invensys

I/A Series® SMART™

(System Monitoring and Reporting Tool)

Version 2.0

User's Guide

smartware Technologies

Smartware Technologies, Inc.
4 Peuquet Parkway
Tonawanda, NY 14150

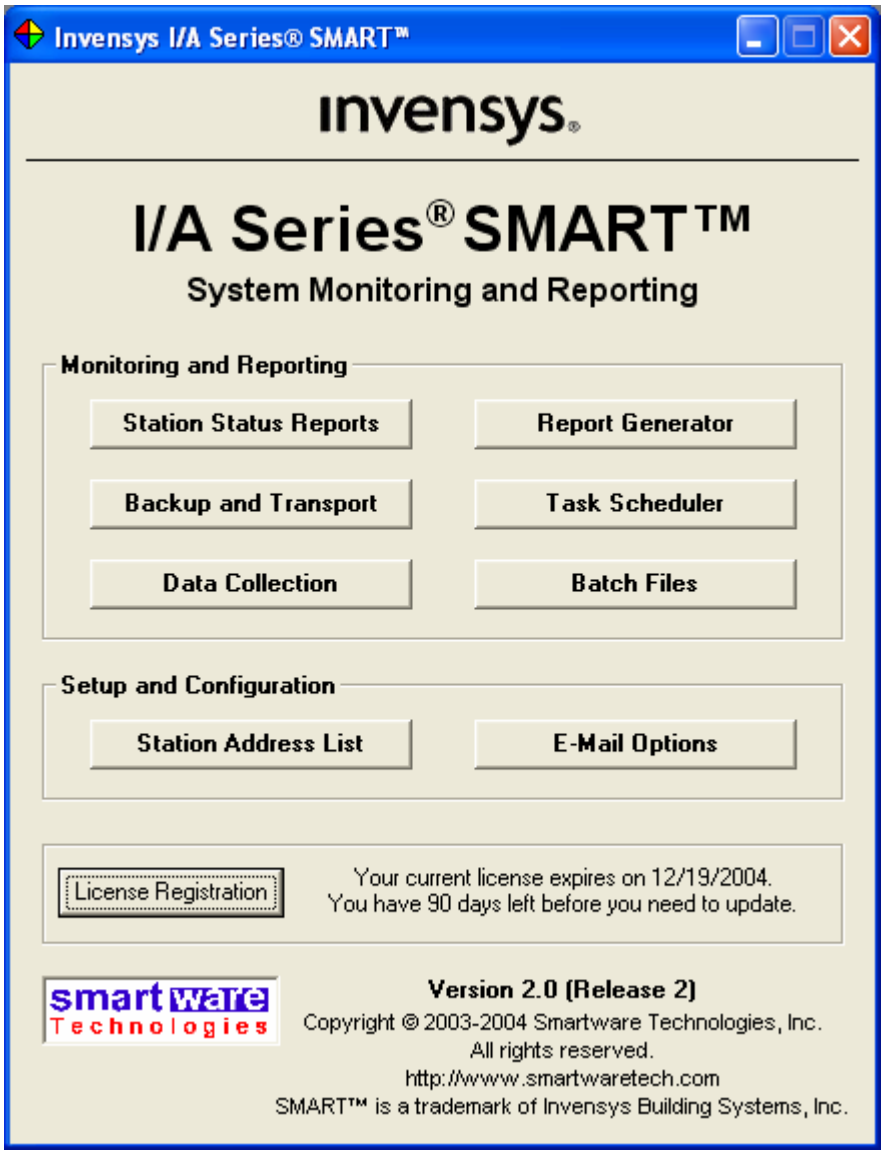
Sales and Support
(716) 213-2222

http://www.smartwaretech.com

Table of Contents

1. Getting Started

Invensys I/A Series SMART is a stand-alone application that is used to monitor and maintain running stations by generating reports, creating backups and collecting data.



The utilities are:

Station Status Reports Used to poll any number of Enterprise Server or UNC stations to confirm and report on their health and status.

Backup and Transport Used to automate the backing up of stations or files and sending them to a file server, FTP site or e-mail address.

Data Collection	Used to query and copy log data from an Enterprise Server Cloudscape archive database or a station backup file to any local database.
Report Generator	Used to create custom Excel-based reports of collected data.
Task Scheduler	Used to help schedule automated execution of the other utilities.
Batch Files	Used to create sets of tasks to be run in sequential order for automated execution.

System Requirements

Invensys I/A Series SMART is designed to run on almost any Windows platform currently in use today. There are a few important pre-requisites that you should be aware of to avoid running into problems when running the software.

Microsoft Windows

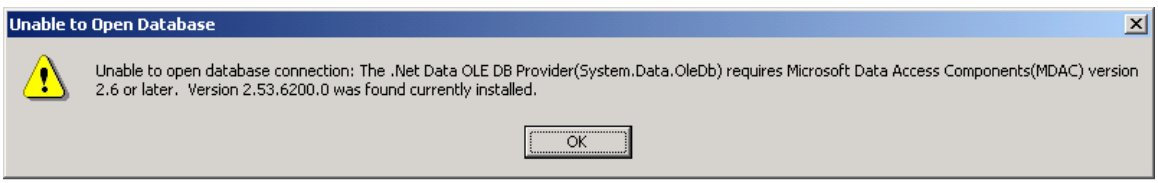
Microsoft Windows XP is recommended. Microsoft Windows 2000 is supported. Microsoft Windows 98 is not officially supported, though it has been tested and there are no specific incompatibilities known.

Microsoft .NET Framework (Version 2.0)

You should have installed on your machine all available updates and service packs from Microsoft. Specifically, you must have Version 2.0 of the Microsoft .NET Framework installed. You can tell if you have it installed by going to ADD OR REMOVE PROGRAMS from the Windows Control Panel and looking for its entry.

Microsoft Data Access Components (MDAC) Version 2.6 (or later) – (Windows 2000 Only)

To use the *Data Collection* or *Report Generator* Utilities, you must have at least version 2.6 of the Microsoft MDAC drivers installed. While most machines do have them, they are not one of the suggested installations during Windows Update. If you attempt to run one of these utilities with an older version of MDAC, you will receive a message such as this:



The latest version of MDAC is 2.8, and it can be downloaded from the Microsoft support web site (http://msdn.microsoft.com) by searching on the keyword "MDAC". You can also download the MDAC 2.8 installer from our website in the same location as the WorkPlace Pro Utilities install file. Windows XP automatically updates the MDAC drivers itself.

Microsoft Excel

To use the *Report Generator Utility*, you will need Microsoft Excel installed on your machine. Excel 2000, Excel 2002 (XP), and Excel 2003 are all compatible.

Installing Invensys I/A Series SMART

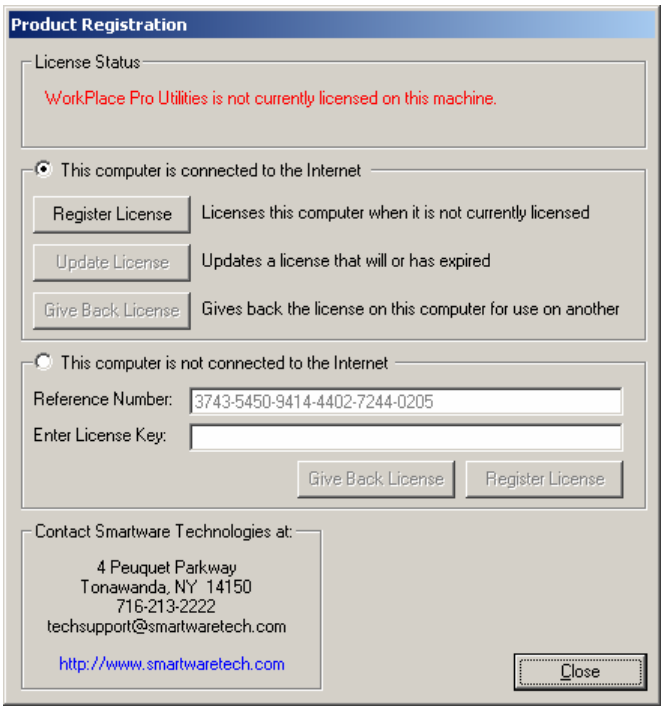
To install Invensys I/A Series SMART, simply download the latest installation program from our web site and run it. If you are upgrading from an earlier version of the program, you should choose to install over the previous version into the same directory. You do not need to uninstall the previous version first.

License Registration

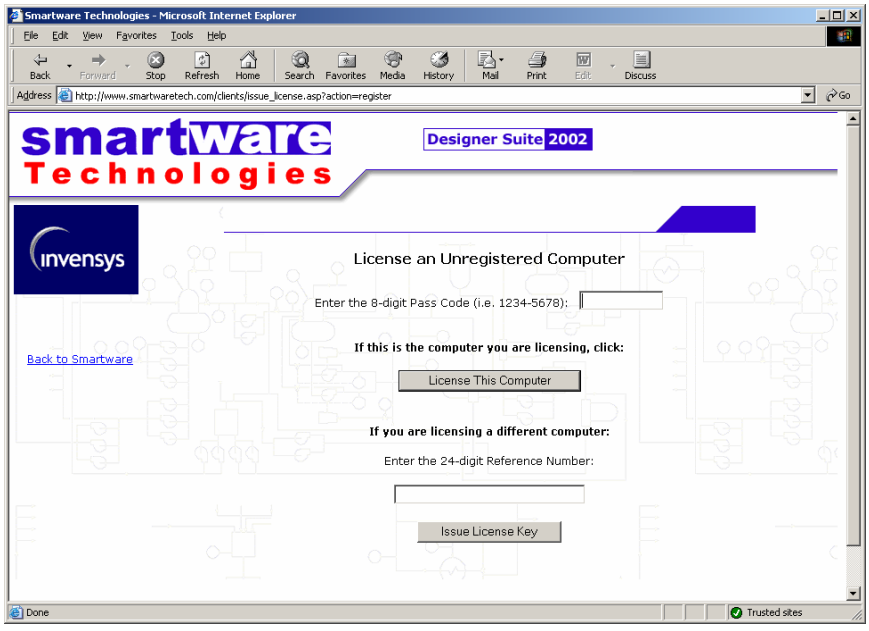
Each copy of Invensys I/A Series SMART needs to be licensed before it can be used. To activate a license on a machine you will need an 8 digit passcode from our sales and support department.

Licensing a New Computer

Click on the LICENSE REGISTRATION button on the main screen. This will bring up the License Registration dialog:



Assuming the machine is connected to the internet, click on the REGISTER LICENSE button. This will automatically open up a browser window and take you directly to our client login page. After you enter your user name and password, you will be directed automatically to our LICENSE AN UNREGISTERED COMPUTER page.



Enter the passcode and click LICENSE THIS COMPUTER. When you return to Invensys I/A Series SMART and close the License Registration dialog, the main screen should the new license. A passcode can only be used one time.

Updating a License

You will need to update your license periodically. This allows us to automatically check for program updates and to enable new features. To update an existing license, simply click the UPDATE LICENSE button in the License Registration dialog. This time you will automatically be logged in to the web site and the license will be automatically be updated. You do not need a passcode to update your license.

Moving a License to Another Computer

To move your license from one computer to another (from an office desktop to a home laptop, for instance), you simply give back the license from the desktop by clicking the GIVE BACK LICENSE button in the License Registration dialog. This will automatically log you in to web site, deactivate the license on the computer, and return to you a new 8 digit passcode. You can then use this new passcode to license a different computer using the procedures detailed in *Licensing a New Computer*.

Licensing a Computer That is Not Connected to the Internet

While it is much easier to perform the licensing actions when the machine is connected to the internet, it is still possible to do so if it is not. You will, however, still need access to the internet from some machine.

Go to our website and log in. From the main menu, select LICENSE REGISTRATION. There you will find a menu of choices for installing, updating and giving back licenses. Follow the instructions on the bottom section of each page, which require you to enter the 24 digit reference code from the License Registration dialog, and return a longer code to enter back into the machine being licensed.

If you have any problems with the licensing procedures, please contact us immediately at (716) 213-2222 or at www.smartwaretech.com.

2. Using the SMART Utilities

Invensys I/A Series SMART includes six utilities that are dedicated to the monitoring and maintenance running stations.

These utilities include:

- The *Station Status Report Utility* – generates and e-mails a report from a set of stations indicating their status and health, or lack thereof.
- The *Backup and Transport Utility* – automates the backup of an Enterprise Server, stand-alone UNC or even a set of Windows files. Also sends backups, collected data, and generated reports to users and administrators via e-mail, FTP or the internet.
- The *Data Collection Utility* – queries and copies data from the Cloudscape database to a local database.
- The *Data Reporting Utility* – generates Excel reports based on collected data.
- The *Task Scheduler Utility* and the *Batch File Utility* – automate any of the other utilities to run at specified times and in specific orders.

Each of these utilities is covered separately in the chapters that follow. The rest of this chapter will discuss features that are common to more than one utility.

Saving and Loading a SMART Data (.WPx) File

Each one of the utilities allows you to save its configuration parameters to a file for later use. This file contains the settings for the specific task, such as the stations to monitor or the location to save backup files.

The configuration files saved by the utilities have the following extensions:

.WPQ	Station Query
.WPB	Backup and Transport
.WPC	Data Collector
.WPR	Report Generator
.WPH	Batch Files

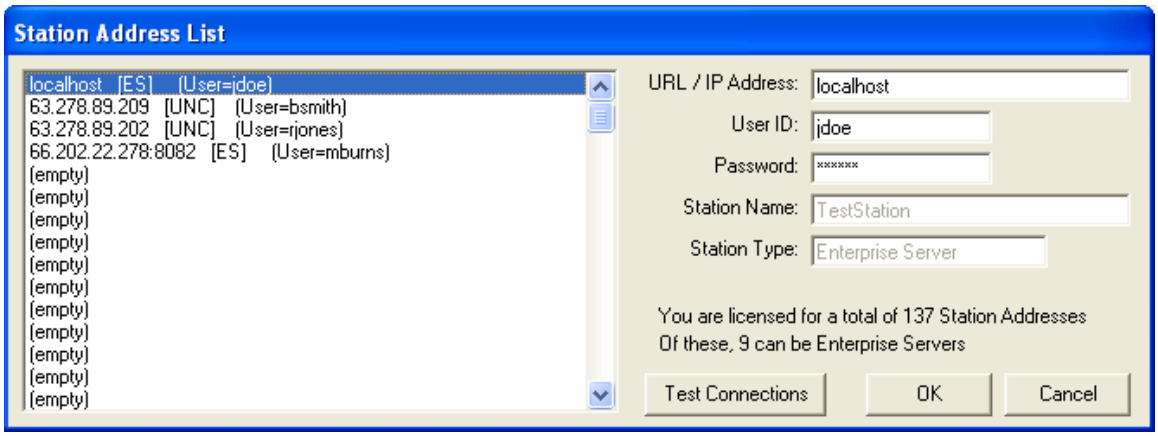
Each of the utilities has a SAVE and LOAD command on the FILE menu for their specific configuration file.

Launching a WPx File from Windows Explorer

The Invensys I/A Series SMART installation program creates file associations for each of the WPx file types. This allows you to double-click on any of these files in Windows Explorer and have Invensys I/A Series SMART automatically run, launch the correct utility and load the appropriate WPx and XML files.

The Station Address List

Invensys I/A Series SMART allows you to maintain the list of stations you access with the monitoring utilities in a global list, called the Station Address List. You can access this list from the STATION ADDRESS LIST button on the Main Menu form.



To add or edit a station to the list, simply select one of the empty slots in the list on the left and enter the IP ADDRESS, USER ID and PASSWORD for the station on the right. When you click the OK button to close the form, or if you click the TEST CONNECTIONS button, an attempt will be made to contact each station to get its STATION NAME and STATION TYPE (Enterprise Server or UNC). If the station cannot be contacted, its STATION TYPE will be shown as "???".

The number of available slots for stations depends on how your license account was set up. Your license will allow you to have a certain number of universal stations (Enterprise Server or UNC) and a certain number of stations that can only be UNCs. In the example shown, the license reflects 10 universal stations and 10 UNCs, for a total of 20. You could not, in this example, specify 11 Enterprise Servers at the same time, but you could have 9 Enterprise Servers and 11 UNCs.

If a UNC station cannot be contacted at the time the address is entered in the dialog, it will be treated as an Enterprise Server for licensing purposes. To update such a UNC at a time when it is available, simply go back to the Station Address List dialog and click OK to contact and update the station record.

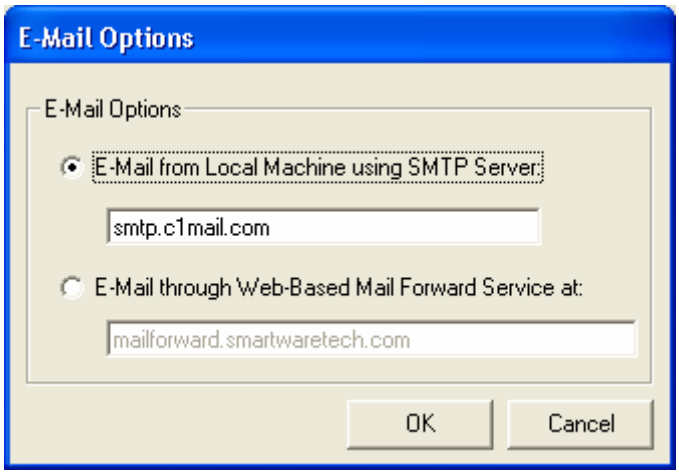
You can change the station addresses as often as you like. The limitations reflect only the number of simultaneous addresses you can have stored for automated use. If you need to support additional stations concurrently, please contact Smartware Technologies directly about upgrading your account.

The address list is stored in a file called *StationAddressList.xml* in the directory where you installed the program. You can copy this file from one machine to another to quickly set up a set of addresses. However, the utilities will not recognize the updated list until you go into the Station Address List dialog box and click OK.

Sending Automated E-Mail

Most of the Utilities have features that automatically send e-mail to users or administrators, sometimes with attachments. For machines that are properly configured to send e-mail via the SMTP protocol, the e-mail can be sent directly. In cases where SMTP is not available, the e-mail can instead be sent through a mail-forwarding web service requiring only a standard internet connection.

To specify how e-mails should be sent for all the utilities, click the E-MAIL OPTIONS button from the Main Menu form:



Configuring an SMTP server is beyond the scope of this document, but in many modern network environments it is not especially difficult. Please contact your network administrator for more information, including the proper SMTP Server name to enter into the E-Mail Options dialog (e.g. *smtp.c1mail.com*).

