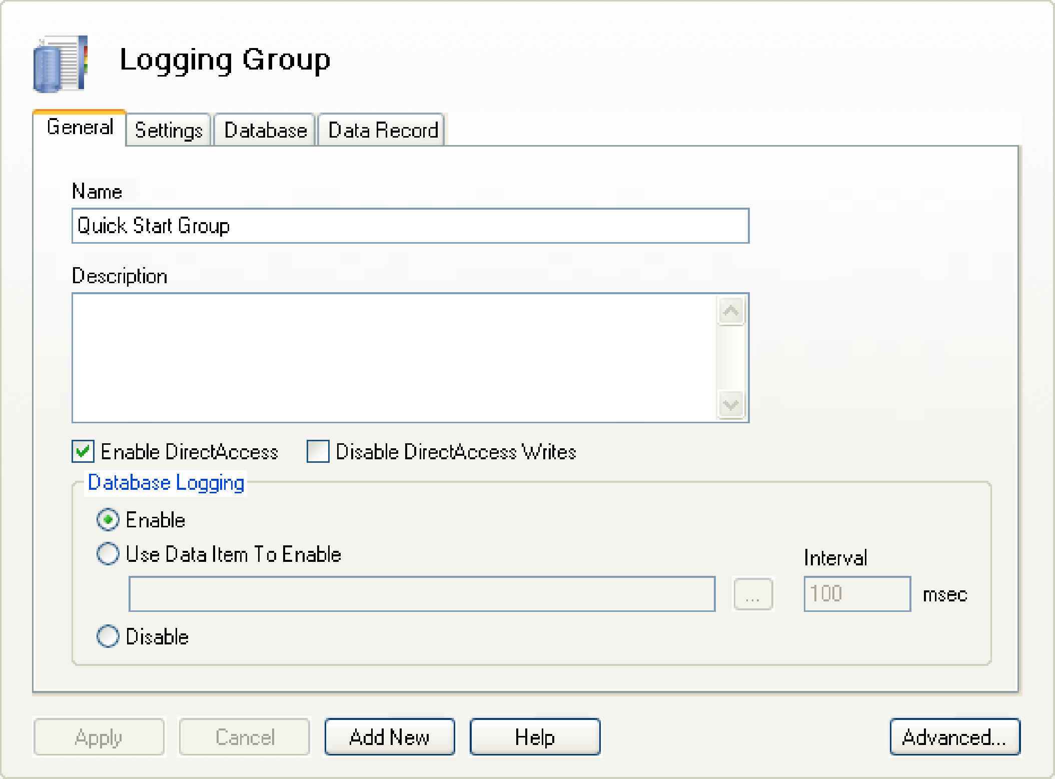
Next, we will Create the Logging Group.

Create the Logging Group

The logging group contains the enable and trigger criteria, as well as the definitions of the records that the logger will create, or the parameters for the stored procedure it will perform.



1. Right-click on the ***Database Operations*** branch and select ***New*** from the context menu, and then select ***Logging Group***.



Logging Group

General | Settings | Database | Data Record

Name

Quick Start Group

Description

☑ Enable DirectAccess □ Disable DirectAccess Writes

Database Logging

● Enable

○ Use Data Item To Enable

Interval 100 msec

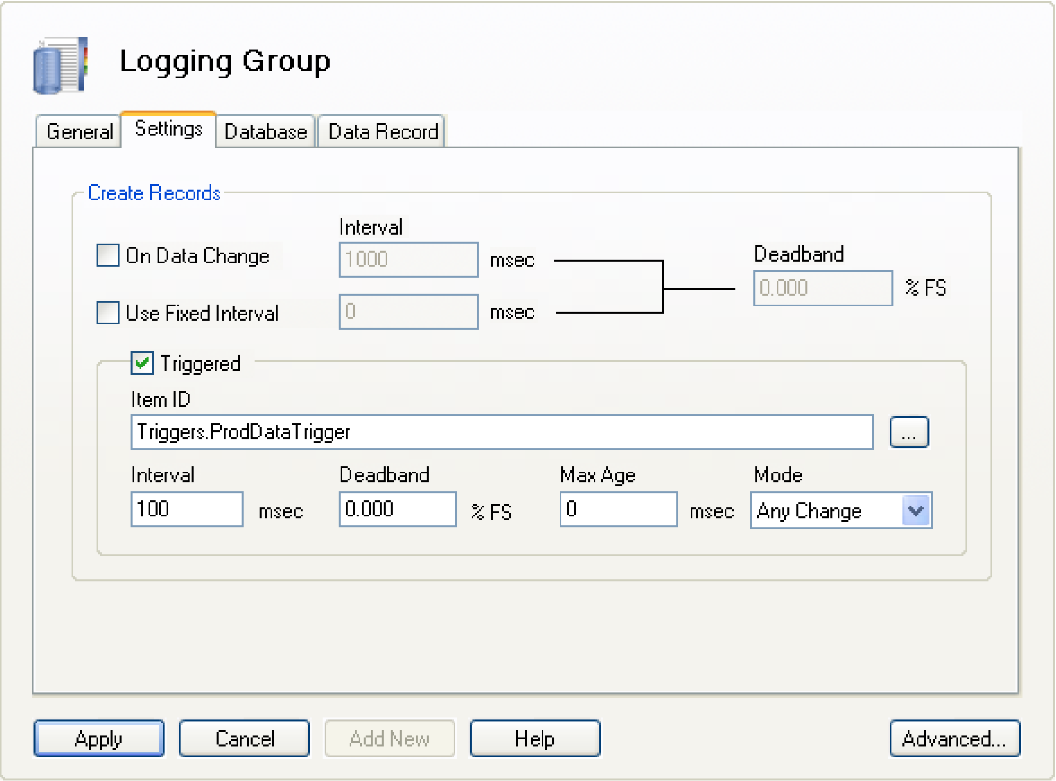
○ Disable

2. In the logging group's editing window, select the ***General*** tab.

3. Enter ***Quick Start Group*** in the ***Name*** field.

4. Under ***Database Logging***, select ***Enable***.

   This will keep the logging group always enabled.



Logging Group

Create Records

☐ On Data Change — Interval: 1000 msec

☐ Use Fixed Interval — 0 msec

Deadband: 0.000 % FS

☑ Triggered

Item ID: Triggers.ProdDataTrigger

Interval: 100 msec

Deadband: 0.000 % FS

Max Age: 0 msec

Mode: Any Change

5. Select the ***Settings*** tab.

6. Uncheck ***On Data Change*** and ***Use Fixed Interval***.

7. Check the ***Triggered*** box.

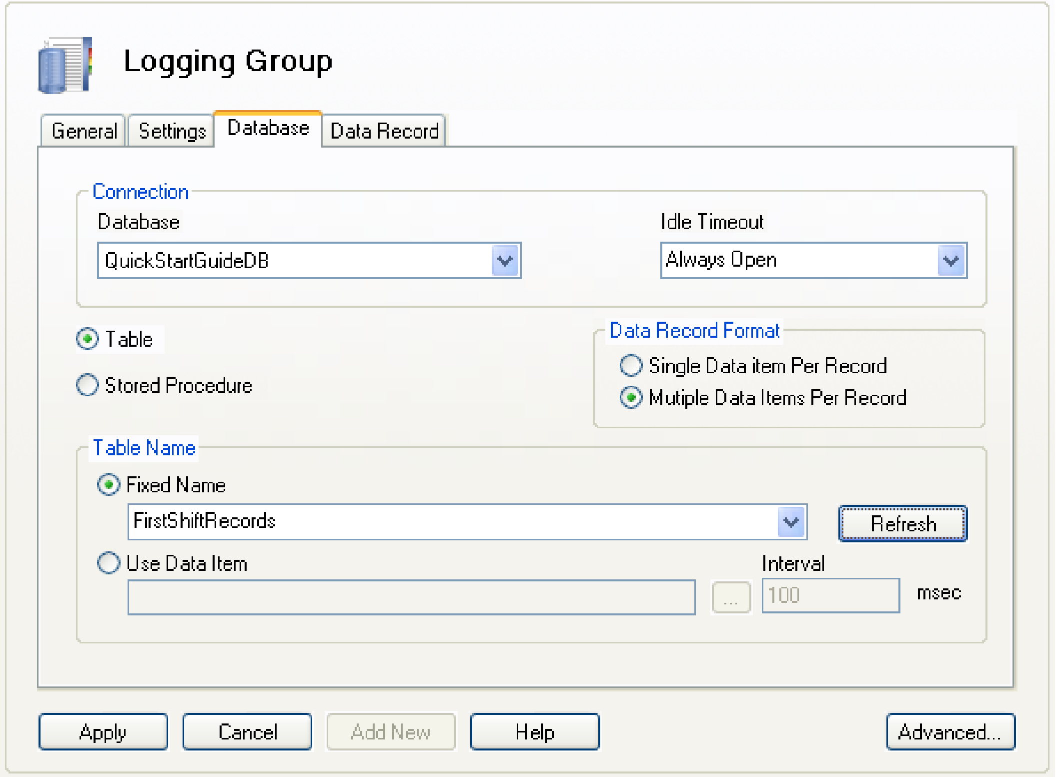
8. Type ***Triggers.ProdDataTrigger*** in the ***Item ID*** field. If you prefer, you can instead click the browse button to open a window that will allow you to browse for the item ID.

9. Enter ***100*** msec in the ***Interval*** field.

10. Enter ***0.000*** %FS in the ***Deadband*** field.

11. Enter ***0*** msec in the ***Max Age*** field.

12. Select ***Any Change*** in the ***Mode*** field.



Connection

Database: QuickStartGuideDB

Idle Timeout: Always Open

● Table

○ Stored Procedure

Data Record Format

○ Single Data item Per Record

● Mutiple Data Items Per Record

Table Name

● Fixed Name

FirstShiftRecords

○ Use Data Item

Interval 100 msec

13. Select the ***Database*** tab.

14. In the ***Connection*** group, select ***QuickStartGuideDB*** as the ***Database*** and ***Always Open*** as the ***Idle Timeout***.

15. For the ***Data Record Format***, select ***Multiple Data Items Per Record***.

16. In the ***Table Name*** group, select ***Fixed Name*** and then click the ***Refresh*** button. The editor will locate the tables within the database.

17. Under ***Fixed Name***, you can now select the desired table.

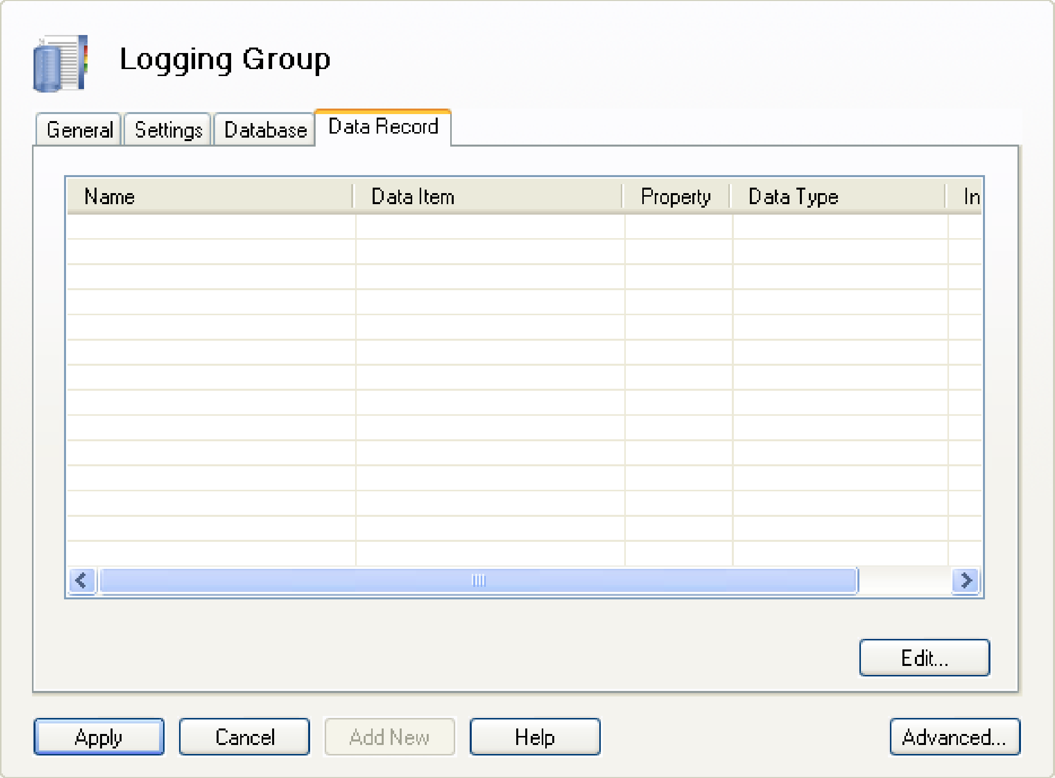
18. Next, you must define the data records you want the logger to create. To do this, go to Configure the Data Record.

Configure the Data Record

You will configure the data record on the logging group's Data Record tab. This is where you specify what data the logger should place in the record.

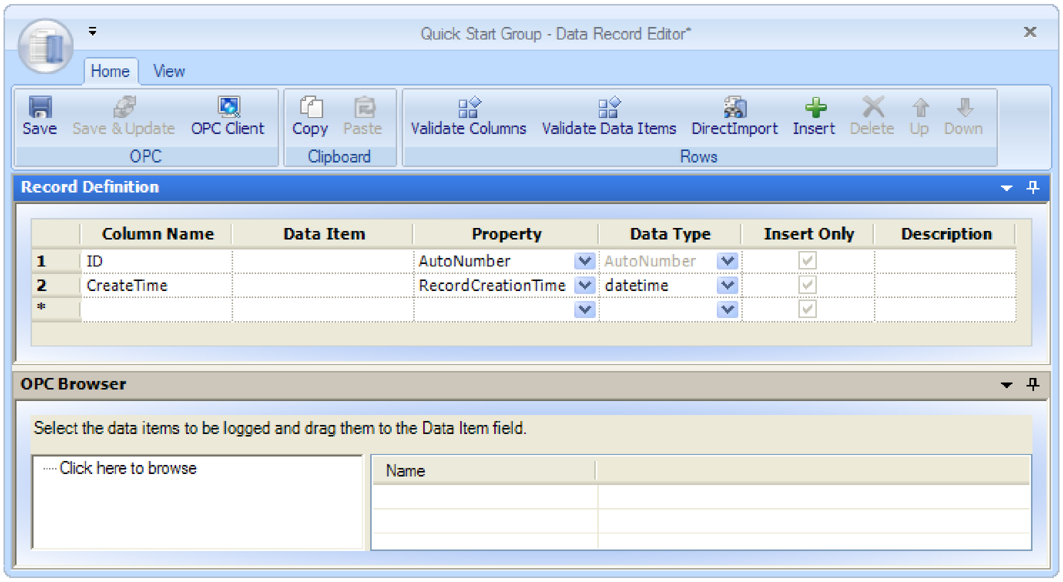
Note

Before you begin to configure the data records, you must create the OPC data items you want to log. Refer to Initial Preparation for more information about this requirement.



1. Select the ***Data Record*** tab.

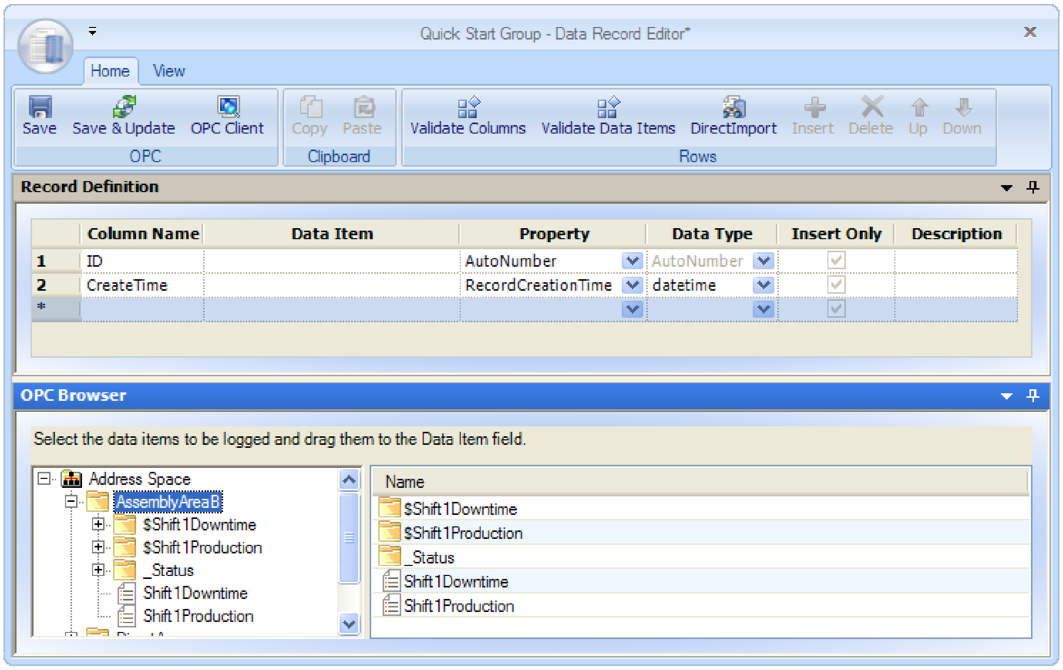
2. Click the ***Edit...*** button to open the Data Record Editor.



	Column Name	Data Item	Property	Data Type	Insert Only	Description
1	ID		AutoNumber	AutoNumber	☑	
2	CreateTime		RecordCreationTime	datetime	☑	
*					☑	

Select the data items to be logged and drag them to the Data Item field.

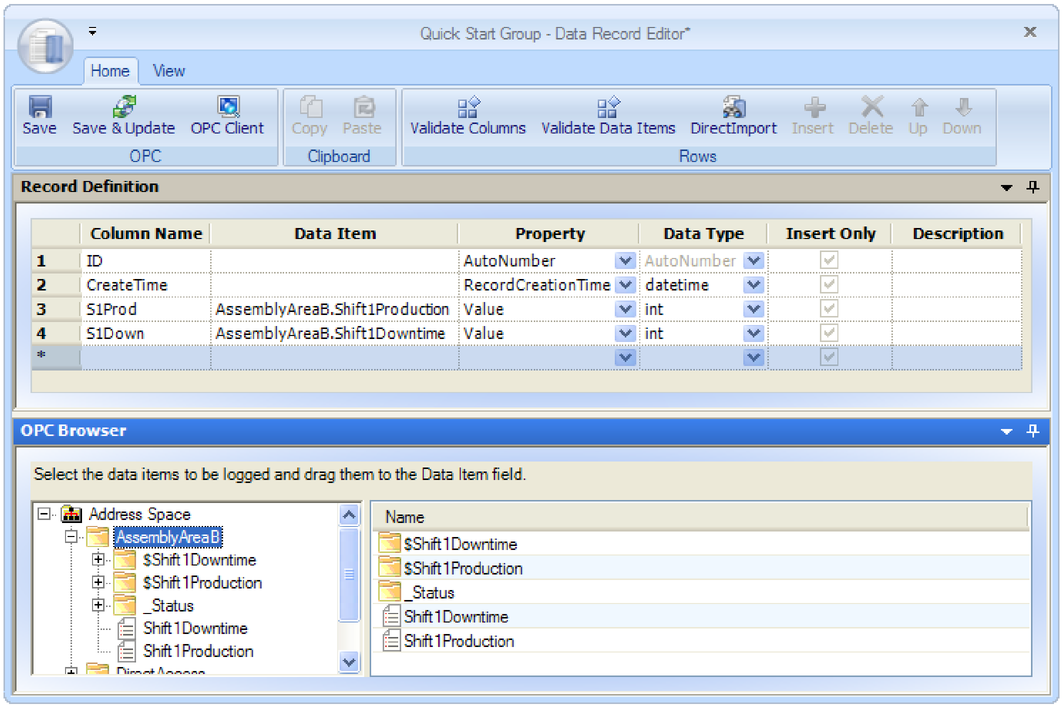
3. In the ***OPC Browser*** pane, click ***Click here to browse***.



