

WorkPlace Pro™ Utilities

Version 2.0

User's Guide



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

Sales and Support
(716) 213-2222

<http://www.smartwaretech.com>

All material is Copyright © 2003-2006 Smartware Technologies, Inc. All rights reserved.

WorkPlace Pro is a trademark of Tridium, Incorporated. Microsoft is a registered trademark and Visio is a trademark of Microsoft Corporation. Oracle is a registered trademark of Oracle Corporation and/or its affiliates. MySQL is a registered trademark of MySQL AB in the United States, the European Union and other countries. Cloudscape is a trademark of International Business Machines Corporation. Invensys and I/A Series are registered trademarks of Invensys plc. Designer Suite 2002 is a trademark of Smartware Technologies, Inc.

Table of Contents

PART I: INTRODUCTION

1. GETTING STARTED	9
<i>The Station Building Utilities</i>	<i>10</i>
<i>The Station Monitoring Utilities.....</i>	<i>10</i>
<i>Other Utilities.....</i>	<i>11</i>
System Requirements	11
<i>Microsoft Windows.....</i>	<i>11</i>
<i>Microsoft .NET Framework (Version 1.1).....</i>	<i>12</i>
<i>Microsoft Data Access Components (MDAC) Version 2.6 (or later).....</i>	<i>12</i>
<i>Microsoft Excel.....</i>	<i>12</i>
Installing WorkPlace Pro Utilities	12
License Registration	12
<i>Licensing a New Computer.....</i>	<i>13</i>
<i>Updating a License.....</i>	<i>14</i>
<i>Moving a License to Another Computer.....</i>	<i>14</i>
<i>Licensing a Computer That is Not Connected to the Internet.....</i>	<i>15</i>

PART II: THE STATION BUILDING UTILITIES

2. USING THE STATION BUILDING UTILITIES	17
Working with the WorkPlace Pro XML Station Files	17
<i>Opening a Niagara XML File.....</i>	<i>17</i>
Selecting a Container.....	18
Saving and Loading a WorkPlace Pro Utilities (.WPx) File	18
<i>Launching a WPx File from Windows Explorer.....</i>	<i>19</i>
<i>Using the WPx File as a Template</i>	<i>19</i>
<i>The Default Template File.....</i>	<i>19</i>
Gx Object Property Sets	20
<i>Working with Property Sets.....</i>	<i>20</i>
<i>Editing a Property Set.....</i>	<i>21</i>
3. THE TREE BUILDER UTILITY	23
Importing a WorkPlace Pro File	23
The Node Properties	24
Modifying the Tree Structure	24
Generating the JavaScript for the Tree	26
<i>Using the Security Specific Versions of the JavaScript File.....</i>	<i>26</i>
Saving the Tree Structure	27
Determining Where the Page Will Be Displayed	27
<i>Setting the Default Target for the Entire Tree</i>	<i>28</i>
<i>Setting the Target for a Single Tree Node</i>	<i>28</i>
Merging and Updating Trees.....	28
Updated DemoR2 Files	29
<i>Faster Navigation Trees (tree.js).....</i>	<i>29</i>
<i>Unable to run the Tree on a UNC (LeftFrIE.html).....</i>	<i>30</i>

4.	THE CONTAINER DUPLICATOR UTILITY.....	31
	Selecting the Source Container.....	31
	Selecting a Secondary Source Container.....	34
	Specifying the New Container Names.....	36
	<i>Importing the New Container List From a File</i>	37
	Duplicating the Containers.....	37
	Saving the Files.....	38
	Special Considerations.....	39
	<i>Shadow Objects</i>	39
	<i>Links Outside of Source Containers</i>	39
5.	THE ALARM PAGE BUILDER UTILITY.....	41
	Selecting the Source Containers for Alarm Controls.....	41
	<i>Selecting the Control Types to Include</i>	42
	<i>Selecting the Parent Container</i>	42
	Selecting and Ordering the Alarms on the Page.....	44
	Specifying the Gx Page for the Alarm Screen.....	45
	Setting the Page Layout Properties.....	46
	Setting the Properties of the Gx Objects.....	47
	Previewing the Page Layout.....	49
	Generating the Alarm Pages.....	50
	Loading and Saving the WPA File.....	50
6.	THE VALUE TABLE BUILDER UTILITY.....	51
	Selecting the Source Containers for the Table Rows.....	51
	Selecting the Table Rows.....	53
	Selecting Secondary Source Containers for Table Rows.....	54
	Selecting the Table Columns.....	55
	Creating Multiple Rows Per Container.....	56
	Specifying the Gx Page for the Value Table.....	57
	Setting the Page Layout Properties.....	57
	Previewing the Page Layout.....	58
	Generating the Value Table.....	59
	Loading and Saving the WPV File.....	59
7.	THE PROPERTY SHEETS UTILITY.....	61
	Notes about Microsoft Excel.....	61
	Selecting the Source File and Container to Report.....	61
	Selecting the Object Types to Include.....	62
	Selecting the Properties to Include.....	63
	Filtering on Object Names.....	63
	Creating the Property Sheet Report.....	65
	<i>Special Property Values</i>	66
	<i>Using a Report Template</i>	67
	Editing Property Values with an Edit Report.....	68
	<i>Unprotecting the Edit Report</i>	70
	<i>Checking for Invalid Property Values</i>	70
	Renaming Objects with an Edit Sheet.....	71
	Loading and Saving the WPS File.....	71
	Running a Batch Report.....	72
	Adding to the Module and Object Database.....	72
	<i>Issues with Missing Modules and Objects</i>	73
	<i>Adding a Jar File Map for a New Module</i>	73

