



Your comments and suggestions on the operation of this software are welcome. Please address them to:

ICONICS

100 Foxborough Blvd.

Foxborough, MA 02035

Tel: 508-543-8600

Fax: 508-543-1503

E-mail: support@iconics.com

Web: www.iconics.com

© ICONICS, Inc. All rights reserved.

Contents

Introduction

Installation

Starting the Alarm Logger Configuration

Running the Logger

Introduction

The AlarmWorX32 Logger provides a permanent copy of alarm and event notifications produced by any OPC Alarm and Events server, including the AlarmWorX32 Server. The alarm and event data may be stored in an OLE-DB or ODBC database and/or sent to a printer. The logger has been tested with the Microsoft Access (Jet) database and with Microsoft SQL Server.

The Logger Application (AWXLog32.exe) that provides the runtime storage and printing has no user interface and may optionally be run as a service on Windows NT. The logger typically reads its configuration information from a Microsoft Access database file.

The Alarm Logger Configurator (AWXLogCfg.exe) is used to make changes to the Microsoft Access Database file that the Logger uses for configuration information.

Installation

The following are the major files that comprise the Logger component of AlarmWorX32:

AWXLog32.exe	Logger application/service.
AWXLogCfg.exe	Logger Configurator.
AWXLog32.mdb	Logger sample configuration database

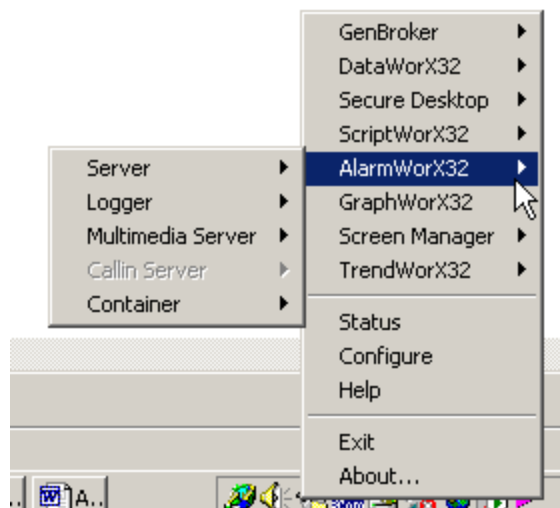
The files are installed in the Bin directory chosen during the install. Program icons are installed for the Configurator in the AlarmWorX32 program group.

By default the Logger (AwxLog32.exe) is registered as a normal application, but can be registered to run as a service under Windows NT. Running as a Service has the following advantages:

- The Logger can be set to start as soon as the machine is powered (before a user has logged on).
- The Logger will remain running even if a user Logs off of NT.

To register the Logger as a Windows NT Service:

1. Run the **GenTray** utility from the Windows **Start** menu by selecting **Programs > ICONICS GENESIS-32 > Tools > GenTray**.
2. This opens GenTray and puts the **GenTray** icon in the taskbar system tray. Click the **GenTray** icon in the system tray and select **AlarmWorX32 > Logger > NT Service**, as shown in the figure below. To have the service start automatically upon system startup, click the **GenTray** icon in the system tray and select **AlarmWorX32 > Logger > Auto Start**.



Running AlarmWorX32 Logger From GenTray

The **GenTray** utility can also be used to manually start and stop the logger.

Note: The Alarm Logger does not use standard COM rules for activation/deactivation. If the server is started for any reason (via GenTray or by activation by an OPC Alarm Client), it will remain running until both of the following conditions are met:

- The GenTray utility has issued the **Stop** command to the Server.
- All OPC alarm clients have disconnected from the server.

Since NT Services are not supported in Windows 95 and Windows 98, GenTray can be used to facilitate the starting of the Alarm Logger on startup:

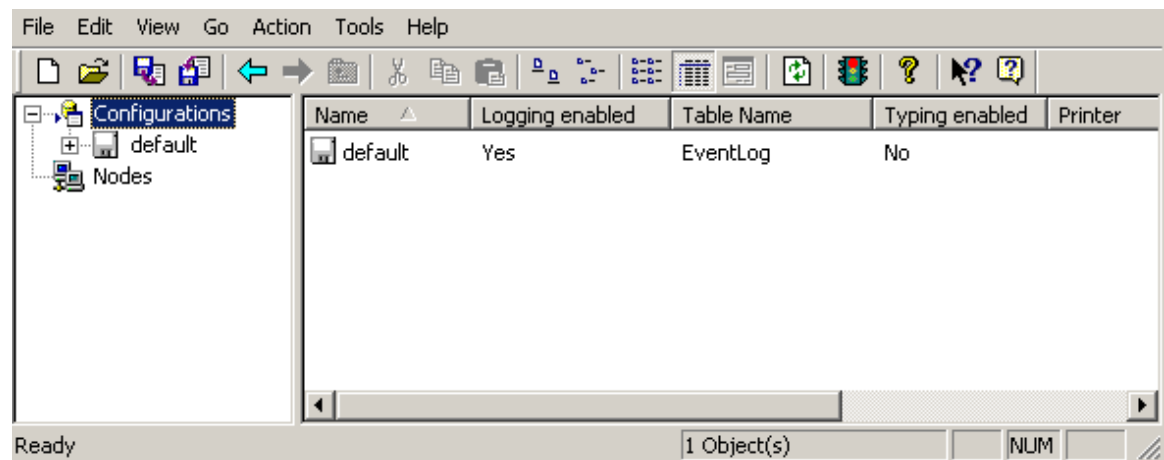
3. Place a shortcut to the **GenTray** utility in the startup group so **GenTray** will start automatically when Windows 95 or Windows 98 boots up.
4. To have the Alarm Logger start automatically when GenTray starts, click the **GenTray** icon in the system tray and select **AlarmWorX32 > Logger > Auto Start**.

Starting the Alarm Logger Configuration

The AlarmWorX32 Logger Configurator writes configuration information to a configuration database. This configuration information is read by the AlarmWorX32 Logger during runtime to determine what alarms and events to log and where to log them. A single database can contain multiple named configurations to be run by one or more nodes on a network. A single node can also run multiple configurations simultaneously.

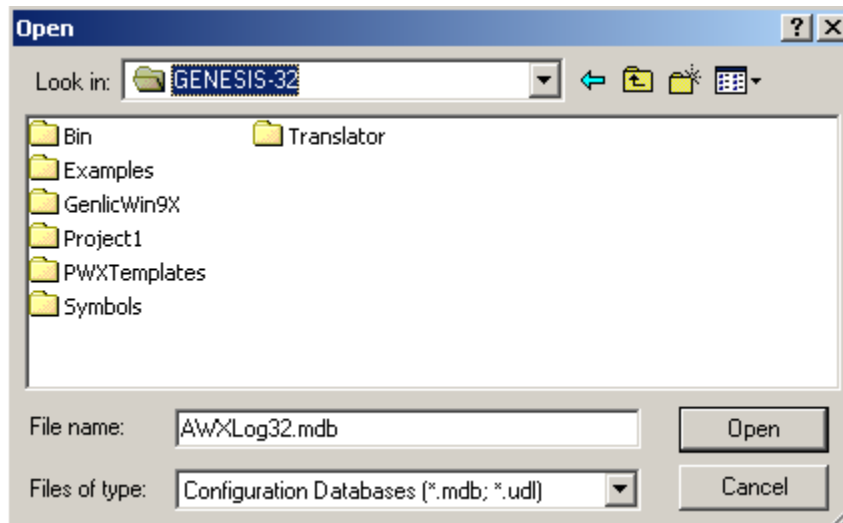
To launch the AlarmWorX32 Logger Configurator from the Windows **Start** menu:

1. Select **Programs > ICONICS GENESIS-32 > AlarmWorX32 > Alarm Logger Configurator**.
2. The **AwxLog32.mdb** Microsoft Access database file opens in the Alarm Logger Configurator, as shown in the figure below. The screen consists of a split window with a tree control view in the left-hand pane and a configuration view in the right-hand pane. The Configurator provides a standard format for the configuration database, as well as a sample (default) configuration project. The Configurator also includes a toolbar and menus with many command functions.



Alarm Logger Configurator Screen

3. To open a different database file, select **Open** from the **File** menu. This opens the **Open** dialog box, shown below, which enables you to browse for configuration files.



Selecting a Configuration Database

Configurations

Database Logging Tab

Printer Logging

Translation Tab

Columns

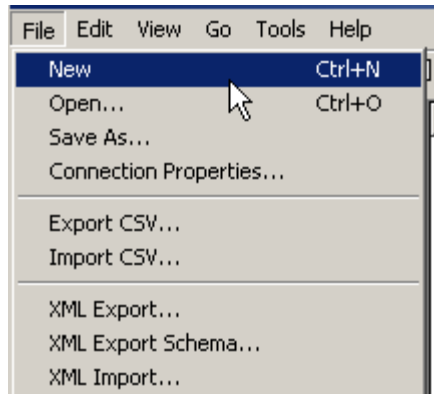
Nodes

Assigned Configurations

Creating Configuration Databases

The Configurator provides a Configuration Database Wizard for creating new Microsoft Access and SQL Server configuration databases. To create a new configuration database in the Configurator:

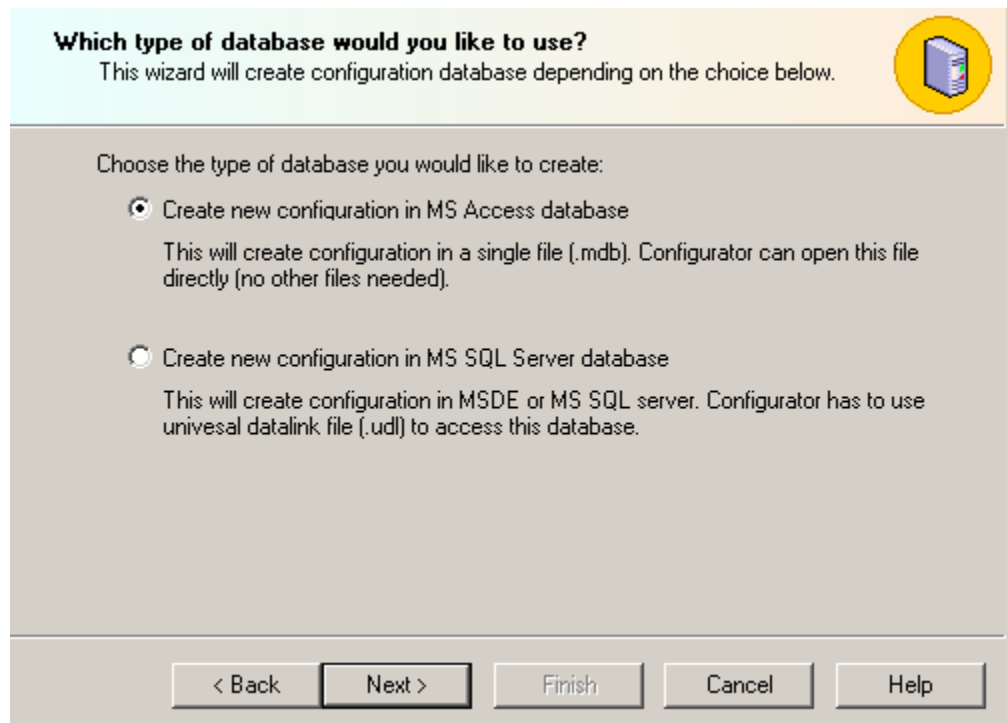
1. Select **New** from the **File** menu, as shown in the figure below.



Creating a Configuration Database

2. The introduction screen for the Configuration Database Wizard appears. Click the **Next** button to continue.
3. You have two options for creating your new database, as shown in the figure below:
 - **Create a new Microsoft Access configuration database:** For a Microsoft Access database, the Configurator uses a single .mdb file
 - **Create a Microsoft SQL Server database:** The Configurator uses Universal Data Link (.udl) files to connect to a Microsoft SQL Server or MSDE database.

Select the database type you want to create, and then click **Next**.

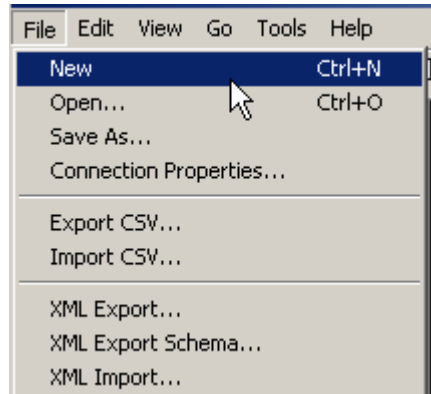


Choosing the Type of Database To Create

Creating a Microsoft Access Configuration Database

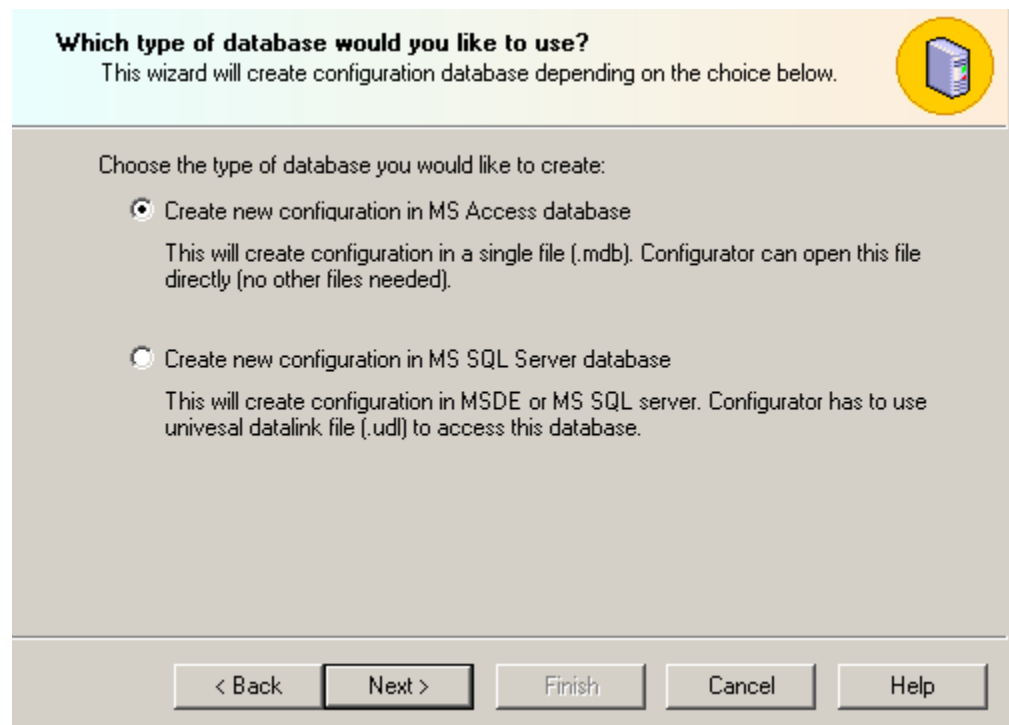
To create a new Microsoft Access configuration database in the Configurator:

1. Select **New** from the **File** menu, as shown in the figure below.



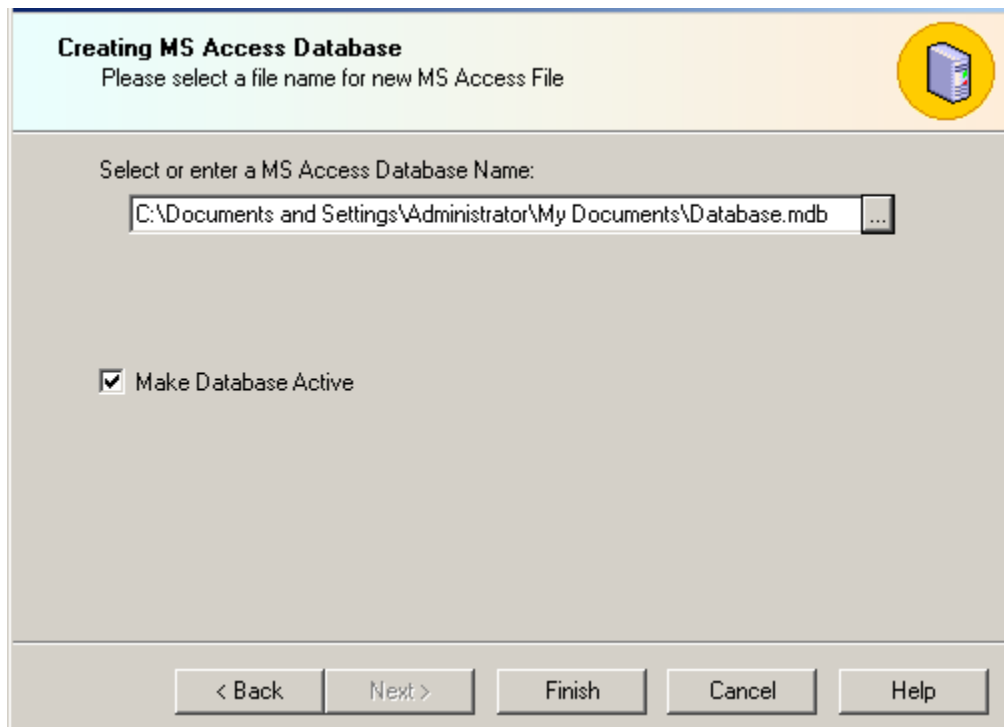
Creating a Configuration Database

2. The introduction screen for the Configuration Database Wizard appears. Click the **Next** button to continue.
3. You have two options for creating your new database, as shown in the figure below. Select **Create new configuration in MS Access database**. Click the **Next** button to continue.



Creating a New Microsoft Access Database

- Specify the directory path and file name for the new database, as shown in the figure below. Click the ... button to browse for a directory. If you want this new database to be the active configuration database, check **Make Database Active**.



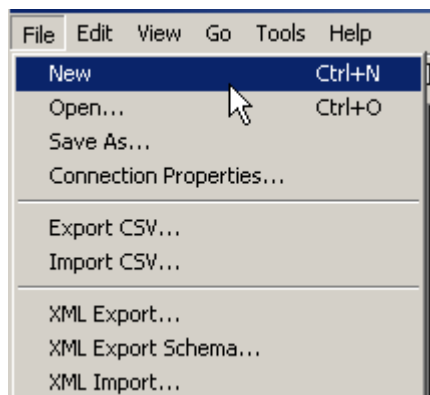
Naming the New Access Database

- Click the **Finish** button. The new database is created and opened in the Configurator.

Creating a Microsoft SQL Server Configuration Database

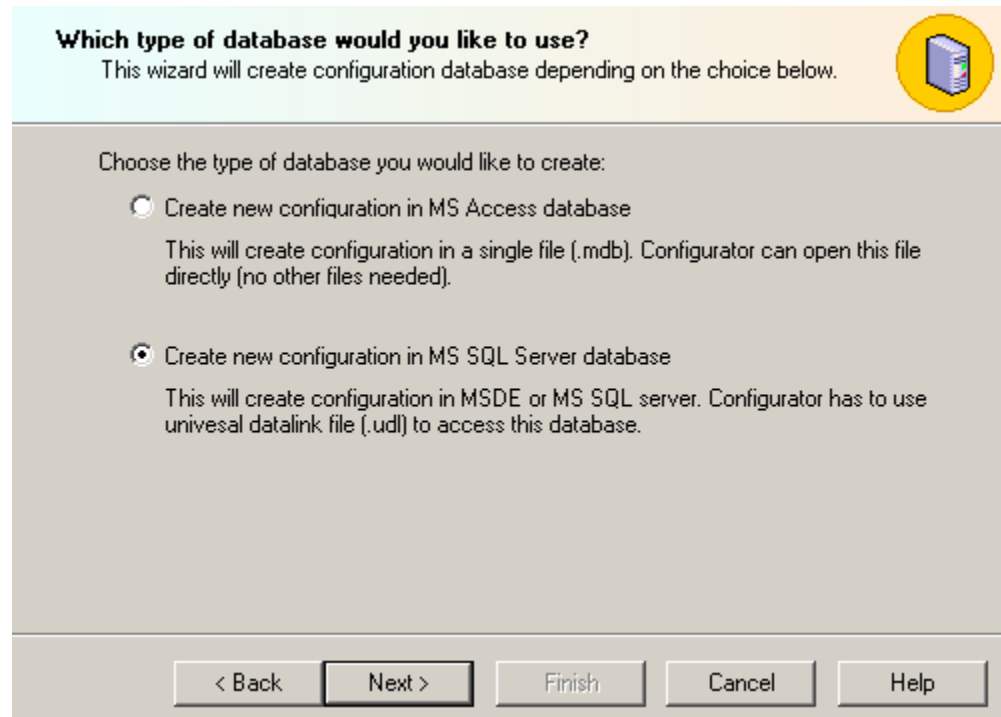
To create a new Microsoft SQL Server configuration database in the Configurator:

- Select **New** from the **File** menu, as shown in the figure below.



Creating a Configuration Database

2. The introduction screen for the Configuration Database Wizard appears. Click the **Next** button to continue.
3. You have two options for creating your new database, as shown in the figure below. Select **Create new configuration in MS SQL Server database**. Click the **Next** button to continue.



Creating a New Microsoft SQL Server Database

4. To connect to a SQL Server database, either select an existing database from the **Database Name** drop-down list, or a type a new name to create a new database, as shown in the figure below. If you choose to use an existing database, you have the option of adding (auto-appending) the configuration to the existing database. (For more information about the auto-append function, please see "Adding the Configuration to an Existing Database.")

In the **SQL Server Name** field, select the local SQL Server on which to create the database. If necessary, enter a user name and password to log on to the SQL Server. (It is recommended that you use Windows NT integrated security.)

Note: Usually you have only one instance of SQL Server running on the local node. In this case, the drop-down list under **SQL Server Name** has only one option: "(local)." However, it is possible to run multiple SQL Server instances on the local node, in which case the **SQL Server** field lists all those SQL Server instances: "(local)" for the default instance and "node_name/instance_name" for all others. The drop-down list may show SQL Server instances on other nodes as well.