	Column Name	Data Item	Property	Data Type	Insert Only	Description
1	ID		AutoNumber	AutoNumber	✓	
2	CreateTime		RecordCreationTime	datetime	✓	
*					✓	

Select the data items to be logged and drag them to the Data Item field.

4. In the Address Space tree, locate the data items you want to log. In this example, we will use *Shift1Production* and *Shift1Downtime*.



	Column Name	Data Item	Property	Data Type	Insert Only	Description
1	ID		AutoNumber	AutoNumber	✓	
2	CreateTime		RecordCreationTime	datetime	✓	
3	S1Prod	AssemblyAreaB.Shift1Production	Value	int	✓	
4	S1Down	AssemblyAreaB.Shift1Downtime	Value	int	✓	
*					✓	

Select the data items to be logged and drag them to the Data Item field.

5. Drag ***Shift1Production*** to the ***Data Item*** column and drop it there.

6. In the ***Property*** column, select ***Value***.

7. In the ***Data Type*** column, select ***int***.

8. Repeat steps 5 through 7 for ***Shift1Downtime***.

9. For the ***Column Name***, enter ***S1Prod*** for ***Shift1Production***, and ***S1Down*** for ***Shift1Downtime***.

10. Click ***Validate Data Items***. Data Logger will verify that the data items exist and that it can access them.

11. Click ***Validate Columns***. Data Logger will verify that the syntax of the columns is correct.

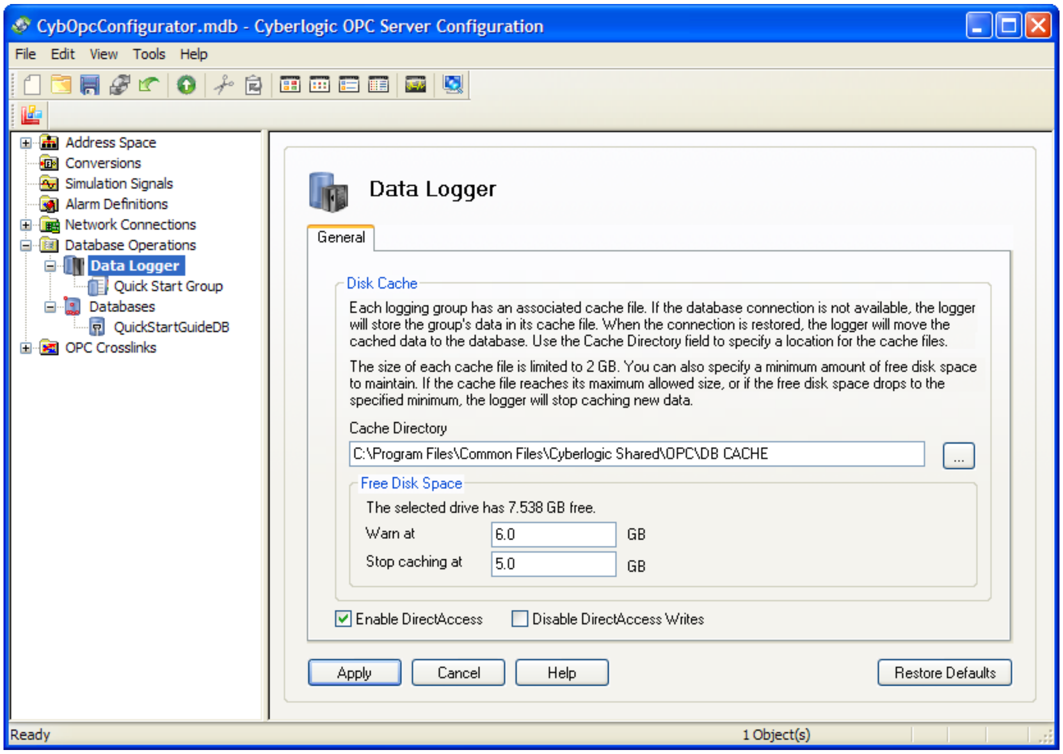
12. Click ***Save*** to save the record definition and then close the editor.

In the next step, you will Configure the Disk Cache.

Configure the Disk Cache

When Data Logger creates a record, it stores it in a memory cache until it can write it to the database. If the database is not available for an extended period of time, the logger will temporarily save the records in a disk cache. This prevents the memory cache from filling up, and also avoids the risk of losing data in case of a system crash or other failure.

For more information, refer to the section on the General Tab of the Data Logger branch.



Disk Cache

Each logging group has an associated cache file. If the database connection is not available, the logger will store the group's data in its cache file. When the connection is restored, the logger will move the cached data to the database. Use the Cache Directory field to specify a location for the cache files.

The size of each cache file is limited to 2 GB. You can also specify a minimum amount of free disk space to maintain. If the cache file reaches its maximum allowed size, or if the free disk space drops to the specified minimum, the logger will stop caching new data.

Cache Directory: C:\Program Files\Common Files\Cyberlogic Shared\OPC\DB CACHE

Free Disk Space

The selected drive has 7.538 GB free.

Warn at 6.0 GB

Stop caching at 5.0 GB

☑ Enable DirectAccess ☐ Disable DirectAccess Writes

1. Select the ***Data Logger*** branch.

2. Leave ***Cache Directory*** at the default value. This is where the logger will store the disk cache file.

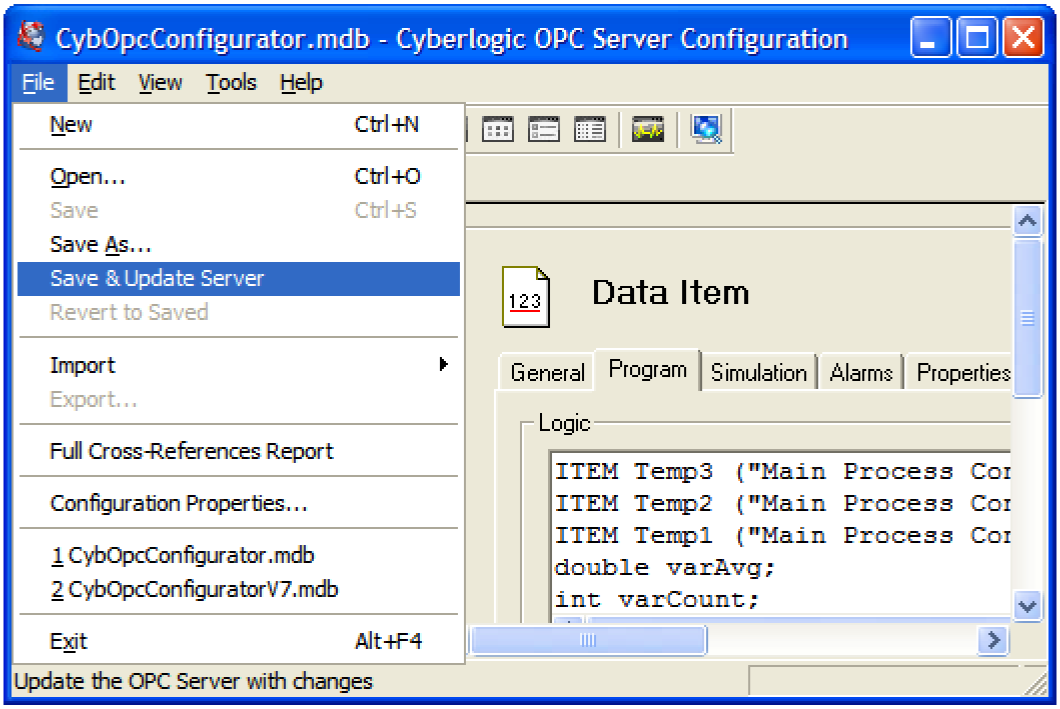
3. Under ***Free Disk Space***, note the amount of free space on the drive. Set the ***Warn at*** and ***Stop caching at*** fields to values that are appropriate for the amount of free space you need to maintain on the drive. For more information about these settings, refer to the section on the General Tab of the Data Logger branch.

4. Click ***Apply***.

Finally, we must Save the Configuration and Update the Server.

Save the Configuration and Update the Server

Whenever you edit the OPC server's configuration, you must save the configuration and update the server for the changes you have made to take effect. Otherwise, the server will still be running with the old configuration.



```
ITEM Temp3  ("Main Process Co
ITEM Temp2  ("Main Process Co
ITEM Temp1  ("Main Process Co
double varAvg;
int varCount;
```

1. Open the ***File*** menu and select ***Save & Update Server***.

2. Be sure to repeat this step every time you change the configuration. You can also do this by clicking the Save & Update Server button on the toolbar.

Your Data Logger configuration is complete. This concludes the Quick-Start Guide.

To learn more about the features of the server, refer to the Theory of Operation section. To learn more about configuration, refer to the Configuration Editor Reference. To learn more about verifying your configuration and troubleshooting tools, refer to the Cyberlogic OPC Server Help.

Configuration Editor Reference

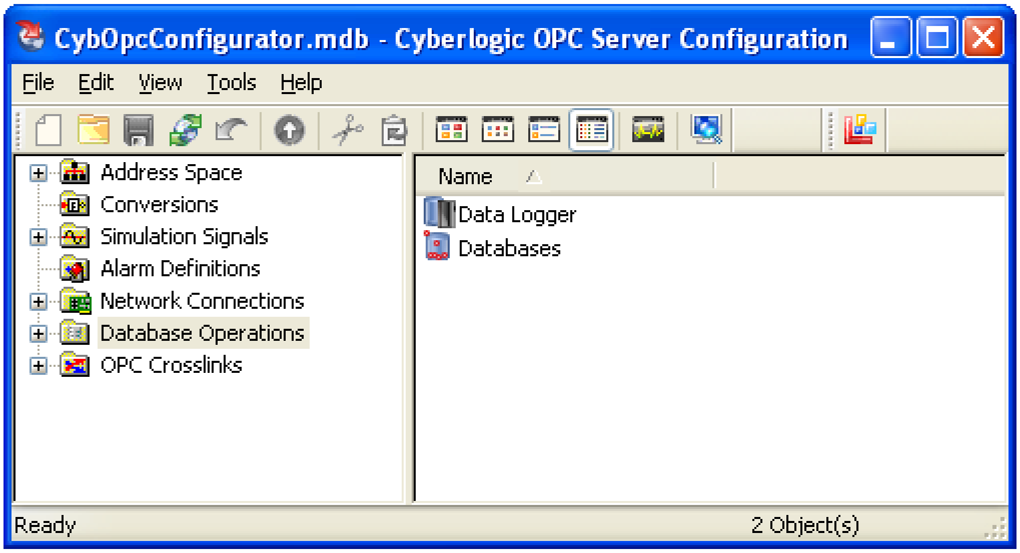
Before you can use the OPC server, you must configure it by using the OPC Server Configuration Editor. Data Logger users must configure the Database Operations and Network Connections trees. The remaining trees (Address Space, Conversions, Simulation Signals, Alarm Definitions and OPC Crosslinks) are optional features used by some systems.

This section describes the editor features that you will use to configure Data Logger. If you are a new user and want a procedure to guide you through a typical configuration session, refer to the Quick Start Guide.

Caution!

> After you edit the configuration, you must open the ***File*** menu and select ***Save & Update Server***, or click the ***Save & Update Server*** toolbar button, for the changes you have made to take effect. Otherwise, the server will still be running with the old configuration.

To launch the editor, open the Windows ***Start*** menu and navigate to ***Cyberlogic Suites***. From there, open the ***Configuration*** submenu and select the ***OPC Server*** menu item.



The left pane of the main workspace window includes the seven main configuration trees. The first part of this reference explains how to create and configure OPC Logging in the Database Operations tree. This includes two sections:

- Databases Branch
- Data Logger Branch

Following that is a very brief description of the other trees:

- Address Space
- Conversions
- Simulation Signals
- Alarm Definitions
- Network Connections
- OPC Crosslinks

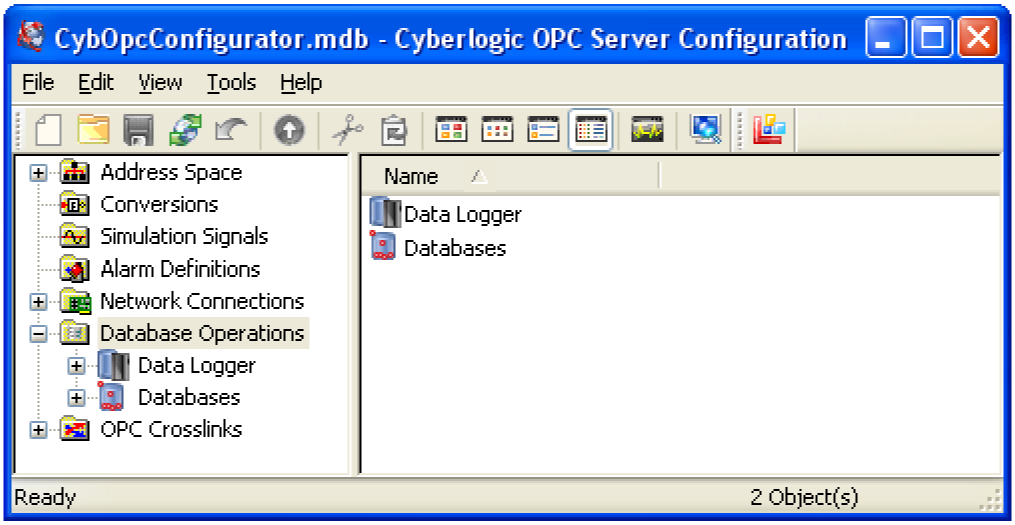
The last sections in this configuration editor reference cover important tips and techniques for:

- Saving and Undoing Configuration Changes
- Configuration Import/Export
- Editor Options

For a full discussion of these and other important configuration topics, please refer to the Cyberlogic OPC Server Help.

Database Operations

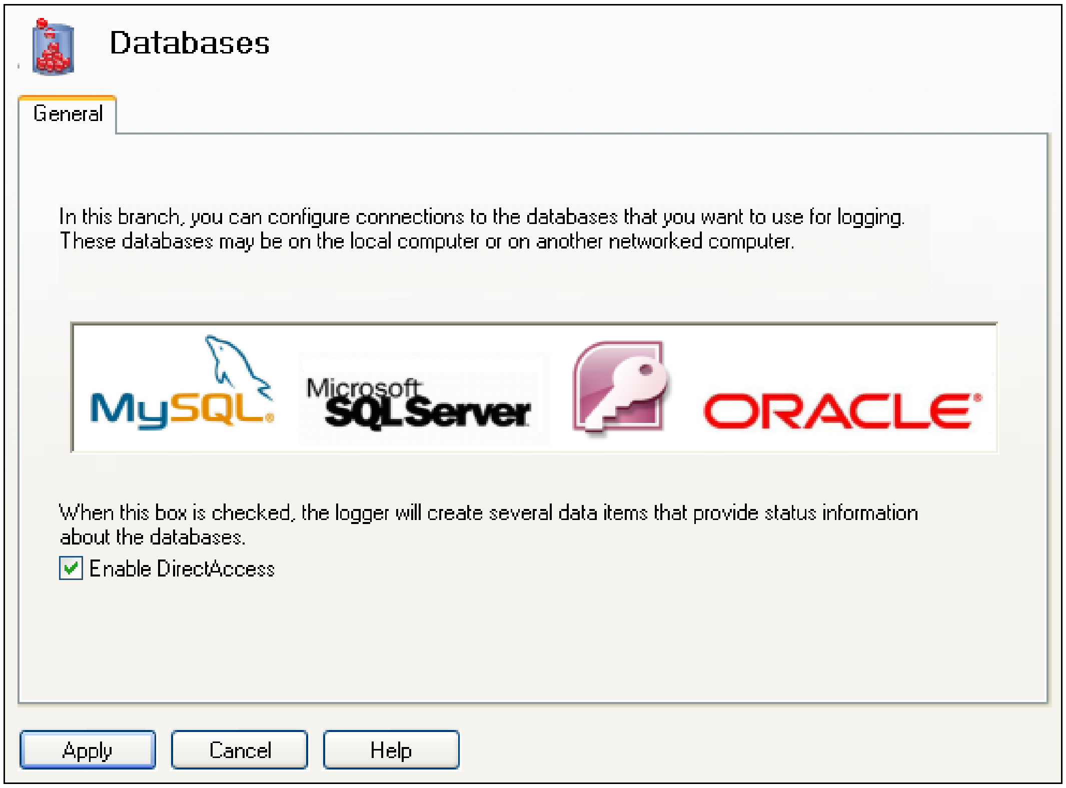
The Database Operations tree is where you will configure Data Logger. It contains two branches: Data Logger and Databases. In the Data Logger Branch, you will configure the logging operations that will store data in the databases that are defined in the Databases Branch.



Databases Branch

The Databases branch is where you will configure connections to the databases that you will use for logging.

General Tab



Databases

In this branch, you can configure connections to the databases that you want to use for logging. These databases may be on the local computer or on another networked computer.

When this box is checked, the logger will create several data items that provide status information about the databases.

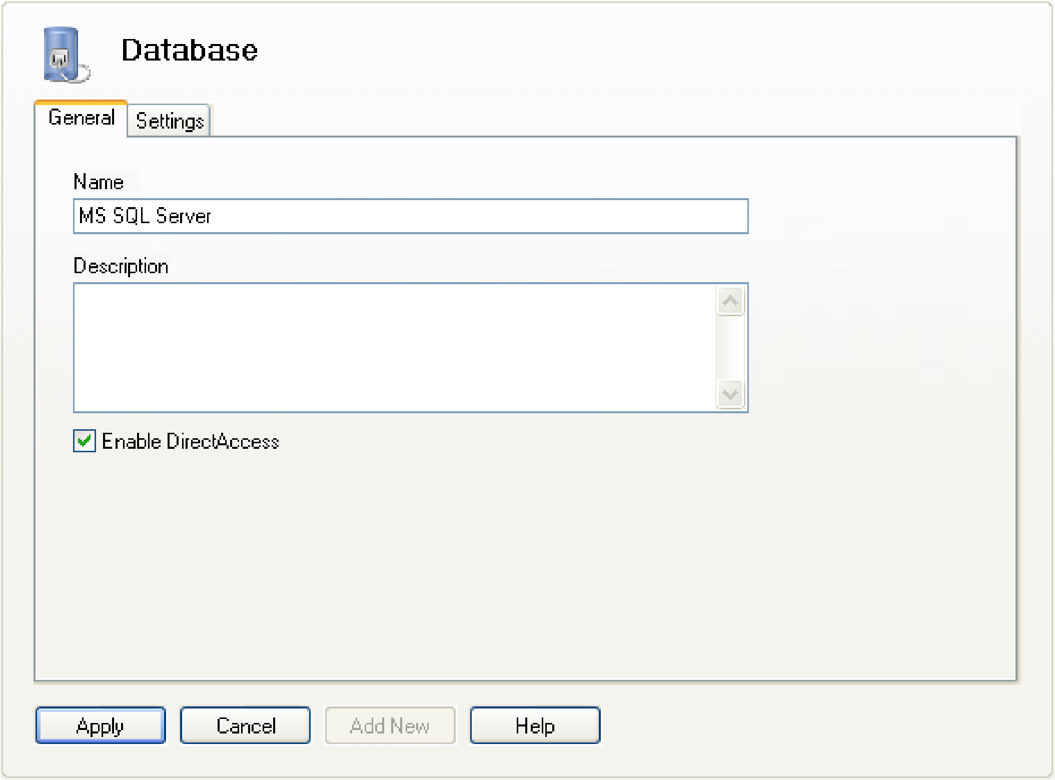
☑ Enable DirectAccess

Enable DirectAccess

When Enable DirectAccess is checked, the logger will create several status items related to the databases. These status items will be available to OPC clients, Math & Logic, and other features, just as are ordinary data items.