8. THE PROFILE UPGRADE UTILITY.....	75
The Upgrade Process.....	75
Using the Utility.....	77
The Upgrade Process.....	78

PART III: THE STATION MONITORING UTILITIES

9. USING THE STATION MONITORING UTILITIES	79
The Station Address List.....	79
Sending Automated E-Mail.....	80
<i>E-Mailing to Multiple Recipients.....</i>	<i>81</i>
<i>Spam Filter Considerations.....</i>	<i>81</i>
Selecting a Local Database.....	82
<i>The Microsoft Database Technologies</i>	<i>83</i>
<i>Using the Database Selection Dialog</i>	<i>83</i>
<i>Connecting to a Microsoft Access Database</i>	<i>85</i>
<i>Connecting to a Microsoft SQL Server Database.....</i>	<i>86</i>
<i>Connecting to a MySQL, Oracle or another OLE DB Database.....</i>	<i>87</i>
<i>Connecting to an ODBC Database.....</i>	<i>88</i>
<i>Customizing the Database Connection String</i>	<i>88</i>
<i>Testing the Database Connection.....</i>	<i>88</i>
10. THE STATION STATUS REPORT UTILITY	89
Defining a Report.....	90
Running a Status Report.....	90
Using the Exception Report.....	91
Loading and Saving a Report Definition (.WPQ) File	92
Running an Automated Report.....	92
11. THE BACKUP AND TRANSPORT UTILITY.....	93
Selecting the Backup Source.....	94
<i>Backing up a Niagara Station.....</i>	<i>94</i>
<i>Backing up a Single File.....</i>	<i>94</i>
<i>Backing up Multiple Files From a Single Folder</i>	<i>94</i>
<i>Backing up Multiple Files From Multiple Folders.....</i>	<i>94</i>
Saving Backups Locally or to a Network Location.....	95
<i>Retaining Multiple Versions of the Backup File</i>	<i>95</i>
Storing Backups on an FTP Site.....	95
E-Mailing Backup Files and Log Reports	95
<i>E-Mailing the Backup File.....</i>	<i>96</i>
<i>E-Mailing the Log Report.....</i>	<i>96</i>
Loading and Saving a Backup Definition (.WPB) File	96
Running an Automated Backup.....	96

12. THE DATA COLLECTION UTILITY.....	99
Loading a Station File	100
<i>The Node Properties</i>	100
<i>Using Alias Names to Shorten Table Names</i>	101
Selecting a Local Database.....	102
<i>Creating and Using a Microsoft Access Database</i>	102
Selecting the Data Collection Source	103
<i>Collecting from an Enterprise Server</i>	103
<i>Collecting from a Station Backup (XML) File</i>	103
<i>How Data Records are Collected</i>	104
<i>Retaining Records in the Local Database</i>	104
Selecting the Logs to Collect From	105
Notification Tables (Alarms and Alerts)	106
Collecting Data.....	107
Loading and Saving the Data Collection (.WPC) File.....	108
Working with Individual Logs	108
<i>Checking for Existing Tables in the Local Database</i>	108
<i>Manipulating the Local Database Table</i>	109
Synchronizing to an Updated Station Tree.....	110
Running an Automated Data Collection.....	112
13. THE REPORT GENERATOR UTILITY.....	115
Types of Reports	115
Defining a Report	116
Defining an SQL Query.....	117
Defining a Log Report Set.....	117
<i>Defining the Reports</i>	118
<i>Using the Log Report Set Template</i>	119
Running a Report.....	120
Loading and Saving the Report Definition (.WPR) File.....	120
Using the Station Log Data Report Templates	120
Query and Reporting Techniques.....	121
<i>Working with Timestamps</i>	121
<i>Copying and Pasting Table Names into a Query</i>	121
<i>Adding Header Fields to a Report</i>	122
<i>Formatting Field Values in the Template</i>	122
Creating a Report Template File	123
<i>Field Codes</i>	123
<i>Report Sections</i>	124
<i>The Control Column</i>	124
<i>Grouping</i>	124
<i>Adding Filters</i>	125
<i>Using Formulas</i>	125
<i>Aggregate Formulas</i>	126
<i>Grand Totals</i>	126
Running an Automated Report	127
14. THE TASK SCHEDULER UTILITY	129
The Task Scheduler Utility View	130
Adding a New Scheduled Task	130
Testing Scheduled Events.....	133
Scheduling Multiple Sequential Tasks	133

15. THE BATCH FILE UTILITY.....	135
Creating a Batch File.....	136
Loading and Saving the Batch (.WPH) File.....	136
Running a Batch File.....	137
Nesting Batch Files.....	137
Running an Automated Batch.....	137

PART IV: OTHER UTILITIES

16. THE WORKPLACE TECH 4.0 SNVT UTILITIES	139
Installing the Smartware Utilities Stencil.....	139
Generating an LNC Data String.....	139
Generating a Snvt Table.....	142
17. THE CUSTOMER NAVIGATION WEB SITE	143
Installing the Web Site.....	143
Adding Your Customer Sites to the Navigation Tree.....	144
Customizing the Web Page.....	145

1. Getting Started

WorkPlace Pro Utilities is a stand-alone application that works with Niagara stations to provide new and additional functionality during all phases of development and maintenance.