The web-based mail forward service is being provided by Smartware Technologies at no additional cost to our customers. To use it, you need only specify the internet address of the web site, which is mailforward.smartwaretech.com. This service is intended for light traffic, such as station status reports or output logs from automated operations such as data collection. It will support the attachment of files from the *Backup and Transport*

Utility, however there is a practical limit on the size of such attachments to prevent the service from being unavailable to others.

DISCLAIMER

Smartware Technologies, Inc. makes no warranty whatsoever, either explicit or implied, to customers who use the Mail Forwarding Service, with regard to its availability or reliability now or in the future. We cannot and will not be held liable for any damages or losses, monetary or otherwise, arising from this Service being unavailable or otherwise failing to deliver any information or perform as expected. We also cannot guarantee that this service will continue to be available at no cost, or even at all, in the future.

E-Mailing to Multiple Recipients

To send an e-mail to more than one recipient, specify a list of e-mail addresses separated with semi-colons.

Spam Filter Considerations

For the utilities that send e-mails, along with the "To" and "Subject" fields you are able to specify the "From" field that will appear on the e-mail. While this may seem to allow you to "spoof" (impersonate) anyone's e-mail address, the actual mail headers will likely have other information about the actual address from where the mail was sent. Accordingly, it is possible that some e-mail servers configured with spam filters might quietly reject your e-mail and throw it out with the weight-loss and get-rich-quick offers that comprise most of the world's internet traffic.

The automated nature of these e-mails also serves as a red flag for spam filters, and as security programs are tightened in response to new virus schemes, even the existence of certain words in the body of the message can cause the e-mail to be thrown out. For example, the word "complete", used innocently in the phrase "Data Collection Complete," was recently shown to trigger the latest updates to Symantec's heuristic spam filter to reject the message (the word was changed to "Finished").

The easiest way to deal with these issues is to test all your e-mailing, and if the messages appear to go through but yet do not arrive, check the spam filters on the receiving e-mail server and clients machines. In many cases they can be configured to allow mail from a specific e-mail address (real or otherwise), which you can specify as the "From" field. In Microsoft Outlook 2003, for example, you can select ACTIONS → JUNK MAIL → JUNK EMAIL OPTIONS → SAFE SENDERS.

Selecting a Local Database

The *Data Collection Utility* and the *Report Generator Utility* both require you to select a local database system to store and retrieve data. The word "local" in this case refers to a database other than the Cloudscape archive database maintained by an Enterprise Server. It can, in fact, be located anywhere on a network, including over an internet connection for those databases, such as Microsoft SQL Server, that support it.

There are a large number of database systems in the world, and almost as many different Microsoft technologies available to connect to them. Invensys I/A Series SMART was built to take advantage of the most current methodologies, while still providing access to the full range of databases and legacy systems in use, by using the latest Microsoft .NET providers that can be configured for older systems yet optimized for more modern database packages. It has also be designed to make it as simple as possible to connect to the most common database configurations by specifying the minimum amount of detail, yet still accepting advanced connection properties to optimize for speed and other provider-specific features such as connection pooling.

Note: *Due to variations in the SQL syntax amongst different database systems, you may encounter problems with those that have not yet been tested and certified compatible. Refer to the separate Release Notes document installed with the program for the latest list of recommended systems and known incompatibilities*.

The Microsoft Database Technologies

The software used to connect to and interface with a database system is commonly referred to as a provider. All the providers eventually connect to the database with parameters specified in a *Connection String*. There are currently three categories of database providers in use with Windows systems:

OLE DB Forgetting the archaic nature of the acronym itself, OLE DB represents the most common method still in use today. Generally speaking, a driver file (referenced by a Provider parameter, such as "Microsoft.Jet.OLEDB.4.0" for Microsoft Access databases or "MySQLPROV" for MySQL) is installed with the database and specified in the connection string. Invensys I/A Series SMART has been programmed to know the most common OLE DB providers.

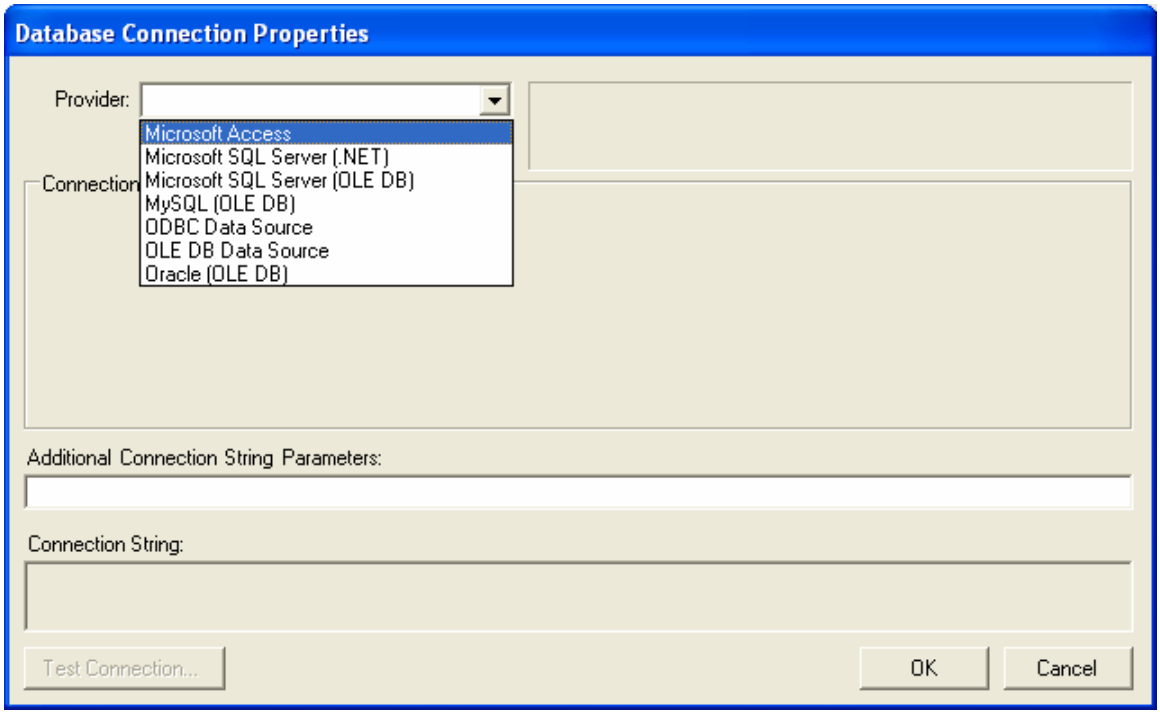
.NET Providers The Microsoft .NET Framework, which is being used to create the latest generation of software including Invensys I/A Series SMART, contains specific optimized drivers for certain database systems, including Microsoft SQL Server (Version 2000 or higher) and Oracle. As other .NET providers are made available, they will be added to Invensys I/A Series SMART.

ODBC The oldest of the technologies still in use, ODBC connections are generally configured directly in Windows through the ODBC Data Source Administrator in the Control Panel's Administrative tools. The connection is given a global identifier (the Data Source Name, or DSN), and all the parameters specific to the database system are specified as well. When an application needs to connect to the database, usually only the DSN will need to be referenced.

Using the Database Selection Dialog

Invensys I/A Series SMART cuts through the confusion and changes that Microsoft has laid out over the years by customizing its selection dialog to recognize the most popular database systems and require only the most commonly used properties. Advanced users, or those using less common database systems, still have the ability to provide whatever information is required to access their system.

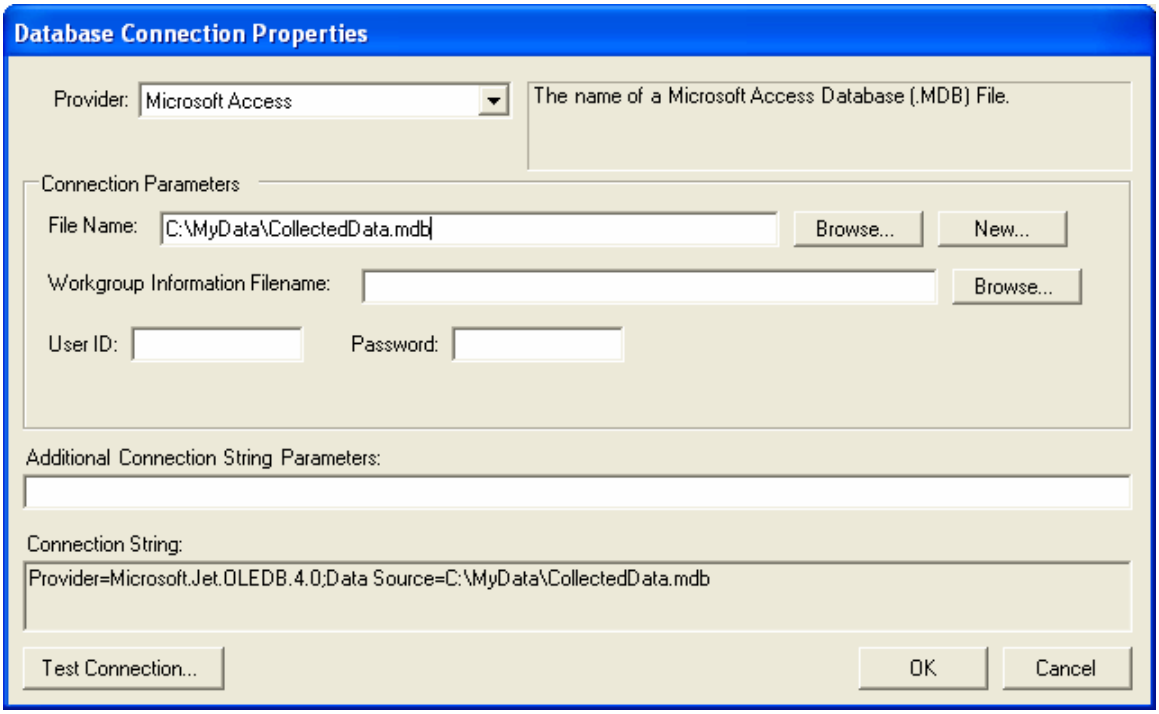
When you need to select a database connection, you will be presented with the Database Connection Properties dialog:



The PROVIDER list box contains a list of the various database systems and connection methods, some more specific than others. In some cases, such as with Microsoft SQL Server, you can connect using more than one type of connection. If your specific provider is not listed, you can use the more generic OLE DB or ODBC providers.

Connecting to a Microsoft Access Database

By far the simplest database to use is Microsoft Access. It does not require any special drivers or software to use, nor does it even require the Microsoft Access application itself. All the data is stored in a single file with a ".mdb" extension.

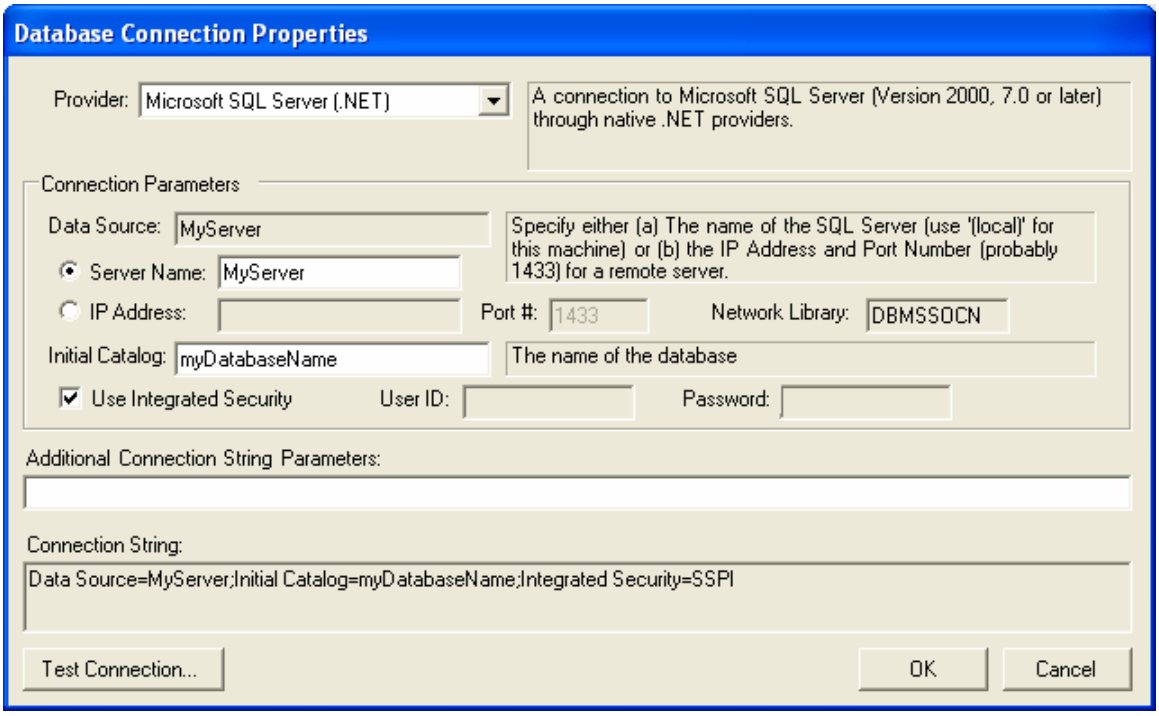


To use a Microsoft Access database, simply specify or browse to the appropriate .mdb file. You can also use the NEW… button to create a blank database file.

Depending on how the database was configured with regard to security, you may need to specify a USER ID and PASSWORD, and also possibly the name of a *Workgroup Information File* (typically ending with ".mdw"). Unless the database was created to require these, you can leave all three of these parameters blank.

Connecting to a Microsoft SQL Server Database

Not surprisingly, the most directly supported database is Microsoft's own SQL Server. For later versions, specifically Versions 7.0, 2000 (including the MSDE) or later, the native .NET provider will offer the best performance. For SQL Server 6.5 or earlier, you can use the OleDb Provider.

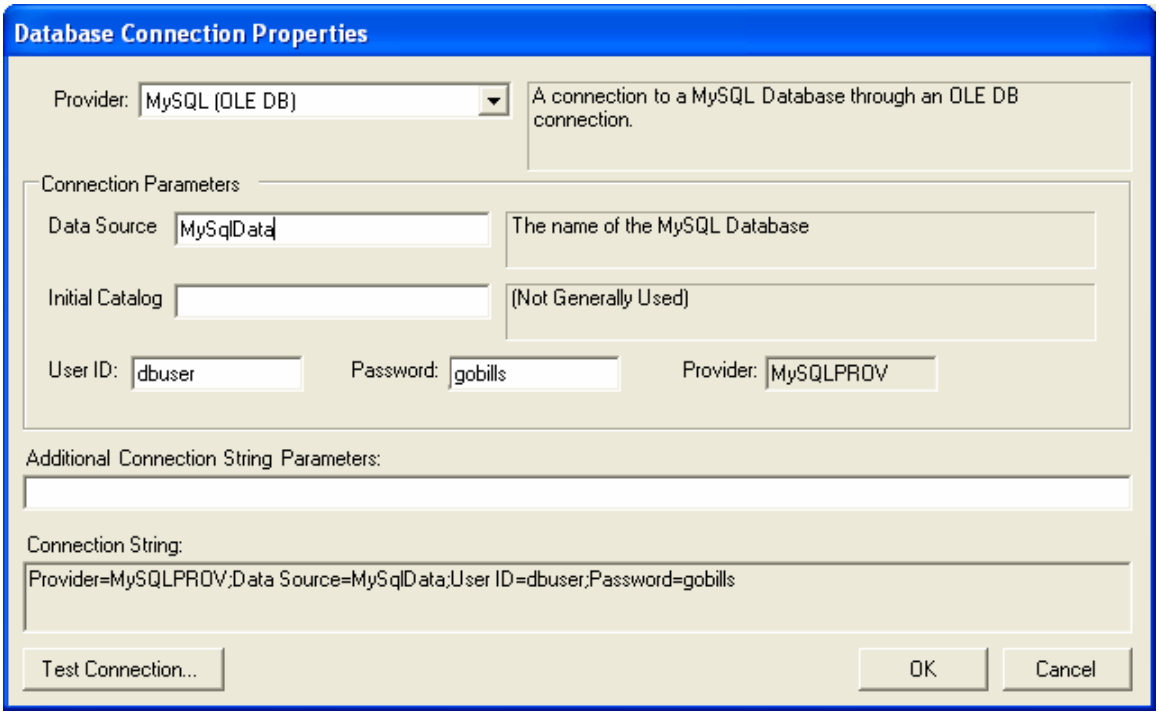


When connecting to Microsoft SQL Server, you must provide (at a minimum)

- The *Server Name* or *IP Address* (with a port, generally 1433)
- The name of the specific database on the server (also known as the *Initial Catalog*)
- The *User ID* and *Password*, or an indication that you will be using *Integrated Security* (where the user's Windows login is used to authenticate them to the database).

Connecting to a MySQL, Oracle or another OLE DB Database

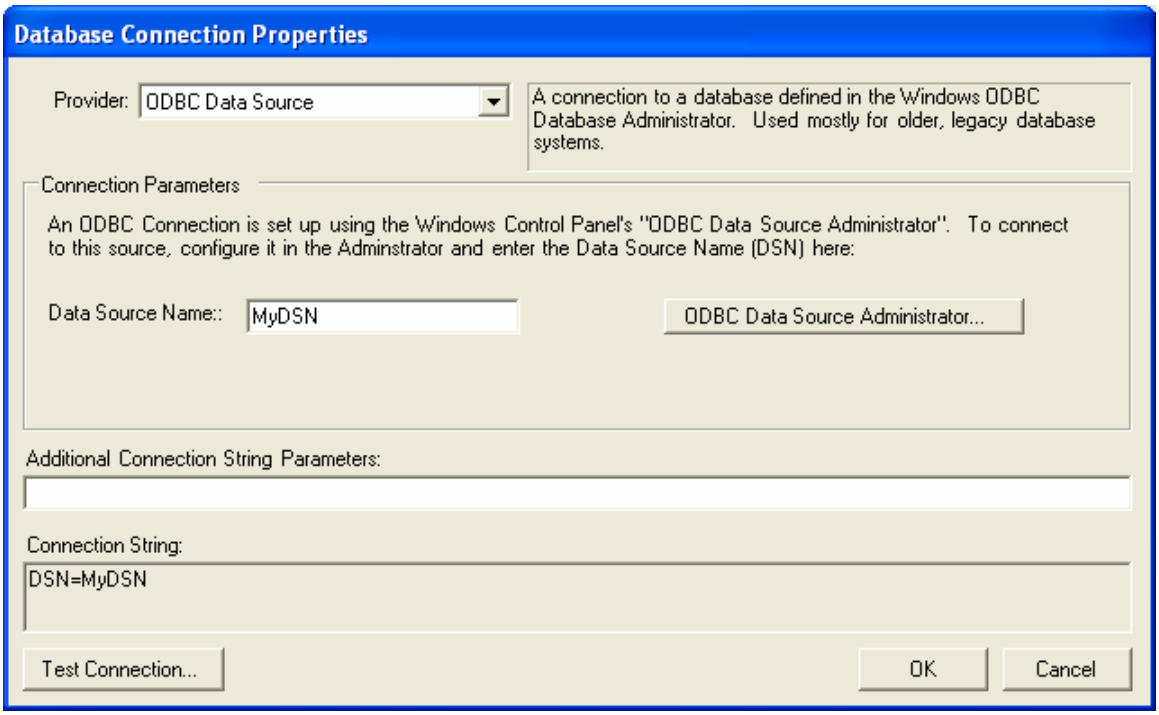
For OLE DB database connections, you need to provide the following information:



Data Source	This is usually the name of the database. The naming conventions vary from database system to system.
Initial Catalog	For some systems, such as Microsoft SQL Server, this is the actual name of the database (with the *Data Source* field being used to identify the server machine instead). For many other providers, this field does not need to be provided.
User ID, Password	These are the security parameters, as needed
OLE DB Provider Name	This is automatically filled in when selecting a known database. For other databases, consult the database's documentation or other internet resources.