SQL Server Database Connection
Please specify the following to connect SQL server

Select or enter a database name:

☐ Auto-append configuration into existing database

Select or enter a server name:

Enter information to log on to the server:
☒ Use Windows NT integrated security (recommended)
☐ Use a specific user name and password:
 User name:
 Password:
☐ Blank Password

Connecting to a SQL Server Database

- Specify a directory path location in which to create the database, as shown in the figure below. You can either use the default SQL Server database folder, or you can click the ... button and browse for a specific folder.

SQL Server Database Creation
Please specify the following to create database files

Specify database files location:
☒ Use default SQL server database folder (recommended)
☐ Use specific folder (local node only)

Database files properties:

Database File		Log File	
Initial Size (MB):	1	Initial Size (MB):	1
File Grow:	☒ AutoShrink	File Grow:	☐ In MB
	☐ In MB 1		☐ In MB 1
	☒ By Percent 20		☒ By Percent 20

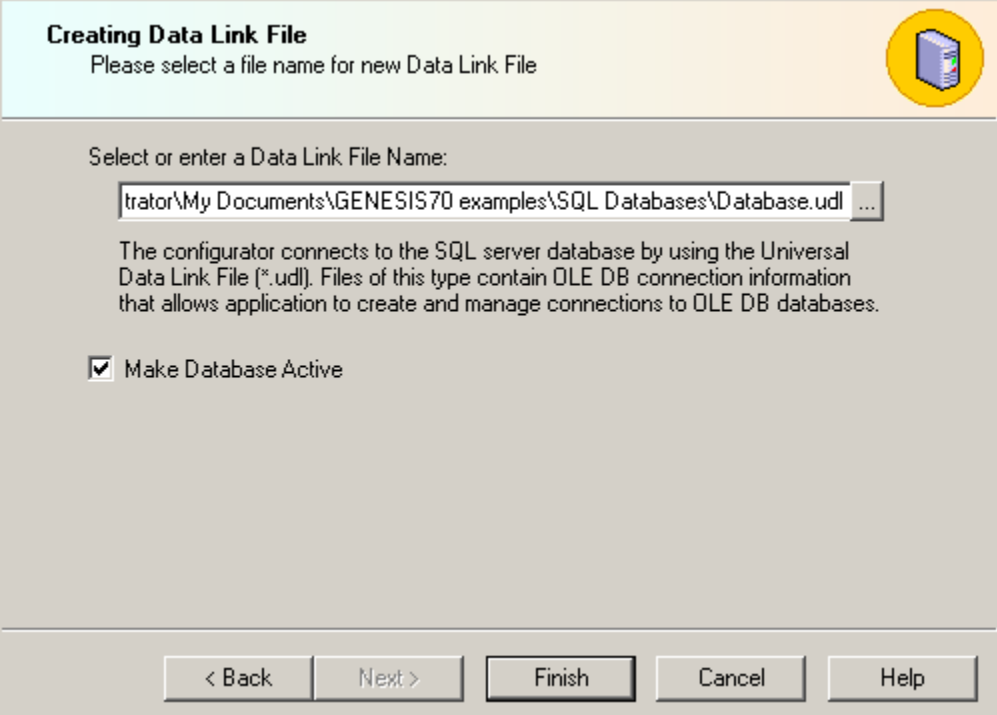
Specifying the Database Location and Properties

Under the **Database Properties** section, specify an initial size for the database, which should be as large as possible. You can also specify a **Database Growth** option (in megabytes) or as a percentage of the total size. MSDE servers are capable of growing the database on the fly to store more data. However, if this operation is performed frequently, the overall system performance may decrease. Choosing an initially large database size and a corresponding database growth option can drastically improve system performance.

Under the **Log File Properties** section, you can also modify the settings for the database transaction log file. Specify a **Log File Growth** option (in megabytes) or as a percentage of the total size. Again, a sufficient initial size setting can greatly improve performance. The default options should be adequate for most applications with a small to medium size load.

Click the **Next** button to create the new SQL Server database.

6. The Configurator uses Universal Data Link (.udl) files to connect to the Microsoft SQL Server database. These .udl files contain OLE database connection information that allows the Configurator to create and manage connections to OLE databases. Enter a name and directory path for the new .udl file in the **Data Link File Name** field, as shown in the figure below. You can browse for a directory by clicking the ... button to the right of this field. Click the **Next** button to continue. If you want this new database to be the active configuration database, check **Make Database Active**.



Creating Data Link File
Please select a file name for new Data Link File

Select or enter a Data Link File Name:

trator\My Documents\GENESIS70 examples\SQL Databases\Database.udl ...

The configurator connects to the SQL server database by using the Universal Data Link File (*.udl). Files of this type contain OLE DB connection information that allows application to create and manage connections to OLE DB databases.

☒ Make Database Active

< Back Next > Finish Cancel Help

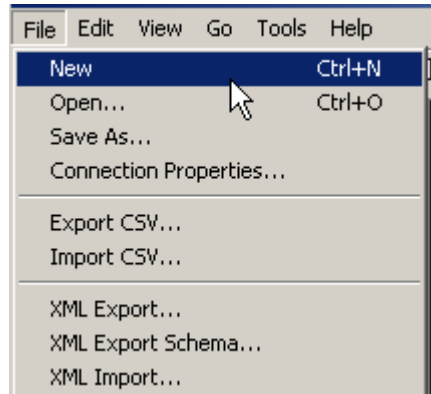
Creating a Universal Data Link File

7. Click the **Finish** button. The new database is created and opened in the Configurator.

Adding a Configuration Database to an Existing SQL Server Database

The Configuration Database Wizard also gives you the option to add the configuration database structure to an existing Microsoft SQL Server database:

1. Select **New** from the **File** menu, as shown in the figure below.



Creating a Configuration Database

2. The introduction screen for the Configuration Database Wizard appears. Click the **Next** button to continue.
3. You have two options for creating your new database, as shown in the figure below. Select **Create new configuration in MS SQL Server database**. Click the **Next** button to continue.

Which type of database would you like to use?
This wizard will create configuration database depending on the choice below.



Choose the type of database you would like to create:

☐ Create new configuration in MS Access database
This will create configuration in a single file (.mdb). Configurator can open this file directly (no other files needed).

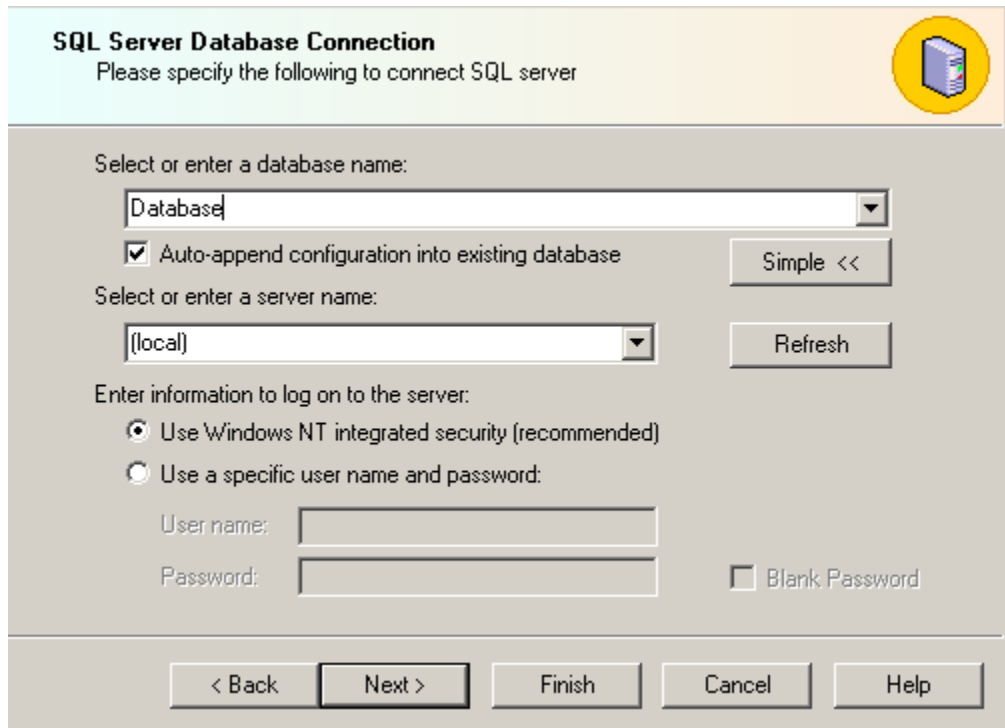
☒ Create new configuration in MS SQL Server database
This will create configuration in MSDE or MS SQL server. Configurator has to use univesal datalink file (.udl) to access this database.

< Back Next > Finish Cancel Help

Creating a Microsoft SQL Server Database

4. To connect to a SQL Server database, select the existing database from the drop-down list, as shown in the figure below. Check the **Auto-append configuration into existing database** check box. In the **SQL Server** field, select the local SQL Server on which to create the database. Type a name for the database in the **Database Name** field. If necessary, enter a user name and password to log on to the SQL Server. (**Note:** It is recommended that you use Windows NT integrated security.) Click the **Next** button to continue.

Note: Usually you have only one instance of SQL Server running on the local node. In this case, the drop-down list under **SQL Server Name** has only one option: "(local)." However, it is possible to run multiple SQL Server instances on the local node, in which case the **SQL Server** field lists all those SQL Server instances: "(local)" for the default instance and "node_name/instance_name" for all others. The drop-down list may show SQL Server instances on other nodes as well.



SQL Server Database Connection
Please specify the following to connect SQL server

Select or enter a database name:

☒ Auto-append configuration into existing database

Select or enter a server name:

Enter information to log on to the server:
☒ Use Windows NT integrated security (recommended)
☐ Use a specific user name and password:
 User name:
 Password:
☐ Blank Password

Connecting to an Existing SQL Server Database

5. If the existing database already has configuration tables, you have the following options, as shown in the figure below:
- **Select a different database name (recommended):** This option allows you to rename the database without affecting the existing database as well as create a new Data Link (.udl) file.
 - **Use the existing database structure:** This option preserves the content of the existing database and allows you to create a new Universal Data Link (.udl) file.
 - **Override all existing configuration tables:** This option expunges the content of the existing database and allows you to create a new Universal Data Link (.udl) file.

Note: Overriding the database may disable other applications that also use the database.

Click the **Next** button to continue.

One or more configuration tables already exist in chosen database
 What would you like to do?



☒ **Select different database name (recommended)**
 This will provide you to specify different database name. Use this option when you are not sure what is the content of chosen database.

☐ **Use database structure as it is**
 This will not change anything in selected database. Use this option if you want to create a new datalink (.udl) file only.

☐ **Override all existing configuration tables**
 Caution: This may cause that other application will stop working. Use this option when you are absolutely sure that the database is not use by other applications.

< Back

Next >

Finish

Cancel

Help

Adding the Configuration to an Existing SQL Server Database

- If you chose **Select a different database name (recommended)**, you are directed back to the SQL Server Database Connection dialog box, as shown in the figure below. Enter a new name for the database, and then click **Next**.

SQL Server Database Connection
 Please specify the following to connect SQL server



Select or enter a database name:

Renamed Database

☒ Auto-append configuration into existing database

Simple <<

Select or enter a server name:

(local)

Refresh

Enter information to log on to the server:

☒ Use Windows NT integrated security (recommended)
☐ Use a specific user name and password:

User name:
 Password:

☐ Blank Password

< Back

Next >

Finish

Cancel

Help

Renaming the Existing SQL Server Database

7. Specify a directory path location in which to create the database, as shown in the figure below. You can either use the default SQL Server database folder, or you can click the ... button and browse for a specific folder.

SQL Server Database Creation
Please specify the following to create database files

Specify database files location:

☒ Use default SQL server database folder (recommended)
☐ Use specific folder (local node only)

C:\Program Files\Microsoft SQL Server\MSSQL\Data\ ...

Database files properties: Simple <<

Database File
Initial Size (MB): 1
File Grow: ☒ AutoShrink
☐ In MB 1
☒ By Percent 20

Log File
Initial Size (MB): 1
File Grow: ☐ In MB 1
☒ By Percent 20

< Back Next > Finish Cancel Help

Specifying the Database Location and Properties

Under the **Database Properties** section, specify an initial size for the database, which should be as large as possible. You can also specify a **Database Growth** option (in megabytes) or as a percentage of the total size. MSDE servers are capable of growing the database on the fly to store more data. However, if this operation is performed frequently, the overall system performance may decrease. Choosing an initially large database size and a corresponding database growth option can drastically improve system performance.

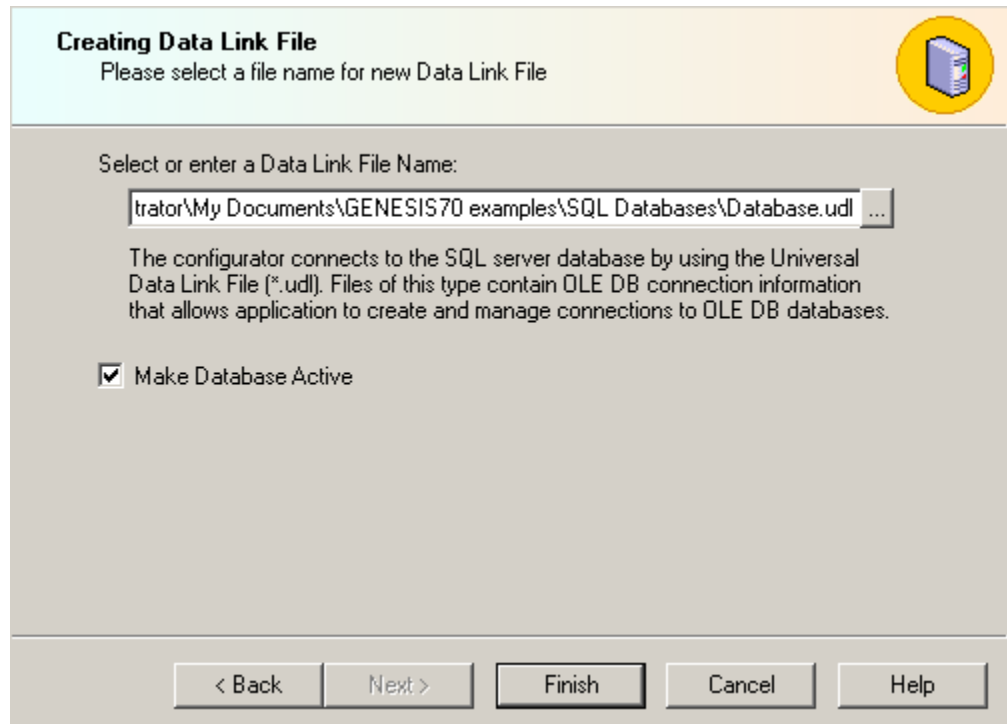
Under the **Log File Properties** section, you can also modify the settings for the database transaction log file. Specify a **Log File Growth** option (in megabytes) or as a percentage of the total size. Again, a sufficient initial size setting can greatly improve performance. The default options should be adequate for most applications with a small to medium size load.

Click the **Next** button to create the new SQL Server database.

8. The Configurator uses Universal Data Link (.udl) files to connect to the Microsoft SQL Server database. These .udl files contain OLE database connection information that allows the Configurator to create and manage connections to OLE databases. Enter a name and directory path for the new .udl file in the **Data Link File Name** field, as shown in the figure below. You can browse for a directory by clicking the ... button to the right of this field.

Click the **Next** button to continue. If you want this new database to be the active configuration database, check **Make Database Active**.

Note: If you chose to **Use the existing database structure** or to **Override all existing configuration tables**, you will still need to create a new Data Link file.



Creating a Universal Data Link File

9. Click the **Finish** button. The new database is created and opened in the Configurator.

Toolbars

The Alarm Logger Configurator contains two toolbars: a **Standard** toolbar and a **Data Manipulation** toolbar.

Standard Toolbar

To show or hide the **Standard** toolbar, select **Toolbars > Standard Buttons** from the **View** menu. The **Standard** toolbar, shown below, contains the following command buttons.

New: Creates a new configuration database.

	Open: Opens an existing configuration database.
	Export Data: Exports configuration data to a text file (.txt) or a Microsoft Excel file (.csv).
	Import Data: Imports configuration data from a text file (.txt) or a Microsoft Excel file (.csv).
	Back: Moves the cursor back to the previously selected item in the tree control.
	Next: Moves the cursor to the next item in the tree control.
	Up One Level: Moves up one level in the tree control.
	Cut: Deletes current selection, sending it to the clipboard.
	Copy: Copies the current selection to the clipboard.
	Paste: Pastes the current contents of the clipboard.
	Large Icons: Displays items as large icons.
	Small Icons: Displays items as small icons.
	List: Displays items as a list.
	Details: Displays items as a list with details.
	Dialog View: Displays additional configuration options.
	Global Refresh: Refreshes the data for the entire Configurator screen.
	Start/Stop Logger: Toggle runtime mode for the Alarm Logger.
	About: Displays information about the application.
	Help: Displays context-sensitive help.
	Help Topics: Launches online Help for the application.

Data Manipulation Toolbar

To show or hide the **Data Manipulation** toolbar, select **Toolbars > Data Manipulation Buttons** from the **View** menu. The **Data Manipulation** toolbar, shown below, contains the following command buttons.

	New Configuration: Creates a new Logger configuration under the Configurations tree.
	New Column Definition: Creates a Column Definition for a Logger configuration.
	New Node: Creates a new node under the Nodes tree control.
	New Link to Configuration: Creates a new link to a Logger configuration.
	Multiply: Creates a copy of the selected item in the tree control.
	Delete: Deletes the selected item in the tree control.

Menus

The menu bar of the Alarm Logger Configurator contains the following menus:

- File
- Edit
- View
- Go
- Action
- Tools
- Help

Note: You can also access many of the menu commands by right-clicking items in the tree control of the Configurator and selecting command functions from the pop-up menus.

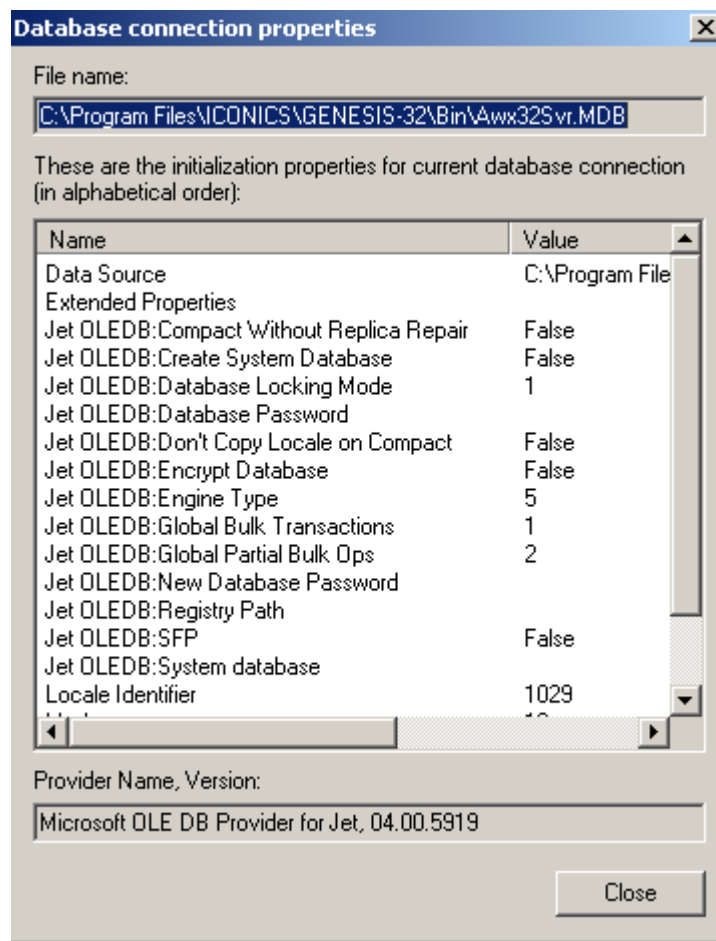
File Menu

The **File** menu of the AlarmWorX32 Logger Configurator contains the following commands:

Command	Shortcut Key	Function
New	CTRL+N	Creates a new configuration database.
Open	CTRL+O	Opens a new .mdb (Microsoft Data Access) or .udl (Microsoft Data Link) file.
Save As		Saves the database under a different name.
Connection Properties		Opens the Database Connection Properties dialog box.
CSV Export		Exports data from a selected .csv file.
CSV Import		Imports data from a selected .csv file.
XML Export		Exports configuration data to an XML file.
XML Export Schema		Exports configuration data to an XML Schema file.
XML Import		Imports configuration data from an XML file.
XML Validate		This feature does not import an XML data file, but it will try to validate its structure using stored XML schema. Once it passes this validation, the XML file is acceptable for import by the Configurator.
Make Active		Activates the current configuration database for use by the alarm server.
Exit		Closes the applications and stops data logging if it started within the Configurator.

Database Connection Properties

Selecting **Connection Properties** from the **File** menu opens the **Database Connection Properties** dialog box, shown below, which lists the initialization properties for the current database connection.