There are three categories of utilities:

Station Building Utilities Used to help speed the development of Niagara stations by automating redundant tasks and automatically creating Gx Pages and trees.

Station Monitoring Utilities Used to monitor and maintain running stations by generating reports and collecting data.

Other Utilities Other utilities, such as a Snvt Generator for WorkPlace Tech 4.0.

The Station Building Utilities

The station building utilities are:

Tree Builder Used to automate the creation of the JavaScript file required to build navigation trees for front end applications, this utility will import the tree structure from an existing Niagara application (or start from scratch), and then easily move, copy, rename, restructure and relink the nodes to underlying Gx Pages or internet IP addresses.

Container Duplicator Useful in situations containing containers, logic bundles or Gx Pages that are typical of many systems, this utility duplicates the containers, renaming and relinking them appropriately.

Alarm Page Builder Used to create a set of master Alarm Pages for a system, this utility compiles the alarm inputs from one or more containers or logic bundles and populates a Gx Page with boolean objects, captions and device headings.

Value Table Builder Used to automatically build a multi-page table of values, where the rows of the table represent containers and the columns represent composite inputs and outputs from the containers.

Property Sheets Used to view and edit a set of property values for a set of objects within a container or station.

Profile Upgrade Used to upgrade the shadow objects of controllers being replaced with newer profile versions.

The Station Monitoring Utilities

The station monitoring 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.

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

Other Utilities

The other utilities include:

WorkPlace Tech Snvt Used with WorkPlace Tech 4.0 drawings to generate LNC Data Strings and tables of Snvt information.

In addition, WorkPlace Pro Utilities also includes a pre-built web site:

Customer Navigation Web Site: A complete web-site that can be used as a central launch point for all of your customer's Niagara sites. This page can then be used by your engineers and technicians to easily access your customer's sites without remembering their internet addresses. The *Tree Builder Utility* is used to easily customize the navigation tree without any web-design experience necessary.

System Requirements

WorkPlace Pro Utilities 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

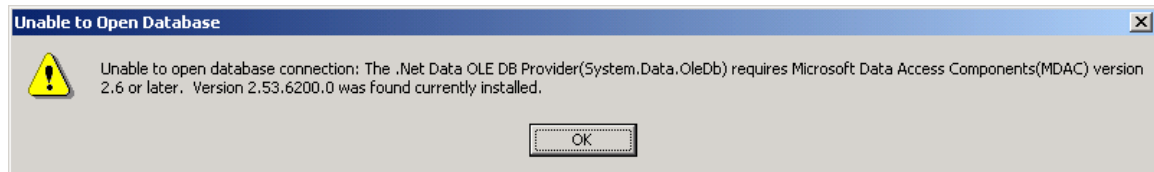
Microsoft Windows XP is recommended. Microsoft Windows 2000 is supported. Microsoft Windows 98 is not officially supported, though it has been tested and there are no specific incompatibilities known.

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 *Property Sheets Utility* or 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 WorkPlace Pro Utilities

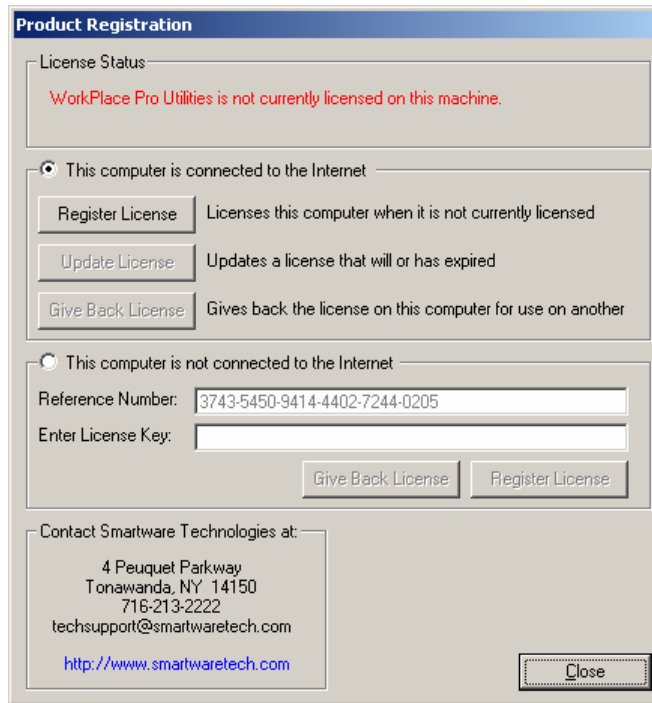
To install WorkPlace Pro Utilities, simply download the latest installation program from our web site and run it. If you are upgrading from an earlier version of the program, you should choose to install over the previous version into the same directory. You do not need to uninstall the previous version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

License Registration

Each copy of WorkPlace Pro Utilities 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:



The image shows a 'Product Registration' dialog box with a blue title bar. It contains several sections: a 'License Status' section with a red error message; two radio button options for internet connectivity; a section with three buttons ('Register License', 'Update License', 'Give Back License') and their descriptions; a section for entering a 'Reference Number' and 'License Key'; and a 'Contact Smartware Technologies at:' section with address, phone, email, and website information. A 'Close' button is located in the bottom right corner.