Connecting to an ODBC Database

Since ODBC connection parameters are configured in the Windows ODBC Data Source Administrator, you need only provide the *Data Source Name* (DSN) to connect to it.



For convenience, you can access the ODBC Data Source Administrator directly from this dialog.

Customizing the Database Connection String

In some cases, such as when optimizations are required or when using one of the generic providers, you may be required to specify additional connection parameters in order to access the database. These parameters are usually detailed in the documentation for the database itself. To add these parameters, simply enter them in the ADDITIONAL CONNECTION STRING PARAMETERS field towards the bottom of the form. Parameters are usually specified in the format "*Parameter*=*Value*", with semi-colons separating each pair.

For example:

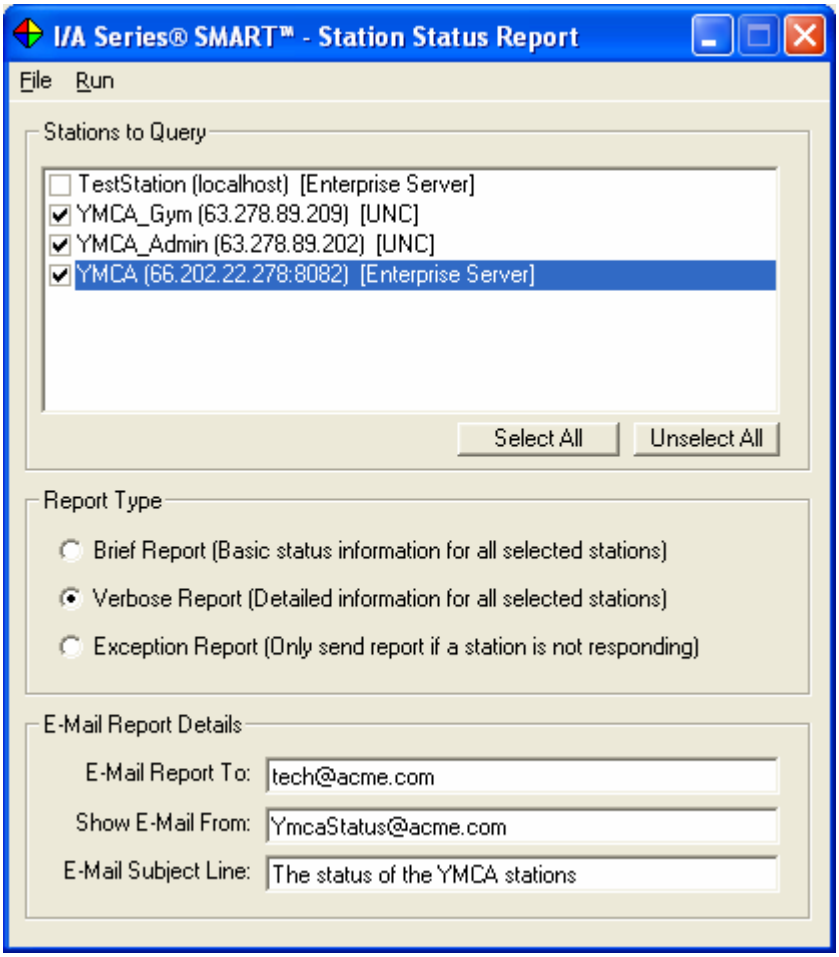
```
Pooling=true;Min Pool Size=0;Max Pool Size=100
```

Testing the Database Connection

For any of the providers, you can click the TEST CONNECTION button to confirm that the parameters are set correctly. If the connection fails, a detailed error message, as generated by the database provider, will be displayed.

3. The Station Status Report Utility

The *Station Status Report Utility* provides a simple way to generate immediate and automated reports on the health, or lack thereof, of one or more stations. It is very simple to configure, and when used in conjunction with the *Task Scheduler Utility*, can provide an ongoing assurance to customers and technicians that everything is working properly, and can also serve as a first-response system for stations that have gone down.



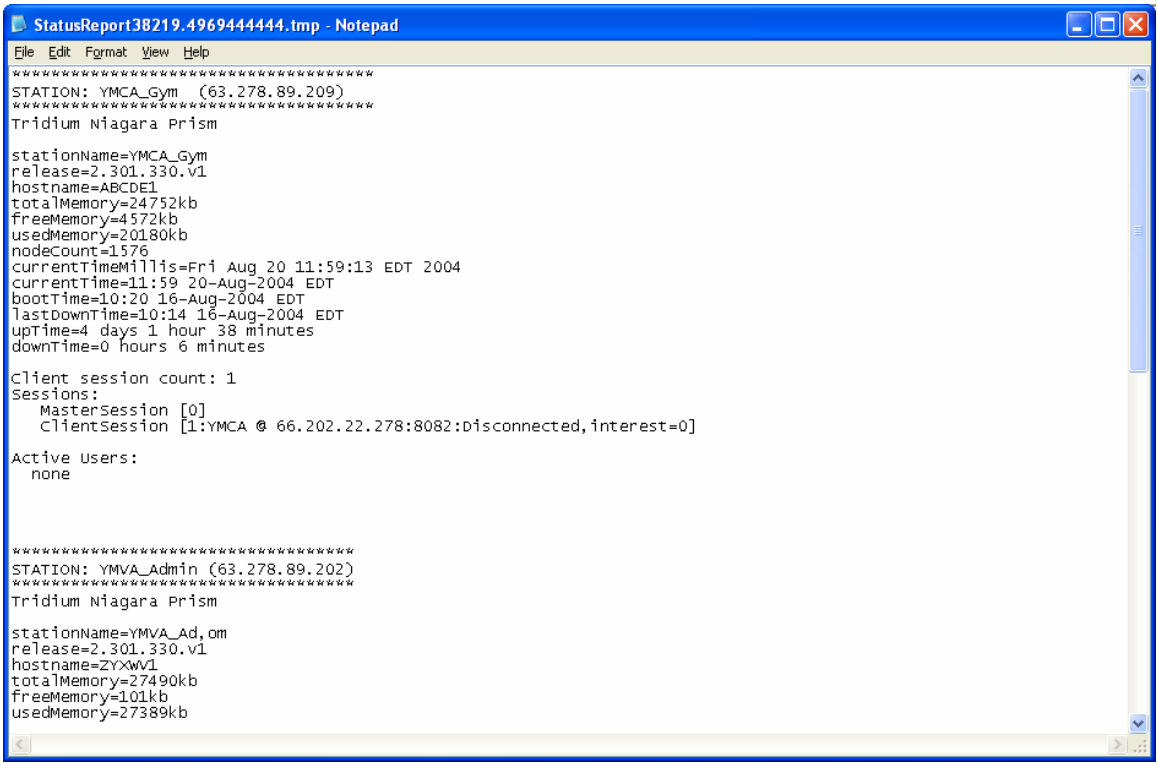
Defining a Report

To set up a Station Status Report, select the following information:

STATIONS TO QUERY	From the list of those that are registered in the Station Address List, simply check the ones you want to contact for status information.
REPORT TYPE	Choose between *Brief* and *Verbose* reports, which provide response data for each station queried, or an *Exception* Report, which reports only on stations that are not currently responding.
E-MAIL REPORT DETAILS	Specify where the report should be sent. For running interactively, the report will be displayed on the screen. In an automated scenario, the e-mail will be the only result of the status query. For more information on e-mailing, refer to the earlier chapter *Using the SMART Utilities*.

Running a Status Report

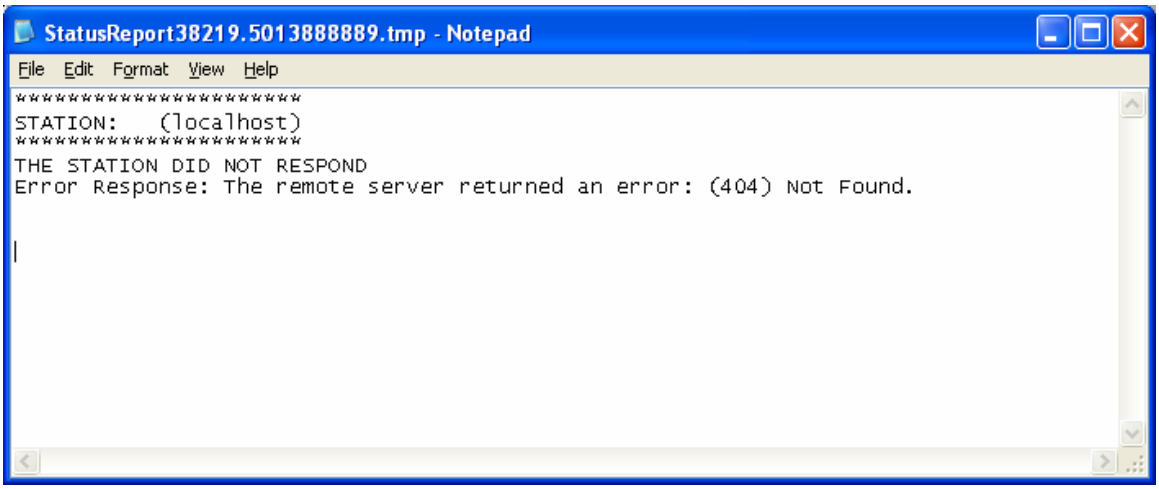
To query the selected stations and run the report, select STATUS REPORT from the RUN menu:



The report shows the information provided by the Prism servlet on each station. The *Brief* report truncates the data to include the most dynamic status properties, while the *Verbose* report includes all the information including more static items such as installed services and modules.

Using the Exception Report

When you run an exception report, each station is queried, but only those that did not respond properly are noted in the report.



If all the stations are up, the report will be blank and no e-mail will be sent. This is particularly useful in an automated scenario, where the stations can be queried once a day, once an hour, or even every minute to ensure that someone will be notified the moment the station no longer responds.

You should be aware, however, that a negative response may occur due to problems other than the station being down. Bad connections between the station and the internet, or the querying workstation and the internet, will also prevent the query from succeeding.

Also recognize that if you rely on a workstation to poll the stations and report problems, the status of that workstation is also important. If the workstation went down or was otherwise unable to send the exception reports, nobody would be notified if a station it was monitoring went non-responsive. There are a number of workarounds, however, such as having two separate workstations in different locations monitoring the same stations, or by having the monitoring workstation also send a regular status report on a less frequent, but regular basis. The absence of this second report would indicate a problem with the monitoring workstation.

Loading and Saving a Report Definition (.WPQ) File

You can save the configuration of your report as a file (with a .WPQ extension) by selecting SAVE WPQ FILE from the FILE menu. You can then reload the settings by selecting OPEN WPQ FILE from the FILE menu.

Running an Automated Report

In most cases, you will probably want the status reports to be run on a regular, automated basis. Invensys I/A Series SMART accepts command line arguments that allow you to have it launch the program, open a WPQ file, generate a Report, e-mail the report and then automatically shut itself down. This is particularly useful when specified as part of a batch file or as a scheduled task.

The *Task Scheduler Utility* and *Batch File Utility* can handle all of the details of selecting a .WPQ file and setting the appropriate command line arguments. For reference, the format for the command line is:

```
WPUtilities.exe  Report.wpq  RunReport
```

The first argument is the name of the Invensys I/A Series SMART executable file. The second argument should be the full path of the WPQ file that contains the report parameters. The third argument is the command to execute, which must the single word "RunReport".

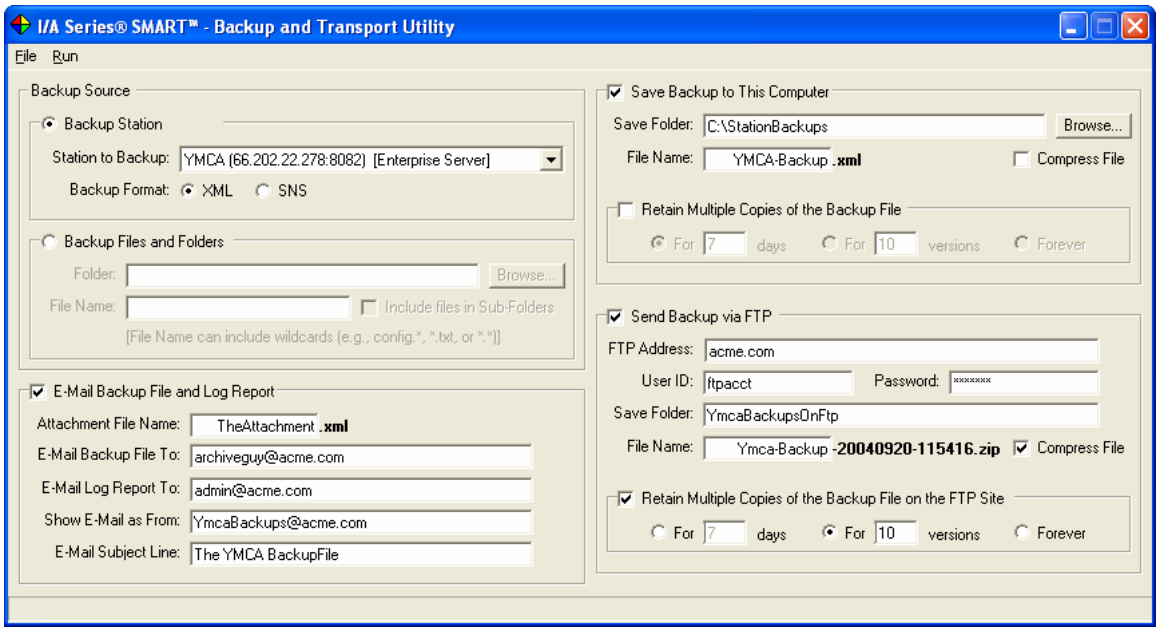
For example:

```
C:\Program Files\Invensys IA Series SMART\WPUtilities.exe
                            "C:\My Reports\StatusReport.wpq" RunReport
```

Depending on where its used and how your environment variables are set up, you may need to use the full path for the executable file. In all cases you should use the full path of the report definition file, including drive letters. If the path contains a space anywhere within it, you must enclose the entire path in double quotes.

4. The Backup and Transport Utility

The *Backup and Transport Utility* is a versatile tool for backing up stations, as well as regular files and directories, and transporting them to other hard drives, FTP sites or e-mail accounts. It can be used in conjunction with many of the other utilities, such as the *Data Collection Utility* (backup a station before collecting log data from it; transport the collected data in the form of a Microsoft Access database) or the *Report Generator Utility* (transport the resulting report), or it can be used as a stand-alone backup system for files not related to a station at all. When used in conjunction with the *Task Scheduler Utility*, it is very simple to create an automated, reliable and redundant backup configuration for many different systems.



In the example shown above, the following events would occur when this backup is executed:

- The specified Enterprise Server would be contacted directly and a backup of the station in the XML format requested.
- The XML file would be copied to "C:\StationBackups\YMCA-Backup.xml". By choice, it will not be zipped, and any existing version would be overwritten.
- The XML file will then be zipped up and copied into the "YmcaBackupsOnFtp" folder on the specified FTP site with a time stamped filename (e.g. "Ymca-Backup-20040820-151210.zip"). If there are more than 10 versions of this file in the folder, the oldest ones will be deleted.

- The XML file will be renamed as "TheAttachment.xml", zipped up, and e-mailed to "archiveguy@acme.com" (the e-mail will be from "YmcaBackup@acme.com" with a subject line of "The YMCA Backup File").
- Finally, a log report of all this activity and any problem that may have occurred will be e-mailed to "admin@acme.com".

Selecting the Backup Source

The utility can be configured to request a backup file from a running station, or it can copy and compress one or more files from a local or network hard drive. The resulting file (.xml or .sns for a station backup, .zip for one or more compressed files) will then be transported (locally, by FTP or by e-mail) in accordance with the rest of the parameters specified on the form.

Backing up a Station

To backup a station, simply select it from the list box. Only stations specified in the Station Address List can be selected (refer to the earlier chapter *Using the SMART Utilities* for more information). You can also select between the XML format or the SNS format for the backup.

Backing up a Single File

If you need to configure the utility to backup or transport a single file, first select the BACKUP FILES AND FOLDERS radio button in the BACKUP SOURCE group box. In the FOLDER field, specify the name of the directory where the file is located. This can be a local hard drive, a mapped network drive, or a full network path. Finally, specify the name of the file within the folder in the FILE NAME field.

Backing up Multiple Files From a Single Folder

To backup multiple files together into a single .ZIP file, use standard filename wildcards in the FILE NAME field. For example, you could use "*.doc", "config.*", or the age-old favorite "*.*" to include all the files in the folder.

Backing up Multiple Files From Multiple Folders

If you check the INCLUDE FILES IN SUB-FOLDERS check box, the utility will search for all files that match the FILE NAME (or the wildcard pattern, if specified). The search will include the specified FOLDER, as well as every folder beneath it, regardless of the names of the folders.

Saving Backups Locally or to a Network Location

The first possible destination for delivery of the backup file is to a local or network hard drive. Check the SAVE BACKUP TO THIS COMPUTER checkbox, and then specify the fully qualified name of the local or mapped hard drive (such as "F:\Temp") or the network path to a file server (such as "\\FS1\BackupFiles") in the SAVE FOLDER field.

In the FILE NAME field, enter the prefix (without the file extension) of the name you want to use when saving the backup. In general, the file extension (which is shown on the form) will be based on the source of the backup (.XML or .SNS for uncompressed station backups) and whether the file is being compressed (.ZIP) explicitly (by selecting the COMPRESS check box) or because there is more than one file in the backup. In the specific case of a single file being copied but not compressed, you will need to specify a file extension yourself.