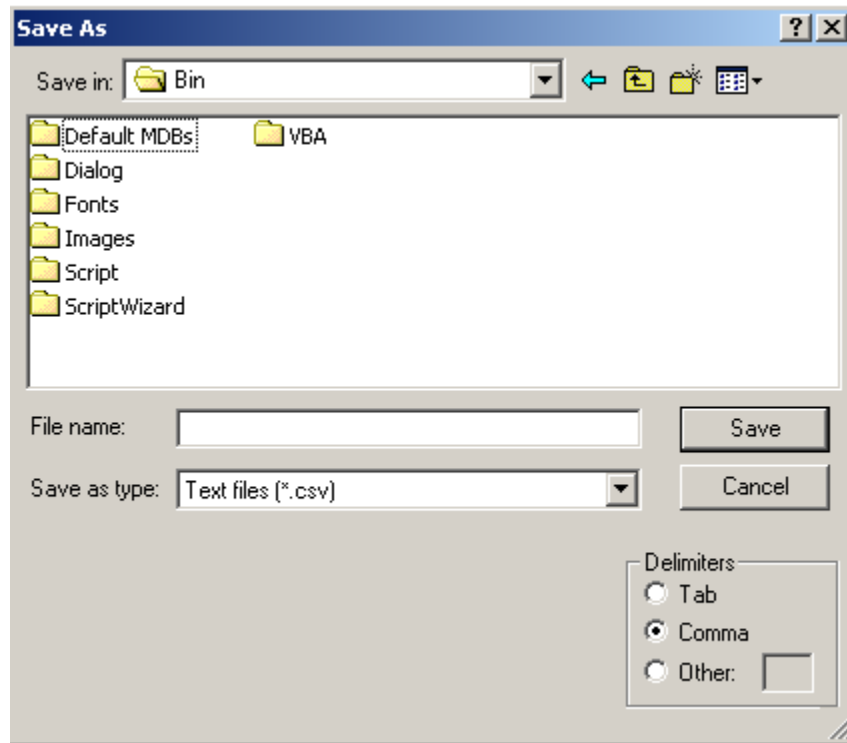


Database Connection Properties Dialog Box

Exporting Configuration Data

Exporting Data to a Text or CSV File

The Configurator offers the flexibility of exporting data from your configuration database to a text (.txt) file or a Microsoft Excel (.csv) file. To export data, select **Export** from the **File** menu. This opens the **Export Configuration Data to File** dialog box, as shown in the figure below. You can then specify the delimiters for exporting the data. Unless you specify delimiters in the **Export Configuration Data to File** dialog box, the file uses **Commas** as delimiters by default. Each group contains headings and columns that provide information about each item, such as descriptions and associated translations and expressions. It also provides the "tree" pathway for each item. Choose the directory to which you want to export the data from your database. In the **Save As Type** field, choose the file type (.txt or .csv) that you would like to save.



Exporting Configuration Data

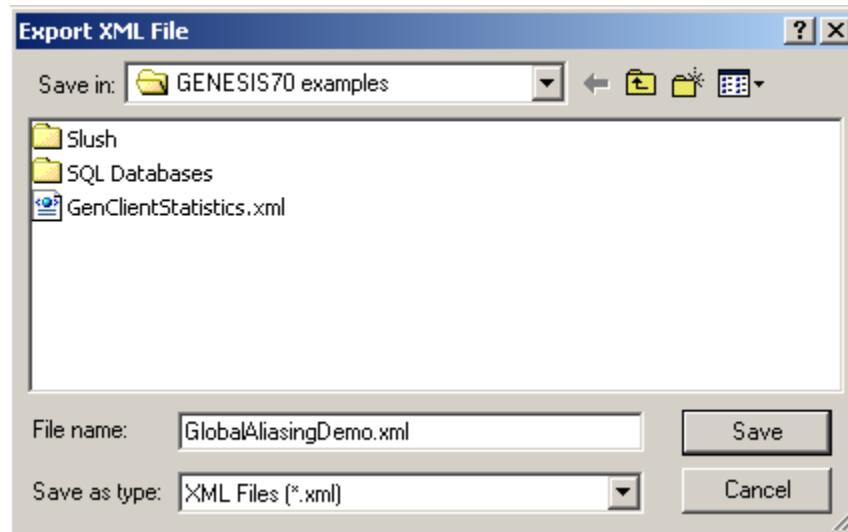
Exporting Data to an XML File

The Configurator also allows you to export data from your configuration database to an XML file. The XML export/import functionality was mainly developed for Windows platforms that do not support databases (e.g. Windows CE and Windows Embedded). XML has the following advantages over the CSV import/export function:

- XML has a standardized format, unlike the text/CSV format, which uses various delimiters (e.g. TAB instead of commas, strings could not accept all characters, etc.)
- XML is language-independent, whereas CSV converts date/time, floats, and currency fields according to local settings in Windows. For example, using CSV, you cannot export data on German Windows and import it on English windows without making changes
- Windows has an installed automation object that has the capability to work with XML. Thus, programmers can create/modify their configurations outside the Configurator using Visual Basic, if desired.
- XML supports schemas. A **schema** is a special XML file that specifies the data structure of an XML data file.

To export data, select **XML Export** from the **File** menu. This opens the **Export XML File** dialog box, as shown in the figure below. Give the file a name, and then choose the directory to which you want to export the data from your database. Click **Save**.

Note: You can also export configuration data to an XML Scheme file by selecting **XML Export** from the **File** menu.



Exporting Configuration Data to an XML File

Importing Configuration Data

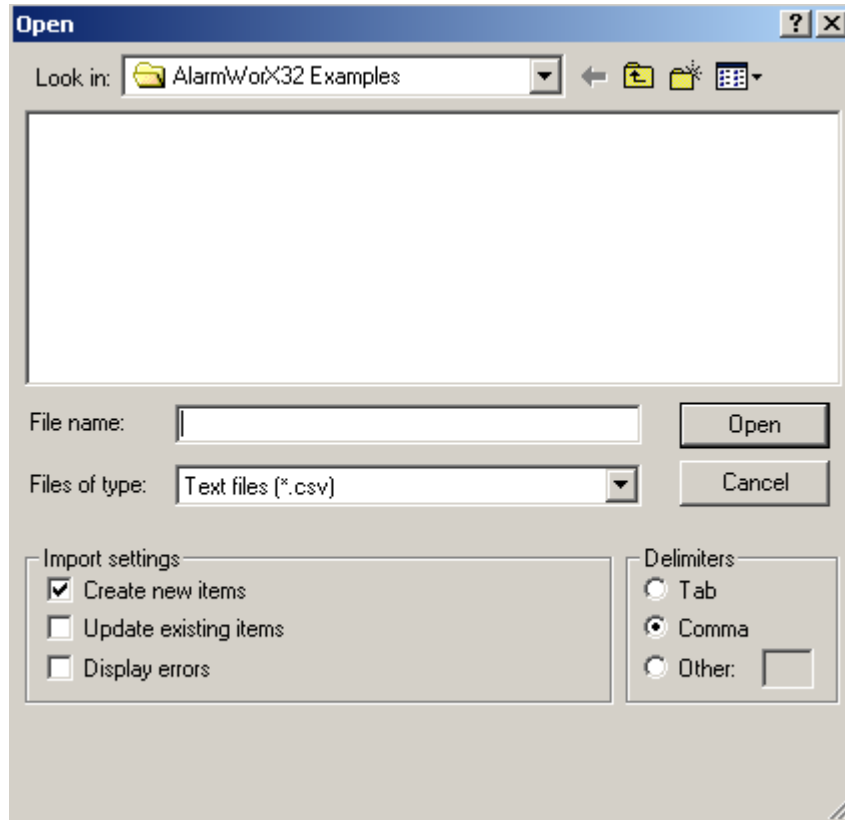
Importing Data From a Text or CSV File

The Configurator offers the flexibility of importing data from a text (.txt) file or a Microsoft Excel (.csv) file to your configuration database. To import data, select **Import CSV** from the **File** menu. This opens the **Import Configuration Data From File** dialog box, shown below. You can then specify the delimiters and choose from the following import settings:

- **Create new items.** When the import file contains items that are not yet in the configuration database, then it creates them. Otherwise it skips these items.
- **Update existing items.** When the import file contains items that are in the configuration database, then it updates them using data from the import file. Otherwise it skips these items.

Note: Either **Create new items** or **Update existing items** must be selected. Otherwise there is nothing to import.

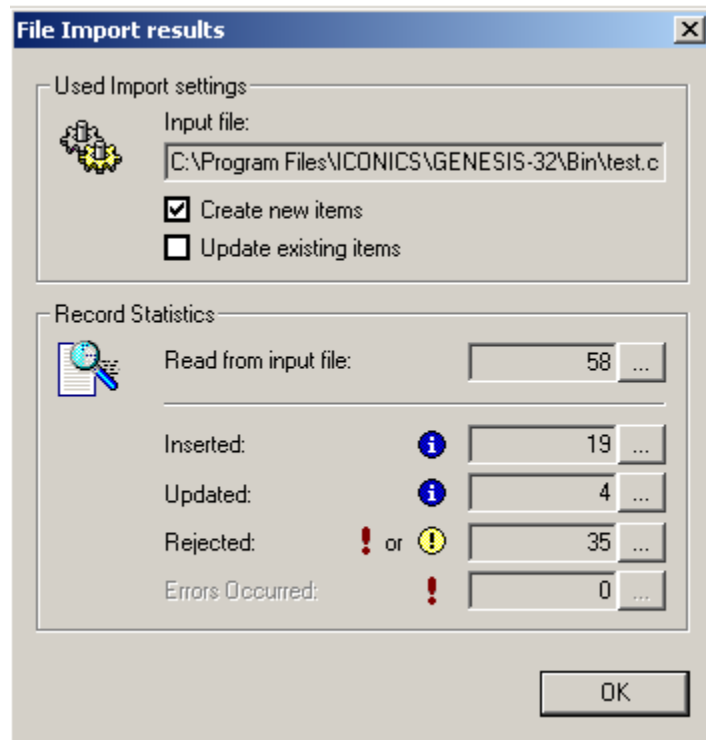
- **Display errors.** When this item is checked, the Configurator shows a dialog box if an error occurs, and then asks you if you want to proceed with the import. When it is not checked, it skips all items where an error occurred.



Importing Configuration Data

When you have selected a file to import, click **Open**. When the import is completed, the **File Import Results** dialog box opens, as shown below. This shows the import settings, including the input file name. It also provides a summary of the import, including how many items were inserted, updated, or rejected, and shows how many errors occurred.

Click the ... button to the right of each field to get the details view of the import results, as shown below. This view shows the specific items that were inserted, updated, or rejected, as well as a description of any errors that occurred.



File Import Results Dialog Box

Importing Data From an XML File

The Configurator allows you to import data from your configuration database to an XML file. The XML export/import functionality was mainly developed for Windows platforms that do not support databases (e.g. Windows CE and Windows Embedded). XML has the following advantages over the CSV import/export function:

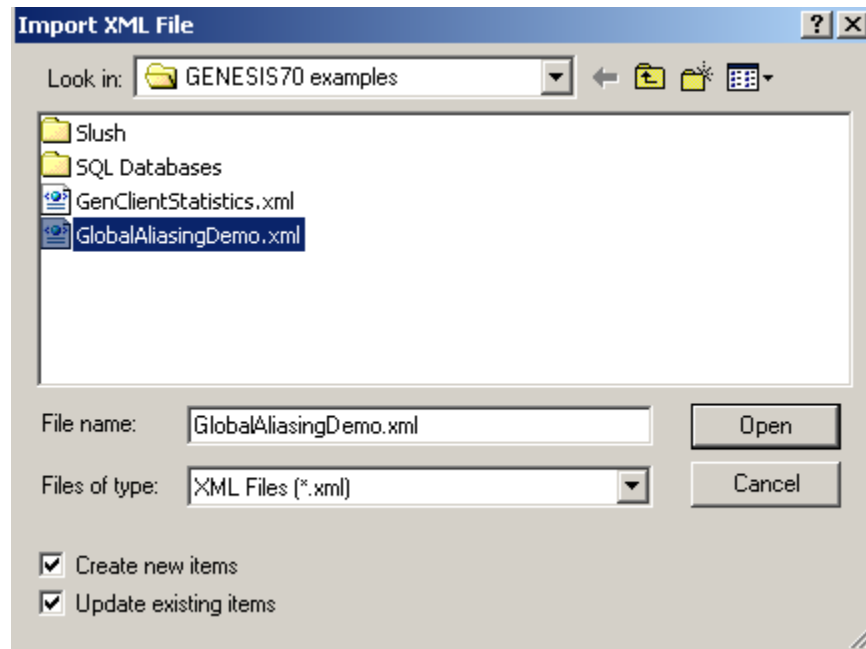
- XML has a standardized format, unlike the text/CSV format, which uses various delimiters (e.g. TAB instead of commas, strings could not accept all characters, etc.)
- XML is language-independent, whereas CSV converts date/time, floats, and currency fields according to local settings in Windows. For example, using CSV, you cannot export data on German Windows and import it on English windows without making changes
- Windows has an installed automation object that has the capability to work with XML. Thus, programmers can create/modify their configurations outside the Configurator using Visual Basic, if desired.
- XML supports schemas. A **schema** is a special XML file that specifies the data structure of an XML data file.

To import data, select **XML Import** from the **File** menu. This opens the **Import XML File** dialog box, as shown in the figure below. Give the file a name, and then choose the directory from which you want to import the data. You can then specify the delimiters and choose from the following import settings. Click **Open**.

- **Create new items.** When the import file contains items that are not yet in the configuration database, then it creates them. Otherwise it skips these items.
- **Update existing items.** When the import file contains items that are in the configuration database, then it updates them using data from the import file. Otherwise it skips these items.

Note: Either **Create new items** or **Update existing items** must be selected. Otherwise there is nothing to import.

Note: Selecting **XML Validate** from the **File** menu does not import an XML data file, but it will try to validate its structure using stored XML schema. Once it passes this validation, the XML file is acceptable for import by the Configurator.

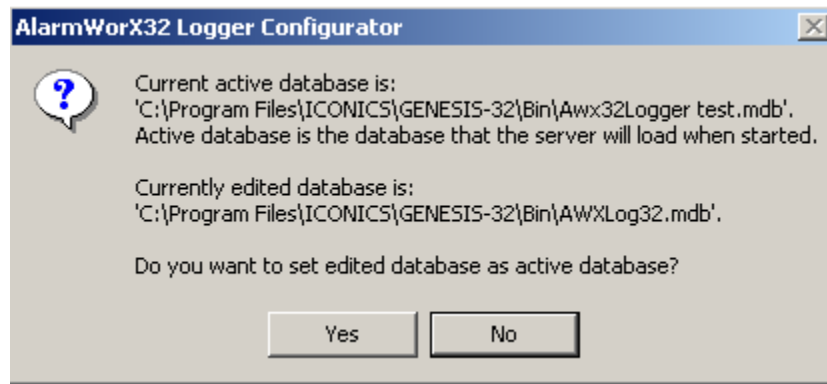


Importing Configuration Data From an XML File

Activating the Database

Once your configuration is complete, you need to make sure that it is the active database. The database that is currently active is the one that the alarm server will use. To make the current database active, select **Make Active** from the **File** menu. If the **Make Active** selection is grayed out, then the current database is already the active database.

A dialog box appears, as shown in the figure below, showing both the current active database and database that is currently being edited, as shown in the figure below. To set the edited database as the active database, click the **Yes** button.



Activating the Database

Edit Menu

The **Edit** menu commands are listed in the table below.

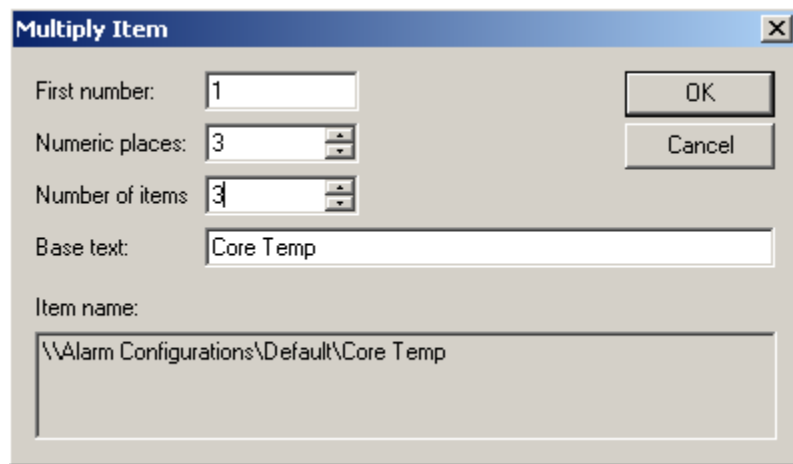
Edit Menu Commands

Command	Shortcut Keys	Function
New		Creates a new item depending on what is selected in the tree control.
Rename	CTRL+R	Renames the selected item.
Multiply	CTRL+M	Opens the Multiply Item dialog box (see below), which allows you to multiply an item in the tree control.
Delete	CTRL+DEL	Deletes the selected object.
Cut	CTRL+X	Cuts the selected object from the view and places it on the clipboard.
Copy	CTRL+C	Copies the selected object to the clipboard.
Paste	CTRL+V	Pastes the last object placed on the clipboard.
Select All	CTRL+A	Selects all objects in a list. The selection is shown in the upper-right-hand section of the viewer.
Invert Selection		Unselects all selected items and selects all unselected items in a list in the upper-right-hand section of the viewer.

Multiplying Items

The Alarm Logger Configurator allows you to multiply items in the tree control. Multiplication provides a simple way of developing configurations where there are many similar items in a given category. To multiply an item:

1. Select the item in the tree control that you wish to multiply.
2. Either right-click the item and select **Multiply** from the pop-up menu, or select **Multiply** from the **Edit** menu. This opens the **Multiply Item** dialog box, shown below.



Multiply Item Dialog Box

3. When the items are multiplied, they are all given a base name followed by a number. The default base text is the name of the item selected for multiplication. To modify the base text, change the **Base Text** field appropriately.
4. In the **First Number** field, specify the number to appear next to the first multiplied item.
5. In the **Number of Items** field, specify how many items you wish to create.
6. In the **Numeric Places** field, specify the minimum length of each number to append. Values that take up less space than the specified amount of numeric places will have zeros before the number.
7. If you want to multiply all subfolders as well, check the **Including Subtree** check box.
8. Click the **OK** button to do the multiplication. The example configuration shown in the **Multiply Item** dialog box above creates three new OPC Data folders with the following names:
 - Core Temp001
 - Core Temp002
 - Core Temp003

All subfolders will also be multiplied.

View Menu

The **View** menu commands are listed in the table below.

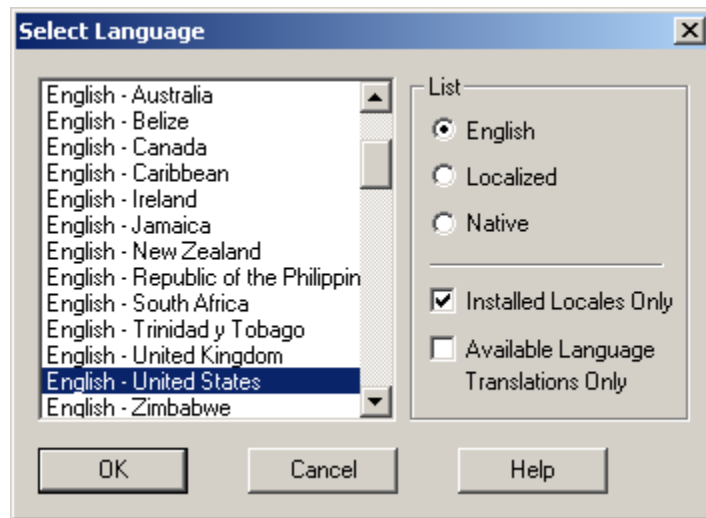
View Menu Commands

Command	Shortcut Keys	Function
Toolbars		Toggles the standard and data manipulation toolbars.
Status Bar		Toggles the status bar.
Large Icons	F7	Displays items as large icons.
Small Icons	F8	Displays items as small icons.
List	F9	Displays items as a list.
Details	F10	Displays items as a list along with detailed information about the configuration of each item.
Dialog View	F11	Toggles the configuration window (right-hand pane).
Sort By		Displays a list of options for sorting the columns in the right-hand pane of the screen. The options listed depend on the level within the view.
Show/Hide Columns		Displays a list of options that you can choose to show or hide in the view.
Select Language		Opens the Select Language dialog box (see below). Choose the language you wish to use for your system (Unicode version only) and click OK . For navigation purposes, use the buttons and check boxes in the List section.
Global Refresh	F5	Refreshes the data for the entire Configurator screen.
Subtree Refresh	CTRL+F5	Refreshes only the data contained in the currently selected subtree.

Selecting Languages

The **Select Language** function on the **View** menu allows you to choose which language to use in your display. Choosing **Select Language** from the **View** menu opens the **Select Language** dialog box, shown in the figure below.

Note: A language resource .dll is required for language switching.



Select Language Dialog Box

Define the parameters listed in the table below. Then click **OK** to return to the work area.

Select Language Parameters

Parameter	Description
List	Lists available languages. Depending on which item you have selected, the view on the left will change. If English is checked, the languages will appear as their English name. If Localized is checked, the languages will appear with the native country in parentheses (for languages with several dialects only). When Native is checked, the languages are displayed the way they would be written in that language.
Installed Locales Only	If this is checked, local languages appear in the box.
Available Language Translations Only	Checking this box allows you to choose from available language translations only.

Go Menu

The **Go** menu commands are listed in the table below.

Go Menu Commands

Command	Shortcut Keys	Function
Back	CTRL+ALT+Left Arrow	Moves the cursor back to the previously selected item in the tree control.
Forward	CTRL+ALT+Right Arrow	Moves the cursor forward to the previously selected item in the tree control.
Up One Level		Moves the cursor up one level in the tree control.
Next Item	ALT+Down Arrow	Moves the cursor to the next item down in the tree control.
Previous Item	ALT+Up Arrow	Moves the cursor to the next item up in the tree view.
Expand Item	ALT+Left Arrow	Expands an item that contains a submenu.
Collapse Item	ALT+Right Arrow	Collapses an item that contains a submenu.
Page Up	ALT+PgUp	Moves the cursor up to the first item in the tree.
Page Down	ALT+PgDown	Moves the cursor down to the last visible item in the tree.
Home	ALT+Home	Moves the cursor up to the first item in the tree.
End	ALT+End	Moves the cursor down to the last visible item in the tree.
Next Pane	F6	Moves the cursor to the next pane.
Previous Pane	SHIFT+F6	Moves the cursor to the last pane used.