Product Registration

License Status

WorkPlace Pro Utilities is not currently licensed on this machine.

This computer is connected to the Internet

This computer is not connected to the Internet

Register License Licenses this computer when it is not currently licensed

Update License Updates a license that will or has expired

Give Back License Gives back the license on this computer for use on another

Reference Number: 3743-5450-9414-4402-7244-0205

Enter License Key:

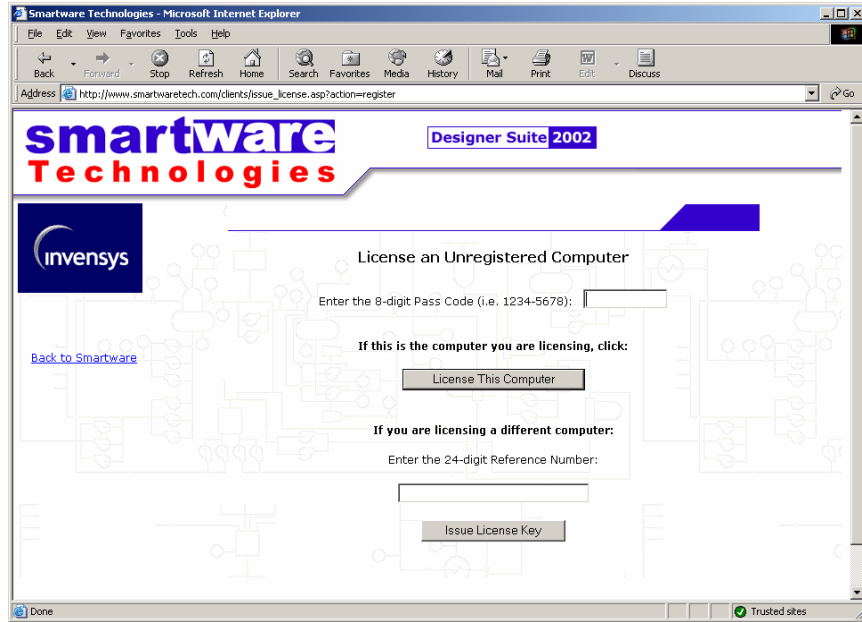
Give Back License Register License

Contact Smartware Technologies at:

4 Pequet Parkway
Tonawanda, NY 14150
716-213-2222
techsupport@smartwaretech.com
<http://www.smartwaretech.com>

Close

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 WorkPlace Pro Utilities and close the License Registration dialog, the main screen should show 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 Station Building Utilities

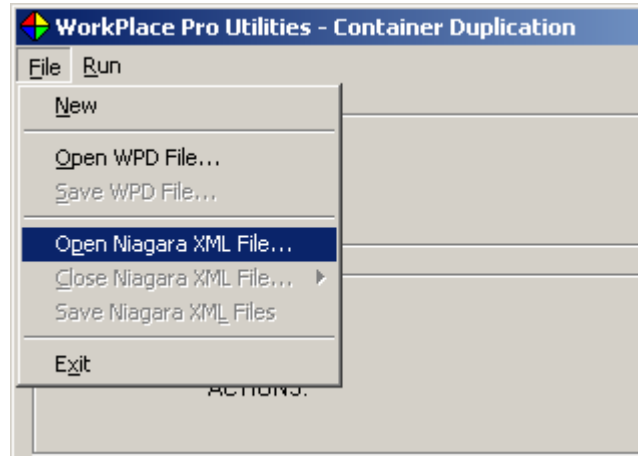
The rest of this manual describes the functionality and use of the individual utilities in depth. This chapter introduces a few topics that are common to several of the utilities. The later chapters will refer back to these sections.

Working with the WorkPlace Pro XML Station Files

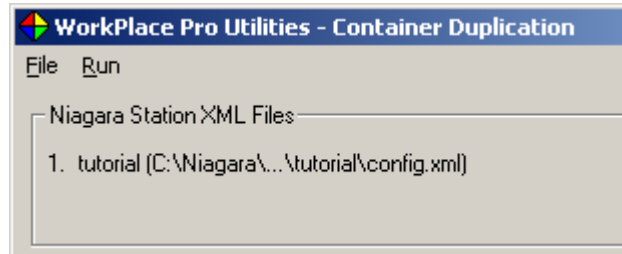
Most of the utilities are designed to read, modify and resave the entire station database in the XML format. You can export a station from Niagara to an XML file (generally named CONFIG.XML) and reimport it back into Niagara using the Admin Tool (on the DBADMIN tab).