Retaining Multiple Versions of the Backup File

For regular backups, it is a common practice to keep more than one version of the backup in case a corruption is detected that might exist in the most recent backup as well. If you check the RETAIN MULTIPLE COPIES OF THE BACKUP FILE checkbox, a timestamp will automatically be appended to the file name (before the file extension). You will then have the option of retaining files for a specified number of days (including fractional days), a specified number of versions or forever. Older versions of the file, based on these criteria, are automatically purged after a new version of the backup is generated.

Storing Backups on an FTP Site

The second possible destination for delivery is an FTP site. You have all the same options as you have when saving to a local or network drive, including retaining multiple copies and automatically purging old ones.

You will need to specify the ADDRESS of the FTP site (omitting the "FTP://" prefix that would normally be used to connect to it), as well as a valid USER ID and PASSWORD for an account with sufficient permissions on that site to create and write to files. If you are retaining files for a specified number of days or versions, you will also need explicit permission to delete files.

E-Mailing Backup Files and Log Reports

There are two possible e-mails that can be generated by the *Backup and Transport Utility*. The first is an e-mail that includes the backup file itself as an attachment. The second is an e-mail of the log report, showing the actions that the utility took in backing up and sending the files.

To send either e-mail, first check the E-MAIL BACKUP FILE AND LOG REPORT checkbox. For more information on e-mailing with Invensys I/A Series SMART, refer to the earlier chapter *Using the SMART Utilities*.

E-Mailing the Backup File

To e-mail the backup file, fill in the ATTACHMENT FILE NAME and E-MAIL BACKUP FILE TO fields. If the result of the backup was not already a .ZIP file, the file will automatically be compressed before it is sent (both for performance issues and to avoid problems with e-mail servers that reject certain types of file extensions). In this case the file name specified as ATTACHMENT FILE NAME will be used for the file within the .ZIP file.

You can specify a list a recipients in the E-MAIL BACKUP FILE TO by separating the e-mail addresses with semi-colons.

There are practical limits to the size of the file you can attach. These will be dependant on your outgoing e-mail server, the recipient's e-mail server, and other parameters set by the system administrators.

E-Mailing the Log Report

For automated backup scenarios, it is also a good idea to e-mail the Log Report file to a technician or administrator by specifying their e-mail addresses in the E-MAIL LOG REPORT TO field. The log report will detail the actions taken by the utility in copying, compressing, uploading and e-mailing the backup files, including the amount of time the operations took. The absence of this report can also serve as an indication that a severe problem occurred, allowing the situation to be noticed and resolved immediately.

Loading and Saving a Backup Definition (.WPB) File

You can save the configuration of your backup task as a file (with a .WPB extension) by selecting SAVE WPB FILE from the FILE menu. You can then reload the settings by selecting OPEN WPB FILE from the FILE menu.

Running an Automated Backup

In most cases, you will probably want the backup to be run on a regular, automated basis. Invensys I/A Series SMART accepts command line arguments that allow you to have it launch the program, open a WPB file, process the backup, e-mail a log report and then automatically shut itself down. This is particularly useful when specified as part of a batch file or as a scheduled task.

The *Task Scheduler Utility* and *Batch File Utility* can handle all of the details of selecting a .WPB file and setting the appropriate command line arguments. For reference, the format for the command line is:

```
WPUtilities.exe  BackupInfo.wpb  RunBackup
```

The first argument is the name of the Invensys I/A Series SMART executable file. The second argument should be the full path of WPB file that contains the backup parameters. The third argument is the command to execute, which must the single word "RunBackup".

For example:

```
C:\Program Files\Invensys IA Series SMART\WPUtilities.exe
                            "C:\My Scripts\Backup.wpb" RunBackup
```

Depending on where its used and how your environment variables are set up, you may need to use the full path for the executable file. In all cases you should use the full path of the backup definition file, including drive letters. If the path contains a space anywhere within it, you must enclose the entire path in double quotes.

5. The Data Collection Utility

The Enterprise Server software includes a facility for archiving data collected from logs into a separate database that it maintains. This archive is stored in an IBM Cloudscape database system, and the data can be accessed using a number of browser-based tools documented in the *Niagara Browser Access Guide* and the *Niagara Web Solutions Guide*.

The *Data Collection Utility* is designed to expand on these capabilities and essentially provide an additional front end to the archive database that allows you to create and maintain snapshots of the archive data to work with locally. This allows you to leverage the wide variety of database tools and report engines available to manipulate, store and disseminate the information as needed. The other utilities, such as the *Report Generator Utility*, *Backup and Transport Utility* and *Task Scheduler Utility* can all be combined with the *Data Collection Utility* to create powerful data collection and reporting solutions without and other software required.

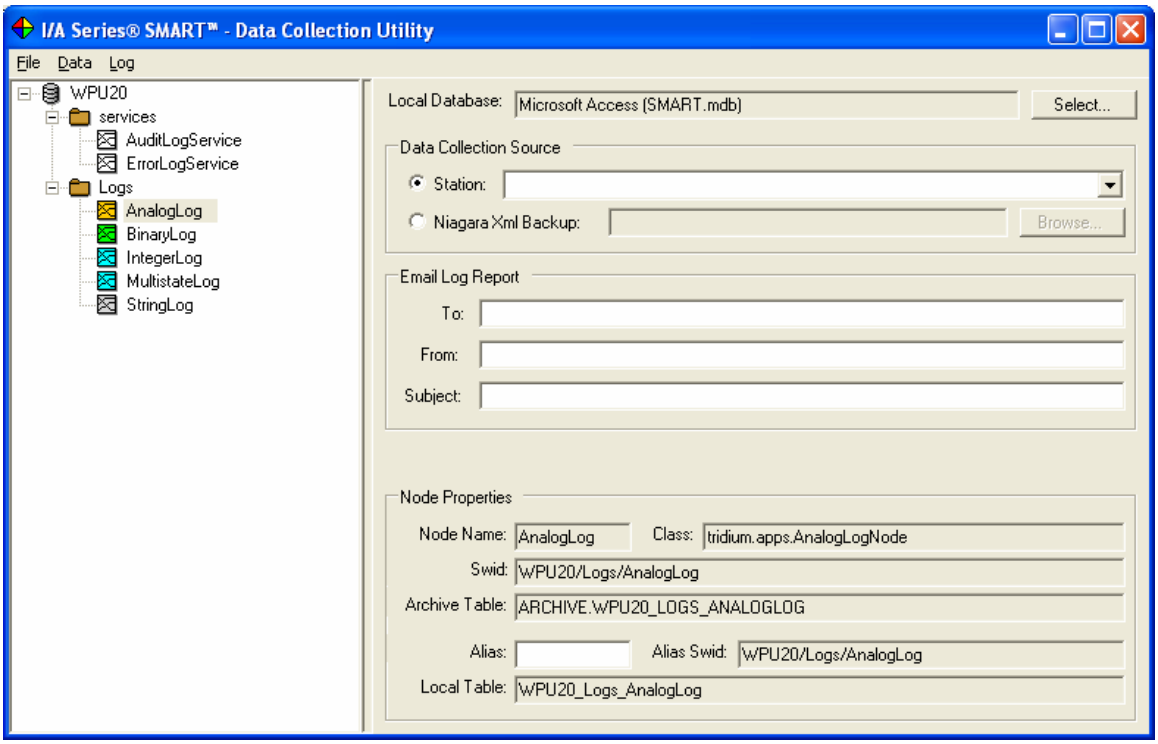
The *Data Collection Utility* can also serve to collect data from stations that are not being archived by WorkPlace Pro by scanning the log data directly from station backup files. This can provide the same archiving capabilities as an Enterprise Server in situations where none exists.

This chapter assumes you are familiar with how WorkPlace Pro stores data in logs, and how those logs can be configured to archive their data.

Loading a Station File

To begin collecting data from a station, you start by backing up the station to an XML file (either with the WorkPlace Pro Admin Tool or by using the *Backup and Transport Utility*). You then load this structure by selecting OPEN NIAGARA XML FILE from the utility's FILE menu.

The structure of the station's containers and logs will be shown in a familiar tree format. Only those containers that contain logs will be included in the tree.



If the structure of the tree changes, you can resynchronize the tree to the new version by using the SYNCH WITH NIAGARA XML FILE command detailed later in this chapter.

The Node Properties

As you click on any of the nodes in the tree, the fields in the lower portion of the right side of the form will be updated with the node's properties.

Node Name	The name of the container or log as defined in WorkPlace Pro.
Class	The WorkPlace Pro object class name.
Swid	The full Swid of the object in WorkPlace Pro
Archive Table	The name used by WorkPlace Pro for the archive table stored in the Cloudscape database. It is very similar to the log's Swid, with underscores replacing slashes and "ARCHIVE." prepended to it.

Alias An alternate name for the node, to help in situations where local database table names become too long. If an alias is specified, it will appear in the tree node, in parentheses, after to the node name.

Local Table The name that will be used for the table stored in the local database.

Using Alias Names to Shorten Table Names

Because the table names used to store the log data in the local database are created based on the names of all the parent containers, they can become quite long if the logs are stored deep within the station's tree structure or if the names of the containers are themselves relatively long. In some cases the maximum length of a table name in your local database may be less than that allowed by WorkPlace Pro or Cloudscape (such as with Microsoft Access, which allows a maximum of 63 characters, compared to approximately 127 characters for the other systems).

The ALIAS field allows you to provide an alternate name for a container or log for the purposes of creating local database table names (and only for this purpose). For example, consider a log that was located within two levels of containers and had the following Swid:

/db/WPU20/ContainerWithAVeryLongName/AnotherLongContainerName/AnalogLog

The table name generated for use in the local database would be:

WPU20_ContainerWithAVeryLongName_AnotherLongContainerName_AnalogLog

You could shorten this by providing aliases to one or both of the containers. If we set the alias on the first container (*ContainerWithAVeryLongName*) to "ShorterName", the local database table name would become:

WPU20_ShorterName_AnotherLongContainerName_AnalogLog

If we further alias the second container (*AnotherLongContainerName*) to "AlsoShort", the local database table name would be:

WPU20_ShorterName_AlsoShort_AnalogLog

You can alias the name of the log itself as well. Using shorter aliased names can also be useful to help reduce the size of the SQL statements that will eventually be used to query the data later. However, keeping the table names representative of the station's structure makes it easier to refer back to the station when changes and additions are made.

Important Note: You should be careful about changing the alias names of any logs or containers once you have started to collect data. Changing the alias of a container will change the target table in the local database, but in most cases will not rename the tables that already exist (though the old tables will not be deleted). You will need to either rename the existing table in the local database itself, discard the existing data or copy the data from the old tables in the new ones.

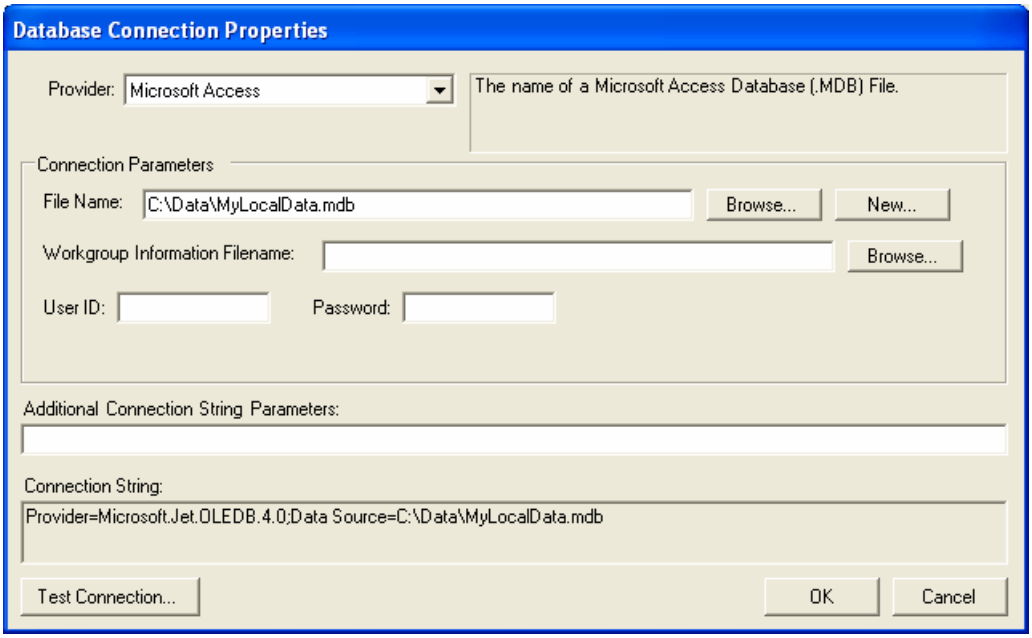
Selecting a Local Database

You can store the collected data in any type of Windows-compatible database system, from Microsoft Access to Microsoft SQL Server to Oracle. You select the database to use by clicking on the SELECT… button next to the LOCAL DATABASE field, or by choosing SELECT DATABASE from the DATA menu. The database selection dialog is described in depth in the earlier chapter *Using the SMART Utilities*.

Creating and Using a Microsoft Access Database

The simplest database to use is a Microsoft Access database file. The data is stored in a single file (.MDB) which is easily backed up and transported to other machines, the format is recognized by almost all data manipulation and reporting tools, and you do not need to own or install any additional software whatsoever to use it. You don't even need the Microsoft Access application itself.

To create and use a new Microsoft Access database, click the SELECT button to bring up the Database Selection dialog. Then select "Microsoft Access" in the PROVIDER drop-down list.



The only parameter you need to specify is the FILE NAME. If you already have an existing Microsoft Access database, click the BROWSE… button to locate it. If you'd like to create a new database file, click on the NEW… button and specify the location and name of the new file.

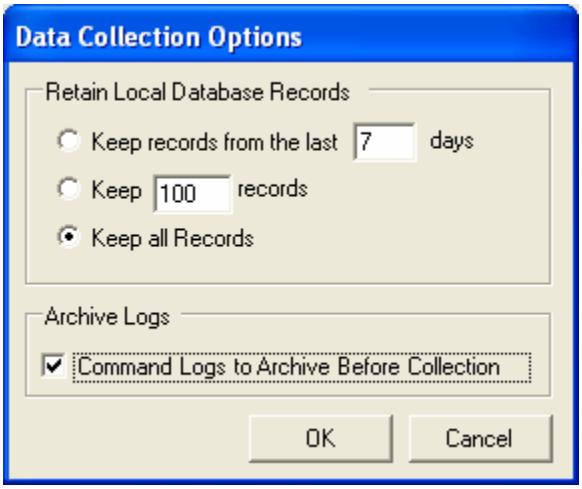
Selecting the Data Collection Source

The *Data Collection Utility* allows you to gather the log records directly from the Cloudscape archive database on an Enterprise Server, or from the actual logs in a snapshot backup of the station itself.

Collecting from an Enterprise Server

To collect the data from the Enterprise Server's Cloudscape archive, choose the STATION radio button in the DATA COLLECTION SOURCE box and choose the station from the listbox. The station must be listed in the Station Address List, as described in the earlier chapter *Using the SMART Utilities*.

The archive will not usually contain the absolute most recent records from the logs, as the logs themselves buffer a user-defined number of records and then archive their data when they become close to full, or on a daily basis, depending on how the log is configured in WorkPlace Pro. You can force the logs to archive their data to the database before they are polled for data collection by choosing EDIT COLLECTION OPTIONS from the DATA menu and checking the COMMAND LOGS TO ARCHIVE BEFORE COLLECTION checkbox.



Collecting from a Station Backup (XML) File

If there is no Enterprise Server in your system, you can choose to collect the log data from a station backup file. Choose the Niagara XML Backup radio button in the DATA COLLECTION SOURCE box and browse to the location of the XML file.

In an automated solution where you want to collect the data on a regular basis, you will need to generate an updated XML file from the station and copy it to this location before the collection occurs. This can be done easily using the *Backup and Transport Utility*. The *Task Scheduler* and *Batch File Utilities* can also be used to ensure that the backup is completed before the data is collected.

How Data Records are Collected

Since the archive databases can become quite large, and querying them can be resource intensive, the *Data Collection Utility* is optimized to acquire only the most recent records. Specifically, it uses the *TStamp* field included in all types of log records to determine the earliest record to collect (i.e., if the last record stored in the local database had a timestamp of *T*, the utility will only request records from the collection source that are time stamped after *T*).

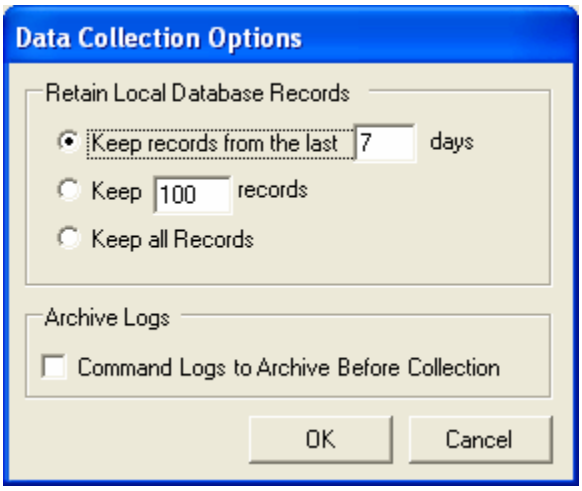
This has a few important implications. Since the station backup files contain the most recent and unarchived records, if you collect from this XML file first and then from the archive database, you will not receive any of the archived records (since they are by definition older than those in the log). This can be useful, especially if you are beginning to collect data from an existing archive and want to set a baseline for collection without bringing in all the historical data.

If, however, you begin collection from the archive and then switch to the backup file, some records could be omitted (depending on when the logs archived themselves and when the backup file was generated). The utility does not try to fill in gaps between the data it has collected and the data in the archive. It will always request newer records only.

Retaining Records in the Local Database

In some data collection scenarios, you will only be interested in a rolling window of the most recent log data. In these cases, the *Data Collection Utilit*y can be configured to automatically purge older records as new ones are collected, keeping the size of the local database to an absolute minimum.