Action Menu

The **Action** menu contains the following commands:

Command	Function
Start Alarm Logger	Starts alarm data logging for the active configuration.
Stop Alarm Logger	Stops t alarm data logging for the active configuration.

Only one of the above options will be available at a time, depending on the state of the Alarm OPC Logger.

Tools Menu

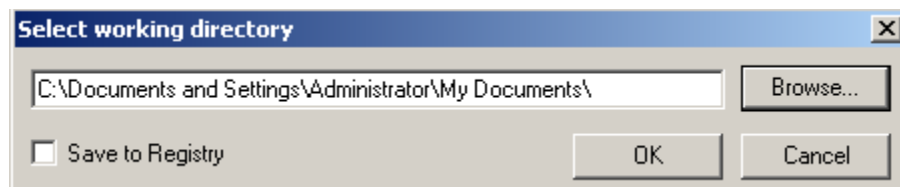
The **Tools** menu commands are listed in the table below.

Tools Menu Commands

Command	Function
Set Working Directory	Sets a working directory for all files relating to the saved configuration.
Options	Launches the Options dialog box.
Compact/Repair MS Access Database	Opens the Compact/Repair MS Access Database dialog box.

Setting the Working Directory

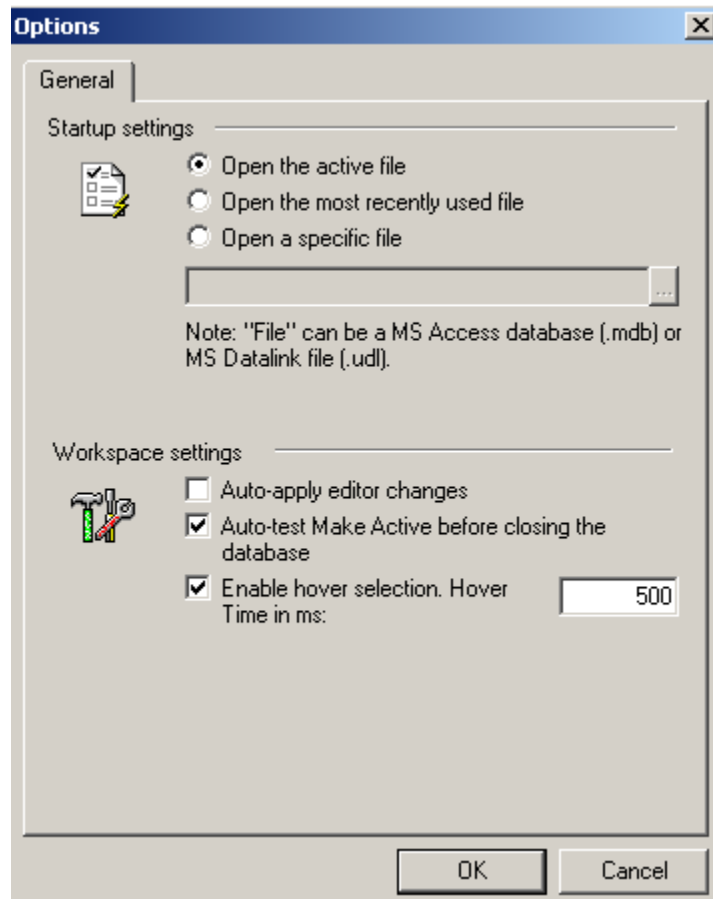
Selecting **Set Working Directory** from the **Tools** menu opens the **Set Working Directory** dialog box, shown below, which enables you to configure a custom directory in which all application configuration files will be stored and retrieved. Click **Browse** to select the directory.



Set Working Directory Dialog Box

Options

To choose additional settings, select **Options** from the **Tools** menu. This opens the **Options** dialog box, shown in the figure below. The **General** tab of the **Options** dialog box, sets the startup and workspace parameters for the Alarm Server Configurator.



Options Dialog Box: General Tab

Startup Settings

The **Startup Settings** options allow you to save regional settings in the registry so that they are applied each time you start the Configurator. This applies to the language settings as well as time and date settings. Select one of the following startup settings:

- **Open the active file:** Launches the currently active database upon startup.
- **Open the most recently used file:** Launches the recently opened database.
- **Open a specific file:** Specifies a Microsoft Access (.mdb) or Microsoft Datalink (.udl) database to launch upon startup. To select a database, click the ... button next to the text box and browse for the file. When this option is selected, the "active" database is overridden by the specified database.

Workspace Settings

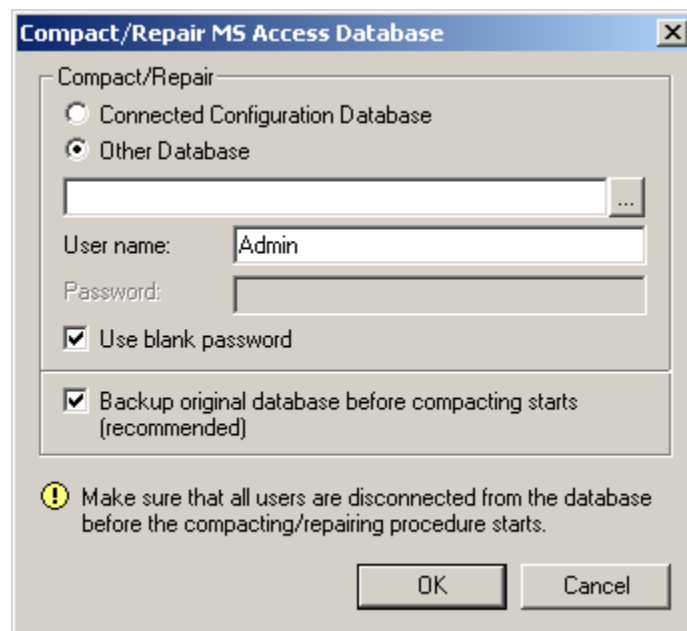
The following **Workspace Settings** are available:

- **Auto-apply editor changes:** Checking this option allows changes to the configuration database to be saved each time you switch dialogs without clicking on the **Apply** button or being shown a message asking if you would like to apply changes.
- **Auto-test Make Active before closing the database:** When this option is checked, each time you exit a currently open (nonactive) database you are asked whether you want to make the database active.
- **Enable hover selection:** Checking this option allows you to highlight an item by moving the mouse pointer over that item and keeping it there for a specified amount of time (in milliseconds).

Compacting and Repairing MS Access Databases

You can compact Microsoft Access databases, which can be either configuration databases or historical databases, using the **Compact/Repair MS Access Database** dialog box, shown in the figure below. To open this dialog box, select **Compact/Repair MS Access Database** from the **Tools** menu. Microsoft Access-based databases are subject to database fragmentation over time, and the support for the database will compact the target database, reclaim unused space, and drastically improve database performance.

Note: It is critical that no users or client applications are connected to the database at the time of compacting and that, if the **Backup Original Database** option is selected, there is plenty of available hard disk space.



Compact/Repair MS Access Database Dialog Box

Help Menu

The **Help** menu contains the following commands:

Command	Shortcut Key	Function
Help Topics	F1	Launches the online help for the Configurator.
What's This?	SHIFT+F1	Displays context-sensitive help topics.
About Application		Launches the ICONICS About Box , which contains information about the product version number, copyright, and available disk space. It also contains information about how to contact ICONICS.

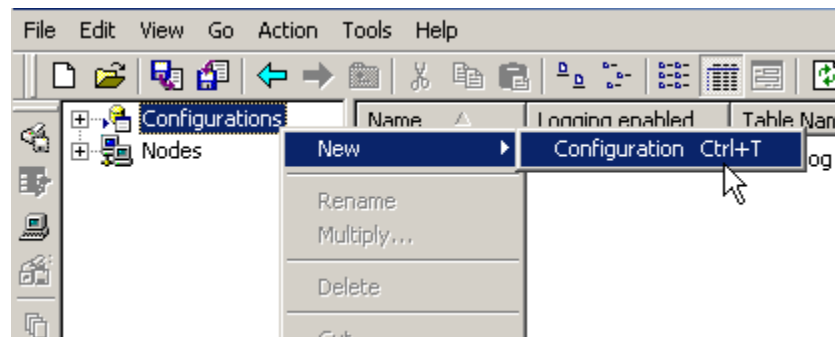
Logger Configurations

The configuration database contains one or more Logger configurations. Each configuration specifies a subset of alarms and events to be logged to a single database table and/or printer. Additionally, the configuration defines which data fields (columns) available are to be stored and/or printed.

Creating a New Logger Configuration

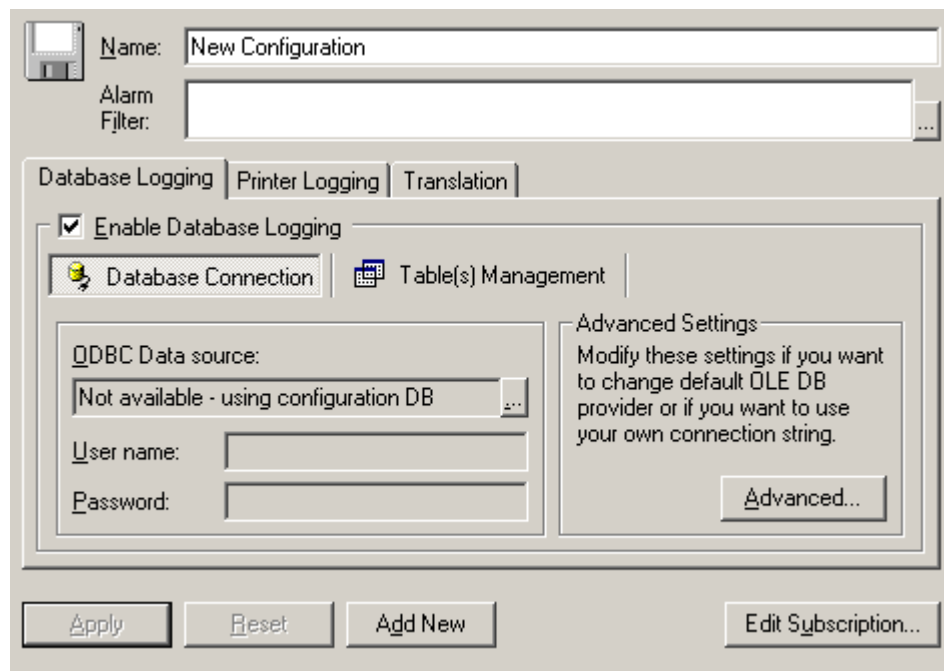
To create a new logger configuration:

1. Right-click on the **Configurations** tree control of the Configurator and select **New > Configuration** from the pop-up menu, as shown in the figure below.



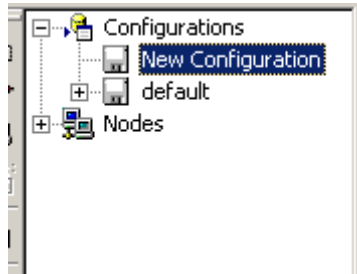
Creating a New Logger Configuration

2. The properties dialog box for the new configuration appears in the right-hand pane of the Configurator, as shown in the figure below.



Setting the Properties for the New Logger Configuration

3. In the **Name** field, type a name for the new configuration.
4. In the **Alarm Filter** field, you can click the ... button to open the **Expression Editor**, which allows you to define alarm filters.
5. When you have finished configuring the Logger configuration properties, click the **Apply** button. The new configuration appears under the **Configurations** tree control, as shown in the figure below.



New Logger Configuration Added to Tree Control

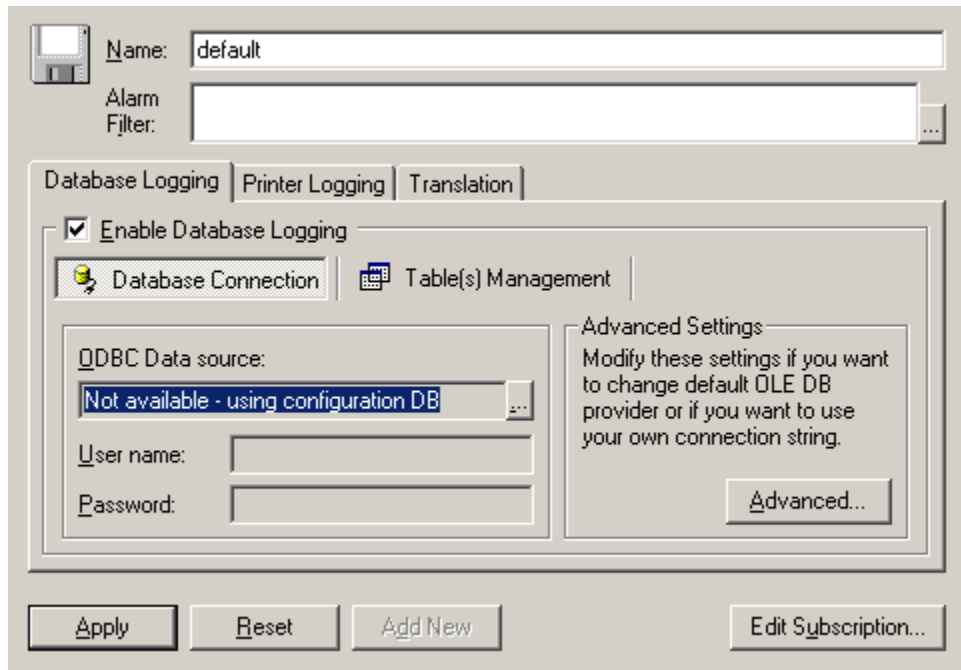
Logger Configuration Properties

Configure the following parameters for Logger configurations, as shown in the figure below.

Note: If **Dialog View** is not selected from the **View** menu, you will not see the configuration properties.

The configuration dialog contains three main tabs:

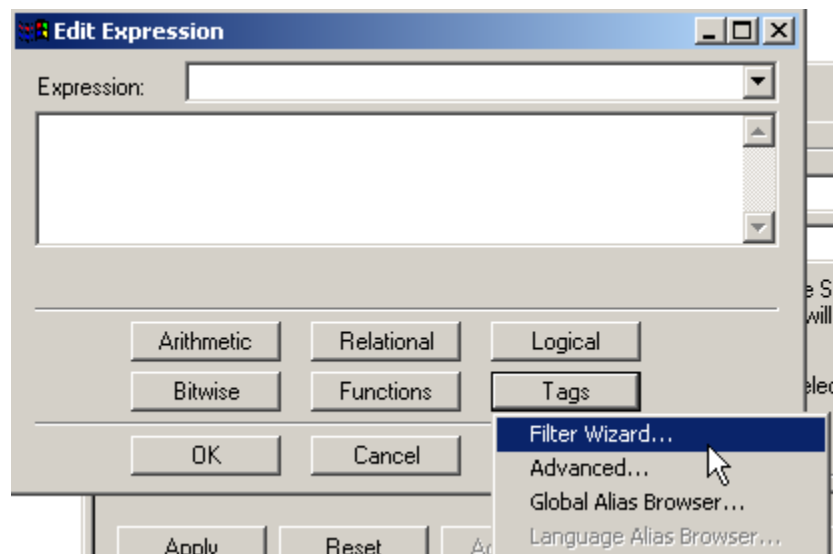
- **Database Logging**
- **Printer Logging**
- **Translation**



Logger Configuration Properties

Name: Type a name for the new configuration. The name must be unique.

Alarm Filter: Click the ... button to open the **Expression Editor**, shown in the figure below, which allows you to define alarm filters for the logger configuration. Please refer to the Expression Editor Help documentation for additional information.

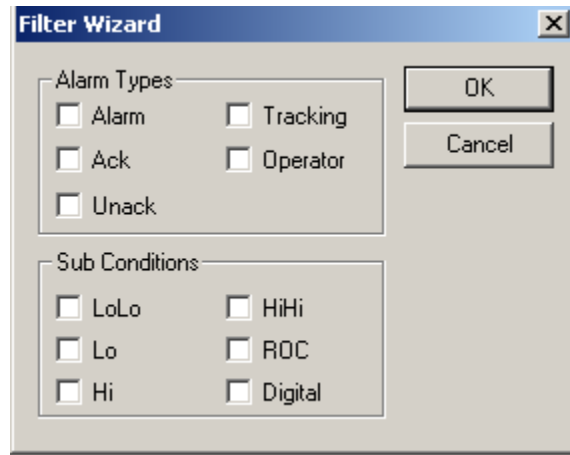


Expression Editor

Click on **Tags** button and select **Filter Wizard** from the pop-up menu. The **Filter Wizard**, shown in the figure below, allows you to choose from the following to items enter in your expression. Select one or more items, and then

click **OK**. The filter string is automatically inserted into the Edit Expression dialog box.

- Alarm Types: Alarm, Ack, Unack, Tracking, and Operator
- Subconditions: LoLo, Lo, Hi, HiHi, Rate of Change, and Digital



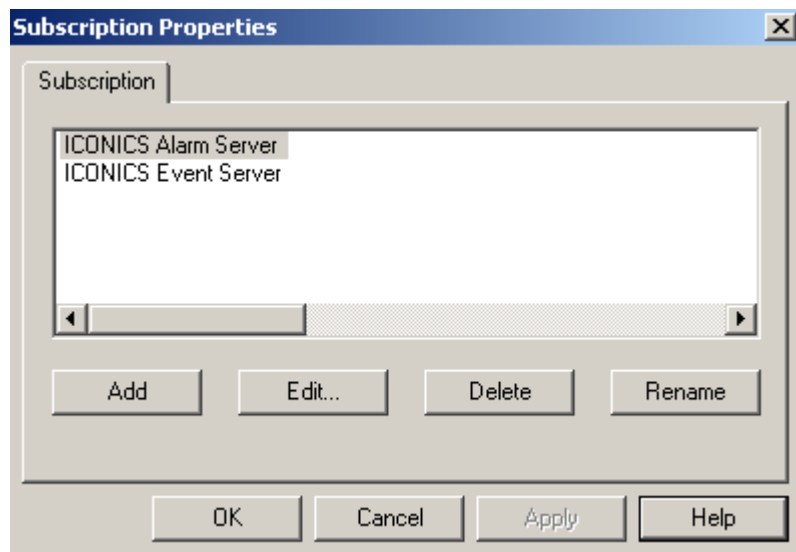
Filter Wizard

Apply Button: Commits changes to the database.

Reset Button: Restores the default configuration settings.

Add New Button: Adds a new Logger configuration to the database.

Edit Subscription Button: Opens the **Subscription Properties** dialog box, shown in the figure below, which allows you to subscribe to one or more OPC Alarm and Events servers and select the desired additional attributes to be included for logging and/or printing. Select a server and click the **Edit** button. This process is identical to the AlarmWorX32



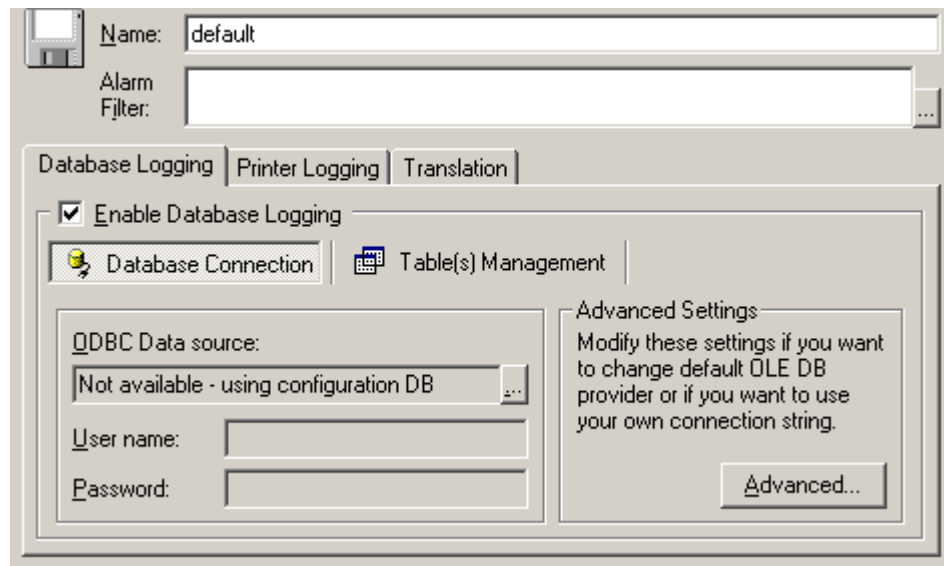
Subscription Properties Dialog

Database Logging Tab

The **Database Logging** tab of the configuration properties, shown in the figure below, contains the following two tabs:

- **Database Connection**
- **Table Management**

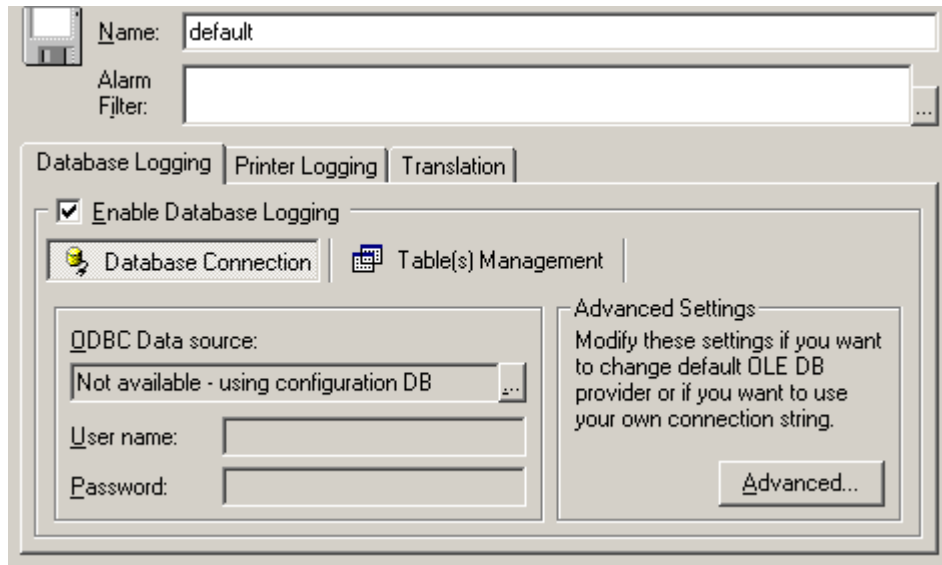
Note: In order to log alarm data to the specified database, you must check the **Enable Database Logging** check box, as shown in the figure below.