Opening a Niagara XML File

Depending on the utility, you may be working on a single file or on multiple files. In each case you begin by loading the XML file into the utility by selecting OPEN NIAGARA XML FILE from the utility's FILE menu:



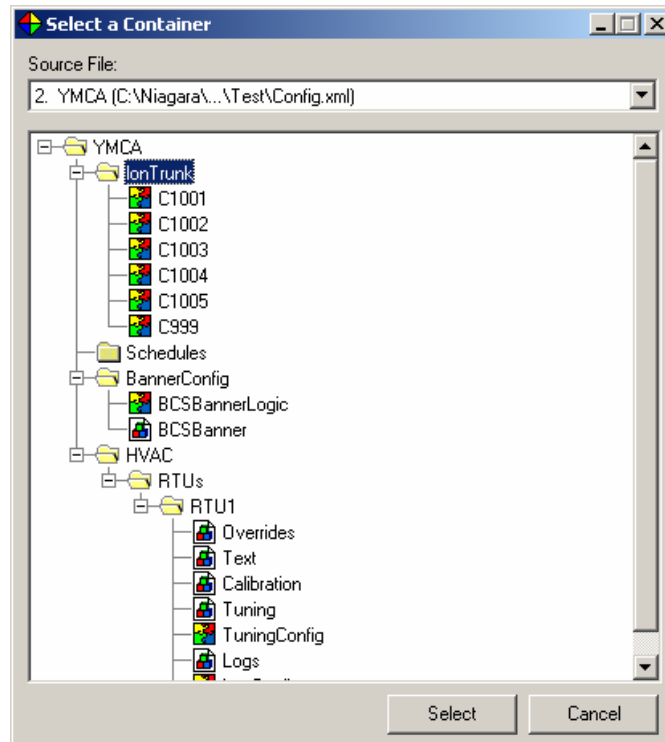
After the file has been loaded, it will appear in the list of open files at the top of the utility's window:



You can close and save the XML files by selecting the appropriate commands on the FILE menu as well. When you save an XML file, a backup copy with a "-1" (etc.) will automatically be saved in the same directory.

Selecting a Container

Many of the utilities work on generic container objects (which include Logic Bundles, Gx Pages and Containers), or need you to specify a set of objects by selecting their container. In these cases, a common Container Selection dialog will be shown:



The drop-down list box at the top of the form lets you choose from any of the open Niagara files. The tree structure in the middle represents the various nodes in the Niagara station. To select a container, simply click on it and press the SELECT button. Depending on the utility, there are times when only certain types of containers can be selected (as when selecting a Gx Page to use as the template for generated Alarm Pages).

Saving and Loading a WorkPlace Pro Utilities (.WPx) File

Aside from manipulating the Niagara station XML files, each one of the utilities allows you to save its configuration parameters to a file for later use. This file contains a list of the open Niagara files, the selected containers, page layout values, property sets, and other utility-specific information (i.e., for the Container Duplicator, the list of new containers to created; for the Value Table Builder, the table rows and column information). It is a good idea to save your work in these files in case you need to rerun the utility on an updated version of the station.

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

.WPD	Container Duplicator
.WPT	Tree Builder
.WPA	Alarm Page Builder
.WPV	Value Table Builder
.WPS	Property Sheets
.WPP	Property Sets Export
.WPJ	Jar File Maps
.WPQ	Station Query
.WPC	Data Collector
.WPB	Backup and Transport
.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 WorkPlace Pro Utilities 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 WorkPlace Pro Utilities automatically run, launch the correct utility and load the appropriate WPx and XML files.

Using the WPx File as a Template

In the case of the page building utilities, these configuration files can also be used as templates. You can start with an empty file, change the page layout and property set information, and then save the file with just this information. When you want to use this template information, simply load the file before loading the XML files and specifying station-specific information.

The Default Template File

Some utilities, such as the Alarm Page Builder and the Value Table Builder, have a default configuration file containing parameters and properties that are loaded automatically. These files (i.e. DEFAULT.WPA and DEFAULT.WPV) are stored in the installation directory for the utility. While you can modify these files, it is strongly recommended that you instead copy them and modify your own version as a template. If you install a new version of WorkPlace Pro Utilities, your modified versions of these default files may get overwritten.

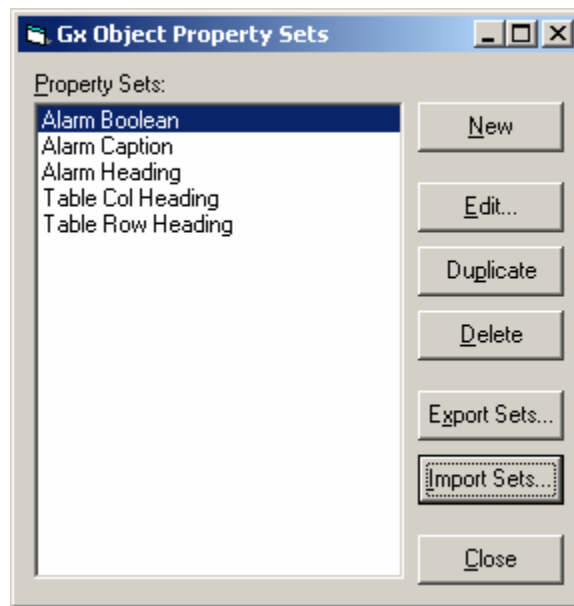
Gx Object Property Sets

The utilities that generate objects on a Gx Page (i.e. the Alarm Page Builder and the Value Table Builder) allow you to specify additional Niagara properties for the graphics objects they create. By including these properties, you can have complete control over the font, style, color, border, images and other visual characteristics of the generated page. You can also use them to set other non-visual properties such as display units or hyperlinks.

Working with Property Sets

To make it easier to reuse sets of properties, the utilities allow you to name these sets, duplicate and edit them, import and export them, and otherwise manipulate them in a manner similar to styles in a word processor.