To specify the amount of records to retain, select EDIT COLLECTION OPTIONS from the DATA menu:

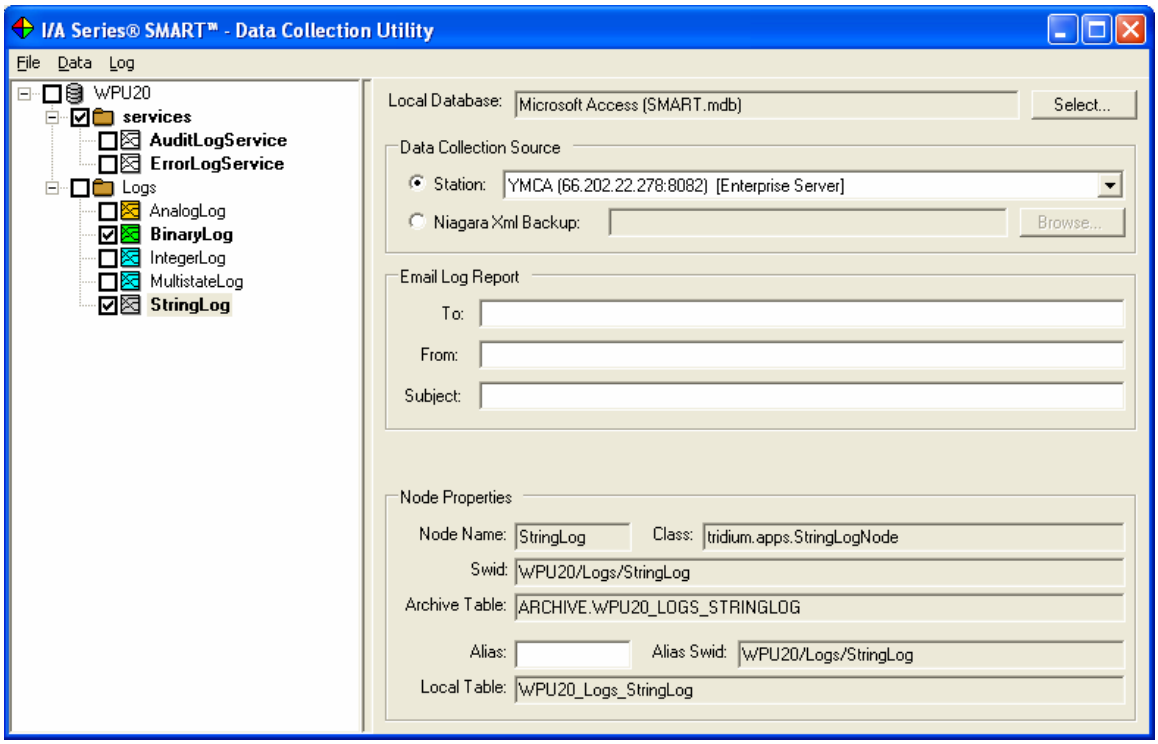


You can choose to keep records for a specific number of days, keep a specific number of records or simply retain all records indefinitely. This setting applies to all tables in the local database, and will be applied to each log the next time the data is collected.

Selecting the Logs to Collect From

Although you can work with a single log at a time, as described later in this chapter, you will most likely want to collect data from a set of logs on a regular, automated basis. To begin selecting the logs you want to collect from, choose SELECT LOGS FOR COLLECTION from the DATA menu.

When you are in selection mode, each of the containers and logs will appear with a checkbox in the station tree:



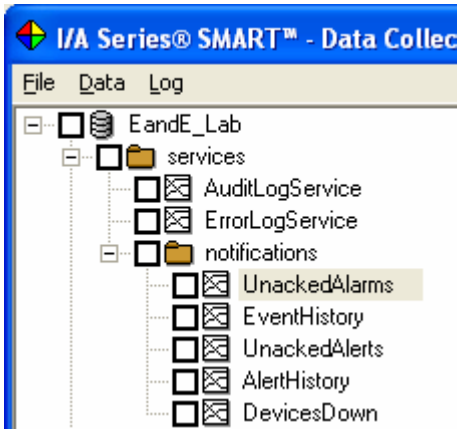
You can choose to collect from a specific log by checking its check box. The name of the log will appear in bold to indicate that it is selected. You can also choose to select all the logs within a container by checking the box next to the container name. In this case the container and all the logs will appear bolded.

Selecting SELECT LOGS FOR COLLECTION from the DATA menu a second time will hide the checkboxes to prevent accidental selection or unselection.

Notification Tables (Alarms and Alerts)

In addition to the log objects in the stations, the *Data Collection Utility* can also collect data from any of the predefined tables that are maintained in the station and in the archive database in association with the Notification Service. The include alarms and alerts, both acknowledged and unacknowledged.

Although they aren't actual objects in the station itself, the *Data Collection Utility* adds these tables to the tree to allow you to select and manipulate them. It creates a folder called *notifications* under the existing *services* container, and adds each of the corresponding tables as child nodes.



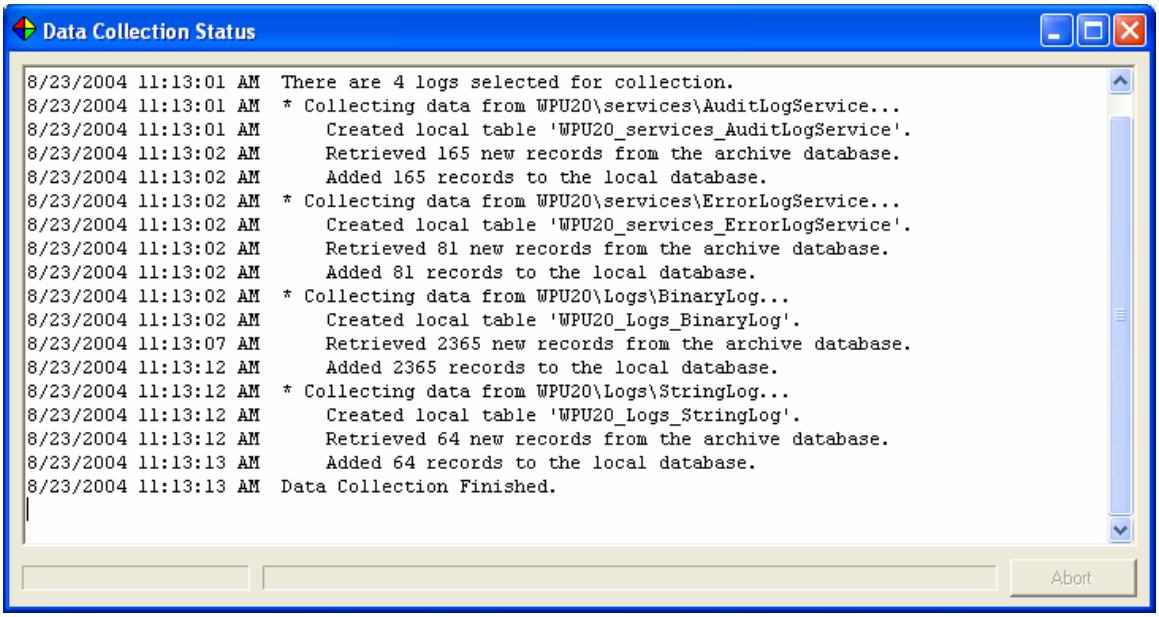
There are five tables that may exist within the station and the archive database:

Unacked Alarms	EVENT.UNACKEDALARMS
Event History	EVENT.EVENTHISTORY
Unacknowledged Alerts	ALERT.UNACKEDALERTS
Alert History	ALERT.ALERTHISTORY
Devices Down	APP.DEVICESDOWN

Collecting Data

To collect the data for all the selected logs, select COLLECT DATA from the DATA menu. For each log, the collection source (the Enterprise Server or the backup XML file) will be queried and the latest records stored in the local database. If the table does not yet exist in the local database, it will be created. If the retention options were set to purge older records, the appropriate records will be deleted.

A status screen will appear during collection to show you the status of the collection activity and to provide an option to abort the process. The resulting log can also be e-mailed to an administrator by completing the E-MAIL LOG REPORT fields on the main form.



Loading and Saving the Data Collection (.WPC) File

You can save the configuration of your data collection task as a file (with a .WPC extension) by selecting SAVE WPC FILE from the FILE menu. You can then reload the settings by selecting OPEN WPC FILE from the FILE menu.

Working with Individual Logs

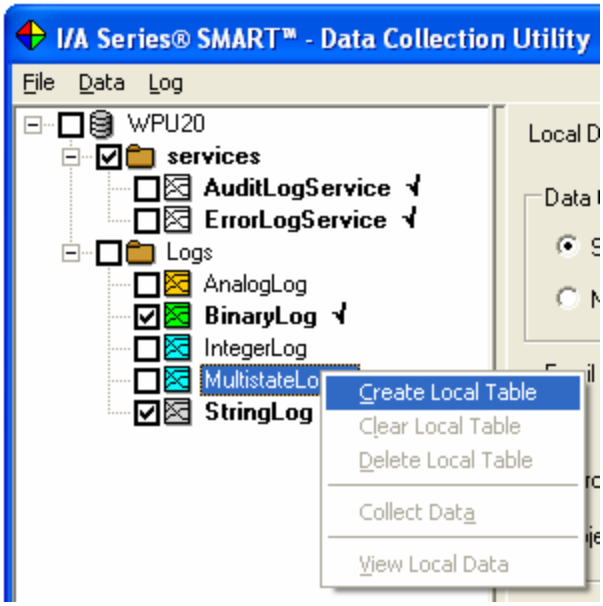
Once you have begun collecting data, you can use the *Data Collection Utility* to view and manipulate the data in the local database directly on a log-by-log basis. These commands are available by selecting a log in the tree and using the commands in the LOG menu. You can also right-click on the log to bring up a pop-up version of the LOG menu.

Checking for Existing Tables in the Local Database

When a .WPC file is loaded, the local database is automatically scanned to see which of the tables already exist. These logs are shown with a check mark after their name in the tree. To update this indicator for all the logs, select CHECK FOR EXISTING TABLES in the DATA menu.

Manipulating the Local Database Table

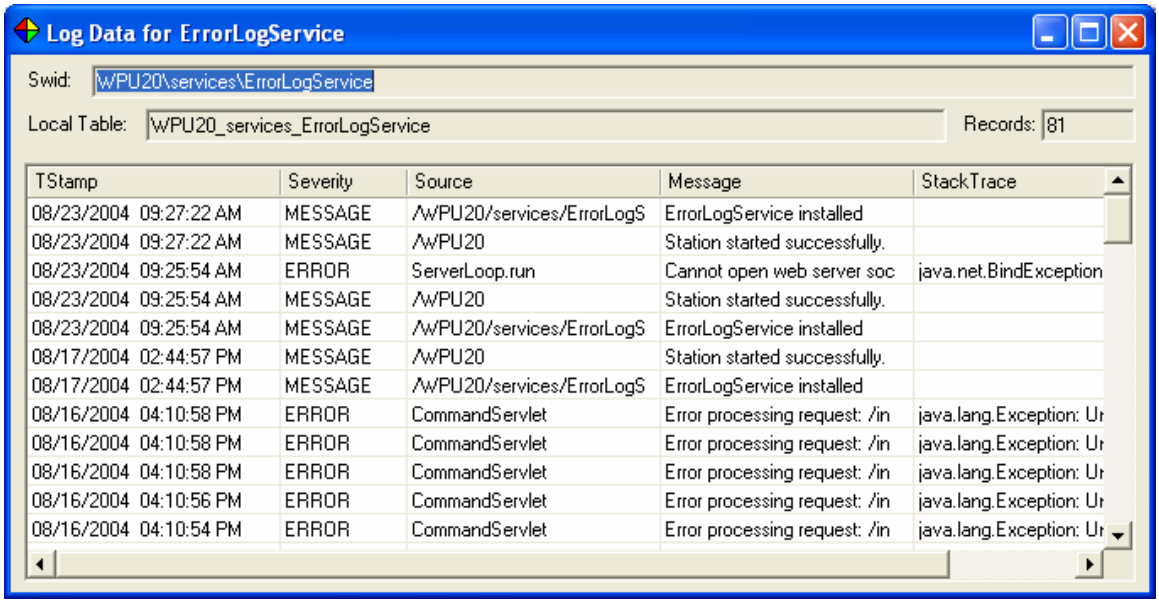
The LOG menu (or pop-up menu) contains commands to manipulate the data in the local database table. If the table does not yet exist (or if the local database hasn't been scanned for existing tables), the only option will be to CREATE LOCAL TABLE.



Once the table exists, you will have the following additional options:

CLEAR LOCAL TABLE	Deletes all the records in the log's table in the local database
DELETE LOCAL TABLE	Deletes the log's table (and all the records) in the local database
COLLECT DATA	Collects the updated records from only the selected log
VIEW LOCAL DATA	Displays the data from the local database

The VIEW LOCAL DATA window provides a quick way to view the contents of the local database table. You can click on the column headings to easily sort the records by any column:

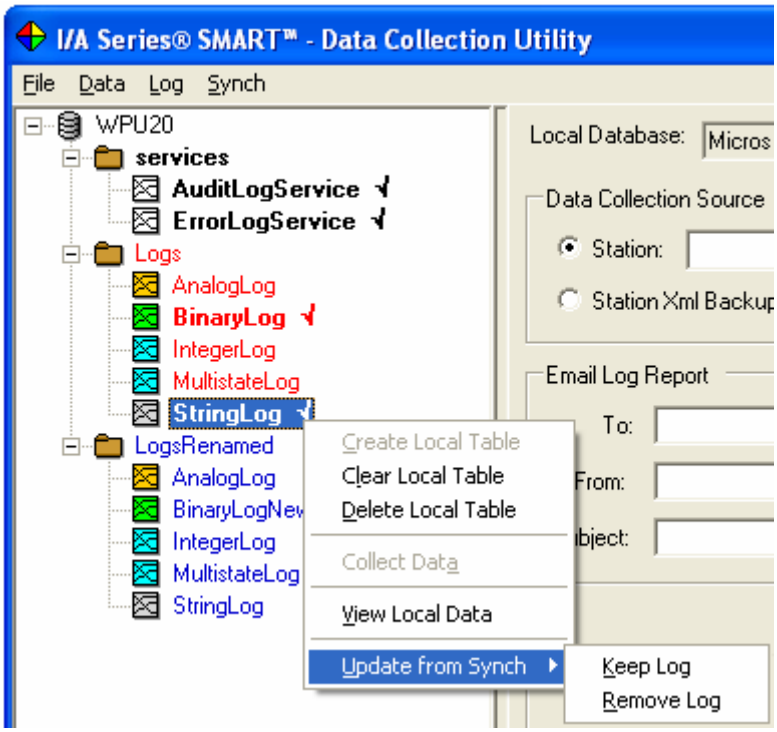


Synchronizing to an Updated Station Tree

If the station you are collecting data from is enhanced or otherwise updated, you will need to update the station tree to reflect these changes. You could simply load this updated station by selecting OPEN NIAGARA XML FILE, but you will lose all the information about which containers and logs are selected, as well as any alias names you may have specified. In these cases you should instead choose SYNCH WITH NIAGARA XML FILE from the FILE menu.

After selecting the updated XML file to synch to, you will receive a message indicating how many nodes (logs or containers) were added or not found (either because they were deleted or renamed):

The station tree will then be updated to reflect these changes. New nodes are displayed in the tree in blue, while missing nodes are shown in red. If the missing nodes had been selected, or if the new nodes were found in a selected container, the names will still appear bold to indicate the selection.

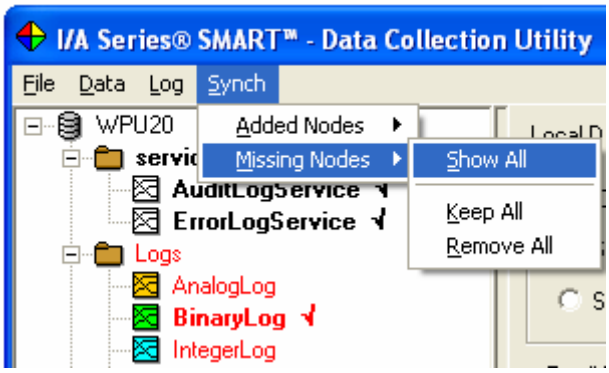


For each conflict, there are several options to resolve it. In each case the changes will be reflected in the station tree used to collect data, and will not cause any changes to the actual station running in WorkPlace Pro or contained in the backup XML file. Also, removing an existing log from the tree will not delete any data that may have already been collected into the local database.

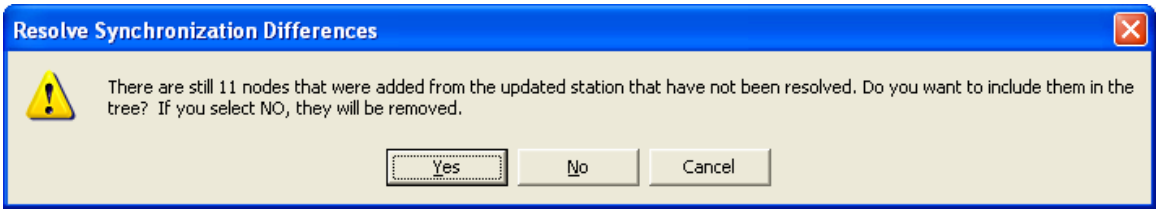
New Logs	You can choose to INCLUDE THE LOG or REMOVE THE LOG from the station tree.
Missing Logs	You can choose to KEEP THE LOG or REMOVE THE LOG
New Containers	You can choose to INCLUDE THE CONTAINER itself, INCLUDE THE CONTAINER AND ALL LOGS within it (including sub-containers), or REMOVE THE CONTAINER (and therefore all sub-containers and logs)
Missing Containers	You can choose to KEEP THE CONTAINER itself, KEEP THE CONTAINER AND ALL LOGS within it (including sub-containers), or REMOVE THE CONTAINER (and therefore all sub-containers and logs)

These options are available by right-clicking on the appropriate node, or by selecting the node and choosing the commands from the LOG or CONTAINER menu.

As long as there are still unresolved nodes, the SYNCH menu will be available on the menu bar. This menu contains commands that let you SHOW ALL added or missing nodes (expanding the appropriate containers to make them all visible). You can also choose to globally accept or reject the changes for all nodes in the tree.



You will need to resolve all the differences before you will be allowed to save to a WPC file. If you attempt to save, you will be prompted to handle each of the two classes of changes:



In general, if you want to update your tree to conform to the updated XML file, you would choose to INCLUDE ALL added nodes and REMOVE ALL missing nodes.

Running an Automated Data Collection

In most cases, you will probably want the data collection to be run on a regular, automated basis. Invensys I/A Series SMART accepts command line arguments that allow you to have it launch the program, open a WPC file, process the data collection, e-mail a log report and then automatically shut itself down. This is particularly useful when specified as part of a batch file or as a scheduled task.

The *Task Scheduler Utility* and *Batch File Utility* can handle all of the details of selecting a .WPC file and setting the appropriate command line arguments. For reference, the format for the command line is:

```
WPUtilities.exe  CollectionInfo.wpc  CollectData
```

The first argument is the name of the Invensys I/A Series SMART executable file. The second argument should be the full path of WPC file that contains the data collection parameters. The third argument is the command to execute, which must the single word "CollectData".