Database Logging Tab

Database Connection

The **Database Connection** section of the **Database Logging** tab, shown in the figure below, contains the following two tabs, establishes a connection to an ODBC data source.



Database Logging: Establishing a Database Connection

The **ODBC Data Source** field specifies the database to use for logging. If this field is blank (the default), logging will take place to the active configuration database. It is highly recommended that you change the default database and log alarm data to a different database from the configuration database.

User Name: Enter the user name to use (if any) when connecting to the logging database.

Password: Enter the password that goes with the **User Name**.

Advanced Button: Clicking **Advanced** opens the **Advanced Database Settings** dialog box, shown in the figure below, which allows the **OLE DB provider name** and the **Connection string** to be edited directly. This capability is provided to allow connection to databases directly via OLE DB (without using ODBC).



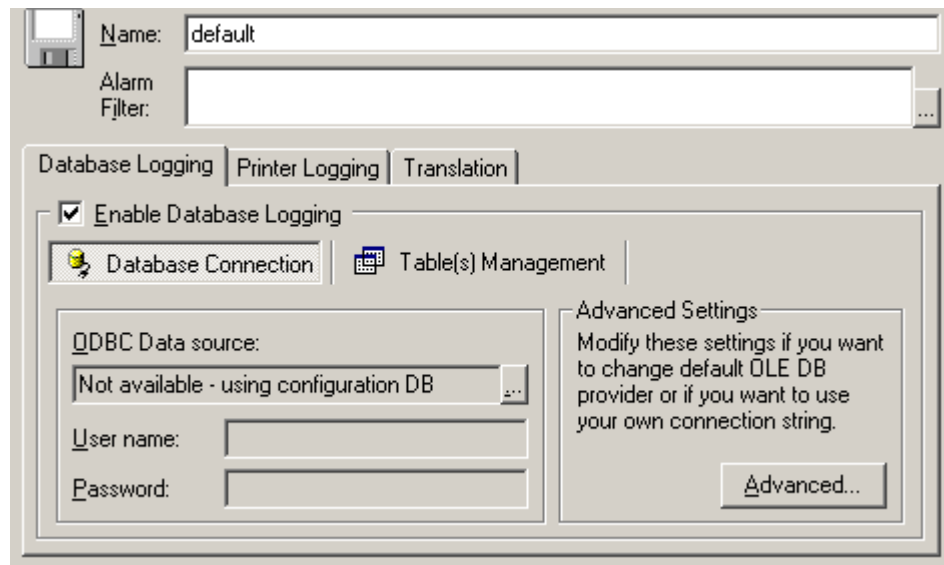
Advanced Database Settings Dialog

Creating a New ODBC Data Source

To set up data logging, you must first configure the underlying database and set up ODBC data sources. This can be done within the Logger Configurator by connecting a database group to an existing database or a new database. It is very critical that the PCs on which the AlarmWorX32 Logger resides have full access to the target database to create, add, delete, and update tables. Otherwise, data logging cannot be accomplished.

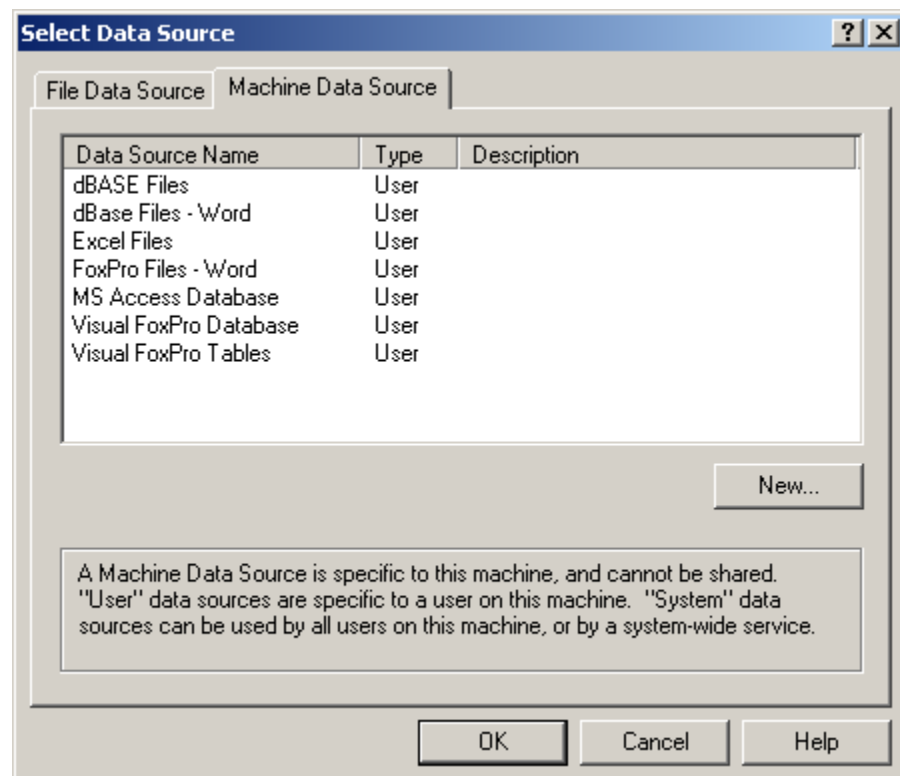
To set up a data-logging source in the Configurator:

1. Click the ... button in the **ODBC Data Source** field of the **Database Connection** section of the **Database Logging** tab, as shown in the figure below.



Database Logging: Establishing a Database Connection

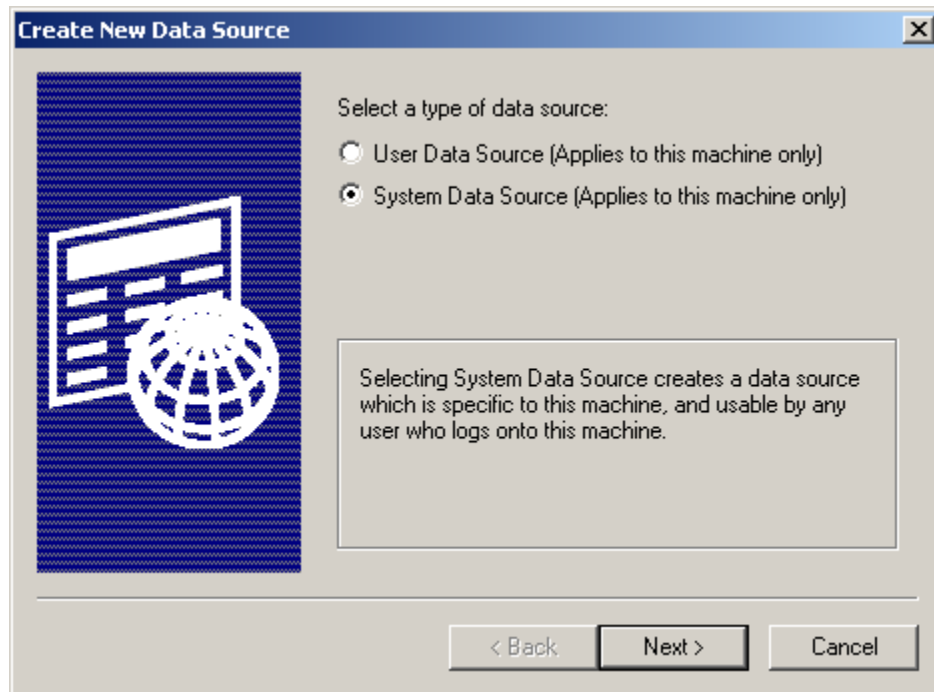
2. This opens the **Select Data Source** dialog box, shown below. Choose the **Machine Data Source** tab and an existing database.



Select Data Source Dialog Box

3. Or you can create a new database by clicking **New** in the **Machine Data Source** tab of the **Select Data Source** dialog box. This opens the

Create New Data Source wizard, shown below, which enables you to configure a new database.



Create New Data Source Wizard

It is recommended that you configure the new database as a **System Data Source** so it is available to any user logged into the specific PC. Click **Next** to select the corresponding ODBC driver, such as Microsoft Access, Microsoft SQL Server, or Oracle.

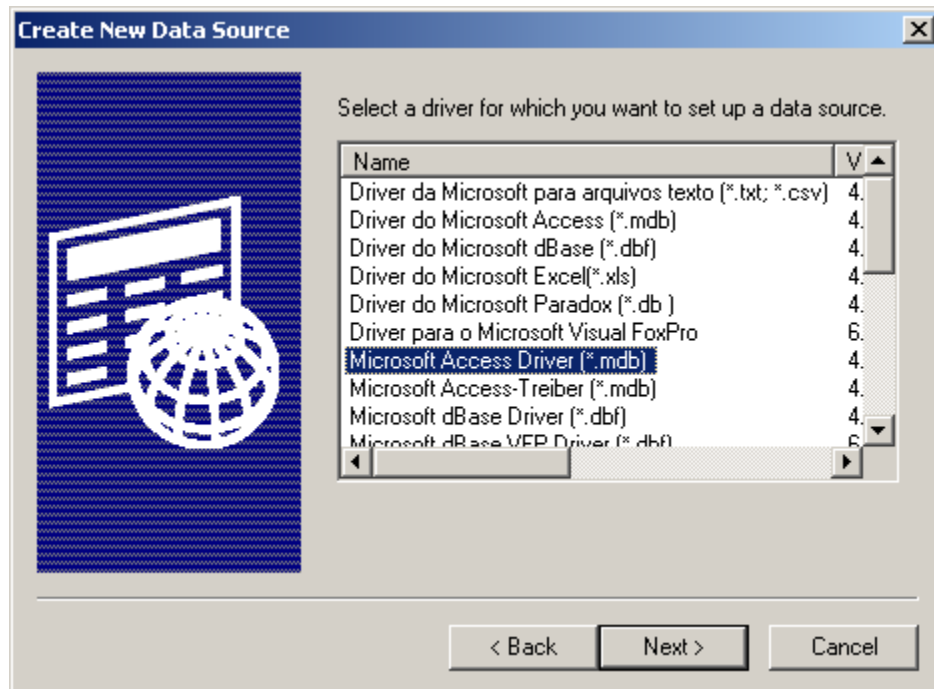
We recommend configuring the new database as a System Data Source so it is available to any user logged on the specific PC station. Next pick the corresponding ODBC driver such as Microsoft Access or SQL Server.

Note: When using a Microsoft SQL Server configuration database, the AlarmWorX32 Logger will not log any data when the **ODBC Data Source** is set as **Not available - using configuration DB** in the **Database Logging** tab, as shown in the figure above. This setting is designed to be used with a Microsoft Access configuration database.

Configuring Microsoft Access Databases

To configure a Microsoft Access database:

1. Select **Microsoft Access Driver (.mdb)** in the **Create New Data Source** wizard, as shown in the figure below.



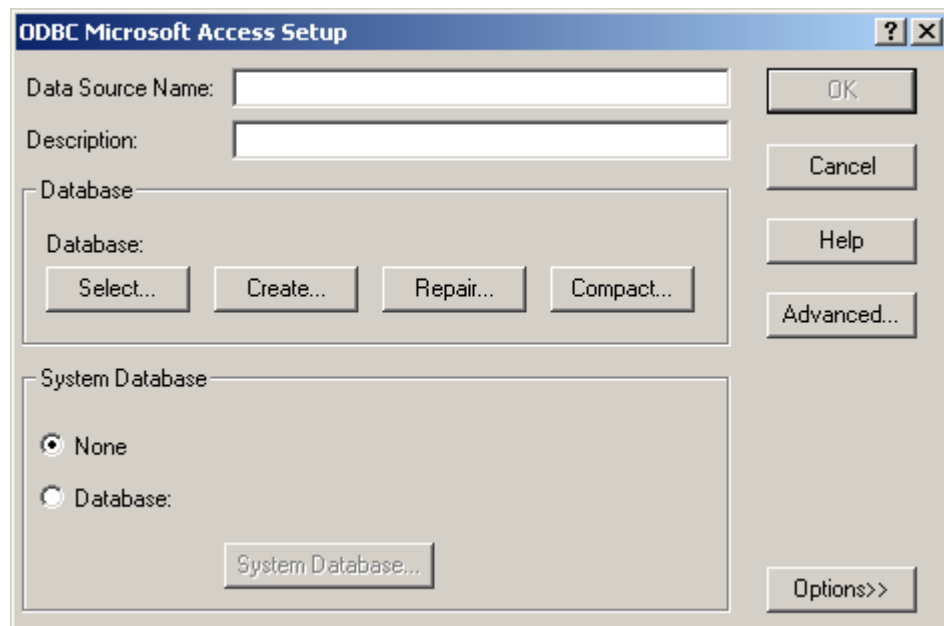
Selecting a Microsoft Access ODBC Driver

- When you have selected the Microsoft Access driver, click **Next** to proceed to the final dialog box in the **Create New Data Source** wizard, shown below.



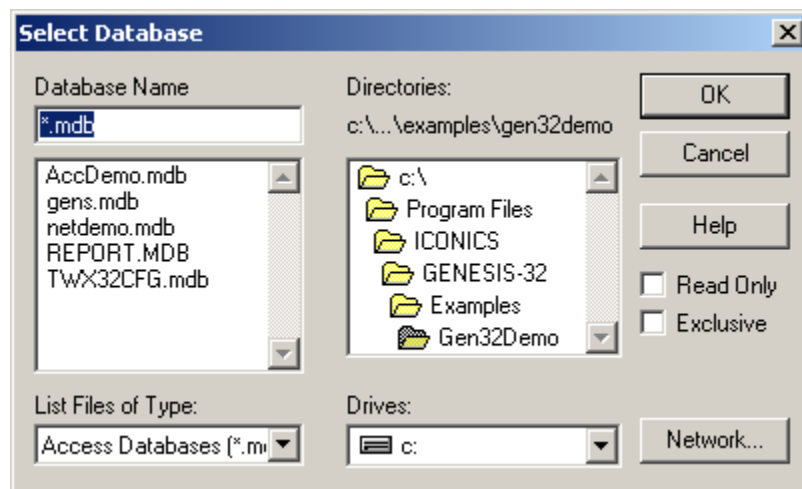
Creating a New Microsoft Access Database Connection

- Click **Finish** to configure your new Microsoft Access database. This opens the **ODBC Microsoft Access Setup** dialog box, shown below. Enter the **Data Source Name** and the **Description**.



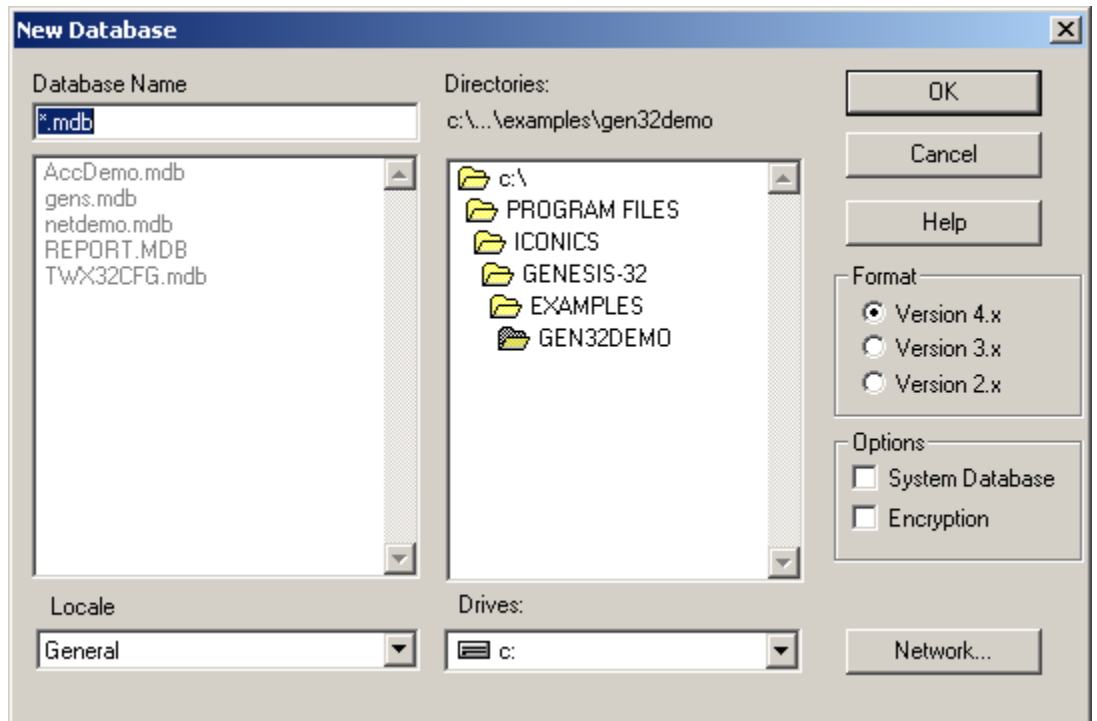
ODBC Microsoft Access Setup Dialog Box

- Click **Select** to choose an existing .mdb file. This opens the **Select Database** dialog box, shown below.



Selecting an Existing Microsoft Access Database File

- Click **Create** to create a new .mdb file. This opens the .mdb file directory in the **New Database** dialog box, as shown below.



Creating a New Microsoft Access Database File

6. You can choose a local hard disk or a networked hard disk as the location of your new .mdb file. Click **OK** to exit the configuration and establish the database connection.

Note: When creating a new .mdb file, the Microsoft Access configuration dialog box provides the options of creating a Version 4.x or a Version 3.x compatible .mdb file. By default, Version 4.x is selected, creating a Jet Engine 4.x compatible .mdb file, which can be opened *only* in Microsoft Access 2000. If you have Microsoft Access 97, select the Version 3.x compatibility. In general, Version 4.x is preferred because it offers greater functionality than the older versions.

Configuring Microsoft SQL Server and MSDE Databases

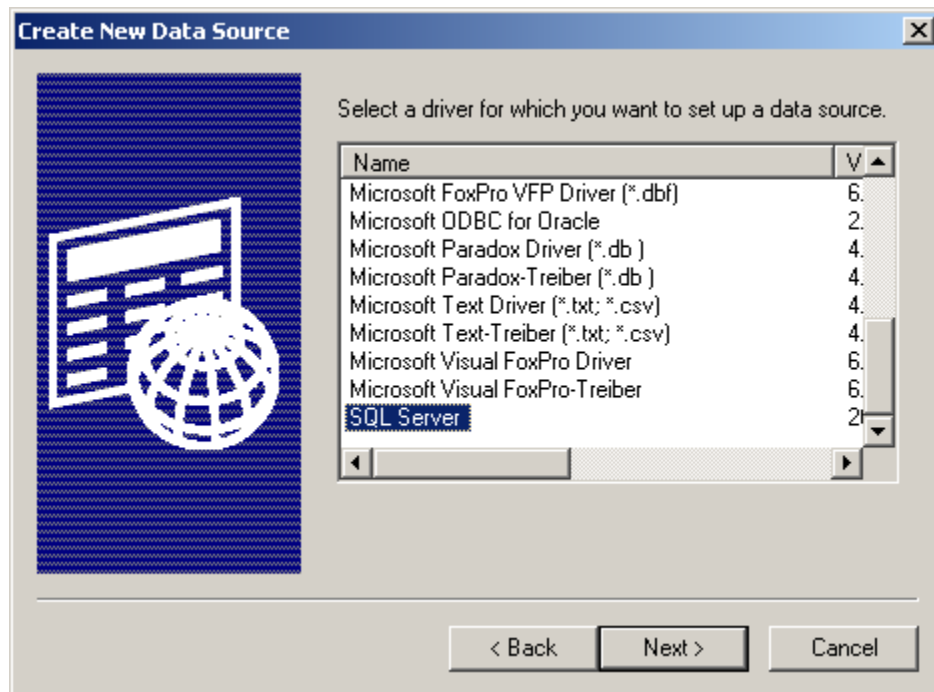
The Alarm Logger includes native support for MSDE (Microsoft Data Engine), which is a Microsoft SQL Server 7.0 or 2000 compatible database engine with the following limitations:

- The Maximum database size is 2 GB.
- Performance can decrease if more than five connections are open at one time.

However, MSDE is a Microsoft SQL Server compatible engine that can be directly upgraded to the full Microsoft SQL Server. Thus, database

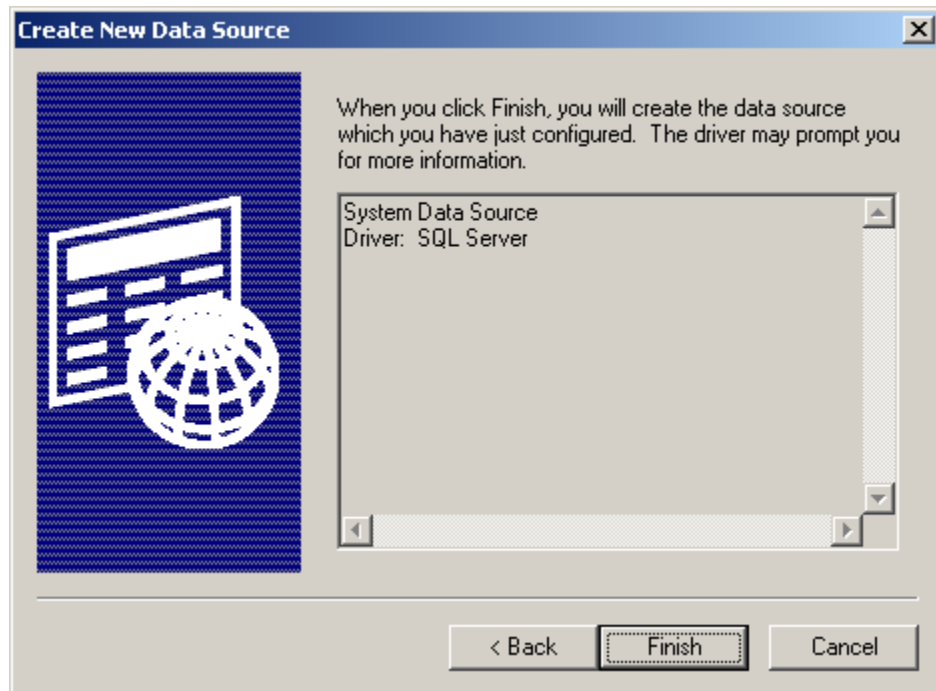
connections to MSDE are identical to the database connections to Microsoft SQL Server.