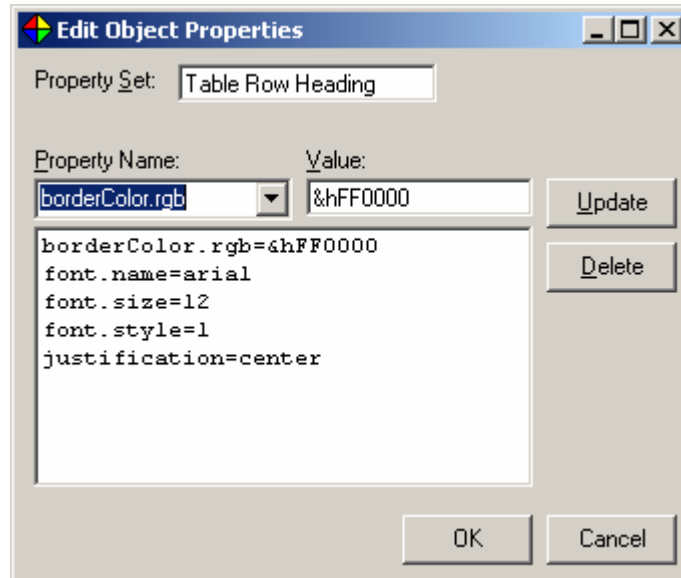
To access the list of Property Sets, select GX OBJECT PROPERTY SETS... from the EDIT menu.



From here you can create, duplicate, delete, import and export the property sets. To rename or modify the actual properties in a set, click the EDIT... button.

Editing a Property Set

The Edit Object Properties dialog allows you to specify the name of the property set, along with the list of properties.



The PROPERTY NAME list box contains a few of the more common properties whose values can be set, but you can enter any property name in there as well. The names and appropriate values of the properties are defined by WorkPlace Pro and used in their XML format. In the above example, which represents the heading properties, the text object will be created as centered, 12 point bold (font.style = 1) text with a red border.

Tip: *Although numeric values are expected to be entered as integers, you can use the "&h" prefix to denote a hexadecimal value. This is particularly useful when specifying RGB values for colors (i.e., &hFF0000 is Red).*

3. The Tree Builder Utility

The Tree Builder Utility is used to design and build a tree structure of nodes to be displayed as a type of menu on an HTML page for use within a controller. Each node in the tree can represent a link to a Gx Page or a link to another URL on the local network or internet.

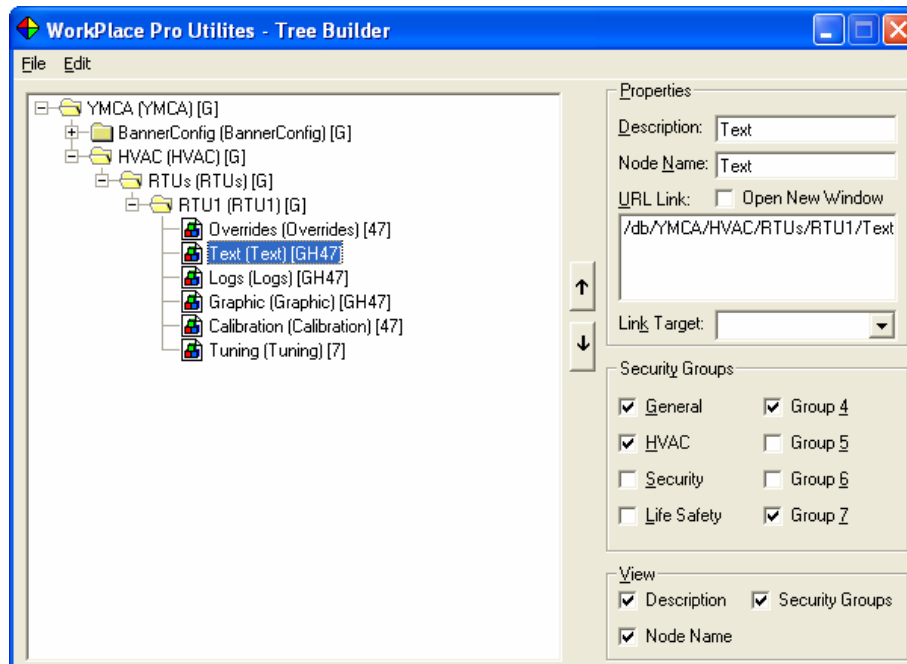
A tree can be generated from scratch or a WorkPlace Pro file can be imported as a starting point. The Tree Builder Utility does *not* make any changes to the WorkPlace Pro XML file.

The Tree Builder Utility works with the Navigation system distributed by Tridium as part of WorkPlace Pro. You can find more information about the system and how to integrate it into your applications in the *Niagara Web Solutions Guide (Chapter 5, DemoR2 Navigation)*

We have also created updates to two of the files found in the DemoR2 station. One resolves problems related to large trees and adds additional functionality (*tree.js*) and another fixes a problem that occurs when running the application on certain UNC devices (*LeftFrIE.html*). Refer to the *Updated DemoR2 Files* section later in the chapter.

Importing a WorkPlace Pro File

To begin with, select IMPORT XML FILE... from the FILE menu and select the appropriate file exported from the WorkPlace Pro Admin Tool.