For example:

```
C:\Program Files\Invensys IA Series SMART\WPUtilities.exe
                              "C:\My Scripts\DataCollection.wpc" CollectData
```

Depending on where its used and how your environment variables are set up, you may need to use the full path for the executable file. In all cases you should use the full path of the backup definition file, including drive letters. If the path contains a space anywhere within it, you must enclose the entire path in double quotes.

6. The Report Generator Utility

The *Report Generator Utility* is a simple and easy-to-use Report Engine that provides a powerful mechanism for creating Microsoft Excel-based reports from any database system. It is included specifically to report on data gathered using the *Data Collection Utility*, but it can be used on any type of data from any source.

Types of Reports

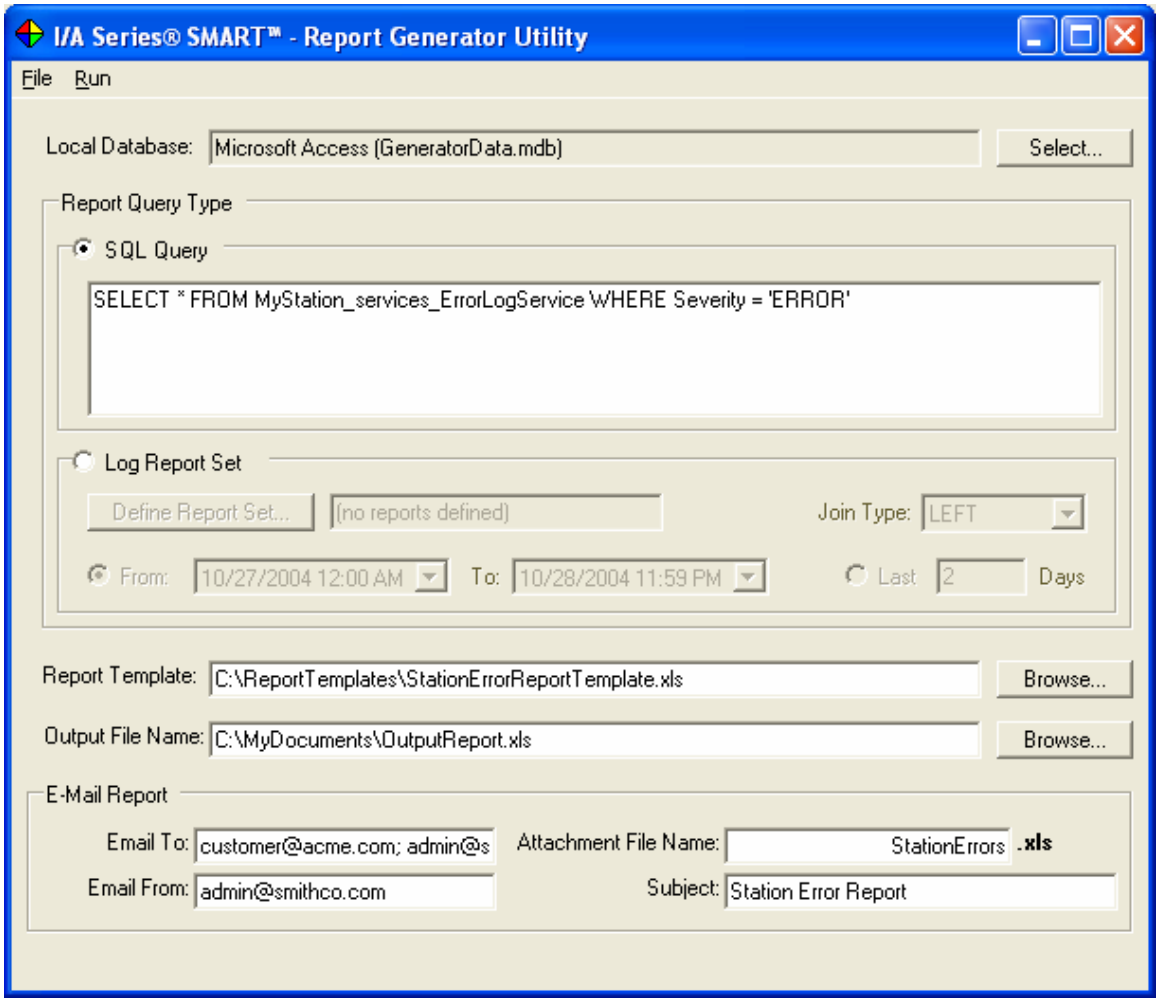
There are two types of reports that can be generated:

SQL QUERY	Generates a report from one or more tables based on a standard SQL Query Statement. Allows the most flexibility in customizing, grouping and sorting report data.
LOG REPORT SET	Generates one or more reports showing data from one or more logs side-by-side. Perfect for generating trend data for commissioning and verification.

While the SQL Query report allows you to create almost any type of report by using any of the features of the SQL language, the Log Report Set is specifically designed to report on collected log data. Similar to the log report features available using a station's Web Services, the Log Report Set allows you to define and save the report definitions by selecting the logs from the station tree. You can then run the report one or more times to update the data for use in checkout and testing procedures.

Defining a Report

A report is defined by specifying the database, the query or report set information, and the name of an Excel file that represents the template for the report. You can also optionally specify where you want the generated report file to be save on your local system, and where you want the file e-mailed.



LOCAL DATABASE — Click SELECT… to select the database to be queried. Refer to the earlier chapter *Using the SMART Utilities* for complete details on how to select from any Windows-compatible database, include Microsoft Access, Microsoft SQL Server and MySql.

REPORT TEMPLATE — Specify or click BROWSE… to select a Microsoft Excel file that contains the template for the report. A report template is just a regular Excel worksheet that contains some layout tags and placeholders for the data rows and values. Template files are described in detail later in this chapter.

Included with the utility is a set of pre-defined report templates for the various types of logs. These are also described later in this chapter

OUTPUT FILE NAME — If you want the resulting Excel file to be saved to a local or network hard drive, specify the name of the file. If the file already exists when the report is run, it will be overwritten automatically. To retain multiple versions of the report, or to store them on an FTP site, use the *Backup and Transport Utility* and select this output file as the source to backup and transport.

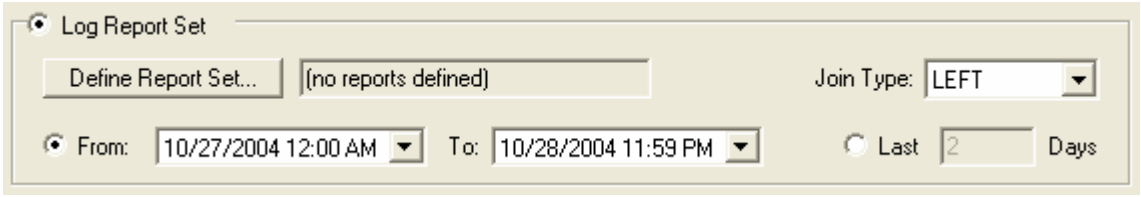
E-MAIL REPORT — If you want the generated report to be e-mailed, specify the name of the ATTACHMENT FILE NAME (the file will be compressed automatically before it is attached), along with the list of e-mail addresses of the recipients, the e-mail address to use as the FROM field, and the text for the e-mail SUBJECT line. For more details on e-mailing files from WorkPlace Pro Utilities, refer to the earlier chapter *Using the Station Monitoring Utilities*.

Defining an SQL Query

To create a report based on any SQL Query, simply select the SQL Query radio button and enter the SQL query statement. Any SQL command supported by the source database can be used. For complex queries, you may want to consider creating a stored procedure in the database itself.

Defining a Log Report Set

There are a few additional parameters to specify when creating a Log Report Set:



FROM / TO — To specify a specific date and time range on which to report, enter the values in the FROM and TO boxes.

LAST N DAYS — To generate a report on the most recent N days, check the appropriate radio button and enter the number of days.

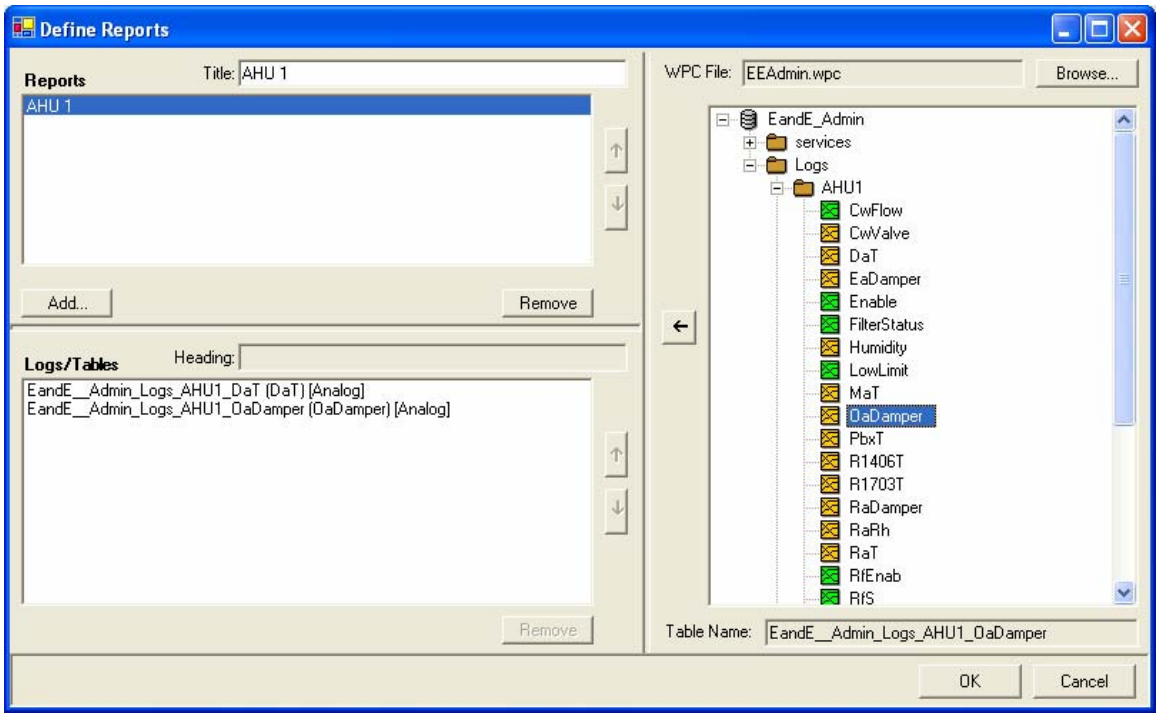
JOIN TYPE — The Log Reports will combine log records from multiple log tables based on their timestamps. The Join Type tells the system what to do in cases where the timestamps don't align exactly.

A LEFT Join will include all the records from the first log, and show empty values for timestamps that don't exist in the subsequent logs.

An INNER Join will only include the records for which there is data in all the specified logs. Records that don't exist in all the logs will be omitted completely.

Defining the Reports

To define the actual reports with the list of log tables, click on the DEFINE REPORT SET… button. This will bring up the Define Reports dialog:



There are three panes in this dialog:

REPORTS — The upper left pane shows the list of reports. To add a report, click on the ADD… button. To rename a report, select it and edit the TITLE in the text box. To reorder the reports, click on the UP and DOWN arrow buttons. To delete a report, click the REMOVE button.

LOGS/TABLES — The bottom left pane shows the list of log tables in the selected report. The name of the table in the local database is shown, followed by the Column Heading (in parentheses) and the Log Data Type (in brackets). To change the Column Heading, select the table and edit the HEADING in the text box. To reorder the log tables in the report, click on the UP and DOWN arrow buttons. To remove a log table, select it and click the REMOVE button.

STATION TREE — The right pane shows a station tree to use to select the tables to include in the report. To view a tree, you must first create a .WPC file using the Data Collection Utility. This allows you to keep information about Alias Names used to rename long names into shorter table names in the local database.

To open a .WPC file, click the Browse… button and select it. To add a table to the current report, simply double-click it, or select it and click the Left arrow button.

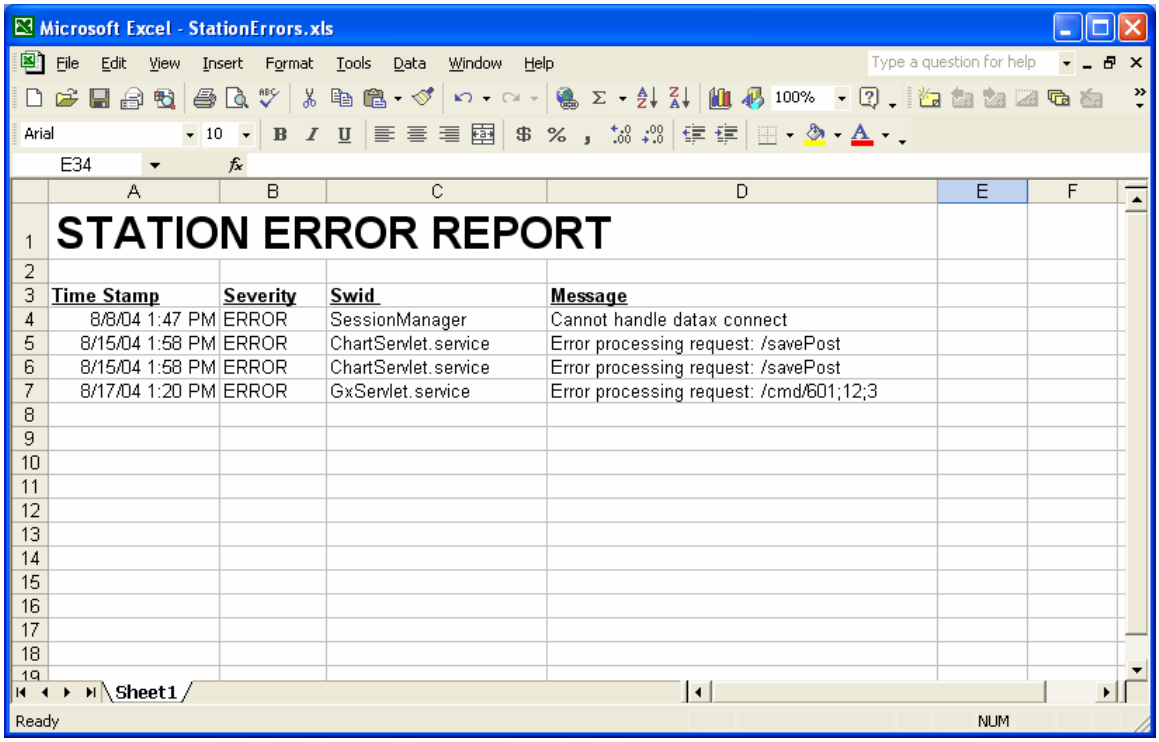
Using the Log Report Set Template

When creating a Log Report Set, you should use the *LogReportSet.xls* template file, or copy it and modify it to suit your needs. Unlike report templates for SQL Queries, this template doesn't actually define the data columns (since they can vary for each report in the set). Instead, there are five sample columns, one for each log data type (Analog, Binary, Integer, String and MultiState). For each generated column of log data, a copy of the appropriate sample column is used as that column's template. The column heading is changed based on the column heading specified in the Report Set.

If you modify this template, be sure that the field codes used for the sample columns are not changed. They must be <AnalogValue>, <BinaryValue>, <IntegerValue>, <StringValue> and <MultiStateValue>.

Running a Report

To run the report, select GENERATE REPORT from the RUN menu.



The generated report will automatically be opened in Excel. If you specified an OUTPUT FILE NAME, the file will be saved. If you specified the E-MAIL REPORT parameters, a copy of the file will be e-mailed.

Loading and Saving the Report Definition (.WPR) File

You can save the configuration of your report task as a file (with a .WPR extension) by selecting SAVE WPR FILE from the FILE menu. You can then reload the settings by selecting OPEN WPR FILE from the FILE menu.

Using the Station Log Data Report Templates

When creating report of data collected from station logs, you can use one of the pre-defined report templates found in the "Reports" subdirectory of the folder where the program was installed ("C:\Program Files\Invensys IA Series SMART" by default). Each of these reports correspond to one of the standard log types (Binary, Analog, Audit, etc.).

You can also use these reports as a starting point, customizing the template to suit your needs. Simply copy the template file and modify the copy. Refer to the later section detailing all of the features of the report templates later in this chapter.

Query and Reporting Techniques

A full discussion of the types of reports you can create and the SQL queries you can build could fill another book. Here are a few techniques you might find useful when working with station log data.

Working with Timestamps

The one field that all logs have in common is the timestamp. This is a date-time value, and it is common to want to select and range or cutoff in your query.

In common SQL, a date can be represented by a string literal in the format:

```
'yyyy-mm-dd hh:mm:ss'
```

For example:

```
'2004-12-25 17:45:00'
```

You can use this in a comparison in your query. For example:

```
… WHERE TStamp > '2004-07-01 00:00:00'
```

When using Microsoft Access, dates are represented in a more common format, with pound signs (#) as delimeters:

```
#mm/dd/yy hh:mm:ss#
```

For example:

```
#12/25/2004 17:45:00#
```

Copying and Pasting Table Names into a Query

When you are building a query to get records from a specific table created by the *Data Collection Utility*, you can copy and paste the table name from the LOCAL TABLE field in the NODE PROPERTIES window of the Data Collection window. Simply select the log in the tree, highlight the local table name, right-click the mouse and choose COPY, and then paste the name into the SQL QUERY window of the *Report Generator Utility*.

Adding Header Fields to a Report

As described in the later section on Report Templates, your report can include a header section that is shown once at the top of the report (or at the top of each page, if so desired). If you are using the same template for multiple reports, you may want to add information to this header section that is specific to the query (such as the Swid of the log whose data is being shown).