1. To establish an ODBC connection to this database, when configuring a new ODBC data source for Microsoft SQL Server or MSDE, select the appropriate ODBC driver. To configure a Microsoft SQL Server or MSDE database, select the **SQL Server** driver in the **Create New Data Source** wizard, as shown in the figure below.



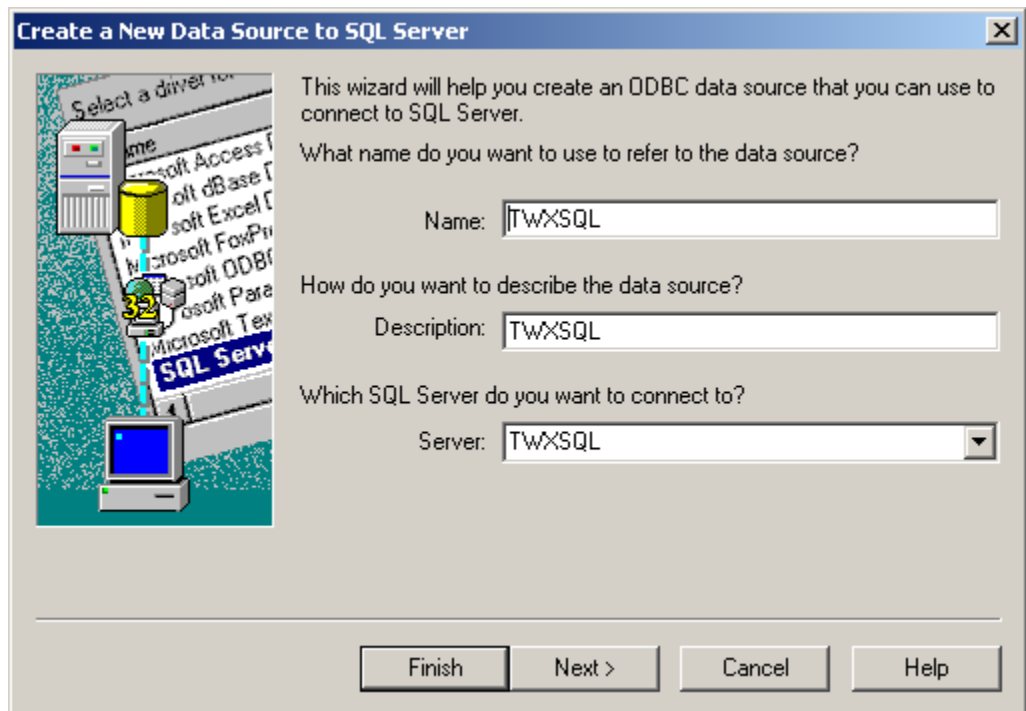
Selecting a Microsoft SQL or MSDE Server ODBC Driver

2. When you have selected the Microsoft SQL Server driver, click **Next** to proceed to the final dialog box in the **Create New Data Source** wizard, shown below.



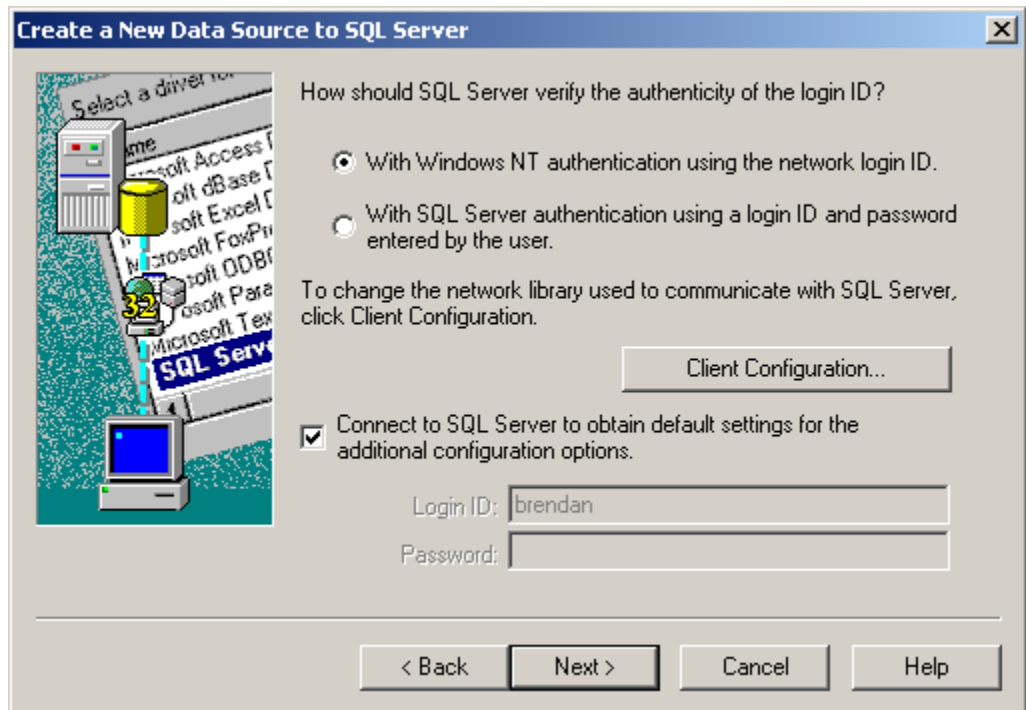
Creating a Microsoft SQL Server or MSDE Database Connection

3. Click **Finish** to configure your new database. This opens the **Create a New Data Source to SQL Server** dialog box, shown below. Enter the **Data Source Name**, the **Description**, and the name of the **Server** on which the Microsoft SQL Server or MSDE engine resides.



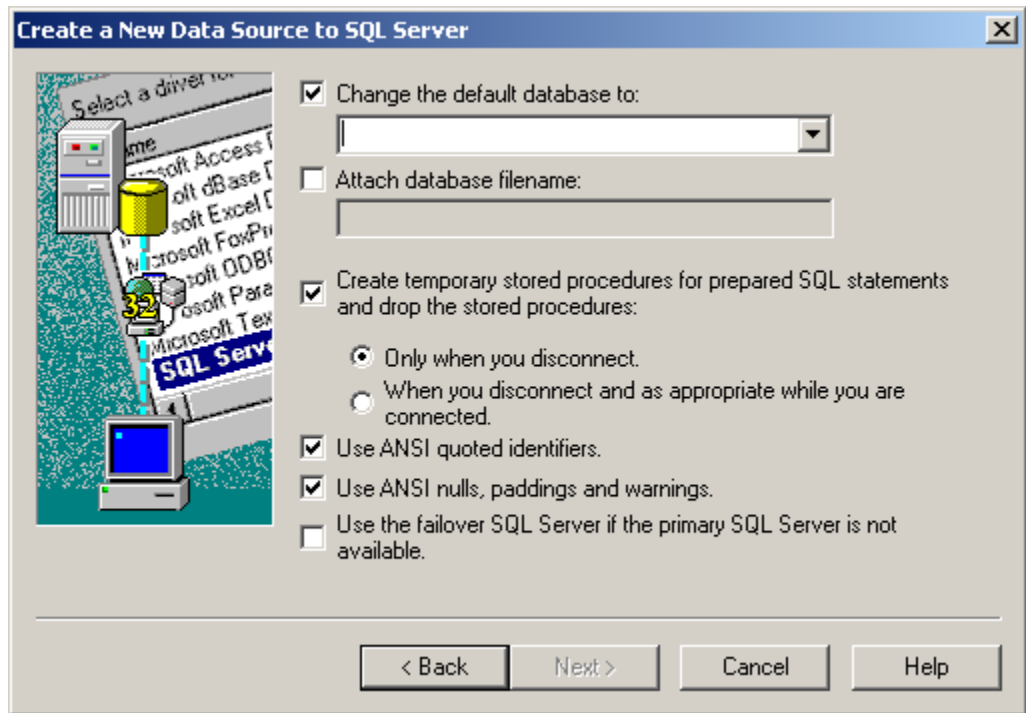
Create a New Data Source to SQL Server Dialog Box

4. If the engine resides on the same PC node as the AlarmWorX32 Logger, select "local." Otherwise select or specify the appropriate server name. Click **Next** to continue with the Microsoft SQL Server Security configuration and Connection type. It is recommended that you use the Windows NT authentication mechanism, which eliminates the need for maintaining individual user accounts and passwords.



Configuring a Microsoft SQL or MSDE Database Connection

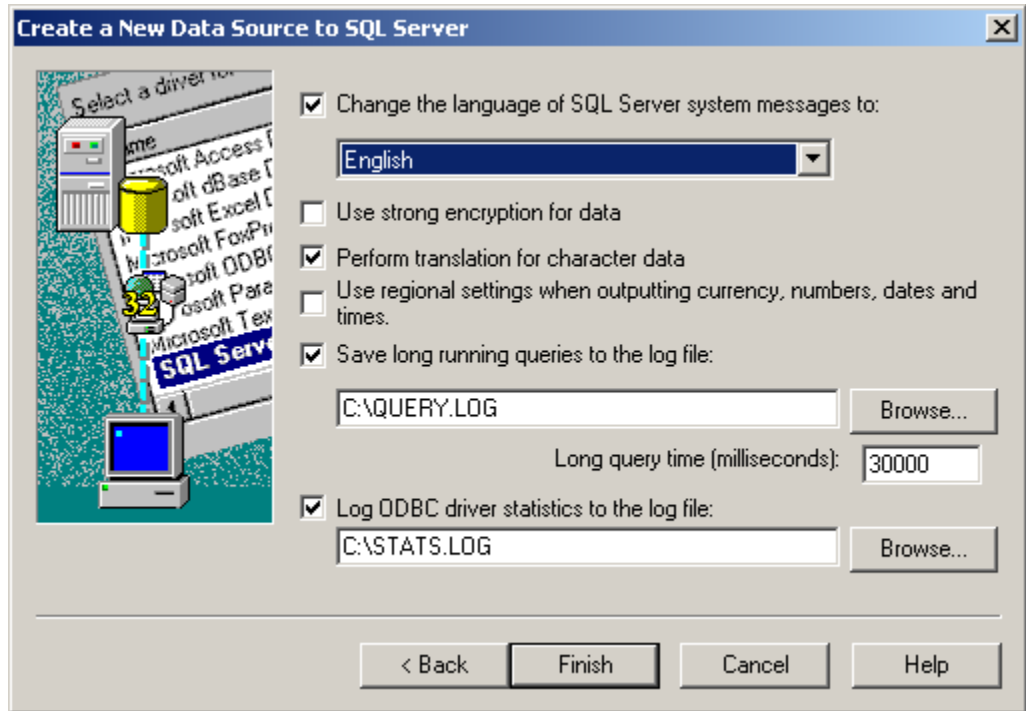
5. Click **Next** to continue with the selection of the default database for the connection. Make sure it is not the master database, but the one created for the purpose of data logging.



Setting the Default Microsoft SQL Server or MSDE Database

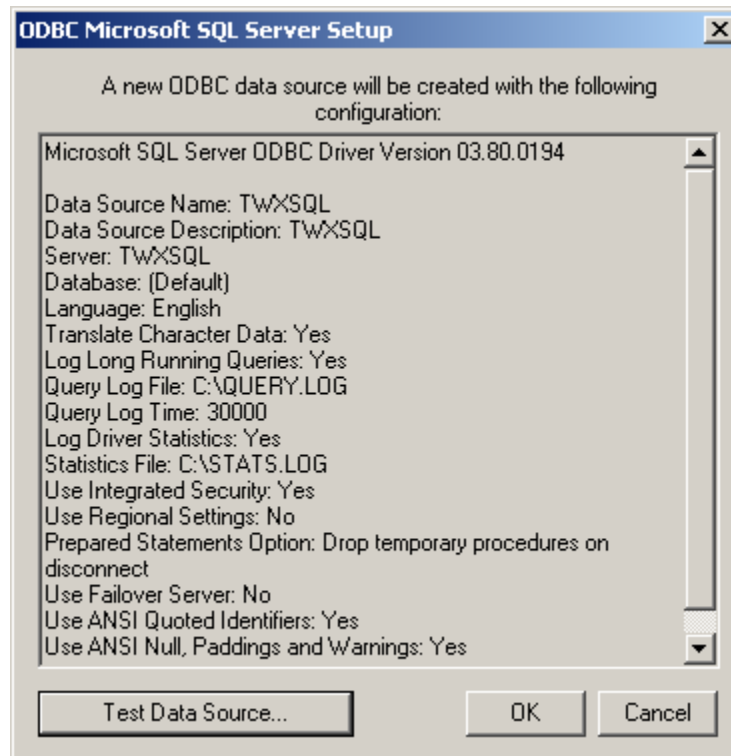
Note: When connecting to an Microsoft SQL Server or MSDE database, make sure that the desired database for historical data logging is selected in the **Change the Default Database** field, as shown in the figure above. Otherwise, you will establish a connection to the master database, which is not recommended.

6. Click **Next** to accept or modify the default settings, as shown in the figure below.



Configuring the Default Settings

7. Complete the connection configuration by clicking **Finish**. It is recommended that you test the established connection by clicking **Test Data Source**, as shown in the figure below.



Testing a Microsoft SQL Server or MSDE Database Connection

Note: Regardless of the database type, all AlarmWorX32 modules, which interact with databases, will create, modify, drop, and update tables. Therefore, it is critical that the currently logged-in user has permission to perform these tasks.

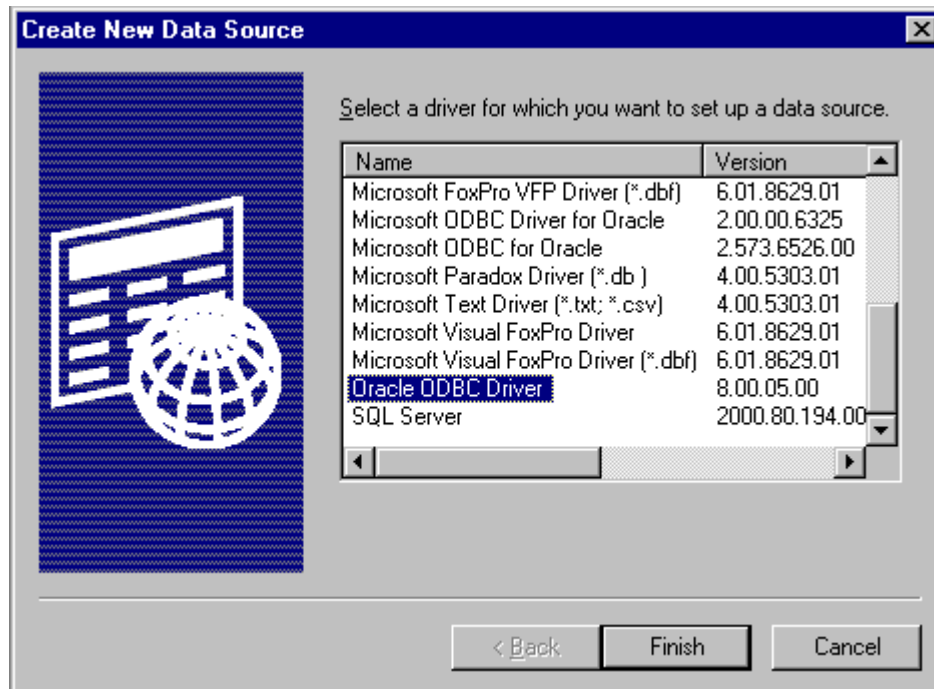
Configuring Oracle Databases

The Alarm Logger includes support for logging data as well as replaying data to and from Oracle databases. The support for Oracle was developed under the assumption that:

- ODBC database connections to Oracle are established using the Oracle ODBC Driver 8.05.60, which is provided by Oracle and is available on the ICONICS product CD under the Tools directory.
- Oracle is running on a Microsoft Windows NT station.

To configure an Oracle database:

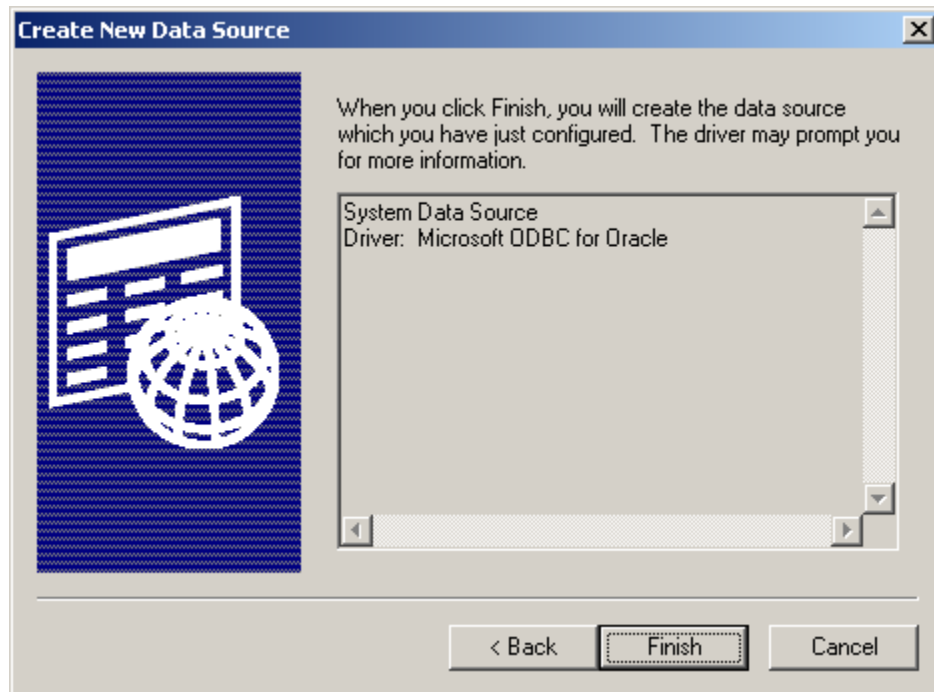
1. Select the **Oracle ODBC Driver** in the **Create New Data Source** wizard, as shown in the figure below.



Selecting the Oracle ODBC Driver

2. When you have selected the ODBC for Oracle driver, click **Next** to

proceed to the final dialog box in the **Create New Data Source** wizard, shown below.



Creating an Oracle Database Connection

3. Click **Finish** to configure your new database. This opens the **Oracle ODBC Driver Setup** dialog box, shown below. Enter the **Data Source Name**, the **Description**, the **Service Name**, and the **User ID**, and configure the other desired settings.

Oracle8 ODBC Driver Setup [X]

Data Source Name:

Description:

Data Source

Service Name:

UserID:

Database Options

Connect to database in Read only mode ☐

Prefetch Count:

WorkAround Options

Force Retrieval of Long Columns ☐

Application Options

Enable Thread Safety ☒ Enable LOBs ☒ Enable Result Sets ☒

Enable Failover ☒ Retry Count: Delay:

Enable Query Timeout ☒

Translation Options

Option:

Library:

Configuring an Oracle ODBC Database Connection

Table Management

Alarm data are logged to one master active table. If no restrictions are placed on the size of the active table, then logging will continue to go to the active table as long as the Alarm Logger is running. If, however, a restriction is placed on the size of the active table, then all logged data beyond the active table size will get logged to archive tables. The archive tables also have a maximum size determined by a number of records or time interval. When one archive table is full, archiving goes on to the next table. When all archive tables are full, the oldest archive table is deleted and a new one is set up. The name of each archive table is the name of the active table followed by the date and time of the first alarm in the table. In this system, the Active Table will always have the most recent alarm data.

The **Table Management** section of the **Database Logging** tab, shown in the figure below, configures the following parameters for the active table.

Database Logging: Table Management

Table Name: Enter the name of the active table to log data to.

Active Table: Here you can specify a limit on the active table size. The limit can be by the **Maximum Number of Records**, or by the **Maximum Time Interval** (select from hours, days, months, or years by clicking the ... button). If both items are checked, then the table is full when the first of those two conditions is met. The **Interval** refers to the time difference between the first item logged (not the logging start time) and the most recent item logged. If neither item is checked, then no archiving will take place.

Archive Tables: In this section, you can enter in the quantity of archive tables and the condition(s) upon which each archive table is full.