The default setting is *checked*, enabling DirectAccess.

Database Branch



Database

Name

MS SQL Server

Description

☑ Enable DirectAccess

Each database has a General tab and a Settings tab. The General tab is the same for all database types.

General Tab

Name

Enter a name for the database.

Description

Enter a description of the database or what it is used for. This field is optional.

Enable DirectAccess

When this box is active and checked, the logger will create several status items related to this database. These status items will be available to OPC clients, Math & Logic, and other features, just as are ordinary data items.

The default setting is *checked*, enabling DirectAccess.

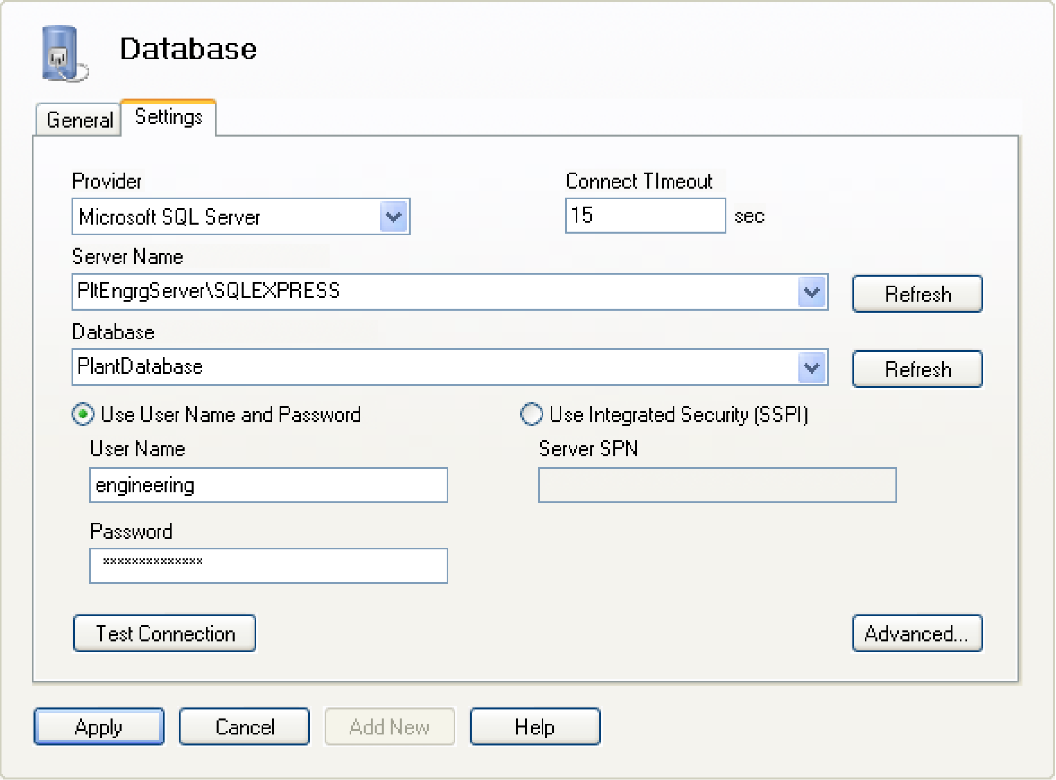
When this box is grayed-out, whether checked or unchecked, it means that DirectAccess for all database has been disabled at the Databases.

Settings Tab

The Settings tab is different for each type of database. The possibilities are:

- Microsoft SQL Server Database Settings
- Oracle Database Settings
- MySQL Database Settings
- Microsoft Access Database Settings

Microsoft SQL Server Database Settings



Provider: Microsoft SQL Server

Connect TImeout: 15 sec

Server Name: PltEngrgServer\SQLEXPRESS

Database: PlantDatabase

● Use User Name and Password

User Name: engineering

Password: **************

○ Use Integrated Security (SSPI)

Server SPN:

Provider

Choose the type of database you want. In this case, pick ***Microsoft SQL Server***.

Connect Timeout

Enter a length of time in seconds. This specifies how long the logger will wait to connect to the database before declaring a timeout failure.

Server Name

Use the drop-down list to select the computer and MS SQL Server instance where the database resides. Click ***Refresh*** to refresh the list of servers.

Database

Select the desired database from the drop-down list. Click ***Refresh*** to refresh the list of databases.

Use UserName and Password

Pick this radio selection to specify a user name and password.

User Name

Enter the user name that the database requires.

Password

Enter the password that the database requires.

The default is blank (no password).

Use Integrated Security (SSPI)

Pick this radio selection to access the database using the credentials of the account that is running the OPC server.

Server SPN

This field is optional. If you want to enter an SPN, the syntax is:

<service class>/<host>:<port>/<service name>

where *<service class>* and *<host>* are required, and *<port>* and *<service name>* are optional.

Example: *SqlServer/PltEngrgServer*

Refer to Microsoft's documentation for more information.

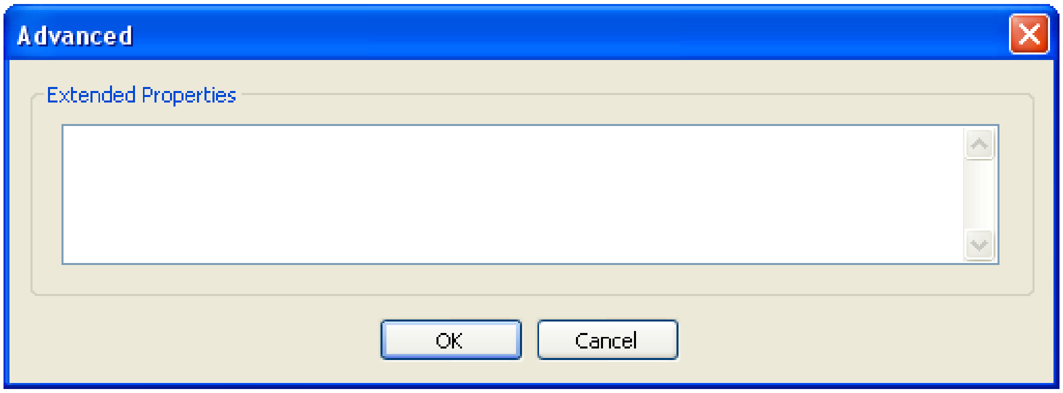
Test connection

After you complete the configuration, click this button. The logger will verify that it can connect to the specified database. For this test, it will impersonate the OPC server account.

Advanced…

Click this button to open the Advanced window.

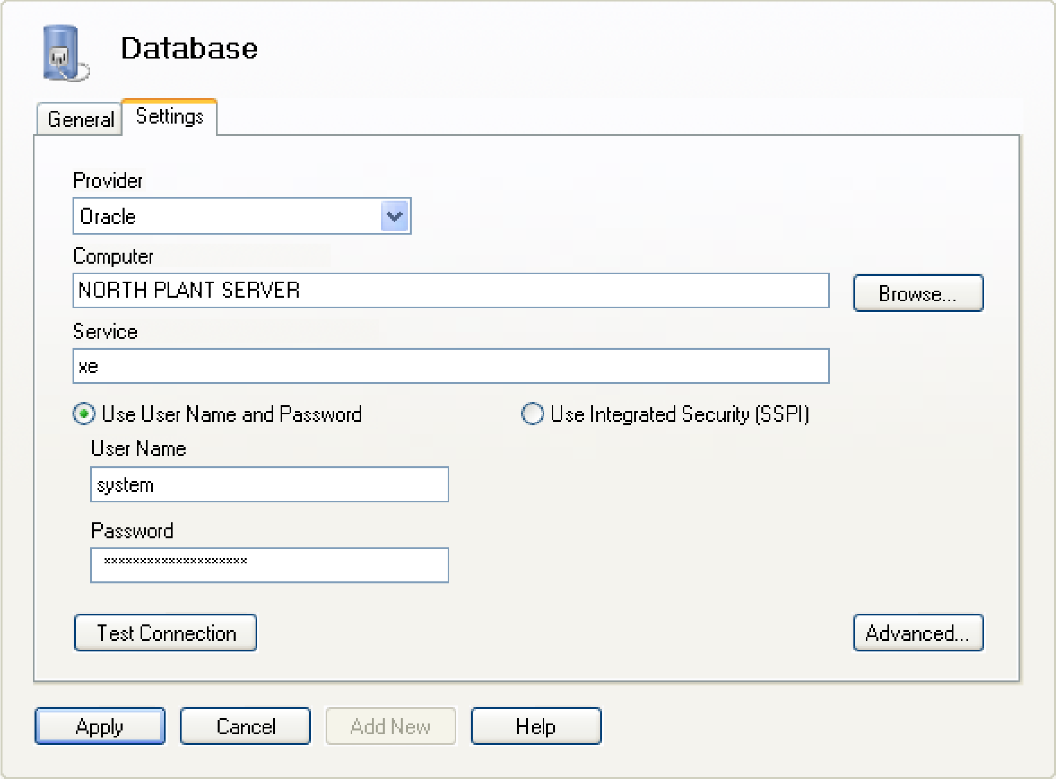
Advanced



Extended Properties

Here you can enter a string that will be appended to the connection string that the logger uses to access the database.

Oracle Database Settings



Provider: Oracle

Computer: NORTH PLANT SERVER

Service: xe

● Use User Name and Password ○ Use Integrated Security (SSPI)

User Name: system

Password: ********************

Provider

Choose the type of database you want. In this case, pick ***Oracle***.

Computer

Enter or browse for the name of the computer where the database resides.

Service

Enter the service name that identifies the edition of Oracle database software as it should be included in the connection string. (For example, *XE* stands for Oracle Express Edition).

This parameter is optional. If it is left blank, the logger will connect to the default service on the host computer.

Use UserName and Password

Pick this radio selection to specify a user name and password.

User Name

Enter the user name that the database requires.

Password

Enter the password that the database requires.

The default is blank (no password).

Use Integrated Security (SSPI)

Pick this radio selection to access the Oracle server using the credentials of the account that is running the OPC server.

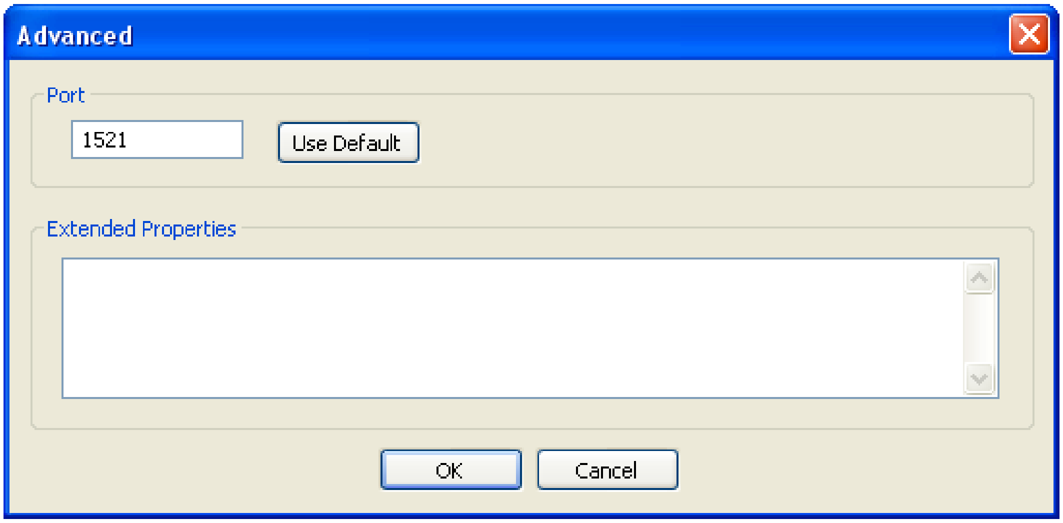
Test connection

After you complete the configuration, click this button. The logger will verify that it can connect to the specified database. For this test, it will impersonate the OPC server account.

Advanced…

Click this button to open the Advanced window.

Advanced



Port: 1521

Port

Oracle uses a single port for all databases. In this field, enter the port that Oracle is configured to use.

The default is *1521*.

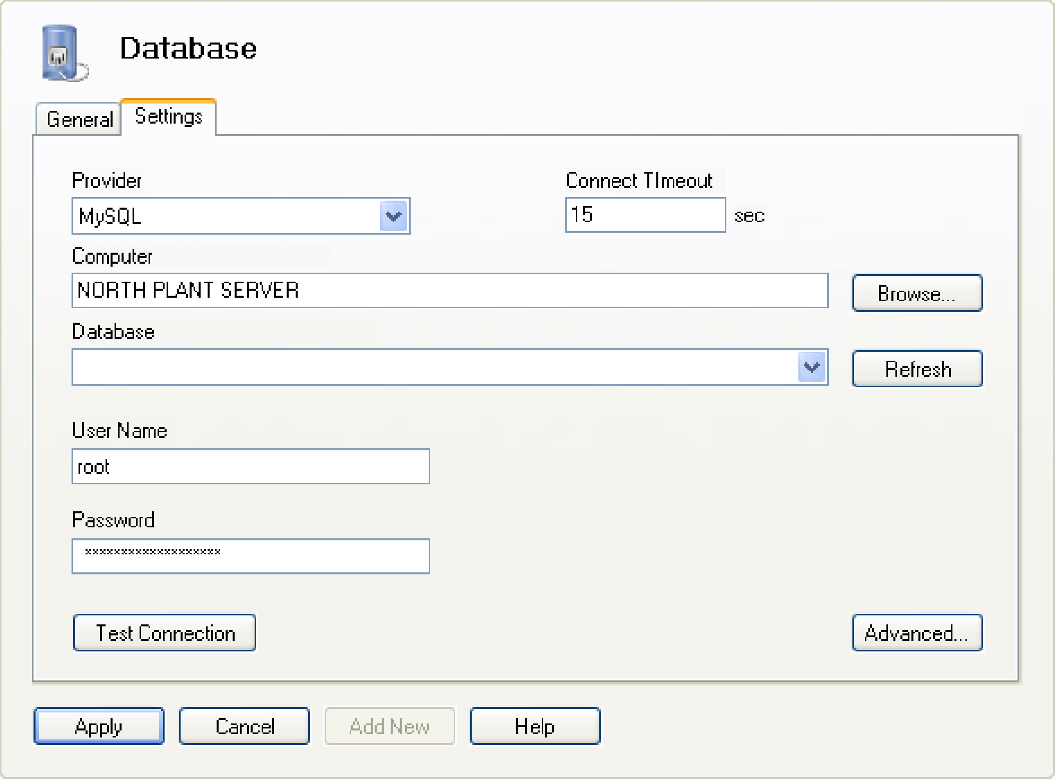
Use Default

Click this button to restore the Port setting to its default value.

Extended Properties

Here you can enter a string that will be appended to the connection string that the logger uses to access the database. Refer to the Oracle documentation for information on the allowed syntax.

MySQL Database Settings



Provider: MySQL

Connect TImeout: 15 sec

Computer: NORTH PLANT SERVER

Database:

User Name: root

Password: ******************

Provider

Choose the type of database you want. In this case, pick ***MySQL***.

Connect Timeout

Enter a length of time in seconds. This specifies how long the logger will wait to connect to the database before declaring a timeout failure.

Computer

Enter or browse for the name of the computer where the database resides.

Database

This drop-down list shows the MySQL databases that are available on the specified computer. Select the one you want to use.

Refresh

Click this button to refresh the Database list.

User Name

Enter the user name that the database requires.

Password

Enter the password that the database requires.

The default is blank (no password).

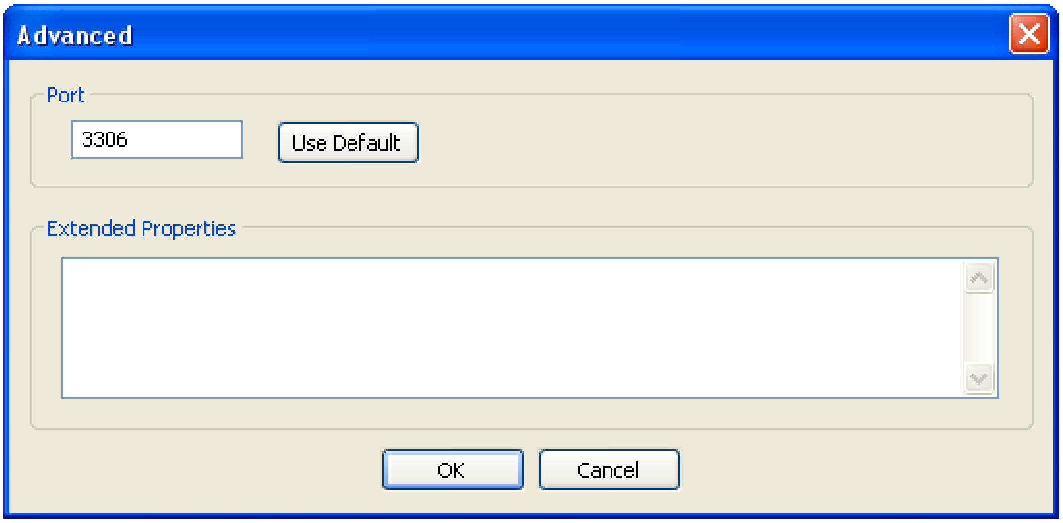
Test connection

After you complete the configuration, click this button. The logger will verify that it can connect to the specified database. For this test, it will impersonate the OPC server account.

Advanced...

Click this button to open the Advanced window.

Advanced



Port

3306

Port

Each MySQL database on a given system uses a different port for its connections. In this field, enter the port for the database you want to use.

The default is ***3306***.