The Node Properties

The Tree Builder will locate all the Gx Pages in the system and build the folder structure accordingly. Each node (container or Gx Page) has the following properties:

- DESCRIPTION: The text that will be displayed on the tree.
- NODE NAME: The corresponding node in the WorkPlace Pro file (shown in parentheses). If the link field for a Gx Page is blank, these names will be used to create the appropriate link for the page (i.e., the Text Gx Page highlighted above will be created in the tree with the link:
`/db/Root/HVAC/RTUs/RTU1/Text`
- URL LINK: The URL for nodes that do not represent Gx Pages, or the full path of a link that does not correspond to the page's position in the tree. For external links you should include the appropriate <http://> prefix.
- NEW WINDOW: A checkbox that indicates whether the link will cause a new browser window to be opened when it is selected in the web page. Refer to the later section on *Determining Where the Page Will Be Displayed* for more information.
- LINK TARGET: The HTML target for the link. Determines which frame or window the linked page will be loaded in. Possible choices include *_blank*, *_self*, *_parent* or the name of any frame in the enclosing frameset. If left blank, the default value (which can be specified by selecting OPTIONS from the EDIT menu) will be used. Refer to the later section on *Determining Where the Page Will Be Displayed* for more information.
- SECURITY GROUPS: Flags that indicate which of the eight security groups the node belongs to. You will have the option to create different versions of the tree based on these security flags.

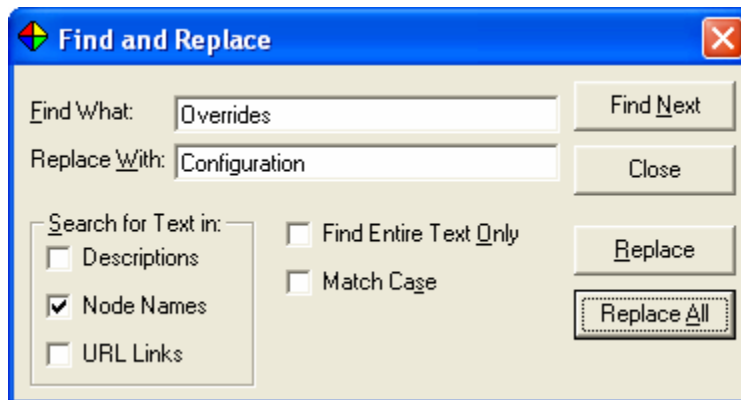
You can change the properties of any node by selecting it in the tree and changing the values in the text boxes and check boxes on the right.

You can change the display to show only the description or the node name, and to turn the security flags on and off, with the check boxes on the lower right of the form.

Modifying the Tree Structure

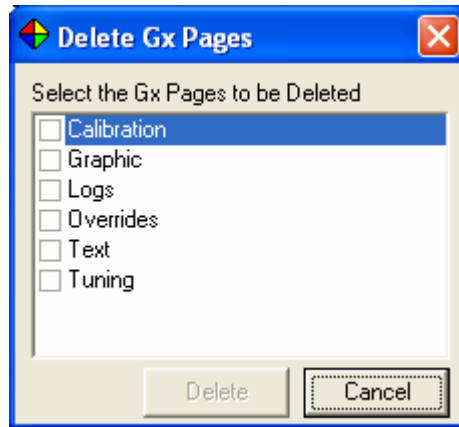
You can adjust the structure of the nodes and folders in a large variety of ways. Most of these options are available by right clicking the mouse on the appropriate node in the tree to bring up a pop-up menu.

- Adding a Node:** Right click on the folder into which the node should be added and select **NEW ITEM**. The new node will be added after any existing nodes in that folder. If the selected node is not a folder, it will be converted to one.
- Copying a Node:** Right click on the folder or Gx Page and select **COPY**. Then right click on the folder into which the copy should be added and select **PASTE**. The node and all the nodes beneath it will be copied.
- Moving a Node:** Right click on the folder or Gx Page and select **CUT**. Then right click on the folder into which the copy should be added and select **PASTE**. The node and all the nodes beneath it will be moved.
- Deleting a Node:** Right click on the folder or Gx Page and select **DELETE**. The folder and all the nodes beneath it will be deleted. This action cannot be undone.
- Reordering Nodes:** To change the order of the nodes within a folder, select one of the nodes and click the up or down arrow.
- Find/Replace Text:** To search for and/or replace any or all of the description or name text, select **FIND/REPLACE** from the **EDIT** menu. This will bring up a standard Find and Replace dialog box:



You have the option of searching for entire matches only or for pieces of text within the descriptions and node names. You can also choose for the search to be case sensitive, and to search and replace within the descriptions, node names or the links.

Delete Gx Pages: To remove all the Gx Pages with the same name, select **DELETE GX PAGES** from the **EDIT** menu. This will bring up a list of all the Gx Pages in the tree:



Select the pages you wish to remove and press **DELETE**. All Gx Pages with the indicated names will be removed.

Generating the JavaScript for the Tree

To create the JavaScript file for the tree, select **GENERATE JAVASCRIPT** from the **FILE** menu. You will be prompted for the name of the file to generate, which is usually *treeloader.js*. This file will contain all the nodes in the tree, regardless of the security flags.

You will then be given the option of generating security-specific versions of the files. In these versions, only nodes which have a certain security flag set, and for which all the folders above it have the same flag set, will be contained in the tree. Each security flag in use will have its own file, named with the security flag (i.e., if the file to generate is *treeloader.js*, the version for the HVAC security will be *treeloader-H.js*).