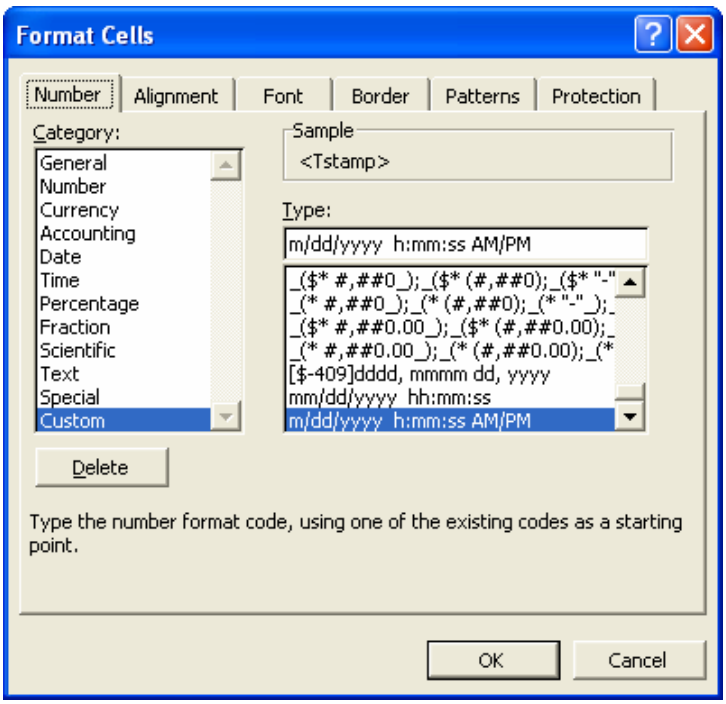
When field codes are used in the header section, the value for the field is taken from the first record generated by the query. The simplest way to get a constant value into the query results is to add it to the select statement:

```
SELECT *, '/AHU1/CvFlow' AS Swid FROM MyStation_AHU1_CvFlow
```

This query will select all the records from the local version of the specified log, but will also add a column (called "Swid") to all the records. For each record the value of this column will be the same string value ("/AHU1/CvFlow"). You can then add a field code tag in the header section ("<Swid>") to display this value.

Formatting Field Values in the Template

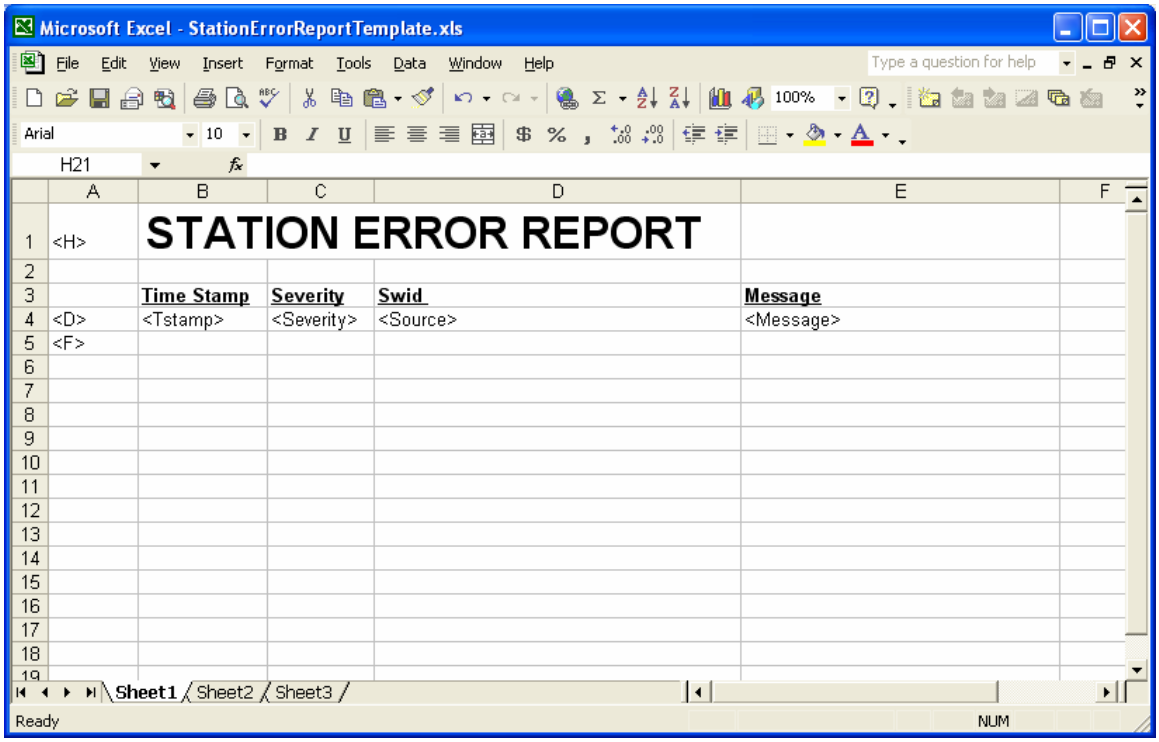
As described in the later section on Report Templates, the actual values generated by the query are inserted into the worksheet in place of the field codes, but all the formatting in the cell remains the same. Therefore, if you want a numeric or date value shown in a specific format, you would set that format on the cell containing the field code placeholder. Select the cell and the select CELLS… from the Excel FORMAT menu.



The custom format string in the picture shows date-time values with both the date and the time (including seconds). Refer to the Excel documentation and help files for more information.

Creating a Report Template File

A Report Template is an Excel worksheet document with field codes added to indicate where data will be inserted when the report is run. It contains a detail section, which is generated once for each output record, and can contain header and footer sections above and below the detail. There are also facilities for section breaks, multiple worksheet reports and aggregate formulas such as totals.



Field Codes

Field Codes are entered as placeholders in cells by enclosing them in angle brackets, such as “<TStamp>” or “<Message>”. The fields that are available will depend on the query used in the report.

When the report is run the field code is removed and replaced with the value for that field. However, all formatting in the cell remains the same. This includes font, style (bold, italic, underline), size, color, and alignment, as well as numeric and date formats. When you design a report, you should put a sample value into the cell, format it appropriately, and then replace it with the field code.

Report Sections

A report will generally consist of three sections: a set of header rows, a single detail row, and a set of footer rows. When the report is run, the header section will appear once, while the detail row will be repeated for each record in the data set. The values for field codes that appear in the header are taken from the first output record (see the section on *Query and Reporting Techniques* for more details).

If not specified explicitly with a Control Column, the detail row is assumed to be the last row in the sheet that contains a field code. All rows above it are assumed to be the header, and all rows below it are assumed to be the footer.

The Control Column

You can explicitly define the sections by including a Control Column. If cell A1 contains an <H> tag, the entire first column is assumed to contain section control information for the report, and will be removed when the report is generated. You can use the following tags:

<H> Begin header section
<D> Begin detail section
<F> Begin footer section

After a section begins, it is assumed to continue until the next different tag or the end of the worksheet.

The <H> tag also accepts certain attributes, which appear after the H but before the closing angle bracket:

<H RepeatOnPage> The entire header is repeated at the top of each page

Grouping

You can break up groups of records into multiple pages by using the following header tag attributes:

<H GroupField=*fieldcode* GroupOn=*groupon*>

In the output report, each time the value of the <*fieldcode*> field changes between two records, a new worksheet tab is created with a separate output report. The value of <*fieldcode*> is used for the name of the worksheet on its tab.

Generally speaking, the output needs to be sorted on the <*fieldcode*> field to group on it.

groupon=Sheets Each group goes on a separate worksheet page

groupon=SamePage The detail and footer sections of each group repeat on the same page (the header is suppressed after the first time)

Adding Filters

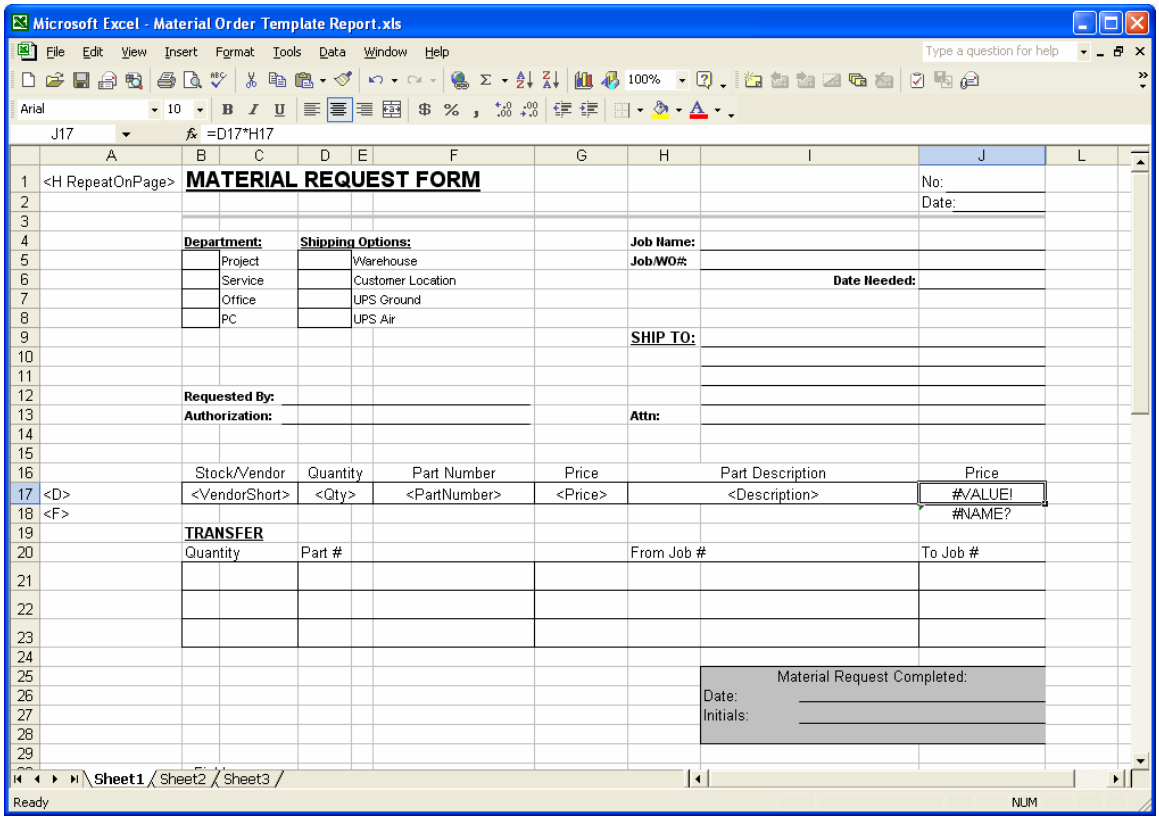
You can have Excel's AutoFilters added to your detail columns by including the following attribute in the detail tag:

<D Filter=Auto>

This will allow you to filter the output rows based on different criteria in each column, using drop down list boxes added to the column headings.

Using Formulas

Use can use any formula that you would normally use in Excel and use cells containing the <*fieldcode*> fields as parameters. The only thing to realize is that since the cells contain text in the template, the result of the formula will often be an error until the report is run.



For example, in the report template above you can see that the detail section (row 17) contains <Qty> and <Price> fields in columns D and H. Column J contains the product of these, with a formula of "`=D17*H17`". Since the text values "<Qty>" and "<Price>" can't be converted to numbers, Excel returns a "#VALUE!" error. However, when the report is run and the field codes are replaced with numbers, the formula will be recalculated and produce the extended price we were looking for.

Aggregate Formulas

To produce totals and other formulas based on the aggregate of multiple rows in the output report, use the following special notation around the cell reference in the formula:

AGG(*cellref*)

When the report is run, the reference to the cell is expanded to include the range of cells for that column in all the output records.

For example, in the sample template, cell J18 in the footer contains the formula "`=SUM(AGG(J17))`". If the report is run with 10 records, they will end up occupying rows 17 to 26 in the output report. The phrase "`AGG(J17)`" in the formula will be expanded to "`J17:J26`", and the resulting formula will be "`=SUM(J17:J26)`", which will properly evaluate to the sum of the values in that column for each row.

It doesn't matter whether the value in J17 was an Excel formula or a field code. Nor are you restricted to the SUM function, as any Excel function will work (e.g., "`=AVG(AGG(J17))`", or even "`=SUM(AGG(J17)) / COUNT(AGG(J17))`". The only restrictions are that the aggregate formula be in the footer section (which requires the use of the Control Column), and that the cell reference be a cell in the detail section.

If the detail section spans more than one row, the aggregate function will attempt to properly build the formula by creating a list of the appropriate cells. For example, if the detail section of a report template occupied rows 30 to 32, and the output report contained 5 records, a reference such as "`=SUM(AGG(F30))`" would be translated into "`=SUM(F30,F33,F36,F39,F42)`", which would evaluate properly. In Excel, however, most functions are limited to 32 parameters, so if there are more than 32 records in the generated report, there is no choice but to translate the cell reference into the entire column (i.e., "`=SUM(F30:F126)`"). This will only be a problem if the detail section contains other numeric values in the same column of its other rows (i.e., cells F31 or F32). As long as those cells are blank or contain only text, the formula will still evaluate properly.

Grand Totals

By adding the *GrandTotal* attribute to the footer, the footer section is repeated one final time, with a Sum() function added for all fields that contained an AGG function in the footer.

```
<F GrandTotal=true>
```

Running an Automated Report

In most cases, you will probably want the report to be run on a regular, automated basis. Invensys I/A Series SMART accepts command line arguments that allow you to have it launch the program, open a WPR file, run the report, save it to a local or network drive, e-mail the report and then automatically shut itself down. This is particularly useful when specified as part of a batch file or as a scheduled task.

The *Task Scheduler Utility* and *Batch File Utility* can handle all of the details of selecting a .WPR file and setting the appropriate command line arguments. For reference, the format for the command line is:

```
WPUtilities.exe  ReportDefinition.wpr  RunReport
```

The first argument is the name of the Invensys I/A Series SMART executable file. The second argument should be the full path of WPR file that contains the report definition. The third argument is the command to execute, which must the single word “RunReport”.

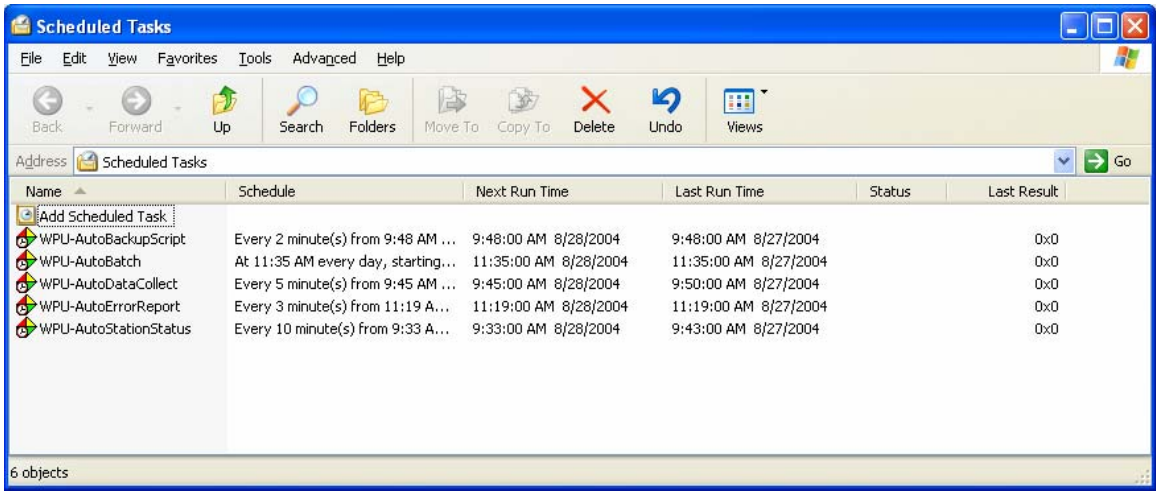
For example:

```
C:\Program Files\Invensys IA Series SMART\WPUtilities.exe
                            “C:\My Scripts\ErrorReport.wpr” RunReport
```

Depending on where its used and how your environment variables are set up, you may need to use the full path for the executable file. In all cases you should use the full path of the report definition file, including drive letters. If the path contains a space anywhere within it, you must enclose the entire path in double quotes.

7. The Task Scheduler Utility

All of the Station Monitoring Utilities are designed to be run on a regular, automated basis. The most efficient method of scheduling these events is to use the often-overlooked Scheduled Tasks facility built into Windows.



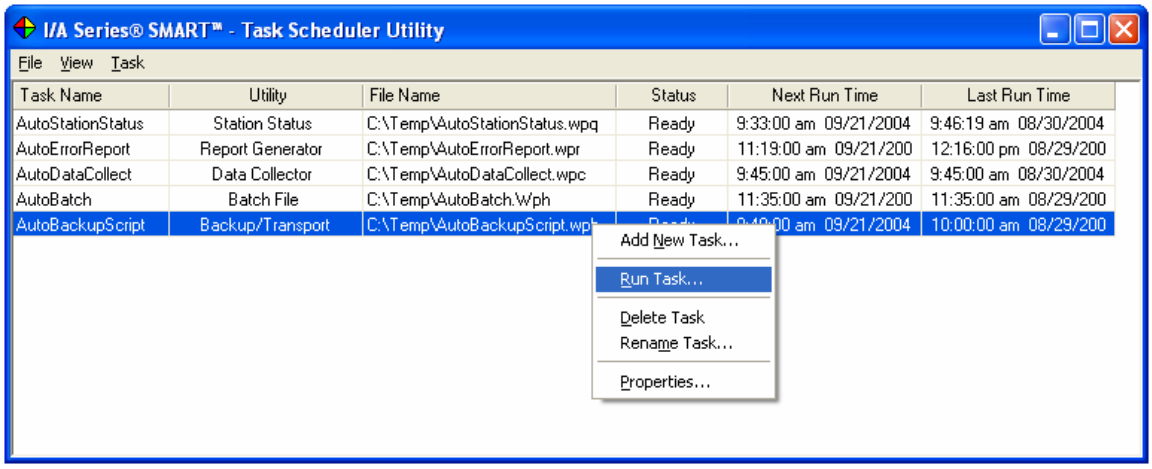
The Scheduled Tasks window, which is normally accessed from the Windows Control Panel, provides a way to set up different *events*, each of which represents a Windows program to run and a schedule of when to run it. The scheduling capabilities are very flexible, and include the ability to run an event on any regular basis, from monthly to once a minute. Since the facility is built into Windows, you do not have to have any programs running in the background to have the events activated at the correct time. You can even specify a User ID and Password for the account to run under, allowing events to be run even if nobody is logged into the workstation.

The only drawback when using the Scheduled Tasks window is that is can be somewhat cumbersome to set up events that require command line arguments to be specified with the program name. All the utilities run in an automated mode by launching the same executable file (*WPUtilities.exe*) and specifying the full name of the .WPx file (such as *"C:\MyReports\StatusReport.wpq"*) and a command (such as *"RunReport"*).

The *Task Scheduler Utility* provides another front-end to the Scheduled Tasks window for the explicit purpose of making it easier to set up events for any of the Station Monitoring Utilities.