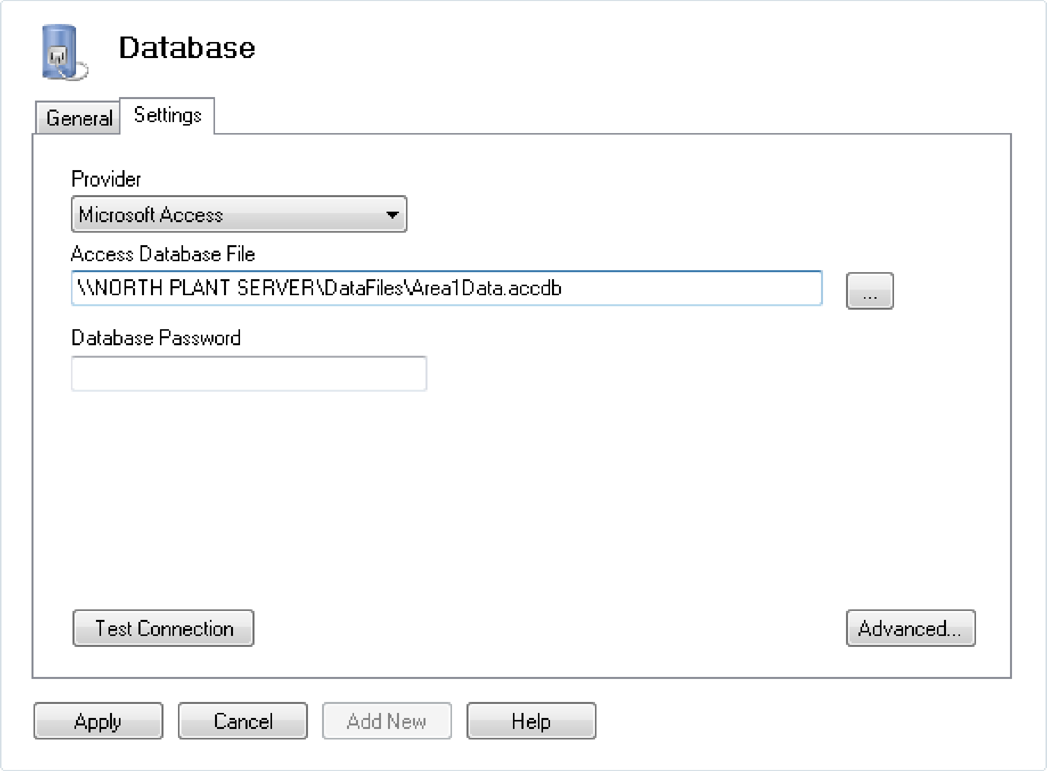
Use Default

Click this button to restore the Port setting to its default value.

Extended Properties

Here you can enter a string that will be appended to the connection string that the logger uses to access the database. Refer to the MySQL documentation for information on the allowed syntax.

Microsoft Access Database Settings



Provider: Microsoft Access

Access Database File: \\NORTH PLANT SERVER\DataFiles\Area1Data.accdb

Database Password:

Provider

Choose the type of database you want. In this case, pick ***Microsoft Access***.

Access Database File

Enter or browse for the path and file name of the specific Access database file you want to use. It may be on the local system or another system on the network.

Database Password

Enter the password if the database file is protected.

The default is blank (no password).

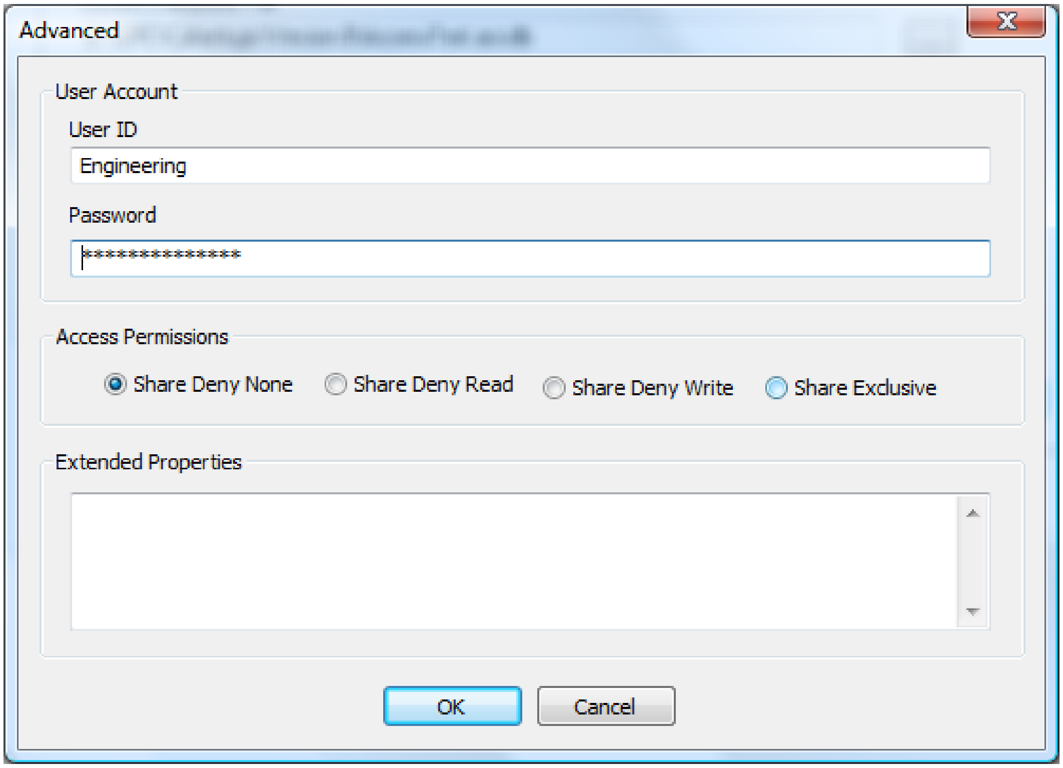
Test connection

After you complete the configuration, click this button. The logger will verify that it can connect to the specified database. For this test, it will impersonate the OPC server account.

Advanced…

Click this button to open the Advanced window.

Advanced



Advanced

User Account

User ID: Engineering

Password: ***************

Access Permissions

● Share Deny None ○ Share Deny Read ○ Share Deny Write ○ Share Exclusive

Extended Properties

User ID

Enter the user account ID that the database requires.

Unprotected database files use *Admin* as the default user account ID.

Password

Enter the password that the database requires.

The default is blank (no password).

Access Permissions

More than one user at a time may work with an Access database. This radio selection lets the Cyberlogic logger limit what others may do. The choices are:

- ***Share Deny None:*** Other users can read and write to the database.
- ***Share Deny Read:*** Other users can write, but cannot read.
- ***Share Deny Write:*** Other users can read, but cannot write.
- ***Share Exclusive:*** Other users can neither read nor write. The Cyberlogic logger has exclusive access.

Extended Properties

Here you can enter a string that will be appended to the connection string that the logger uses to access the database. Refer to Microsoft's documentation for information on the allowed syntax.

Data Logger Branch

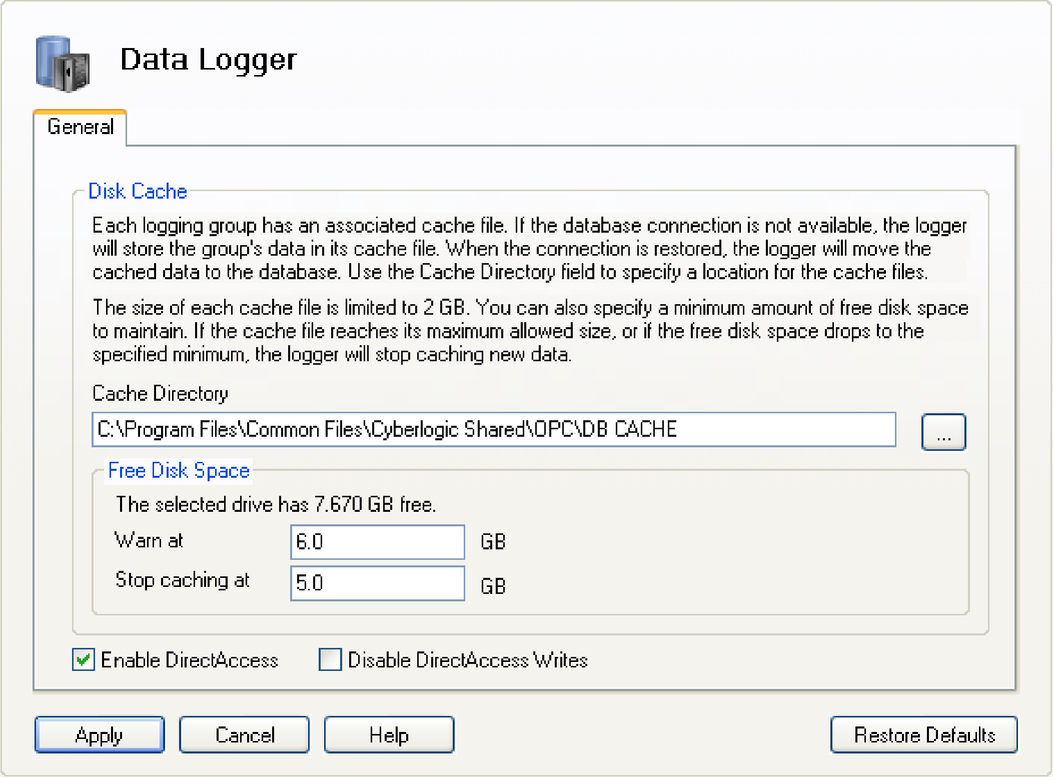
Within this branch, you configure the logging operations. To do this, you will create Logging Groups that define:

- What data will be logged
- When it will be logged
- Where it will be logged

In the main Data Logger branch, you configure the disk cache and DirectAccess permissions.

General Tab

Click on the Data Logger branch to open the General tab.



Disk Cache

Each logging group has an associated cache file. If the database connection is not available, the logger will store the group's data in its cache file. When the connection is restored, the logger will move the cached data to the database. Use the Cache Directory field to specify a location for the cache files.

The size of each cache file is limited to 2 GB. You can also specify a minimum amount of free disk space to maintain. If the cache file reaches its maximum allowed size, or if the free disk space drops to the specified minimum, the logger will stop caching new data.

Cache Directory: C:\Program Files\Common Files\Cyberlogic Shared\OPC\DB CACHE

Free Disk Space

The selected drive has 7.670 GB free.

Warn at: 6.0 GB

Stop caching at: 5.0 GB

☑ Enable DirectAccess ☐ Disable DirectAccess Writes

Normally, the logger puts the records into a memory cache, and then writes them immediately to the database. However, the database may be offline or busy, or there may be a communication problem. In that case, the records will accumulate in the memory cache until the database becomes available. The logger will then write the records to the database, clearing the memory cache as it does.

If the database remains unavailable for a long time, the memory cache may fill up. In addition, you may not want the data to remain in memory for a long period of time, risking data loss in case of a power failure. You can have the logger create a file on the local drive to hold the records temporarily, while the database is unavailable. This file is called the disk cache. Each logging group has its own memory cache and disk cache.

When the database becomes available, the logger will first take the records from the disk cache and write them to the database. After that, it will take the records from the memory cache, and likewise write them to the database.

Cache Directory

Enter or browse for the drive and directory where you want the logger to store the disk cache file.

The default is: *C:|Program Files|Common Files|Cyberlogic Shared|OPC|DB CACHE*

Caution! You can specify a drive that is located on another system. However, we strongly recommend that you choose a drive on the local system. In case of a network problem, a remote disk may not be available for caching, and this would result in loss of data.

Warning Level

The logger can try to preserve a minimum amount of free disk space by limiting the amount of space used for caching. In this field, you can specify the amount of free space at which you want the logger to declare a warning. To do this, it will set status item *DirectAccess.Database Operations.Data Logger._Status.DiskFreeSpaceWarning* to true.

The default warning level is *6 GB*.

Minimum Free Disk Space

Specify the minimum number of gigabytes of free disk space that you want to maintain on the drive that is used for the disk cache. The logger will limit the size of the disk cache files for all logging groups to try to maintain the required amount of free space.

The default minimum free disk space is *5 GB*.

Caution!

If the database remains unavailable long enough for the disk cache to fill up, then additional records must be held in the memory cache. This means they would be lost in case of a power failure or system crash. Furthermore, if the disk cache and memory cache both fill up, any new records will be discarded.

Keep these considerations in mind when you place limits on the cache size.

Note

In addition to the logger, other processes may use space on the disk. This may make it impossible for the logger to maintain the minimum free space, and may force it to reduce the disk cache size to zero. To avoid this, you should take corrective actions as soon as the disk free space warning level is flagged.

Enable DirectAccess

When this box is checked, the logger will create several status items related to OPC Logging. These status items will be available to OPC clients, Math & Logic, and other features, just as are ordinary data items. When the box is unchecked, DirectAccess is disabled for all logging groups, regardless of their individual settings.

The default setting is *checked*, enabling DirectAccess.

Disable DirectAccess Writes

When this box is checked, all status items become "read-only". OPC clients, Math & Logic programs, and other features will not be able to write to them. Checking this box disables DirectAccess writes for all logging groups, regardless of their individual settings.

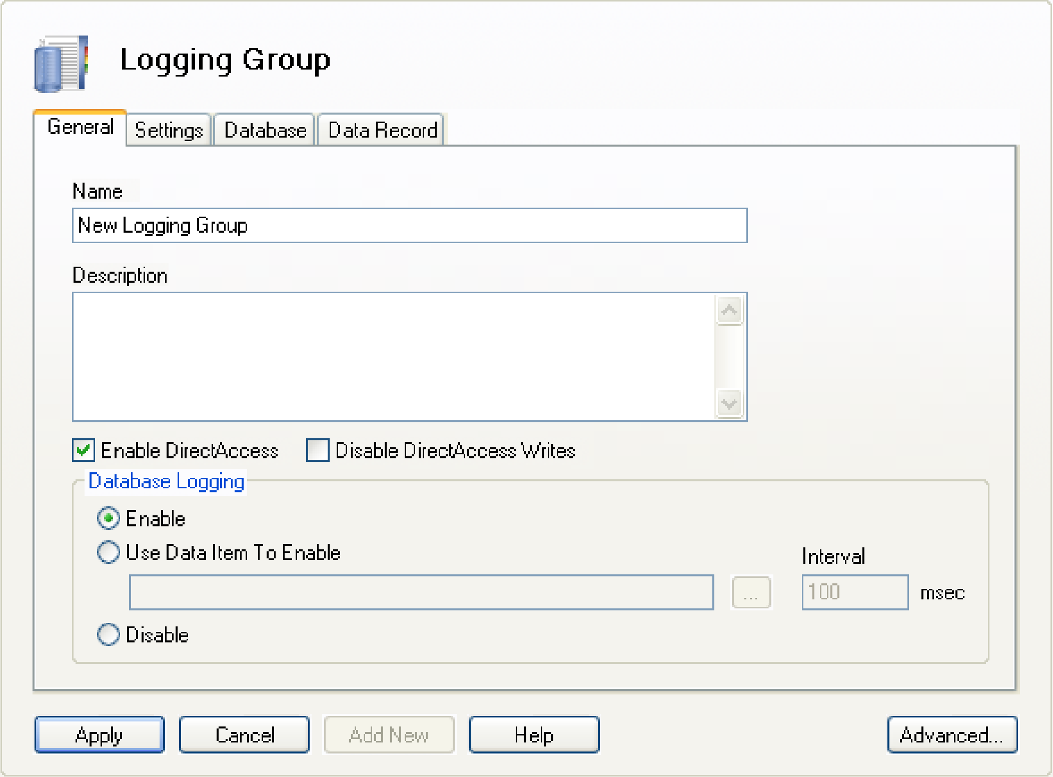
The default setting is *unchecked*, permitting read/write access.

Logging Groups

You create logging groups to specify how you want data to be logged. This is where you choose the database, define the data records, and set the criteria for enabling and triggering the logging.

To create a logging group, right click on the ***Data Logger*** branch. From the context menu, select ***New***, and then ***Logging Group***.

General Tab



Name: New Logging Group

Description:

☑ Enable DirectAccess □ Disable DirectAccess Writes

Database Logging

● Enable
○ Use Data Item To Enable
○ Disable

Interval: 100 msec

This tab lets you name and describe the logging group. It also lets you enable and disable DirectAccess, DirectAccess writes and logging.

Name

Enter a name for the logging group.

Description

Enter a description of the logging done in this group. This field is optional.

Enable DirectAccess

When this box is active and checked, the logger will create several status items related to this logging group. These status items will be available to OPC clients, Math & Logic, and other features, just as are ordinary data items.

The default setting is *checked*, enabling DirectAccess.

When this box is grayed-out, whether checked or unchecked, DirectAccess for all logging groups has been disabled at the Data Logger level.

Disable DirectAccess Writes

When this box is active and checked, all status items become "read-only". OPC clients, Math & Logic programs, and other features will not be able to write to them.

The default setting is *unchecked*, permitting read/write access.

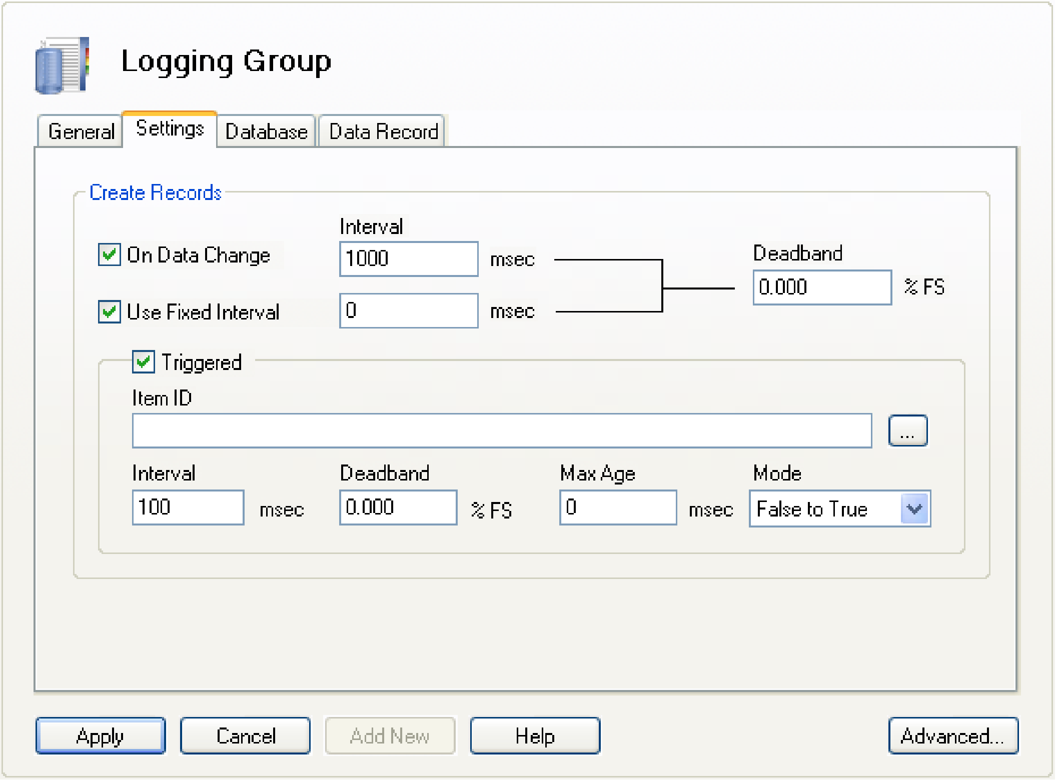
When this box is grayed-out, whether checked or unchecked, DirectAccess writes for all logging groups have been disabled at the Data Logger level.