Note: If, in the **Archive Table(s)** section, the **Max Records** and **Max Interval** fields are both left unchecked, then all archiving will go to one archive table of

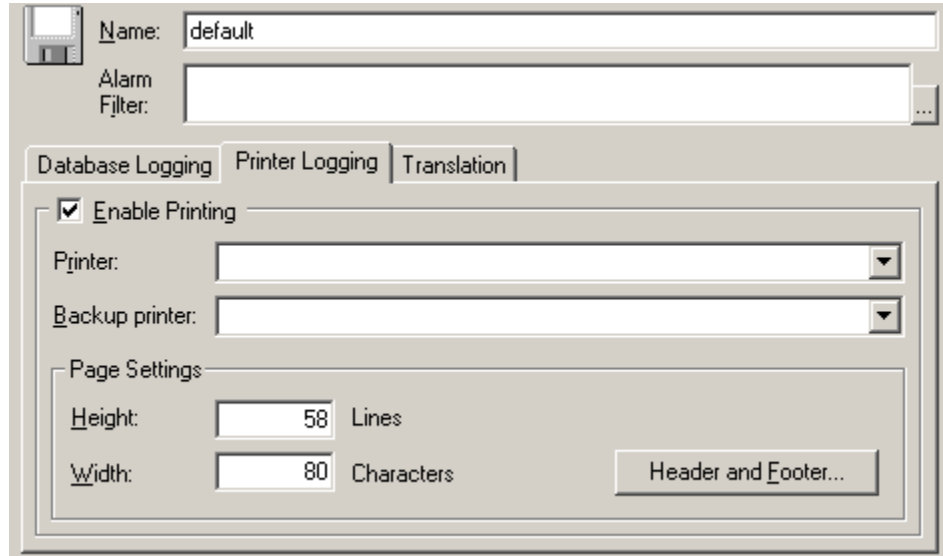
theoretically unlimited length, as no restrictions have been placed on the table size. In essence, the **Max Tables** field will be ignored.

Note: During runtime, the Alarm Logger will create the active table if it does not exist and will also add any columns to the table if they are missing. The logger will not create any indexes. Use a database administration tool to add indexes to columns that are searched and/or sorted by queries.

Note to Microsoft Access Users: If you are connected to a Microsoft Access data base and you are using archive tables, it is very important to note that, though the archiving deletes the oldest table to make room for a new one when all tables are used up, the disk space taken by the deleted table does not get freed up. The end result is that over a long period of time, the log database file will keep growing in size even though you may have specified a finite quantity of records to archive. One solution to this problem is to periodically open the database in Microsoft Access and select **Compact/Repair Access Database** from the **Tools** menu of the Configurator. This frees up unused disk space. **It is very important, however, that you make sure that no other users, including the AlarmWorX32 Alarm Logger, are connected to your database when you use the 'Compact' command.** An alternative solution is to use SQL or MSDE for your data logging instead of Microsoft Access.

Printer Logging Tab

Checking the **Enable Printing** check box on the **Printer Logging** tab of the configuration properties, shown in the figure below, causes the Alarm Logger to type event records to the printer(s) specified in the following fields.

The image shows a screenshot of a configuration window titled "Printer Logging Tab". At the top, there is a "Name:" field with the value "default" and an "Alarm Filter:" field. Below these are three tabs: "Database Logging", "Printer Logging" (which is selected), and "Translation". Under the "Printer Logging" tab, there is a checked checkbox labeled "Enable Printing". Below this checkbox are two drop-down menus: "Printer:" and "Backup printer:". Further down is a "Page Settings" section containing two input fields: "Height:" with the value "58" and the unit "Lines", and "Width:" with the value "80" and the unit "Characters". To the right of these fields is a button labeled "Header and Footer...".

Printer Logging Tab

Printer: Select the **Primary Printer** from the drop-down list. The AlarmWorX32 Logger sends each event directly to the printer as soon as it is received. This works best with a dedicated ink jet or dot matrix printer that will print one line at a time, rather than a page-oriented printer like a laser printer. Laser printers will still work, but the page will not be ejected until it is full.

All event notifications will print on the Primary Printer unless an error is detected (offline, out of paper, etc.), in which case the Backup Printer will be used. Switchover from Backup to Primary occurs automatically when the error condition is cleared on the Primary Printer.

Backup Printer: Select the **Backup Printer** from the drop-down list. The Backup Printer will be used whenever an error is detected (offline, out of paper, etc.) on the Primary Printer. Switchover from Backup to Primary occurs automatically when the error condition is cleared on the Primary Printer.

Height: Enter the number of lines your printer prints on a single page using its default font. The logger prints events "one line at a time," by sending raw text to the printer, bypassing the normal Windows page device context printing. Printers that accept this raw text will use an internal font to render the characters. The number of lines that will fit on a page using this internal font needs to be entered here. If you cannot find this information in the printer manual, you may need to experiment to find an acceptable value. A value of 58 will be safe for most printers.

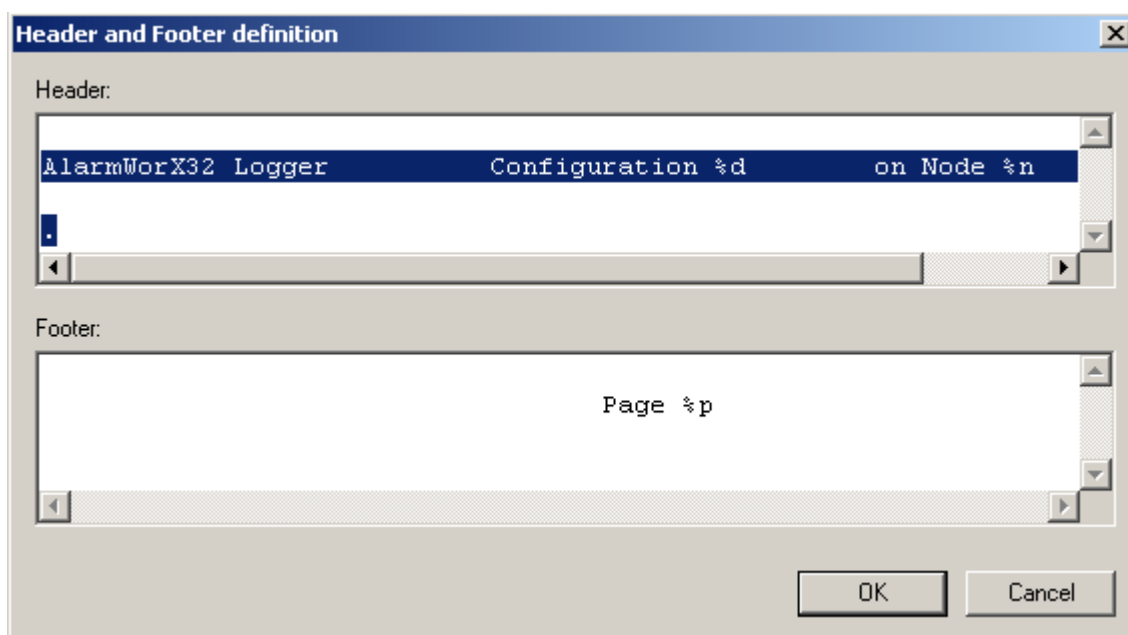
Width: Enter the number of characters the printer is capable of printing on a single line using its default font.

Header and Footer Button: Click the **Header and Footer** button to edit the lines of text that will appear at the top and bottom of each printed page. The header and footer may contain more than one line. The following tokens will be replaced at runtime with their appropriate values:

%d - Name of the configuration.

%n - Computer name the logger is running on.

%p - Page number.



Header and Footer Dialog

International Language Printing

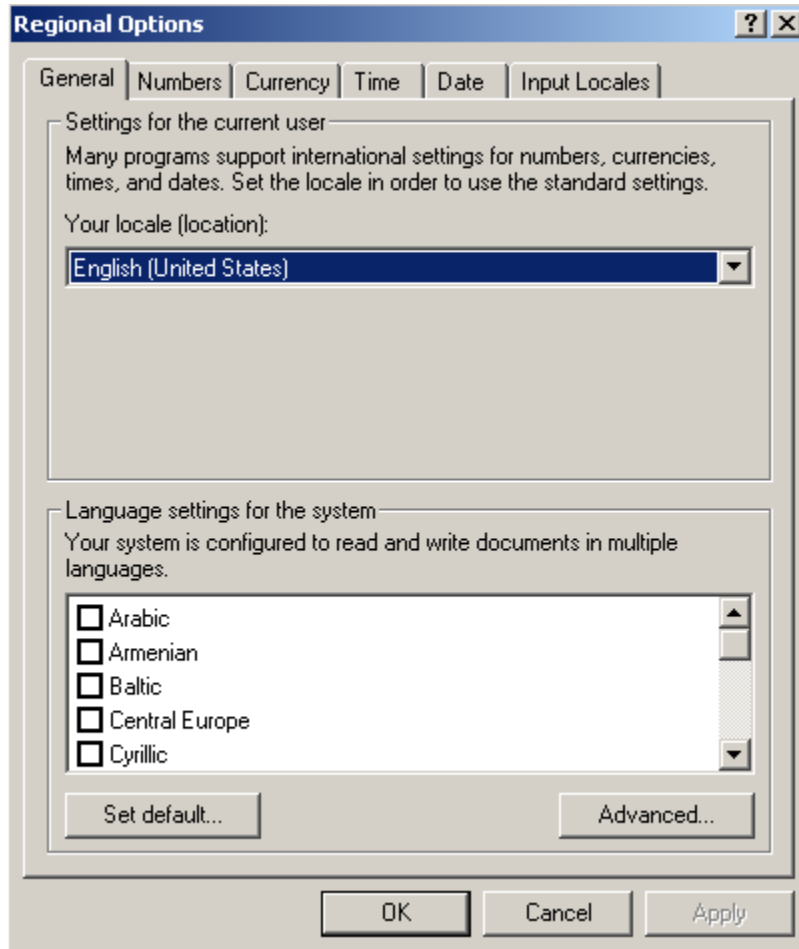
The AlarmWorX32 Logger supports printing using character sets other than the standard 7-bit ASCII codes appropriate for English. Since the logger does not use normal "Windows" printing, the setup to use other character sets is more involved than for other typical Windows programs.

Selecting the Windows Regional Options

First select the desired locale and Language from the Windows Control Panel "Regional Settings":

1. From the Windows **Start** menu, select **Settings > Control Panel**.
2. In the Control Panel, double-click on **Regional Options**. This opens the **Regional Options** dialog box, as shown in the figure below. On the **General** tab, select the desired locale and Language, as shown in the figure below.

Note: You must set the desired language in the Windows Regional Settings. It is not enough to choose the language only within GENESIS32.



Setting the Windows Regional Language Options

Selecting the Character Set (Code Page) in the Printer

By selecting the desired language in Windows, the Logger will convert the UNICODE characters to the 8-bit ASCII version using the DOS code page for the selected language. In order for the characters to print properly, the printer must be configured to use the correct code page.

The code page for the printer is selected by sending the printer special non-printable ASCII characters called control codes. The control codes are listed in the first column (Decimal 0 - 31) of the ASCII chart below.

Ctl	Dec	Hex	Char	Code	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
^@	0	00		NUL	32	20	sp	64	40	@	96	60	`
^A	1	01	␣	SOH	33	21	!	65	41	A	97	61	a
^B	2	02	␣	SIX	34	22	"	66	42	B	98	62	b
^C	3	03	␣	EIX	35	23	#	67	43	C	99	63	c
^D	4	04	␣	EOI	36	24	\$	68	44	D	100	64	d
^E	5	05	␣	ENQ	37	25	%	69	45	E	101	65	e
^F	6	06	␣	ACK	38	26	&	70	46	F	102	66	f
^G	7	07	␣	BEL	39	27	'	71	47	G	103	67	g
^H	8	08	␣	BS	40	28	(	72	48	H	104	68	h
^I	9	09	␣	HI	41	29	)	73	49	I	105	69	i
^J	10	0A	␣	LF	42	2A	*	74	4A	J	106	6A	j
^K	11	0B	␣	VI	43	2B	+	75	4B	K	107	6B	k
^L	12	0C	␣	FF	44	2C	,	76	4C	L	108	6C	l
^M	13	0D	␣	CR	45	2D	-	77	4D	M	109	6D	m
^N	14	0E	␣	SD	46	2E	.	78	4E	N	110	6E	n
^O	15	0F	␣	SI	47	2F	/	79	4F	O	111	6F	o
^P	16	10	␣	SLE	48	30	0	80	50	P	112	70	p
^Q	17	11	␣	CS1	49	31	1	81	51	Q	113	71	q
^R	18	12	␣	DC2	50	32	2	82	52	R	114	72	r
^S	19	13	␣	DC3	51	33	3	83	53	S	115	73	s
^T	20	14	␣	DC4	52	34	4	84	54	T	116	74	t
^U	21	15	␣	NAK	53	35	5	85	55	U	117	75	u
^V	22	16	␣	SYN	54	36	6	86	56	V	118	76	v
^W	23	17	␣	EIB	55	37	7	87	57	W	119	77	w
^X	24	18	␣	CAN	56	38	8	88	58	X	120	78	x
^Y	25	19	␣	EM	57	39	9	89	59	Y	121	79	y
^Z	26	1A	␣	SIB	58	3A	:	90	5A	Z	122	7A	z
^[	27	1B	␣	ESC	59	3B	;	91	5B	[	123	7B	{
^\	28	1C	␣	FS	60	3C	<	92	5C	\	124	7C	
^]	29	1D	␣	GS	61	3D	=	93	5D	]	125	7D	}
^^	30	1E	␣	RS	62	3E	>	94	5E	^	126	7E	~
^_	31	1F	␣	US	63	3F	?	95	5F	_	127	7F	Δ†

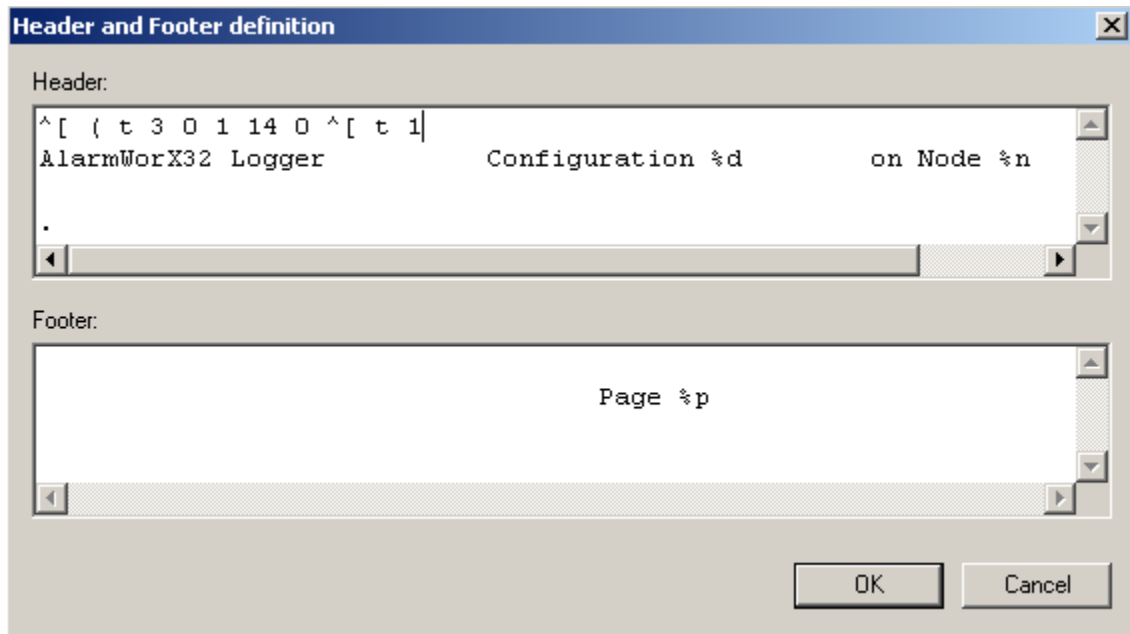
† ASCII code 127 has the code DEL. Under MS-DOS, this code has the same effect as ASCII 8 (BS). The DEL code can be generated by the CTRL+BACKSPACE key.

The configuration of the header and footer for the printed alarms understands the ASCII control codes so for example the character sequence ^[will be interpreted as the ESC code (Decimal 27) instead of the two literal characters ^ and [.

The proper control code sequence to select a particular language varies by printer manufacturer and model. Please refer to the printer manual for your printer to obtain the correct code sequence. For example, the following sequence will assign the PC866 Russian code page to character table 1 and then make character table 1 active for International (non-US) Epson printers:

^[(t 3 0 1 14 0 ^[t 1

The same code sequence is shown below in the header configuration of the AlarmWorX32 Logger. To view the header configuration, click the **Header and Footer** button on the **Printer Logging** tab of the AlarmWorX32 Logger Configurator.



Code Sequence Displayed in Printer Header Configuration

Translation Tab

The **Translation** tab of the configuration properties, shown in the figure below, will appear only if you have the version of AlarmWorX32 with Language Switching installed (Unicode Version only). It is for use with the language translations as configured in the Language Configurator.

The screenshot shows the 'Translation' tab of a configuration window. At the top, there is a 'Name' field with the value 'default' and an 'Alarm Filter' field. Below these are three tabs: 'Database Logging', 'Printer Logging', and 'Translation', with 'Translation' being the active tab. The 'Translation Mode' section on the left has three radio buttons: 'Use Registry', 'No Translation', and 'Custom Translation', with 'Custom Translation' selected. To the right, the 'Custom Translation Settings' section contains a 'Language' dropdown set to 'English', a 'Language Subset' dropdown set to 'US', and a checkbox for 'Synchronize application LCID with selected language' which is unchecked. Below the checkbox is an 'Application LCID' field with the value '1033'. A note states: 'When the Language Subset is empty then default subset will be used.'

Translation Tab

Translation Mode allows you to choose how you would like Language Switching to work with the Alarm Logger for this particular configuration.

Use Registry: This mode uses the registry settings to determine what language to use. This means that if a user changes the language in another application, the Alarm Logger will also use the new language.

No Translation: This mode does not allow for the language to be translated before getting logged. If you have items configured with special characters for use with Language Switching, they will remain with the items and be logged in this way. This is useful if you would like the next phase, such as reporting, to be able to determine which language translation to use.

Custom Translation: This mode allows you to set the language translation here so that it will always be the language used with the Alarm Logger for this particular configuration. The following Custom Translation Settings are available for use with this mode:

- **Language:** Choose the language you wish to use from the list of available languages as configured in the Language Configurator.
- **Language Subset:** If you are using a language subset, enter it here. It must be a language subset that has been defined in the Language Configurator.
- **Synchronize application LCID with selected language:** If using application language converting also, you can synchronize with language switching by checking this check box.
- **Application LCID:** Allows you to choose a language to use for application language converting. This option is only available for use

with languages that have alternate language resources (dynamic link libraries) associated with them. See the document 'International ToolworX.doc' on your ICONICS CD for more information. It can be found in the CD directory in the Tools folder under Translations.

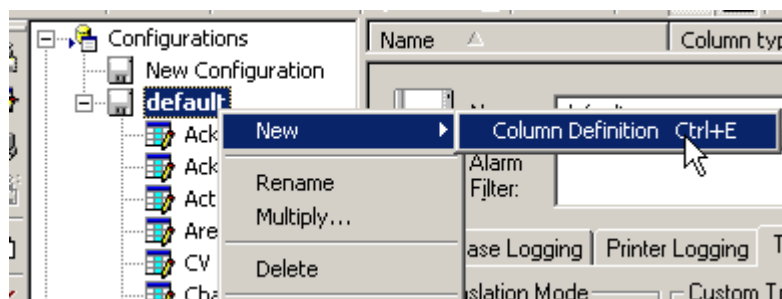
Columns

Each configuration contains one or more columns. The columns defined for a configuration appear as child elements of the configuration in the tree control, and as list items in the list view (top right pane).

Creating a New Column

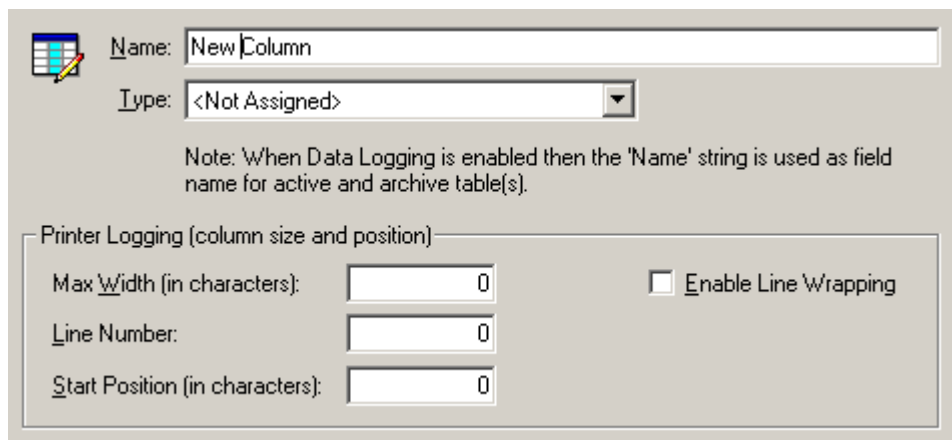
To create a new column:

1. Right-click on a logger configuration in the tree control of the Configurator and select **New > Column Definition** from the pop-up menu, as shown in the figure below.



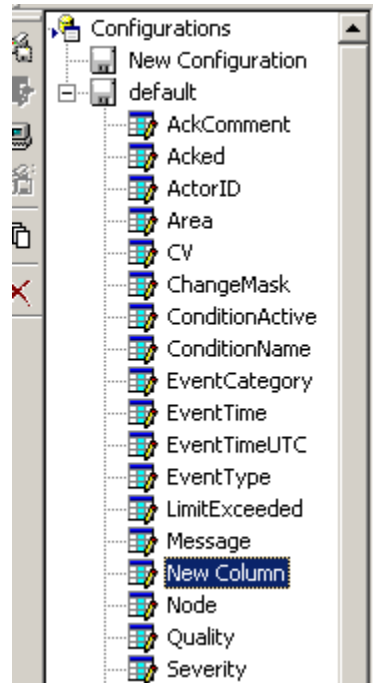
Creating a New Column

2. The properties dialog box for the new tag appears in the right-hand pane of the Configurator, as shown in the figure below.



Setting the Properties for the New Column

3. In the **Name** field, type a name for the new column. This will be the column name used in the logging database and the column heading when printing.
4. In the **Type** field, select a column type from the drop-down list.
5. When you have finished configuring the column properties, click the **Apply** button. The new tag appears under the Logger Configuration tree control, as shown in the figure below.