The Task Scheduler Utility View

When you open the *Task Scheduler Utility*, all the Scheduled Tasks associated with any of the utilities (and *only* those associated with the utilities) are shown:



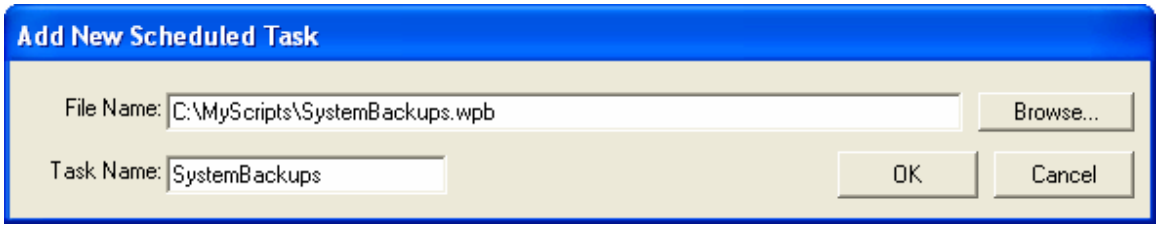
You can do most of the same actions in this window as you can from the Windows Scheduled Tasks Window, such as adding, deleting or running a task, by right-clicking on an item or choosing the command from the TASK menu. You can also access the PROPERTIES window, which allows you to specify the various scheduling and account parameters using the same dialog boxes as in Windows.

The *Task Scheduler Utility's* view differs from the Windows Scheduled Tasks window in a few key elements:

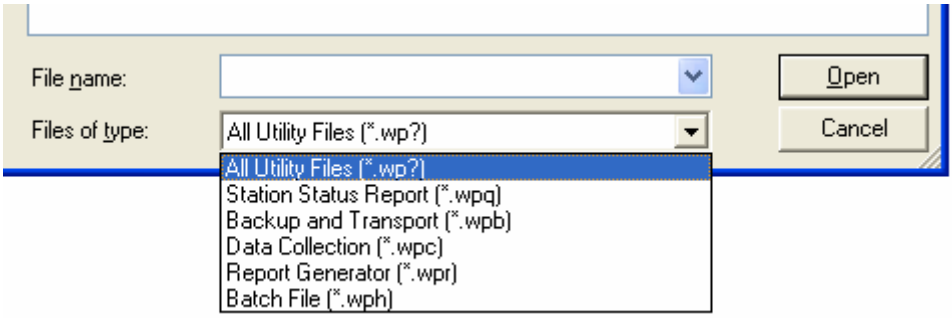
- The TASK NAME, which is used to uniquely identify a task, is actually prefixed with the string "WPU-" in the Windows Scheduled Tasks. This allows the *Task Scheduler Utility* to show only those tasks related to one of the utilities.
- The *Task Scheduler Utility* window shows the name of the UTILITY, along with the FILE NAME that defines the report to run, the data collection or backup task, or the batch file to execute.

Adding a New Scheduled Task

You can create a new scheduled task by selecting ADD NEW TASK… from the TASK menu.

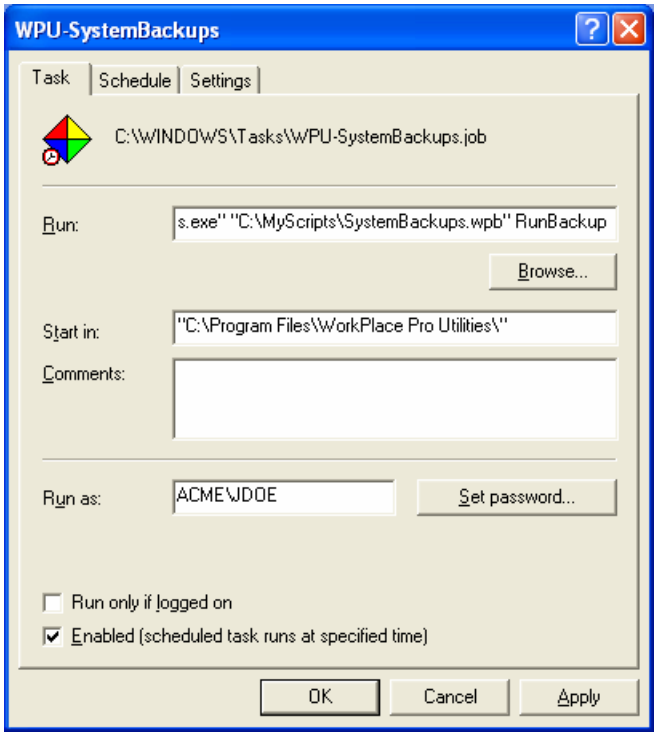


Click the BROWSE… button to select any .WPx file representing the task to be scheduled. The FILES OF TYPE filter at the bottom can be used to make it easier to find a file for a specific utility:



The TASK NAME will default to the name of the file. You can change this, as needed, to ensure that the task names are unique.

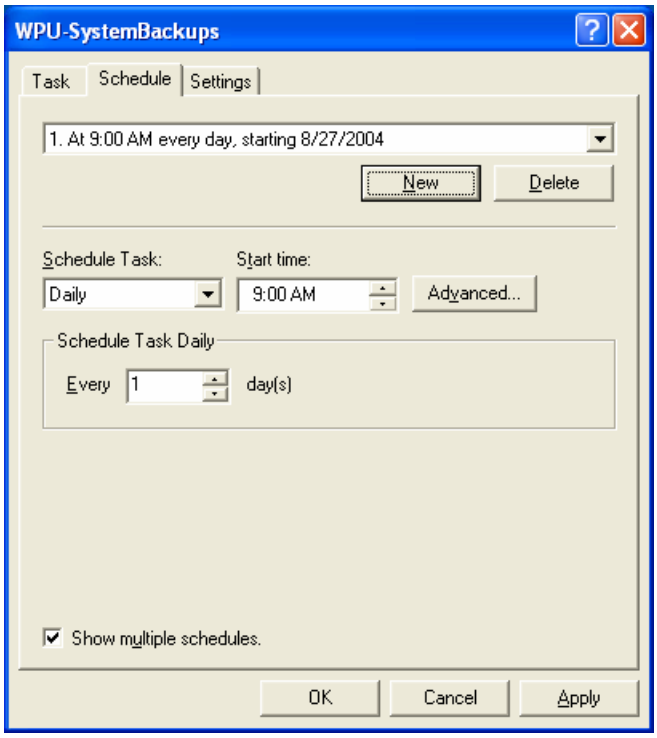
After you click OK, you will automatically be shown the Scheduled Tasks properties window:



You can see that the command line (under RUN) includes the full path to the Invensys I/A Series SMART executable file (*WPUtilities.exe*), along with the full path of the .WPB file (enclosed in quotes) and the appropriate command for the *Backup and Transport Utility*, which is "RunBackup".

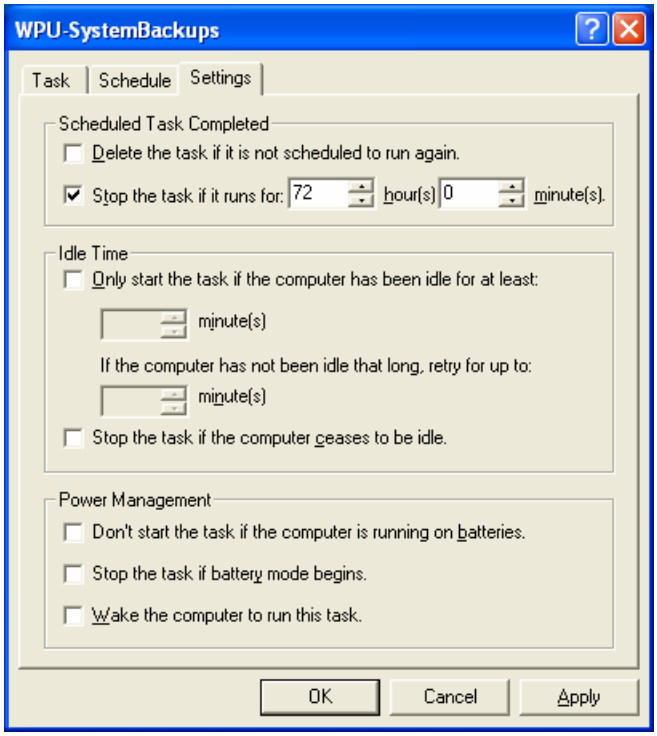
On this TASK tab, you can set the login account information (under RUN AS) and the password (by clicking SET PASSWORD…). On Windows XP, you could instead check the RUN ONLY IF LOGGED ON checkbox and omit the account information. Also note the ENABLED checkbox, which is the easiest way to enable and disable a scheduled task.

If you click on the SCHEDULE tab, you can specify the time and frequency for the task to run. Start by clicking the NEW button to create a schedule entry:



The options for scheduling the event are relatively straight forward. Use the ADVANCED… tab to repeat the task at other than daily intervals. For complete information, refer to the Windows Help File and search for "Scheduled Tasks"

Finally, you can use the SETTINGS tab to specify additional configuration information:



Testing Scheduled Events

You should always test a scheduled event manually before leaving it to run unattended. This will allow you to ferret out any problems, such as incorrect user account and password information (which may cause the event to quietly fail).

To run the task, right-click the entry in the *Task Scheduler Utility* or the Windows Scheduled Tasks window and select RUN. The utility should run, and the LAST RUN DATE should be updated. If there is a problem, you may need to look in the Windows Scheduled Tasks window to see the complete description of the error.

To open the Windows Scheduled Tasks window, select OPEN WINDOWS TASK SCHEDULER from the utility's VIEW menu.

Scheduling Multiple Sequential Tasks

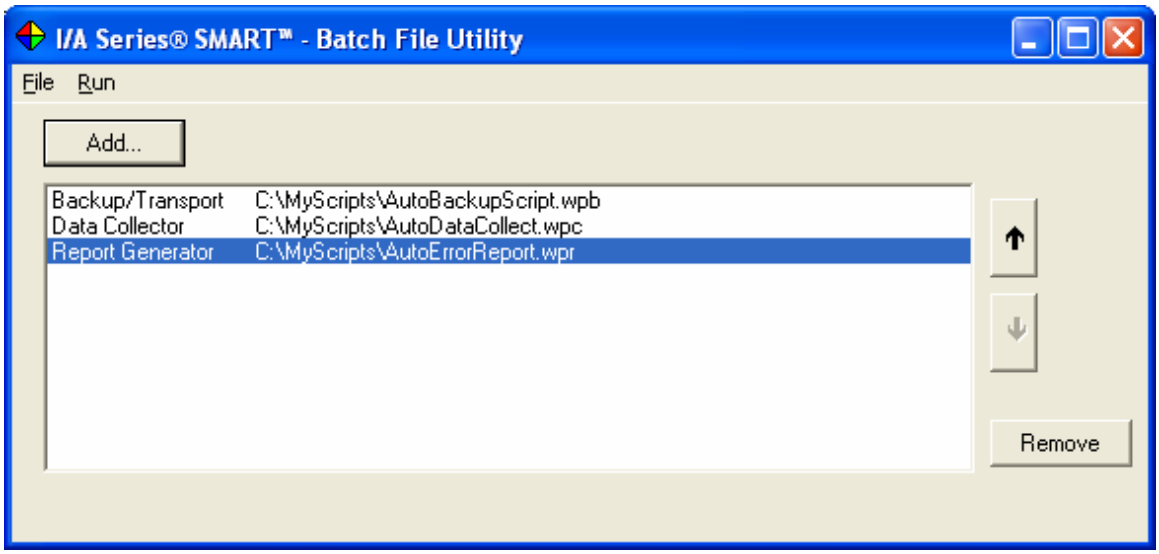
If you need to run multiple tasks in sequence, such as collecting data from a station and then generating a report from that data, you should combine the events into a single batch file using the *Batch File Utility*, and then schedule the Batch File (.WPH File) to run at a specified time. This ensures that each item in the batch file is completed before the next one begins.

8. The Batch File Utility

The SMART Utilities are designed to complement one another, in many cases acting as individual steps of a complete solution. Consider this sample scenario:

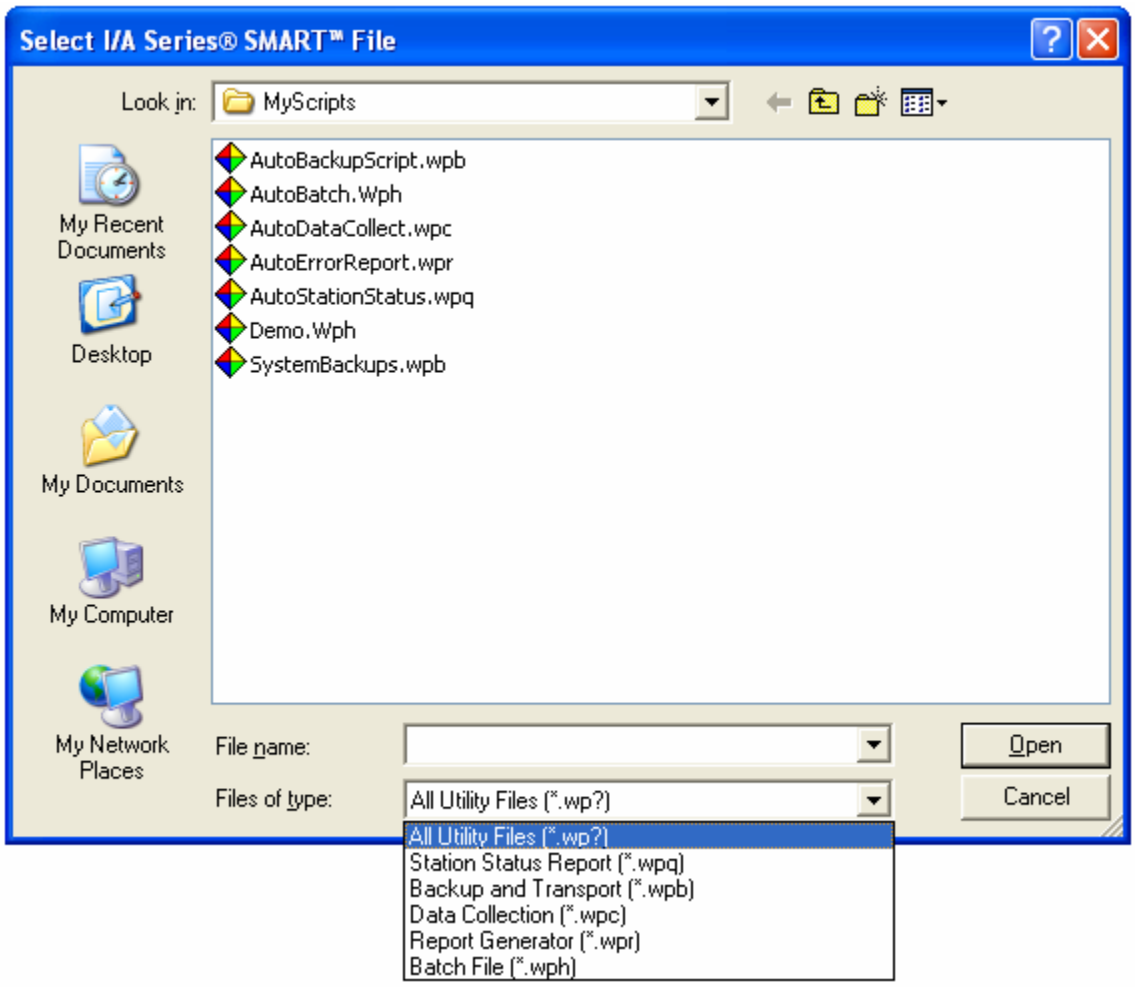
- The *Backup and Transport Utility* downloads an XML backup of a station.
- Next, the *Data Collection Utility* gathers records from the logs in that station backup file and stores them in a local database.
- Then, the *Report Generator Utility* queries the database and creates a report, e-mailing it to the customer and technician.
- Finally, the *Backup and Transport Utility* uploads the report to an FTP server where copies of the latest 90 days worth of reports are maintained.

The *Batch File Utility* allows you to specify a list of ordered tasks to be run in sequence. This can be used to simplify the manual invocation of these tasks, or more importantly, for ensuring the one task be completed before the next begins when the tasks are run at a specific time with the *Task Scheduler Utility*.



Creating a Batch File

Creating a batch file is as simple as selecting the set of Invensys I/A Series SMART (.WPx) files you want to run, and in what order. Click the ADD… button to bring up the File Open dialog:



The FILES OF TYPE filter at the bottom can be used to make it easier to find a file for a specific utility.

After you've added files, you can use the UP and DOWN arrows to reorder the actions, or the REMOVE button to delete one from the batch file.

Loading and Saving the Batch (.WPH) File

You can save the batch file (with a .WPH extension) by selecting SAVE WPH FILE from the FILE menu. You can then reload the settings by selecting OPEN WPH FILE from the FILE menu.

Running a Batch File

To execute each of the actions in the loaded batch file, select RUN BATCH FILE from the RUN menu. For each entry, a new instance of Invensys I/A Series SMART will be launched. The specified file will then be opened automatically, and the action executed. When the action is complete, the instance of Invensys I/A Series SMART will shut itself down, and the next action in the batch file will be executed.

Nesting Batch Files

A batch file can easily contain one or more other batch files to run. When this type of nesting is used, the second batch file will be run as a single action, meaning that all of the actions in the second file will be completed before any other actions in the first file are executed.

Batch1.wph	**Batch2.wph**
a) *BackupTransport1.wpb*	d) *DataCollect1.wpc*
b) *Batch2.wph*	e) *ReportGenerator1.wpr*
c) *BackupTransport2.wpb*	

Consider the above example. When *Batch1.wph* is run, the actions will occur in the following sequence:

a) *BackupTransport1.wpb*
b) *Batch2.wph*
  d) *DataCollect1.wpc*
  e) *ReportGenerator1.wpr*
c) *BackupTransport2.wpb*

There is no limitation on how deep you can nest the batch files. Of course, you should make sure that you don't create any circular references (such as if *Batch2.wph* above contained *Batch1.wph* as an action) or the batch files will keep executing one another until the system runs out of memory.

Running an Automated Batch

In most cases, you will probably want the batch file to be run on a regular, automated basis. Invensys I/A Series SMART accepts command line arguments that allow you to have it launch the program, open a WPH file, execute all the actions in the batch, and then automatically shut itself down.

The *Task Scheduler Utility* can handle all of the details of selecting a .WPH file and setting the appropriate command line arguments. For reference, the format for the command line is:

```
WPUtilities.exe  BatchFile.wph  RunBatch
```

The first argument is the name of the Invensys I/A Series SMART executable file. The second argument should be the full path of WPH batch file. The third argument is the command to execute, which must the single word “RunBatch”.

For example:

```
C:\Program Files\Invensys IA Series SMART\WPUtilities.exe
                              “C:\My Scripts\Batch1.wph” RunBatch
```

Depending on where its used and how your environment variables are set up, you may need to use the full path for the executable file. In all cases you should use the full path of the batch file, including drive letters. If the path contains a space anywhere within it, you must enclose the entire path in double quotes.