Database Logging

This radio selection group lets you choose how to enable and disable logging. Your choices are:

- ***Enable:*** Logging is always enabled.
- ***Use Data Item To Enable***: Enter or browse for the item ID of a data item. Logging is enabled when the data item is true or nonzero, and disabled when it is false or zero. The ***Interval*** field specifies how often you want to check the value of the enable data item.
- ***Disable:*** Logging is always disabled. This is typically used during system debugging.

Settings Tab



☑ On Data Change — Interval 1000 msec

☑ Use Fixed Interval — 0 msec

Deadband 0.000 % FS

☑ Triggered

Item ID

Interval 100 msec — Deadband 0.000 % FS — Max Age 0 msec — Mode False to True

This tab lets you specify the conditions under which a new data record will be created. To create a record, logging must be enabled by the conditions on the General tab, and one or more of the criteria on this tab must be met.

On Data Change

Check this box to create a record when the value in at least one column of the record changes. The ***Interval*** field specifies how often the values in all record columns should be read and checked for data changes.

Use Fixed Interval

Check this box to create a record at regular intervals. The ***Interval*** field specifies how often you want to create a record.

Deadband

Specify the deadband that you want to use to decide if the value has changed. If the value changes by less than the deadband, *On Data Change* triggering will not create a new record. *Use Fixed Interval* will create a record, but the value logged will be the same as the old value.

Caution! The deadband is expressed as a percentage of the engineering unit range for the data item. This applies only to data items that have an OPC dwEUType of Analog. By default, data items are not Analog type. To use a deadband, you must apply a conversion to make the data item analog. Refer to the Cyberlogic OPC Server Help for details on how to use a conversion to do this.

Triggered

Check this box to create a record when a trigger data item changes state.

Item ID

Enter or browse for an item ID. The logger will create a record when this data item changes state.

Interval

Enter a value to specify how often you want to check the value of the trigger item.

Deadband

Specify the deadband that you want to use to decide if the trigger value has changed. If the value changes by less than the deadband, the change is ignored and the logger will not create a new record.

Caution! The deadband is expressed as a percentage of the engineering unit range for the data item. This applies only to data items that have an OPC dwEUType of Analog. By default, data items are not Analog type. To use a deadband, you must apply a conversion to make the data item analog. Refer to the Cyberlogic OPC Server Help for details on how to use a conversion to do this.

Max Age

Enter a maximum acceptable age for the data that is to be logged. If the values in the cache are younger than this limit, the logger will create the record using the cached values. If the values in the cache are older than this, the logger will request an update before it creates the record.

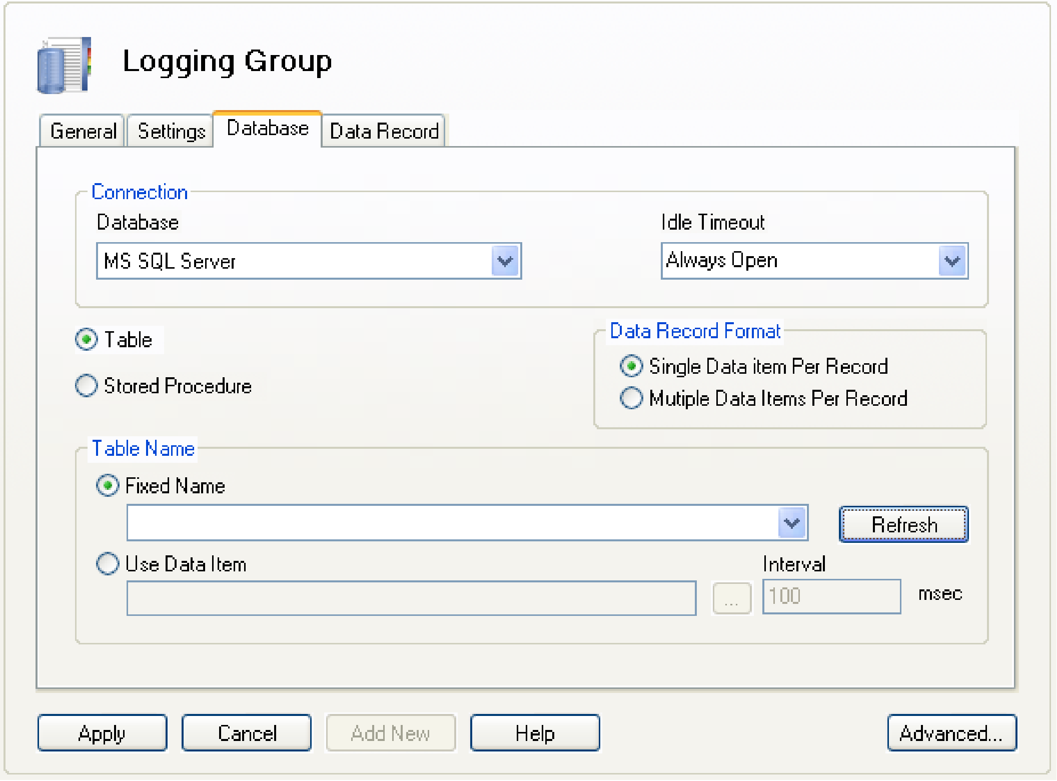
Mode

Specify the type of change that will trigger the creation of a record. Your choices are:

- False to true
- True to false
- Any change

Numeric data items are false when their value is zero, and true when their value is nonzero.

Database Tab



Database: MS SQL Server

Idle Timeout: Always Open

● Table

○ Stored Procedure

● Single Data item Per Record

○ Mutiple Data Items Per Record

● Fixed Name

○ Use Data Item

Interval: 100 msec

On this tab, you specify the database where the data will be logged. You also specify the table and record format to be used, or the stored procedure and its parameters.

Database

Select the database you want to use from the drop-down box. The choices will be the databases you configured in the Databases Branch.

Idle Timeout

Specify how long you want to wait after a transaction before closing the connection to the database. Your choices are:

- ***Always Open:*** The connection remains open indefinitely.
- ***Close After Transaction:*** The connection is closed after each record is logged.
- ***<Enter Number Of Seconds>:*** You must enter an idle timeout in seconds. The connection will be closed after the specified timeout. If the logger writes a record before the timeout expires, then the timeout is restarted.

Table / Stored Procedure

Choose whether the data record describes a row in a database table or the parameters passed to a stored procedure.

Record Format

Choose the format of the data records you want to create. Your choices are:

- ***Single Data Item Per Record:*** Each record contains various property values of one data item.
- ***Multiple Data Items Per Record:*** Each record contains property values for more than one data item.

Table Name / Procedure Name

The name of this radio selection is called *Table Name* if you selected *Table*, and it is called *Procedure Name* if you selected *Stored Procedure*. There are two choices:

- ***Fixed Name:*** Enter a table or stored procedure name or browse for the desired table or stored procedure.
- ***Use Data Item:*** Enter an item ID or browse for the desired data item. This item will contain the name of the table or stored procedure you want to use. Enter an ***Interval***, in milliseconds. This specifies how often you want the logger to check the data item for an updated table or stored procedure name.

When you specify the table or procedure name, you will usually just enter the name. However, advanced users can use a dotted notation to override the database and schema information. The details of the syntax depend on the type of database.

- *MS SQL:* [[Database.]Schema.]Table

  Example: MyDB.dbo.Table1

- *Oracle:* [UserName.]Table

  Example: MyName.Table1

- *MySQL:* [Database.]Table

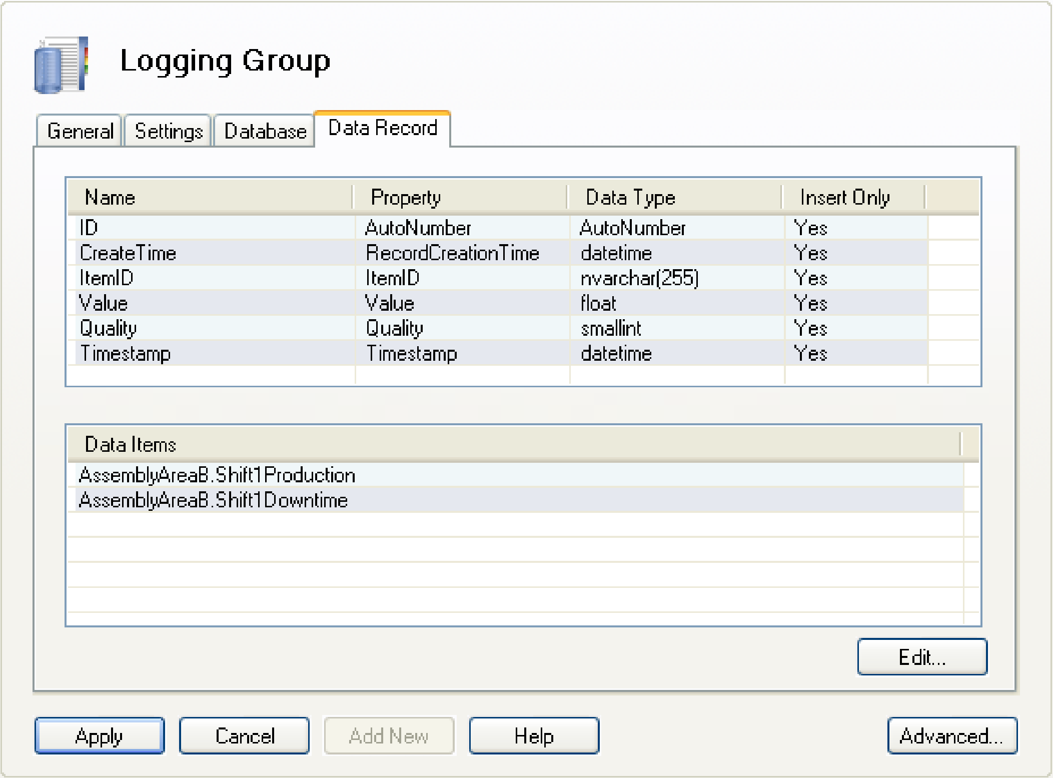
  Example: MyDB.Table1

Any sub-string that includes spaces must be enclosed with double-quotes, for example:

"My Database"."My Schema"."My Table"

This notation is primarily important if you use a data item to specify the table or procedure, but it also applies to fixed names.

Data Record Tab

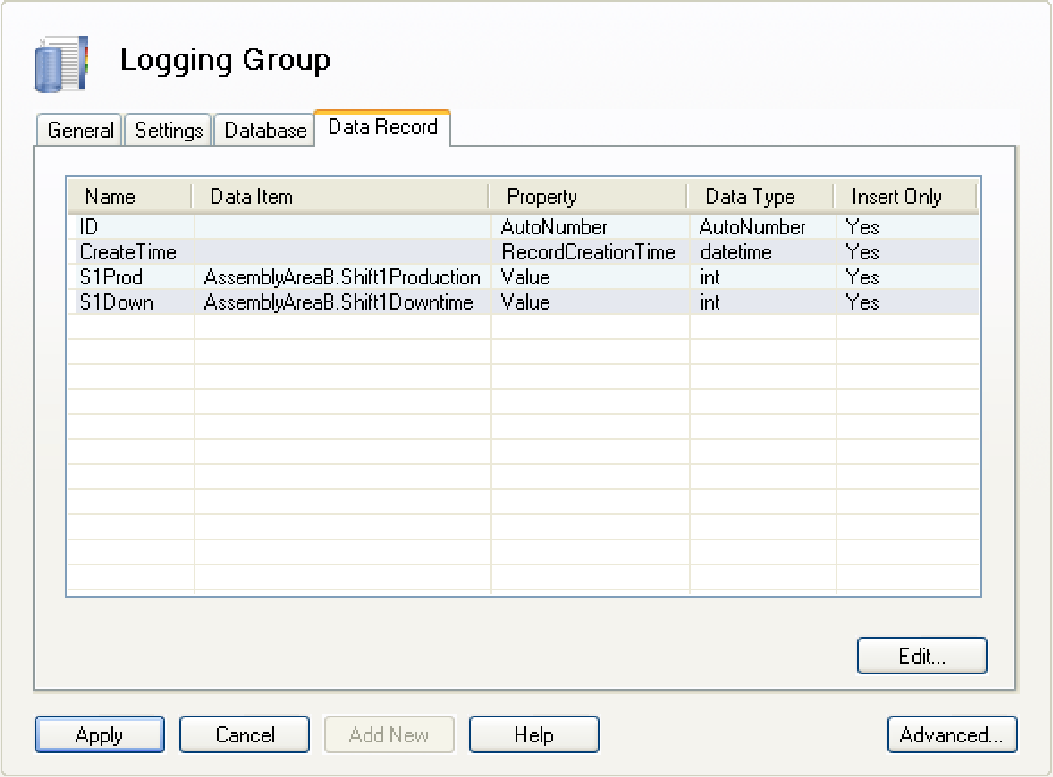


Logging Group

Name	Property	Data Type	Insert Only
ID	AutoNumber	AutoNumber	Yes
CreateTime	RecordCreationTime	datetime	Yes
ItemID	ItemID	nvarchar(255)	Yes
Value	Value	float	Yes
Quality	Quality	smallint	Yes
Timestamp	Timestamp	datetime	Yes

Data Items
AssemblyAreaB.Shift1Production
AssemblyAreaB.Shift1Downtime

On this tab, you can view the structure of the data records. The layout of the tab will vary, depending on the type of data record. The figure above is for *Single Data Item Per Record*, and the figure below is for *Multiple Data Items Per Record*.



Logging Group

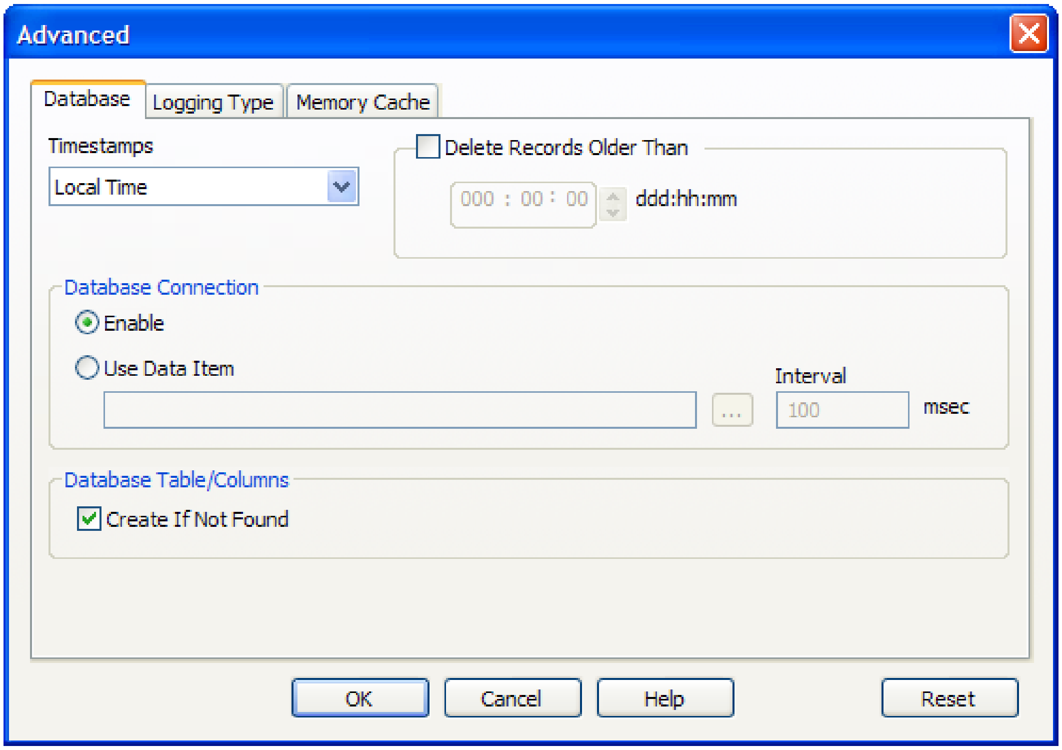
Name	Data Item	Property	Data Type	Insert Only
ID		AutoNumber	AutoNumber	Yes
CreateTime		RecordCreationTime	datetime	Yes
S1Prod	AssemblyAreaB.Shift1Production	Value	int	Yes
S1Down	AssemblyAreaB.Shift1Downtime	Value	int	Yes

If you want to modify the structure of the records, click ***Edit...*** to open the Data Record Editor.

Advanced Features

From any tab in the Logging Group, you can click ***Advanced...*** to open the Advanced settings editor. As the name implies, most users will not need to modify these settings, but rather can leave them at their defaults.

Database Tab



Timestamps: Local Time

□ Delete Records Older Than: 000 : 00 : 00 ddd:hh:mm

Database Connection

● Enable

○ Use Data Item

Interval: 100 msec

Database Table/Columns

☑ Create If Not Found

Timestamps

Select the desired time zone. This selection will apply to all columns and stored procedure arguments that have a property of Timestamp or RecordCreationTime. The choices are:

- Local Time
- UTC (Coordinated Universal Time)

The default is *Local Time*.

Delete Records Older Than

Check this box if you want the logger to purge records from the target database after they reach a specified age. You must then specify the desired maximum age in days, hours and minutes.

The default is *unchecked*.

Database Connection

This radio selection lets you control the connection to the database. Your choices are:

- ***Enable:*** The connection is always enabled.
- ***Use Data Item:*** Enter or browse for the item ID of a data item that will control the connection. If the data item is true or nonzero, the connection is enabled; if the data item is false or zero, the connection is disabled. Enter an ***Interval***, in milliseconds, to specify how often to check the data item's value.

The default selection is *Enable*.

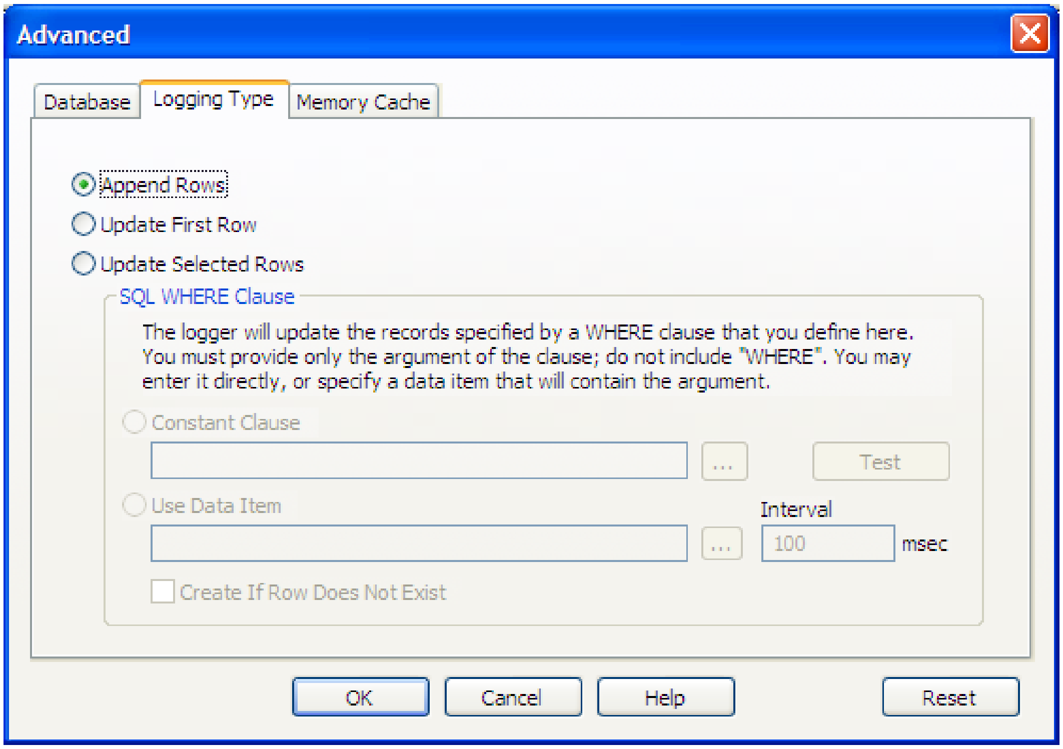
Create Table If Not Found

If this box is checked and the table specified for logging does not exist in the target database, the logger will create a new table to receive the records.