New Column Added to Tree Control

Column Configuration Properties

Configure the following parameters for columns, as shown in the figure below.

Note: If **Dialog View** is not selected from the **View** menu, you will not see the configuration properties.

Column Configuration Properties

The following fields are specified for each column:

Name: Enter the name of the column. This will be the column name used in the logging database and the column heading when printing.

Maximum Width: Enter the number of character spaces for this column when printing.

Line Number: When printing, each event may occupy more than one printed line. Specify the line on which you want the column to print. A value of zero will prevent the column from printing.

Start Position: Enter the number of character spaces from the left to begin printing. The first character space is zero.

Enable Line Wrapping: When printing, field data that are wider than the configured column width will either be truncated or wrapped to the next line depending on this setting.

Column Type: Select the OPC Alarm and Events data item you wish to log in this column from the drop-down list.

The available column types are listed in the table below.

Note: The following column type descriptions are taken directly from the OPC Alarms and Events Specifications. Please refer to the Specifications for complete information.

Column Type	Description
	The following items are present for all event types.
Source	The source of event notification. This Source can be used in the IOPCEventServer::TranslateToItemIDs method to determine any related OPC Data Access itemIDs.
Time	Time of the event occurrence. For conditions, time that the condition transitioned into the new state or sub-condition. For example, if the event notification is for acknowledgment of a condition, this would be the time that the condition became acknowledged.
Active Time	Array of active times corresponding to each Source and ConditionName pair. This parameter uniquely identifies a specific transition of the condition to the active state or into a different sub-condition and is the same as the SubCondLastActive condition attribute. Active times are passed to the client in the ActiveTime member of the ONEVENTSTRUCT by the IOPCEventSink::OnEvent callback. If the condition has become active again or transitioned into a different sub-condition at a later time, this acknowledgment will be ignored.
Message	Event notification message describing the event.
Subscription	Subscription to a given OPC Alarm and Event server.
Server Description	Information about the currently subscribed OPC Alarm and Event server.
Server Node	Node (computer) on which the currently subscribed OPC Alarm and Event server is located.
Event Type	OPC_SIMPLE_EVENT, OPC_CONDITION_EVENT, or

	OPC_TRACKING_EVENT for Simple, Condition-Related, or Tracking events, respectively.
Event Category	Event categories define groupings of events supported by an OPC Event server. Examples of event categories might include "Process Events", "System Events", or "Batch Events". Event categories may be defined for all event types, i.e. Simple, Tracking, and Condition-Related. However, a particular event category can include events of only one type. A given Source may generate events for multiple event categories. Names of event categories must be unique within the event server. The definition of event categories is server-specific.
Severity	Event severity (0-1000). The severity value is an indication of the urgency of the sub-condition. This is also commonly called 'priority', especially in relation to process alarms. Values will range from 1 to 1000, with 1 being the lowest severity and 1000 being the highest. Typically, a severity of 1 would indicate an event that is informational in nature, while a value of 1000 would indicate an event of catastrophic nature which could potentially result in severe financial loss or loss of life.
Number Event Attributes	The length of the specific event attribute array.
Attributes 1-20	User-specified event attributes.
	The following items are present only for condition-related events.
Condition Name	The name of the condition related to this event notification.
SubCondition Name	The name of the current sub-condition, for multi-state conditions. For a single-state condition, this contains the condition name.

Change Mask	Indicates to the client which properties of the condition have changed to have caused the server to send the event notification. It may have one or more of the following values: OPC_CHANGE_ACTIVE_STATE OPC_CHANGE_ACK_STATE OPC_CHANGE_ENABLE_STATE OPC_CHANGE_QUALITY OPC_CHANGE_SEVERITY OPC_CHANGE_SUBCONDITION OPC_CHANGE_MESSAGE OPC_CHANGE_ATTRIBUTE If the event notification is the result of a Refresh, these bits are to be ignored. For a “new event”, OPC_CHANGE_ACTIVE_STATE is the only bit that will always be set. Other values are server-specific. (A “new event” is any event resulting from the related condition leaving the Inactive and Acknowledged state.)
NewState	A WORD bit mask of three bits specifying the new state of the condition: OPC_CONDITION_ACTIVE, OPC_CONDITION_ENABLED, OPC_CONDITION_ACKED.
Quality	Quality associated with the condition state. Values are as defined for the OPC Quality Flags in the OPC Data Access Server specification.
AckRequired	This flag indicates that the related condition requires acknowledgment of this event. The determination of those events that require acknowledgment is server-specific. For example, transition into a LimitAlarm condition would likely require an acknowledgment, while the event notification of the resulting acknowledgment would likely not require an acknowledgment.
Active Time	Time that the condition became active (for single-state conditions), or the time of the transition into the current sub-condition (for multi-state conditions). This time is used by the client when acknowledging the condition (see IOPCEventServer::AckCondition method).
Cookie	Server-defined cookie associated with the event notification. This value is used by the client when acknowledging the condition (see IOPCEventServer::AckCondition method). This value is opaque to the client.
	The following is used only for tracking events and for condition-related events that are acknowledgment notifications.

ActorID	For tracking events, this is the actor ID for the event notification. For condition-related events, this is the AcknowledgerID when OPC_CONDITION_ACKED is set in NewState. If the AcknowledgerID is a NULL string, the event was automatically acknowledged by the server. For other events, the value is a pointer to a NULL string.
---------	---

Event Type Values

Event Type	Value	Description
OPC_SIMPLE_EVENT	1	Simple event.
OPC_TRACKING_EVENT	2	Tracking event.
OPC_CONDITION_EVENT	4	Condition-Related event.

Change Mask Values

Change Mask Item	Value	Description
OPC_CHANGE_ACTIVE_STATE	1	The condition's active state has changed.
OPC_CHANGE_ACK_STATE	2	The condition's acknowledgment state has changed.
OPC_CHANGE_ENABLE_STATE	4	The condition's enabled state has changed.
OPC_CHANGE_QUALITY	8	The ConditionQuality has changed.
OPC_CHANGE_SEVERITY	16	The severity level has changed.
OPC_CHANGE_SUBCONDITION	32	The condition has transitioned into a new sub-condition.
OPC_CHANGE_MESSAGE	64	The event message has changed (compared to prior event notifications related to this condition).
OPC_CHANGE_ATTRIBUTE	128	One or more event attributes have changed (compared to prior event notifications related to this condition).

New State Values

New State	Value	Description
OPC_CONDITION_ENABLED	1	The condition has been enabled.
OPC_CONDITION_ACTIVE	2	The condition has become active.
OPC_CONDITION_ACKED	4	The condition has been acknowledged.

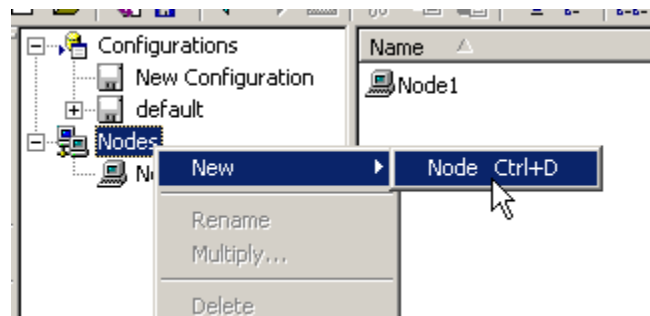
Nodes

Each **Node** represents a computer on the network that is configured to run one or more of the named logger configurations in the active configuration database. When the logger executable is run, it looks for its network computer name in the list of nodes to determine which logger configuration(s) to run. If a logger cannot find its computer name in the list of nodes, it will run the default configuration.

Adding a New Node

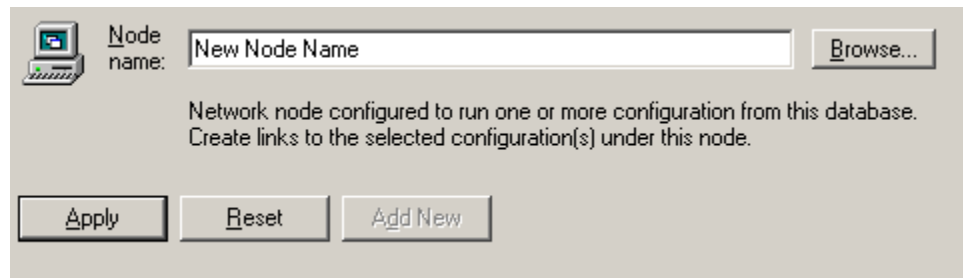
To add a new node to the configuration:

1. Right-click on the **Nodes** tree control of the Configurator and select **New > Node** from the pop-up menu, as shown in the figure below.



Creating a New Node

2. The properties dialog box for the new node appears in the right-hand pane of the Configurator, as shown in the figure below.



Setting the Properties for the New Node

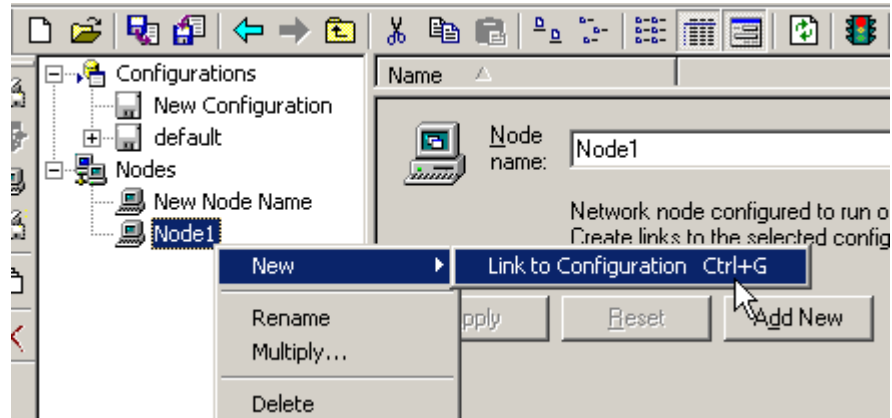
3. In the **Name** field, type a name for the new node, or click the **Browse** button to select a node from the network.
4. When you have finished configuring the nodes, click the **Apply** button. The new area appears under the **Nodes** tree control.

Assigning Configurations to Nodes

Each node has one or more configurations assigned to it. The configurations assigned to a node appear as child elements of the node in the tree control, and as list items in the list view (top right pane). A configuration may be assigned to more than one node.

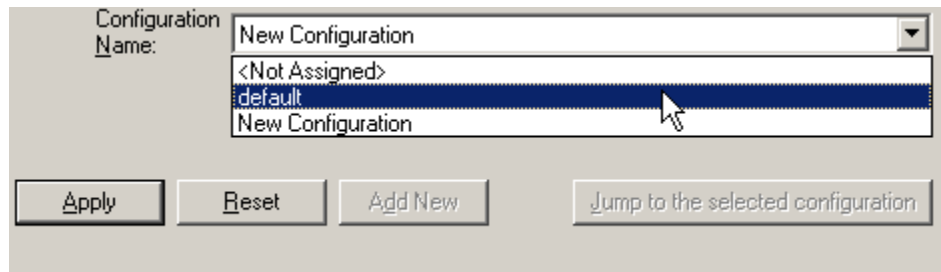
To assign a configuration to a node,

1. Right-click on the **Nodes** tree control of the Configurator and select **New > Link to Configuration** from the pop-up menu, as shown in the figure below.



Linking a Configuration to a Node

2. From the drop-down list under **Configuration Name**, select an existing configuration to assign to this node, as shown in the figure below.



Assigning a Configuration to a Node

3. Click the **Apply** button.

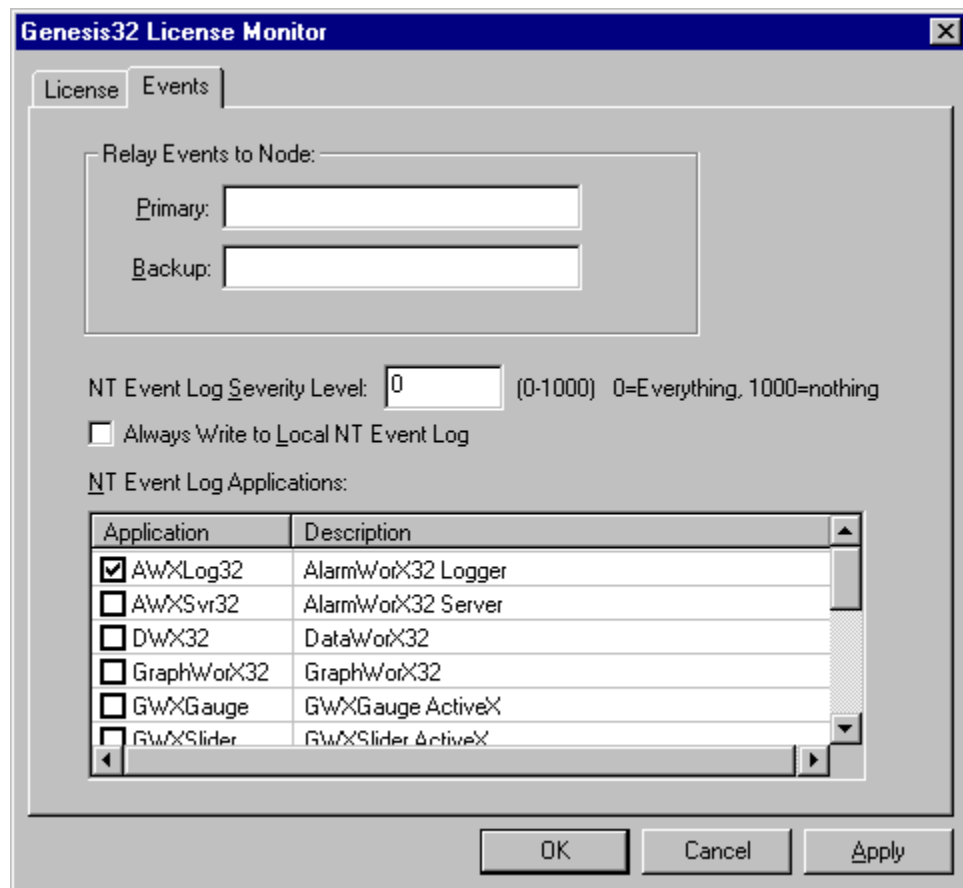
Running the Logger

The Alarm Logger application (AWXLog32.exe) that provides the runtime storage and printing has no user interface and may optionally be run as a service on Windows NT (see the installation section). The logger reads its configuration information from a Microsoft Access database file. The logger reads the configuration for the active database only. To activate a database, select **Make Active** from the **File** menu in the Logger Configurator.

Possible Runtime Errors

Any error conditions encountered by the logger will be reported via the ICONICS Event Server and optionally to the Windows NT Event Log.

To allow the logger to send messages to the NT Event Log, open the License Monitor by selecting Start > Programs > ICONICS **GENESIS32 > License Monitor**. In the License Monitor, select the Events tab, as shown in the figure below. Check **AWXLog32** in the list of applications.



NT Event Log Configuration

The possible error messages are listed in the table below.

SEVERITY	SOURCE	MESSAGE
500	AWXLog32	ICONICS AlarmWorX Logger Started.
500	AWXLog32	ICONICS AlarmWorX Logger Stopped.
500	AWXLog32	Primary Printer "%1" now in service. Printing on Backup Printer "%2" terminated
500	AWXLog32	Printing has resumed
500	AWXLog32	Configuration "%1" INSERT INTO table "%2"

SEVERITY	SOURCE	MESSAGE
		succeeded.
750	AWXLog32	Failed to Open Primary Printer "%1"
750	AWXLog32	Configuration "%1" printer event queue overflow. Messages have been lost. Queue size is %2 events.
750	AWXLog32	Primary Printer "%1" has failed. Now Printing on Backup Printer "%2"
750	AWXLog32	Configuration "%1" filter is inactive due to syntax error(s).
750	AWXLog32	Configuration "%1" logging event queue overflow. Messages have been lost. Queue size is %2 events.
800	AWXLog32	Error opening Configuration Database: Provider: %1 Connection: %2 (Error string from database is in "Comment" Attribute)
800	AWXLog32	Error Querying Configuration Database: Provider: %1 Connection: %2 Query: %3 (COM error string is in "Comment" Attribute)
800	AWXLog32	Configuration Database has no configuration for "Node" and no "Default". Provider: %1 Connection: %2
800	AWXLog32	Error opening Logging Database for Configuration "%1" Provider: %2 Connection: %3 (COM error string is in "Comment" Attribute)
800	AWXLog32	Query of Column definitions failed for configuration "%1" Query: %2 (COM error string is in "Comment" Attribute)
800	AWXLog32	Error opening Configuration Database: Provider: %1 Connection: %2 (COM error string is in "Comment" Attribute)
800	AWXLog32	Configuration Database has no configuration "%1". Provider: %2 Connection: %3 (COM error string is in "Comment" Attribute)
800	AWXLog32	Configuration "%1" subscription to OPC Event Server(s) failed.
800	AWXLog32	Configuration "%1" error assigning value to column "%2"

SEVERITY	SOURCE	MESSAGE
800	AWXLog32	Configuration "%1" INSERT INTO table "%2" failed (Error string from database is in "Comment" Attribute)
800	AWXLog32	Failed to register "%1" with GenRegistrar. Configuration "%2" will not execute.
800	AWXLog32	Printing to both Primary Printer "%1" and Backup Printer "%2" has failed

Using a technology that has been incorporated into all ICONICS products, TraceWorX32 provides online diagnostics and tuning of applications running in the GENESIS32 system. TraceWorX32 is designed expressly for systems integrators, OEMs and customers who want to have tools for doing their own troubleshooting and diagnostics.

TraceWorX32 tracks the runtime activity for each GENESIS32 application and logs the runtime data to a log file based on user-configured trace levels. The log file provides a thorough, color-coded report detailing all activity for the application, including the time, the date, the severity level, and a description of the event or problem.

TraceWorX32 also features several options for reporting issues to technical support. If you are experiencing problems with any applications, the log file deployment options, such as compressing and e-mailing log files, are ideal for tracking and archiving data and sending detailed reports to technical support. Developers can use these reports to identify the source of the problems.

For information about TraceWorX32, please see the TraceWorX32 Help documentation.



Your comments and suggestions on the operation of this software are welcome. Please address them to:

ICONICS

100 Foxborough Blvd.

Foxborough, MA 02035

Tel: 508-543-8600

Fax: 508-543-1503

E-mail: support@iconics.com

Web: www.iconics.com

© ICONICS, Inc. All rights reserved.

Contents

Introduction

Automation Interfaces

Available Control Properties

Introduction

This reference describes the OLE Automation features available in the AlarmWorX32 Logger.

Automation Interfaces

The AlarmWorX32 logger provides a COM interface that allows external programs to manipulate the logger as it is running. The interface is available to all programming languages that support COM including Visual Basic (VB), Visual Basic for Applications (VBA) and Microsoft Visual C++.

To access the Automation interface from VB and VBA, AlarmWorX32 logger must be made available by choosing **Project - References** from the main menu in the VB or VBA development environment and selecting **AWXLog32 1.0 Type Library** in the list of available references.

Available Control Properties

boolean LoggingEnabled

Description

True if logging to database is enabled. Applies to all running configurations that match the current **Filter Property**. Property is read/write.

Remarks

Values written to this property are transient. The value written is not stored in the configuration database and hence is "lost" when the logger is shut down.

Reading this property will throw an error in the following cases:

The logger has just started and the configuration has not been read yet.

Multiple running configurations match the filter and they have differing values (some True, some False).

Example

This example shows how to determine if logging is enabled.

```
Dim bEnabled as Boolean  
bEnabled = AWXLogger.LoggingEnabled
```

This example shows how to turn on/off logging

```
AWXLogger.LoggingEnabled = True
```

boolean PrintingEnabled

Description

True if logging to printer is enabled. Applies to all running configurations that match the current **Filter Property**. Property is read/write.

Remarks

Values written to this property are transient. The value written is not stored in the configuration database and hence is "lost" when the logger is shut down.

Reading this property will throw an error in the following cases:

The logger has just started and the configuration has not been read yet.

Multiple running configurations match the filter and they have differing values (some True, some False).

Example

This example shows how to determine if printing is enabled.

```
Dim bEnabled as Boolean  
bEnabled = AWXLogger.PrintingEnabled
```

This example shows how to turn on/off printing.

AWXLogger.PrintingEnabled = True

long Count

Description

Returns the number of running configurations whose names match the current Filter Property. Property is read-only.

Remarks

Reading this property will throw an error if the logger has just started and the configuration has not been read yet.

Example

```
Dim n as Long  
n = AWXLogger.Count
```

string Filter

Description

Wildcard string that a running configuration name must match in order to be affected by other methods and properties of this object. String property is read/write.

Remarks

If the name of a running configuration does **not** match the filter, then that configuration is ignored by all other methods and properties of this object. The default initial value for this Property is "" meaning all running configurations will match, hence by default, all of the methods and properties will operate simultaneously on all running configurations.

Example

```
Dim sFilter as String  
sFilter = AWXLogger.Filter  
  
'set filter for all configurations that begin with b  
AWXLogger.Filter = "b*"
```

string Item

Description

The name of a running configuration (that matches the **Filter Property**) for a given index value in the array (VB Collection). String property is read-only.

Syntax

Item(long n)

Where **n** is an integer from 1 to **Count**

Remarks

The AlarmWorX32 logger Automation object is designed as a VB Collection. Each element in the collection is the name of an active configuration that matches the current **Filter Property**.

Example

These three examples do the same thing -- print the name of each configuration.

```
Dim AWXLogger As AWXLog32Auto  
Set AWXLogger = New AWXLog32Auto
```

```
' method 1  
For Each ConfigName In AWXLogger  
    Print ConfigName  
Next ConfigName
```

```
' method 2  
For i = 1 To AWXLogger.Count  
    Print AWXLogger(i)  
Next i
```

```
' method 3  
For i = 1 To AWXLogger.Count  
    Print AWXLogger.Item(i)  
Next i
```