If this box is unchecked and the table does not exist, the logger will generate an error and will cache new data records. It will continue to cache records until the cache is full or the user manually creates the desired table.

The default is *checked*.

Logging Type Tab



● Append Rows
○ Update First Row
○ Update Selected Rows

SQL WHERE Clause

The logger will update the records specified by a WHERE clause that you define here. You must provide only the argument of the clause; do not include "WHERE". You may enter it directly, or specify a data item that will contain the argument.

○ Constant Clause

○ Use Data Item

Interval 100 msec

□ Create If Row Does Not Exist

This radio selection allows you to choose how records are added or modified in the database. It applies only to table operations and not to stored procedures . The choices are:

- ***Append Rows:*** The logger inserts a new row each time it writes to the database.
- ***Update First Row:*** The logger updates the first row in the database each time it writes new data.
- ***Update Selected Rows:*** The logger updates one or more rows based on the SQL WHERE clause each time it writes new data. You must provide an SQL WHERE clause that defines which rows you want to update.

The default is *Append Rows*.

SQL WHERE Clause

If you select *Update Selected Rows*, you must use this section to define the rows you want the logger to update. Your choices are:

- ***Constant Clause:*** Enter the WHERE clause you want the logger to use.
- ***Use Data Item:*** Enter or browse for the item ID of a data item that will contain the WHERE clause you want the logger to use. Enter an ***Interval***, in milliseconds, to specify how often to check the data item's value.

Test

After you select *Constant Clause* and enter the desired WHERE clause, click this button to test it.

Note

> The test may fail even if the WHERE clause is valid. For example, it may apply only to a table that does not yet exist because it will not be created until runtime. Therefore, if the test fails, be sure to look at the error message carefully to determine if the failure is really a problem.

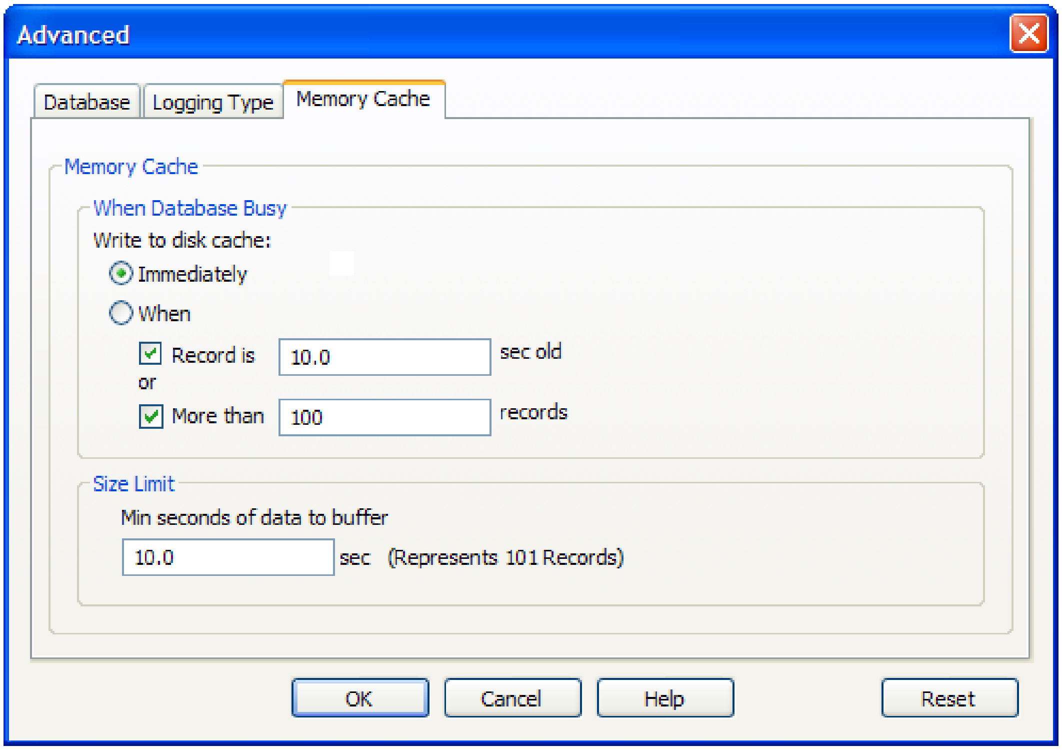
You cannot use the Test button if you configure the WHERE clause as *Use Data Item*, because the data item will not contain the WHERE clause until runtime.

Create If Row Does Not Exist

If this box is checked and no row in the target database table fits the WHERE clause criteria, the logger will append a new row to the table.

If this box is unchecked and no matching row exists, the logger will discard the data and generate an error.

Memory Cache Tab



Memory Cache

When Database Busy

Write to disk cache:

● Immediately

○ When

☑ Record is 10.0 sec old

or

☑ More than 100 records

Size Limit

Min seconds of data to buffer

10.0 sec (Represents 101 Records)

This tab allows you to limit the size of the memory cache and control how often the logger will move data from the memory cache to the disk cache.

When Database Busy

This radio selection lets you choose when to move data records to the disk cache when the database is busy. The choices are:

- ***Immediately:*** If the logger cannot write to the database, it will write new records to the disk cache as soon as it creates them.
- ***When:*** If the logger cannot write to the database, it will hold new records in the memory cache until one of the specified criteria is met. When that happens, the logger will move the records to the disk cache. You must choose at least one of the two criteria, and you may choose both:

  If you check ***Record is _____ sec old***, you must enter an age limit. The logger will keep the records in the memory cache for the specified time, and then will write them to the disk cache. The age limit you provide is in seconds, but may be specified to three decimal places. The default setting is *10 sec*.

  If you check ***more than ______ records***, you must enter the maximum number of records that you want to allow the logger to keep in memory. The logger will keep the records in the memory cache until this limit is reached. After that, it will begin writing records to the disk cache, beginning with the oldest. The default setting is *100 records*.

The default setting is *Immediately*.

Size Limit

Enter the minimum number of seconds of data you want to be able to store in the memory cache. The limit you provide is in seconds, but may be specified to three decimal places.

The logger will calculate the maximum number of records that could be generated in that amount of time, and will display that value as *Represents n records*. It will set that as the maximum number of records it will store in the memory cache.

Caution!

When the logger calculates the number of records to use as the limit, it assumes that it will log data at the maximum possible rate for the current configuration settings. For example, suppose the configuration calls for logging on data change, and checks the data every 100 ms. The logger will assume that the data will actually change every 100 ms, and will use that assumption to calculate the number of records that could be generated in the time interval you specify. The logger will limit the memory cache to that number of records.

If records are actually created at less than the maximum possible rate, then the memory cache will be able to accumulate data for a longer period of time than you specified.

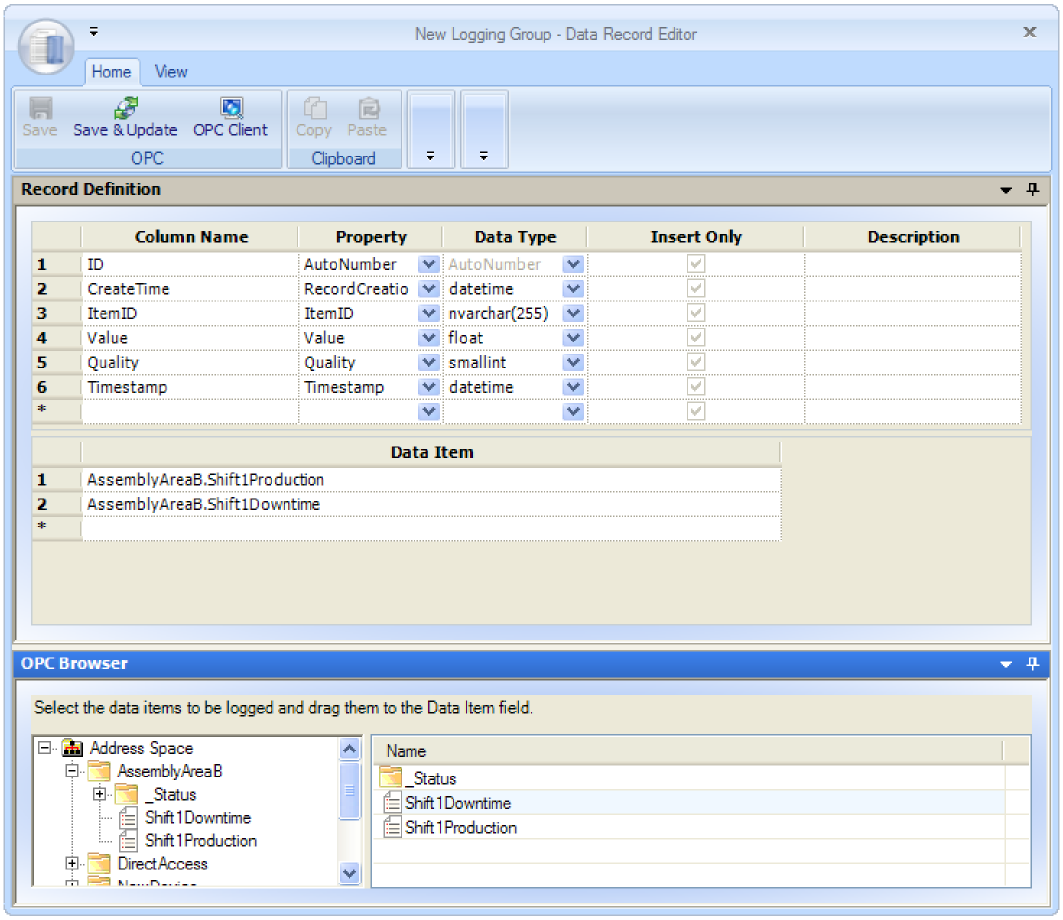
The default setting is *10 seconds*.

Data Record Editor

You use this editor to configure the structure of the records that the logger will produce and store in the database, or the parameters that it will pass to the stored procedure. The appearance of the editing screen will vary, depending on whether you have chosen the *single data item per record* or the *multiple data items per record* record type.

There are two main tabs: Home and View. The ***View*** tab has a single button that lets you change the layout of the information on the screen. Click the button to toggle between views, and pick the layout you prefer.

The editing operations are done on the ***Home*** tab.

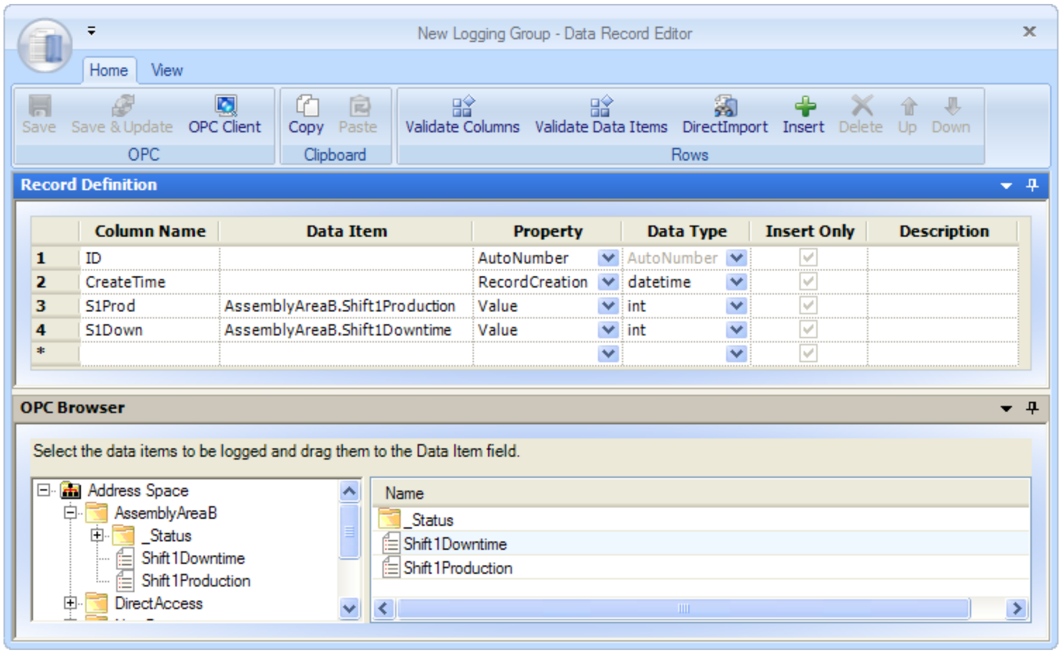


	Column Name	Property	Data Type	Insert Only	Description
1	ID	AutoNumber	AutoNumber	☑	
2	CreateTime	RecordCreatio	datetime	☑	
3	ItemID	ItemID	nvarchar(255)	☑	
4	Value	Value	float	☑	
5	Quality	Quality	smallint	☑	
6	Timestamp	Timestamp	datetime	☑	
*				☑	

	Data Item
1	AssemblyAreaB.Shift1Production
2	AssemblyAreaB.Shift1Downtime
*	

Select the data items to be logged and drag them to the Data Item field.

The Home tab for single data item per record logging



	Column Name	Data Item	Property	Data Type	Insert Only	Description
1	ID		AutoNumber	AutoNumber	☑	
2	CreateTime		RecordCreation	datetime	☑	
3	S1Prod	AssemblyAreaB.Shift1Production	Value	int	☑	
4	S1Down	AssemblyAreaB.Shift1Downtime	Value	int	☑	
*					☑	

Select the data items to be logged and drag them to the Data Item field.

The Home tab for multiple data items per record or stored procedure logging

Record Definition Pane

This area shows the details of each record that the logger will create. Although there are different layouts for the different logging types, the field definitions are the same for both layouts. The record table shows the columns in a record as rows with five or six configurable fields.

Column Name / Argument Name

Enter the name you want to use for the column/argument.

Data Item

This is the item ID of the data item to be logged.

If your logging format is *single data item per record*, the data items are shown in a separate list. The logger will use the record format you defined in the record table to create or update a separate record for each data item in the list.

If your logging format is *multiple data items per record* or *stored procedure*, the data items are included in the record table. The logger will use the record format to create records that each contain all of the selected data items or to create a list of parameters to pass to the stored procedure.

Property

Select the data item property that you want to log. For example, you might choose the value, timestamp or quality.

Data Type

Choose the format in which the data should be logged. Some examples are integer, datetime and string formats.

Note The available data types and the names used for them will depend on the selected database.

Insert Only

If this box is checked, the logger will not overwrite the data in this column when it updates an existing row. However, it will always write the data for all columns when it creates a new row.

This box will be grayed-out if the Logging Group refers to a stored procedure. It will also be grayed-out if you selected *Append Rows* on the Logging Type Tab. With that selection, the logger will never overwrite a row, so this setting is irrelevant.

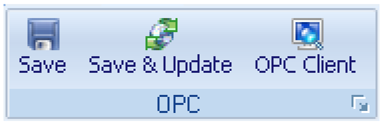
Description

Enter an optional description of the data in this column. This description is displayed here in the editor, but is not part of the data that will be logged.

OPC Browser Pane

Click on the left side of this pane to begin browsing for OPC data items. When you find an item you want to log, drag it to the desired Data Item field.

OPC Group



Save

Click ***Save*** to save the configuration without updating the OPC server. The server will continue to run with the old configuration.

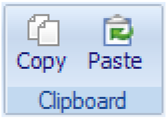
Save & Update

Click ***Save & Update*** to save the configuration and also load it into the OPC server. The server will now run with the new configuration.

OPC Client

Click ***OPC Client*** to open Cyberlogic's OPC client application. This client is useful for testing your server configuration. Refer to the Cyberlogic OPC Client discussion in the Validation & Troubleshooting section for more information.

Clipboard Group



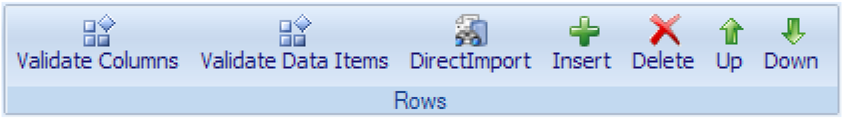
Copy

Select a row and click ***Copy*** to copy the row to the clipboard. You can then paste it as a new record. The keyboard shortcut Ctrl-C also works for this function.

Paste

After you've copied a row to the clipboard, click ***Paste*** to paste it as a new record.

Rows Group



Validate Columns

Click ***Validate Columns*** to have the editor check the column configuration for syntax errors.

Validate Data Items

Click ***Validate Data Items*** to have the editor verify that the item IDs actually point to existing data items.

If the data item is a DirectAccess item, the editor will highlight it in yellow. If it is invalid, it will highlight it in red.

DirectImport

If the table or stored procedure you want to use already exists in the database, you can import its configuration. To do this, click the ***DirectImport*** button. You can then browse and select the desired table or stored procedure.

Caution!

After doing a DirectImport, check the imported configuration for items with data types shown as *Undefined*. The logger does not handle all of the data types that the databases support. If your import includes an unsupported data type, items of that type will appear as an undefined type.

Insert

Click in a row and then click ***Insert*** to insert a row above the selection.

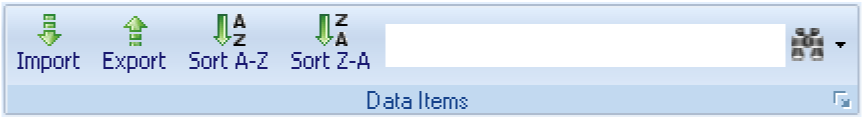
Up

Click in a row and then click ***Up*** to move the row up.

Down

Click in a row and then click ***Down*** to move the row down.

Data Items Group



This group is available only when you edit a logging group that has *Single Data Item Per Record* selected.

Import

Click ***Import*** to import a list of data items from a CSV file.

Export

Click ***Export*** to export the data items list to a CSV file.

Sort A-Z

Click Sort A-Z to sort the data items in ascending order.

Sort Z-A

Click Sort Z-A to sort the data items in descending order.

Find

Enter a search expression in the search field and then click the button. The data items list will show only the itemIDs that match the expression. The expression may contain wildcard characters, and is evaluated according to the following rules:

- A question mark (?) represents any single character.
- An asterisk (*) represents zero or more characters.
- An expression may contain both question marks and asterisks.
- Searches are not case-sensitive.

Click the ***down arrow*** and select ***Clear*** to restore the full list.

Address Space

The Address Space Tree allows you to create data items and define how they get their data from PLCs or other OPC servers. You can also organize the data items into a hierarchical structure that will make sense to you and to those who will configure the clients that connect to your server.

If you have a Premier or Enterprise Suite, this is where you can use Math & Logic to program trigger and enable conditions for the logger. The OPC Datacenter Suite does not include the full Math & Logic feature, but it does include pre-programmed Triggers and Switches that Data Logger can use.

Refer to the Cyberlogic OPC Server Help for a full discussion.

Conversions

The data that the server receives may be process values that are not expressed in engineering units. To simplify operations on the data, the Cyberlogic OPC Server lets you associate a conversion with each data item.

You can define many different conversions, and a number of data items can then use each conversion. As a result, you need not define the same conversion many times over.

Refer to the Cyberlogic OPC Server Help for a full discussion.

Simulation Signals

The server can simulate the data for each of the data items according to a predefined formula. This makes it easy to perform client-side testing without the need for a working connection to a PLC.

You can define many different types of simulation signals, and a number of data items can then use each such signal. As a result, you need not define the same simulation signal many times over.

The server can generate the following types of simulation signals:

- Read count
- Write count
- Random
- Ramp
- Sine
- Square
- Triangle
- Step

Each signal has parameters that define properties such as amplitude, phase and number of steps.

Refer to the Cyberlogic OPC Server Help for a full discussion.

Alarm Definitions

The Cyberlogic OPC Server supports the OPC Alarms and Events specification, permitting it to generate alarms based on the value of data items.

You may define many different alarm conditions, and a number of data items can then use each such condition. As a result, you need not define the same alarm condition many times over.

There are two categories of alarms: digital and limit. Digital alarms are normally used with Boolean data items, and limit alarms are normally used with numeric data items. However, both types of alarms may be used with either data type. Alarms cannot be used with string or array data items, or with bit fields larger than 64 bits.

Refer to the Cyberlogic OPC Server Help for a full discussion.

Note Configuring alarms is meaningful only if your client software also supports the OPC Alarms & Events specification. Consult your client software documentation to see what specifications it supports.

Network Connections

Network connections allow you to configure the networks you will use to communicate to network nodes. The network nodes may be PLCs, other OPC servers, or other devices that provide data to the OPC server.

Refer to the driver agent help files for a full discussion.

OPC Crosslinks

OPC Crosslinks allow you to transfer data from an OPC server or PLC to other OPC servers or PLCs. The data item you read from is called the crosslink input. You may write its value to any number of data items, and these are called crosslink outputs.

Refer to the OPC Crosslink Help for a full discussion.

Saving and Undoing Configuration Changes

The Cyberlogic OPC Server Configuration Editor keeps track of recent configuration changes. Until you save these changes, you can revert to the previously-saved configuration. The editor supports two types of save operations:

- The standard ***Save*** operation saves the changes without updating the server.
- The ***Save & Update Server*** operation saves the changes and also updates the server.

Caution! After you edit the configuration, you must open the ***File*** menu and select ***Save & Update Server***, or click the ***Save & Update Server*** toolbar button, for the changes you have made to take effect. Otherwise, the server will still be running with the old configuration.

Undoing Configuration Changes

To undo configuration changes and revert to the previously saved configuration, open the ***File*** menu and select ***Undo Changes***, or click the ***Undo Changes*** button on the toolbar.

Saving Configuration Changes Without Updating the Server

To save the configuration without updating the server, open the ***File*** menu and select ***Save***, or click the ***Save*** button on the toolbar. The changes will be saved but the server will continue to run with the old configuration.

Saving Configuration Changes and Updating Server

To save the configuration and update the server, open the ***File*** menu and select ***Save & Update Server***, or click the ***Save & Update Server*** button on the toolbar.

Configuration Import/Export

The Import/Export feature lets you:

- Export configuration data to a text file format
- Import configuration data from these exported files
- Import comma separated values (csv) files from other vendors' OPC servers and programming software

For details on this important feature and instructions in its operation, refer to the Cyberlogic OPC Server Help.

Editor Options

The Cyberlogic OPC Server Configuration editor has several options that let you adjust the operation of the editor to suit your preferences. They also allow you to set security levels as needed for communication with client software.

For a full discussion, refer to the Cyberlogic OPC Server Help.

VALIDATION AND TROUBLESHOOTING

The following sections describe features that will help you to verify and troubleshoot your server's operation. The Data Monitor and Cyberlogic OPC Client allow you to view the data as it is received by the server. Each of those tools allows you to view the Status Items

Data Monitor

The Data Monitor lets you monitor the values and status of the data items. Its use is described in detail in the Cyberlogic OPC Server Help.

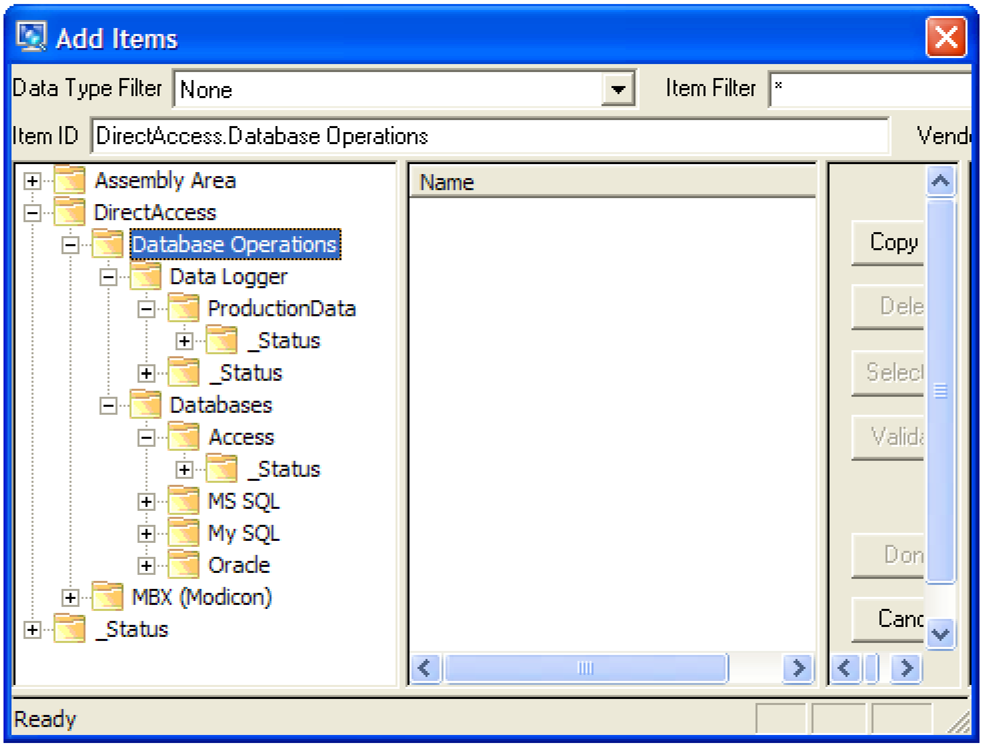
Cyberlogic OPC Client

The Cyberlogic OPC Client is a simple OPC Data Access client that lets you see how the server interacts with a client and lets you test its response to various loads. Its use is described in detail in the Cyberlogic OPC Server Help.

Status Items

The Cyberlogic OPC Server provides status items that any OPC client application can access. These items provide health and performance information about the server itself, its network connections, network nodes, devices and crosslinks.

This section describes status items that are specific to Data Logger. For complete information on the standard status items and how to access them through DirectAccess, refer to the Cyberlogic OPC Server Help.



Data Type Filter: None

Item Filter: *

Item ID: DirectAccess.Database Operations

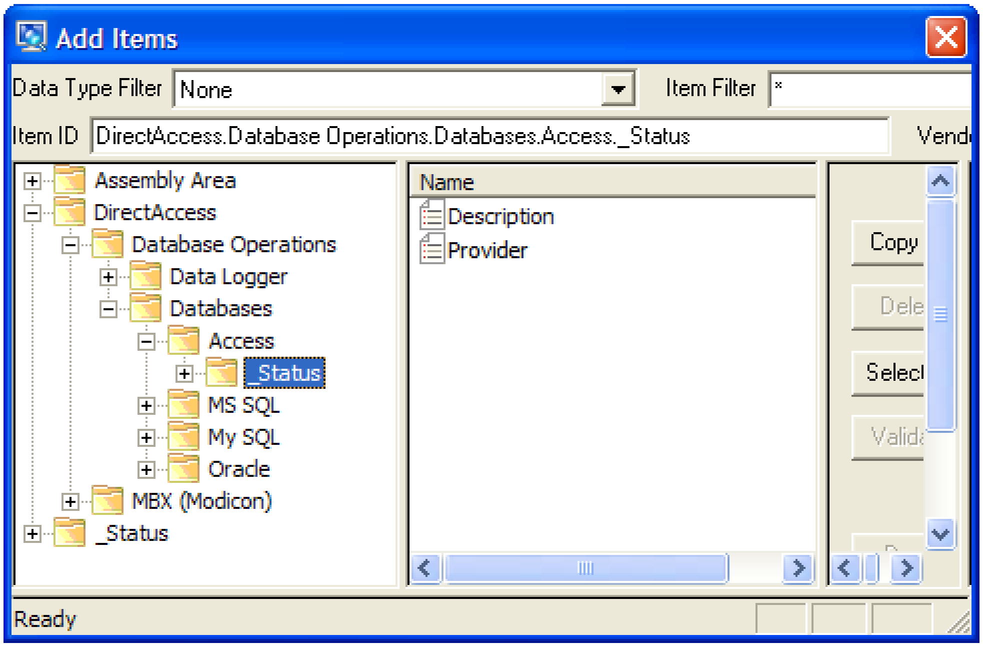
Status item folders are available for Data Logger,
each Logging Group, and each Database.

When you connect to the Cyberlogic OPC Server with a client application and browse for items to display, the status items are shown in folders called *_Status*. The contents of each folder depend on the type of item that it provides status for.

- Databases
- Data Logger
- Logging Group Control Items
- Logging Group Status Items
- Data Record Status Items

Databases

Each database has its own set of status items. If you want to access the status items for a database, you must check the ***Enable DirectAccess*** box on the General Tab for that database.



Data Type Filter None

Item Filter *

Item ID DirectAccess.Database Operations.Databases.Access._Status

When enabled, the status items for a database will be located in a _Status folder directly under the branch for that database in the *DirectAccess.Database Operations.Databases* tree. Refer to the figure above for the location of the _Status folder for a typical Access database.

The database status items are:

Description

This is the description text as configured in the Description field.

Provider

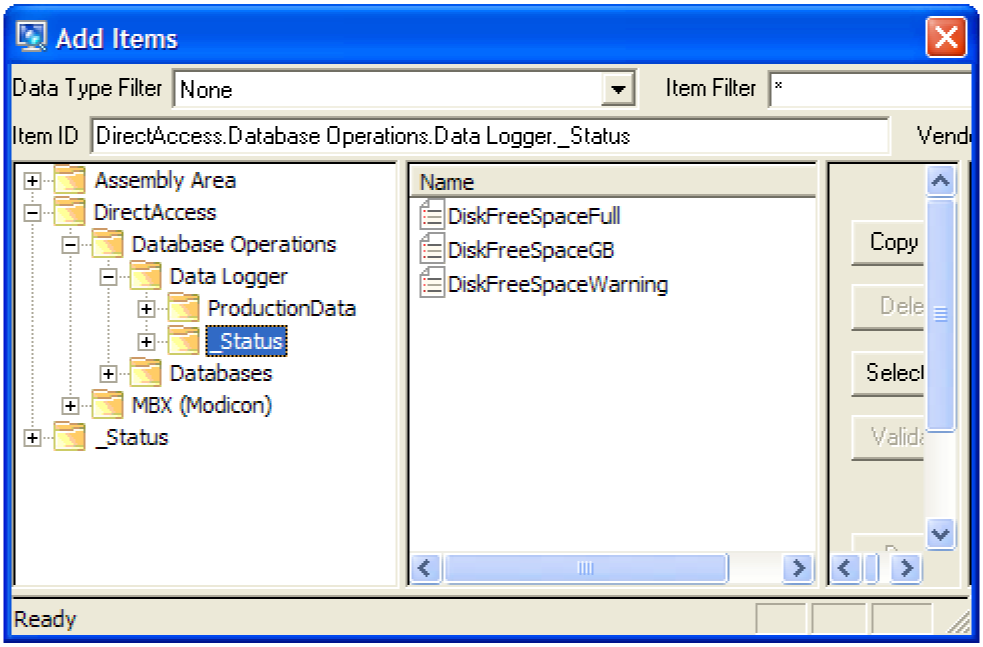
This item provides the name of the data provider associated with this database.

ConnectionString

This item provides the string the OPC server uses to establish a connection to this database.

Data Logger

There is one set of status items for Data Logger. To access these, you must check the ***Enable DirectAccess*** box on the Data Logger's General Tab.



Data Type Filter: None

Item Filter: *

Item ID: DirectAccess.Database Operations.Data Logger._Status

When enabled, the status items are located in a _Status folder directly under the Data Logger branch. Specifically, the folder is located at *DirectAccess.Database Operations.Data Logger._Status*, as shown in the figure above.

The Data Logger status items relate to settings that are configured on the Data Logger General Tab. They are:

DiskFreeSpaceGB

This is the current amount of free space on the disk that is used for the disk cache. The space is shown in gigabytes.

DiskFreeSpaceWarning

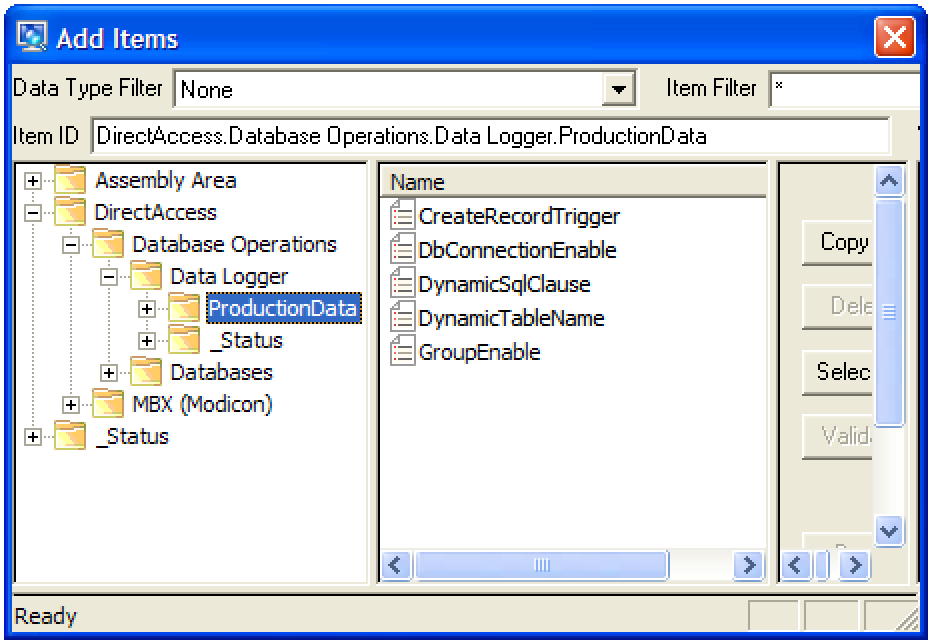
When this flag is ON, it indicates that the amount of free space on the disk is at or below the specified warning level.

DiskFreeSpaceFull

When this flag is ON, it indicates that the amount of free space on the disk is at or below the specified minimum level.

Logging Group Control Items

There is one set of control items for each logging group. To access these, you must check the ***Enable DirectAccess*** box on the logging group's General Tab.



When enabled, the control items are located in the logging group's folder, which is *DirectAccess.Database Operations.Data Logger. {LoggingGroupName}*, as shown in the figure above.

Each control item is present only if certain configuration settings are selected. The definitions below refer you to the screen where you can view or change these settings.

GroupEnable

Refer to the logging group's General Tab.

This control input is present only when *Use Data Item To Enable* is selected. It shows the status of the selected enable input. If DirectAccess writes are enabled, you can write to this input to either enable (ON state) or disable (OFF state) the logging group.

CreateRecordTrigger

Refer to the logging group's Settings Tab.

This control input is present only when *Triggered* mode is checked. It shows the status of the selected trigger input. If DirectAccess writes are enabled, you can write to this input to trigger the creation of a data record.

DbConnectionEnable

Refer to the logging group's *Advanced* settings Database Tab.

This control input is present only when *Use Data Item* is selected for the *Database Connection*. It shows the status of the selected enable input. If DirectAccess writes are enabled, you can write to this input to either enable (ON state) or disable (OFF state) the database connection.

DynamicTableName

Refer to the logging group's Database Tab.

This control input is present only when *Use Data Item* is selected for the *Table/Stored Procedure Name*. It shows the dynamically-selected table or stored procedure name. If DirectAccess writes are enabled, you can write to this input to modify the name.

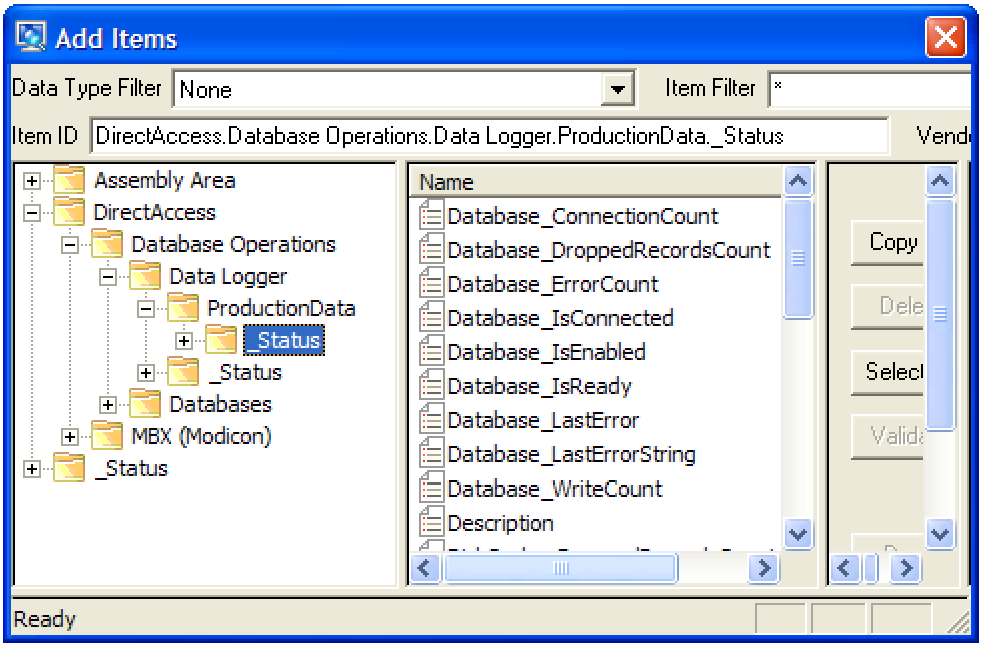
DynamicSqlClause

Refer to the logging group's *Advanced* settings Logging Type Tab.

This control input is present only if the logging type is *Update Selected Rows*, and *Use Data Item* is configured for *SQL WHERE Clause*. It shows the dynamically-selected SQL WHERE clause. If DirectAccess writes are enabled, you can write to this input to modify the clause.

Logging Group Status Items

There is one set of status items for each logging group. To access these, you must check the ***Enable DirectAccess*** box on the logging group's General Tab.



Data Type Filter: None

Item Filter: *

Item ID: DirectAccess.Database Operations.Data Logger.ProductionData._Status

When enabled, the status items are located in a _Status folder directly under the logging group's branch. Specifically, the folder is located at *DirectAccess.Database Operations.Data Logger. {LoggingGroupName}._Status*, as shown in the figure above.

IsEnabled

Indicates whether or not the logging group is enabled. The values are:

- 0 = Disabled
- 1 = Enabled

ResetAllErrorCounts

Set this item to the ON state to reset all data logger error counts to zero. It is automatically reset to the OFF state following the reset operation.

This resets only the counts; the last error information is not cleared. Any OPC client can write to this item.

ResetAllErrorInfo

Set this item to the ON state to reset all data logger error information. This includes the error counts and the last error information. It is automatically reset to the OFF state following the reset operation.

Inputs_UpdateCount

This is the total update count for all inputs associated with this logging group.

This counter is incremented each time an OnDataChange update callback is received for the inputs associated with this logging group.

Inputs_ErrorCount

This is the total error count for all inputs associated with this logging group.

The counter is incremented each time bad quality data is received on any input in the logging group.

MemCache_RecordCount

This is the current number of data records in the memory cache.

MemCache_RecordCountLimit

This is the maximum number of data records that may be stored in the memory cache.

MemCache_RecordCountMax

This is the maximum number of data records that have been stored in the memory cache since the server started, or the last Save & Update operation.

MemCache_DroppedRecordsCount

When the memory cache limit is reached, no new data records can be accepted and new records will be dropped. This item shows how many records have been dropped since the server started.

DiskCache_FileSizeMB

This is the current size of the disk cache for this logging group. The value is in megabytes.

DiskCache_RecordCount

This is the current number of data records in the disk cache.

DiskCache_DroppedRecordsCount

This count indicates the number of dropped data records since the server has started. It is incremented each time a data record cannot be written to the disk cache. This can happen under the following conditions:

- The disk cache is not available because it failed to initialize.
- An error happened during an attempt to write a record into the disk cache file.
- The free disk space dropped below the minimum level that was specified on the Data Logger General Tab.

DiskCache_IsReady

At server startup and each time you perform Save & Update Server, there is a delay while Data Logger prepares the disk cache for use. *DiskCache_IsReady* is OFF during this delay and turns ON when the disk cache is ready to accept data records.

The initialization time depends on the number of records in the disk cache file, and may vary from a few milliseconds to several seconds. During this time, new data records are saved in the memory cache. You should allow for the disk cache initialization time when you set the *Size Limit* on the Memory Cache Tab.

Refer to *DiskCache_InitTimeLast* and *DiskCache_InitTimeMax*.

DiskCache_ErrorCount

This is the total number of errors associated with the disk cache. Refer to *ResetAllErrorCounts* and *ResetAllErrorInfo* for information on how to reset this counter to zero.

DiskCache_LastError

This is the last error code associated with the disk cache. Refer to *ResetAllErrorInfo* for information on how to reset this code.

DiskCache_LastErrorString

This is the last error string associated with the disk cache. Use this information to troubleshoot the problem. Refer to *ResetAllErrorInfo* for information on how to reset this string.

DiskCache_InitTimeLast

At server startup and each time you perform Save & Update Server, there is a delay while Data Logger prepares the disk cache for use. This item shows how long it took to initialize the disk cache the last time. The value is in milliseconds.

DiskCache_InitTimeMax

At server startup and each time you perform Save & Update Server, there is a delay while Data Logger prepares the disk cache for use. This item shows the maximum amount of time that it ever took to initialize the disk cache. The value is in milliseconds.

DiskCache_ClearCache

Setting this item to the ON state removes all data records from the disk cache file. It is automatically reset to the OFF state following the clear operation.

Database_Name

The name of the database the logging group is using. This will be one of the databases configured in the Databases Branch.

Database_IsEnabled

Indicates whether or not access is enabled to the target database.

Database_IsConnected

Indicates whether or not the data logger is connected to the target database.

Database_IsReady

After Data Logger connects to the target database, it prepares the database for use. An ON state of this flag indicates that the database is ready to accept data records.

Database_ConnectionCount

Data Logger increments this counter each time it connects to the target data base.

Database_TimeToConnectLast

This item shows the amount of time it took to connect to the target database last time. The value is in milliseconds.

Database_TimeToConnectMax

This item shows the maximum amount of time that it ever took to connect to the target database. The value is in milliseconds.

Database_WriteCount

Data Logger increments this counter is incremented each time it attempts to write to the database, that is, each time it attempts to insert a row, update a row or execute a stored procedure. The count includes both successful and unsuccessful writes.

Database_WriteOkCount

Data Logger increments this counter is incremented each time it successfully writes to the database, that is, each time it inserts a row, updates a row or executes a stored procedure.

Database_DroppedRecordsCount

If a write to the database fails because of a recoverable problem, such as a communication failure, Data Logger will retry the operation until it succeeds. However, if the write failure is due to a fatal error, such as an illegal table name, it can never succeed. In that case, Data Logger drops the record and increments this counter. The counter indicates the number of data records that Data Logger has dropped since the server started.

Database_ErrorCount

This counter provides the total number of errors associated with the database operations. Refer to *ResetAllErrorCounts* and *ResetAllErrorInfo* for information on how to reset this counter to zero.

Database_LastError

This item contains the last error code associated with a database error. Refer to *ResetAllErrorInfo* for information on how to reset this code.

Database_LastErrorString

This item contains the last error string associated with a database error. Use the information provided here to troubleshoot the problem. Refer to *ResetAllErrorInfo* for information on how to reset this string.

Database_UpdateNoRowsFoundCount

This counter shows the number of update operations that returned no rows to be updated.

This item is present only if the Logging Type is set to *Update Selected Rows*. The Logging Type is set on the Logging Type Tab of the logging group's Advanced properties.

Database_UpdateFailedRowsCount

This counter shows the number of times a row update operation failed. Refer to *ResetAllErrorCounts* and *ResetAllErrorInfo* for information on how to reset this counter to zero.

This item is present only if the Logging Type is set to *Update Selected Rows*. The Logging Type is set on the Logging Type Tab of the logging group's Advanced properties.

Database_UpdateWarningString

This item contains the last error string associated with a row update operation error. Use the information provided here to troubleshoot the problem. Refer to *ResetAllErrorInfo* for information on how to reset this string.

This item is present only if the Logging Type is set to *Update Selected Rows*. The Logging Type is set on the Logging Type Tab of the logging group's Advanced properties.

Database_StoredProcReturnValue

This item contains the returned value from the last stored procedure call. The return value can be of any supported type, however, this items will always present the value as a string.

This item is present only if the Data Record Format is set to *Stored Procedure* on the logging group's Database Tab.

Database_StoredProcOutput1-3

In addition to the returned value, each stored procedure can return up to three output values. The output parameters can be of any supported type, however, these items will always present the values as strings.

GroupEnable_ErrorCount

This counter contains the total error count associated with the *GroupEnable* control input. The counter is incremented each time bad quality data is received for *GroupEnable*.

This item is present only if the *Database Logging* selection is *Use Data Item To Enable*. Refer to the logging group's General Tab for details on this configuration.

GroupEnable_ItemID

This is the Item ID string associated with the *GroupEnable* control input.

This item is present only if the *Database Logging* selection is *Use Data Item To Enable*. Refer to the logging group's General Tab for details on this configuration.

GroupEnable_UpdateCount

This counter contains the total update count for the *GroupEnable* control input. The counter is incremented each time an OnDataChange callback delivers a new value for the *GroupEnable*.

This item is present only if the *Database Logging* selection is *Use Data Item To Enable*. Refer to the logging group's General Tab for details on this configuration.

GroupEnable_LastError

This is the last error code associated with the *GroupEnable* control input.

This item is present only if the *Database Logging* selection is *Use Data Item To Enable*. Refer to the logging group's General Tab for details on this configuration.

GroupEnable_LastErrorQuality

This is the last error quality associated with the *GroupEnable* control input.

This item is present only if the *Database Logging* selection is *Use Data Item To Enable*. Refer to the logging group's General Tab for details on this configuration.

GroupEnable_LastErrorString

This is the last error code string associated with the *GroupEnable* control input.

This item is present only if the *Database Logging* selection is *Use Data Item To Enable*. Refer to the logging group's General Tab for details on this configuration.

CreateRecordTrigger_ErrorCount

This is the total error count associated with the *CreateRecordTrigger* control input. The counter is incremented each time bad quality data is received for this input.

This item is present only if *Triggered* is selected in the logging group's *Create Records* section. Refer to the Settings Tab for details on this configuration.

CreateRecordTrigger_ItemID

This is the Item ID string associated with the *CreateRecordTrigger* control input.

This item is present only if *Triggered* is selected in the logging group's *Create Records* section. Refer to the Settings Tab for details on this configuration.

CreateRecordTrigger_UpdateCount

This counter contains the total update count for the *CreateRecordTrigger* control input. It is incremented each time an OnDataChange callback delivers a new value for the *CreateRecordTrigger*.

This item is present only if *Triggered* is selected in the logging group's *Create Records* section. Refer to the Settings Tab for details on this configuration.

CreateRecordTrigger_LastError

This item contains the last error code associated with the *CreateRecordTrigger* control input.

This item is present only if *Triggered* is selected in the logging group's *Create Records* section. Refer to the Settings Tab for details on this configuration.

CreateRecordTrigger_LastErrorQuality

This item contains the last error quality associated with the *CreateRecordTrigger* control input.

This item is present only if *Triggered* is selected in the logging group's *Create Records* section. Refer to the Settings Tab for details on this configuration.

CreateRecordTrigger_LastErrorString

This item contains the last error code string associated with the *CreateRecordTrigger* control input.

This item is present only if *Triggered* is selected in the logging group's *Create Records* section. Refer to the Settings Tab for details on this configuration.

CreateRecordTrigger_LastValue

This item contains the last data value associated with the *CreateRecordTrigger* control input.

This item is present only if *Triggered* is selected in the logging group's *Create Records* section. Refer to the Settings Tab for details on this configuration.

DynamicTableName_ErrorCount

This counter contains the total error count associated with the *DynamicTableName* control input. It is incremented each time bad quality data is received for this input.

This item is present only if the *Table Name* selection is *Use Data Item*. Refer to the logging group's Database Tab for details on this configuration.

DynamicTableName_ItemID

This is the Item ID string associated with the *DynamicTableName* control input.

This item is present only if the *Table Name* selection is *Use Data Item*. Refer to the logging group's Database Tab for details on this configuration.

DynamicTableName_UpdateCount

This counter contains the total update count for the *DynamicTableName* control input. It is incremented each time an OnDataChange callback delivers a new value for the *DynamicTableName* control input.

This item is present only if the *Table Name* selection is *Use Data Item*. Refer to the logging group's Database Tab for details on this configuration.

DynamicTableName_LastError

This is the last error code associated with the *DynamicTableName* control input.

This item is present only if the *Table Name* selection is *Use Data Item*. Refer to the logging group's Database Tab for details on this configuration.

DynamicTableName_LastErrorQuality

This is the last error quality associated with the *DynamicTableName* control input.

This item is present only if the *Table Name* selection is *Use Data Item*. Refer to the logging group's Database Tab for details on this configuration.

DynamicTableName_LastErrorString

This is the last error code string associated with the *DynamicTableName* control input.

This item is present only if the *Table Name* selection is *Use Data Item*. Refer to the logging group's Database Tab for details on this configuration.

DynamicTableName_LastValue

This is the last data value associated with the *DynamicTableName* control input.

This item is present only if the *Table Name* selection is *Use Data Item*. Refer to the logging group's Database Tab for details on this configuration.

DynamicSqlClause_ErrorCount

This counter contains the total error count associated with the *DynamicSqlClause* control input. It is incremented each time bad quality data is received for this input.

This item is present only if the logging type is *Update Selected Rows* and the *SQL WHERE Clause* selection is *Use Data Item*. Refer to the Logging Type Tab in the logging group's Advanced settings for details on this configuration.

DynamicSqlClause_ItemID

This is the Item ID string associated with the *DynamicSqlClause* control input.

This item is present only if the logging type is *Update Selected Rows* and the *SQL WHERE Clause* selection is *Use Data Item*. Refer to the Logging Type Tab in the logging group's Advanced settings for details on this configuration.

DynamicSqlClause_UpdateCount

This counter contains the total update count for the *DynamicSqlClause* control input. It is incremented each time an OnDataChange callback delivers a new value for the *DynamicSqlClause* control input.

This item is present only if the logging type is *Update Selected Rows* and the *SQL WHERE Clause* selection is *Use Data Item*. Refer to the Logging Type Tab in the logging group's Advanced settings for details on this configuration.

DynamicSqlClause_LastError

This is the last error code associated with the *DynamicSqlClause* control input.

This item is present only if the logging type is *Update Selected Rows* and the *SQL WHERE Clause* selection is *Use Data Item*. Refer to the Logging Type Tab in the logging group's Advanced settings for details on this configuration.

DynamicSqlClause_LastErrorQuality

This is the last error quality associated with the *DynamicSqlClause* control input.

This item is present only if the logging type is *Update Selected Rows* and the *SQL WHERE Clause* selection is *Use Data Item*. Refer to the Logging Type Tab in the logging group's Advanced settings for details on this configuration.

DynamicSqlClause_LastErrorString

This is the last error code string associated with the *DynamicSqlClause* control input.

This item is present only if the logging type is *Update Selected Rows* and the *SQL WHERE Clause* selection is *Use Data Item*. Refer to the Logging Type Tab in the logging group's Advanced settings for details on this configuration.

DynamicSqlClause_LastValue

This is the last data value associated with the *DynamicSqlClause* control input.

This item is present only if the logging type is *Update Selected Rows* and the *SQL WHERE Clause* selection is *Use Data Item*. Refer to the Logging Type Tab in the logging group's Advanced settings for details on this configuration.

DbConnectionEnable_ErrorCount

This counter contains the total error count associated with the *DbConnectionEnable* control input. It is incremented each time bad quality data is received for this input.

This item is present only if *Use Data Item* is selected for enabling the database connection. Refer to the Database Tab in the logging group's Advanced settings for details on this configuration.

DbConnectionEnable_ItemID

This is the Item ID string associated with the *DbConnectionEnable* control input.

This item is present only if *Use Data Item* is selected for enabling the database connection. Refer to the Database Tab in the logging group's Advanced settings for details on this configuration.

DbConnectionEnable_UpdateCount

This counter contains the total update count for the *DbConnectionEnable* control input. It is incremented each time an OnDataChange callback delivers a new value for the *DbConnectionEnable* control input.

This item is present only if *Use Data Item* is selected for enabling the database connection. Refer to the Database Tab in the logging group's Advanced settings for details on this configuration.

DbConnectionEnable_LastError

This is the last error code associated with the *DbConnectionEnable* control input.

This item is present only if *Use Data Item* is selected for enabling the database connection. Refer to the Database Tab in the logging group's Advanced settings for details on this configuration.

DbConnectionEnable_LastErrorQuality

This is the last error quality associated with the *DbConnectionEnable* control input.

This item is present only if *Use Data Item* is selected for eanbling the database connection. Refer to the Database Tab in the logging group's Advanced settings for details on this configuration.

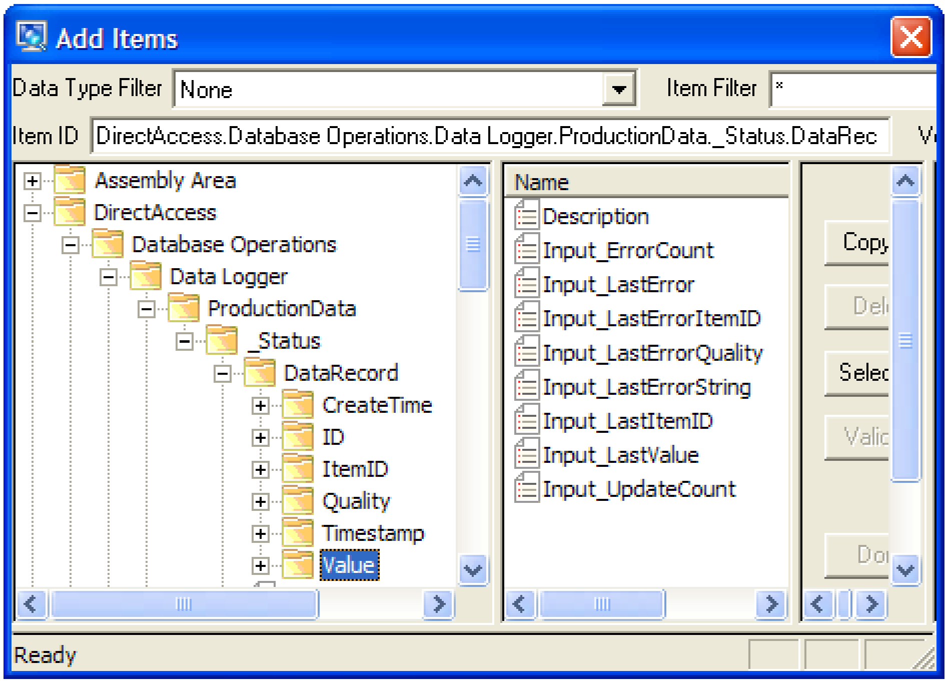
DbConnectionEnable_LastErrorString

The last error code string associated with the *DbConnectionEnable* control input.

This item is present only if *Use Data Item* is selected for enabling the database connection. Refer to the Database Tab in the logging group's Advanced settings for details on this configuration.

Data Record Status Items

There are several status item folders for each configured data record. These are contained in a folder called *DataRecord*, which you can find in the *_Status* folder for the logging group.



Data Type Filter: None

Item Filter: *

Item ID: DirectAccess.Database Operations.Data Logger.ProductionData._Status.DataRec

Specifically, the folder is located at *DirectAccess.Database Operations.Data Logger.{LoggingGroupName}._Status.DataRecord*, as shown in the figure above. Within the DataRecord folder are folders for each column of the record. The column folders have the same names as the columns, and each contains the status items for that column. The data record and its columns are configured on the logging group's Data Record Tab.

Description

This is the description text as configured in the Description field.

Input_ErrorCount

This counter contains the total error count. It is incremented each time Data Logger receives bad quality data from an input associated with this column.

Input_LastItemID

This is the Item ID string associated with the input that provided the last data for this column.

Input_LastError

This is the last error code associated with the input that provided data for this column.

Input_LastErrorItemID

This is the Item ID string associated with the last input that provided data for this column, and that had an error.

Input_LastErrorQuality

This is the quality associated with the last input that provided data for this column, and that had an error.

Input_LastErrorString

This is the error string associated with the last input that provided data for this column, and that had an error.

Input_LastValue

This is the last data value received for this data record's column.

Input_UpdateCount

This counter contains the total update count. It is incremented each time an OnDataChange callback delivers a new value for this column.

Frequently Asked Questions

My database connection attempts time out and fail. What’s the problem?

If you’re using Oracle or MySQL, check the Windows firewall to be sure that the necessary port is open. For Oracle, the default is 1521, and for MySQL it’s 3306.

Every time Data Logger tries to create a new table, it fails. What should I do?

Check the database software configuration to verify that the user account Data Logger is using has the right to create a table.

***I’m using Oracle and I’ve confirmed that the user has the right to create tables, but it still fails with the error message:* Failed creating table <TableName>. Error = “ORA-01536: space quota exceeded for tablespace ‘USERS’” Reported by OraOLEDB.**

Each user has a limited amount of table space, and you’ve gone over your quota. Execute the following command from the system account:

ALTER USER <user> quota unlimited on USERS;

where <user> is the specific user name that needs the larger quota.