Using the Security Specific Versions of the JavaScript File

Although the Tree Builder Utility will create multiple versions of the JavaScript file, the standard versions of the HTML page that display the tree (such as Tridium's *SiteFrIE.html*) have no features to make use of the extra versions. It is up to the developer to create the extended security structure that allows the user to log in, determines which security groups they have permission to access, and then uses the appropriate JavaScript file to display the correct tree.

Saving the Tree Structure

To save the structure of the tree including all the properties, flags and settings for later editing, select SAVE WPT FILE from the FILE menu. To reload the tree at a later time, select OPEN WPT FILE from the FILE menu.

Determining Where the Page Will Be Displayed

Each node in the tree becomes a link in the web browser when its displayed, loading a GxPage, web site or other document depending on the value of the URL Link property for that node. The location in the user's browser window where the page will be displayed is referred to as the *link target*, because it corresponds to the *target* property in the generated HTML.

By default, the target of the links is "Main", which corresponds to the name of the frame in many versions of the *SiteTreeIE.html* file commonly used to display the tree:

```
<HTML>
  <HEAD>
    <TITLE>Niagara Demo Using Tree Navigation</TITLE>
  </HEAD>

  <FRAMESET ROWS="46,* ,25" frameborder="NO" framespacing="0" >
    <FRAME SRC="tabtop2.htm" scrolling="no" resize="no" name="tabtop">
    <FRAMESET COLS="145,*" frameborder="NO" framespacing="0" >
      <FRAME SRC="LeftFrIE.html"
        scrolling="auto" resize="no" name="tabnav">
      <FRAME SRC="/db/demoR2/Sim/Displays/Home"
        scrolling="auto" resize="yes" name="Main">
    </FRAMESET>
    <FRAME SRC="bottom1.htm" scrolling="no" resize="no" name="bottom">
  </FRAMESET>

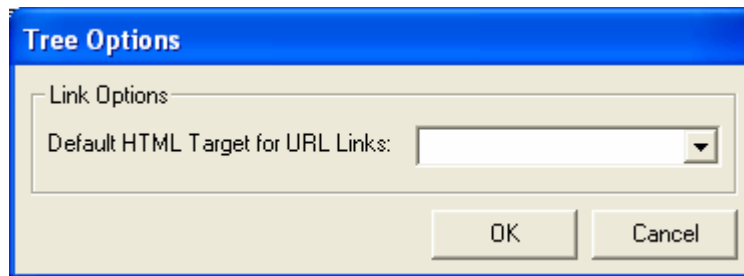
</HTML>
```

You can change this default to match your HTML framesets for the entire tree, or for each individual link on a node-by-node basis.

Note: *If you specify any non-default values for the target, you must use the updated version of the tree.js file found in your C:\Program Files\WorkPlace Pro Utilities\TreeLoader directory.*

Setting the Default Target for the Entire Tree

To set a different name for the target frame, select **OPTIONS** from the **EDIT** menu:



The listbox contains a number of the special case values defined in HTML, such as:

- _blank* Open the page in a new browser window
- _self* Open the page in the frame containing the tree
- _top* Open the page to fill up the entire window include all frames
- _parent* Similar to *_top*, but in cases with nested framesets, open the page to fill up the parent of the frame containing the tree.

You can also type your own value into the combo box, such as “viewFrame”. This target value will now be used for all nodes that do not have a link target value specified explicitly.

Setting the Target for a Single Tree Node

To specify a different target for one or more links, select or enter the value in the **LINK TARGET** field of the node properties. You can also click the **OPEN NEW WINDOW** checkbox, which is the equivalent to setting the link target to *_blank*.

Merging and Updating Trees

There are times when you may need to merge the contents of two different trees. One common case is when a tree is generated from a station, and subsequently additions are made to the station that need to be reflected in the tree.

The Tree Builder handles these needs by allowing you to append an existing tree file into the current one. To do this, select **APPEND FROM WPT FILE...** from the **FILE** menu. The entire tree from the selected file will be added as a new node on the root of the current tree. You can then copy and paste selected nodes or branches of this appended tree into the existing one and delete the rest.

In the specific case of adding new nodes from a station file into an existing tree (saved, for example, as *MyTree.wpt*), you should:

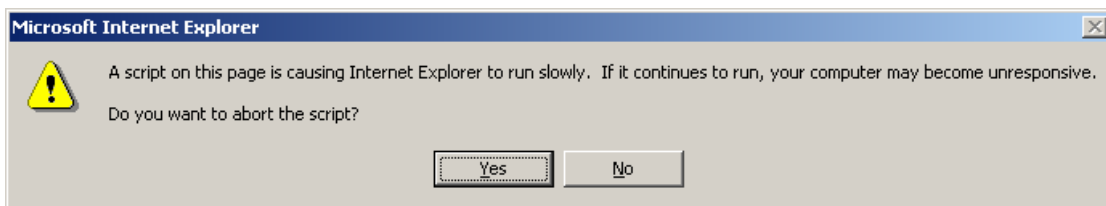
- Run the Tree Builder and import the updated XML file for the station into a new tree by selecting IMPORT XML FILE from the FILE menu.
- Save this tree to a new WPT file by selecting SAVE WPT FILE from the FILE menu. Specify a new file name, such as *Temp.wpt*.
- Open the existing tree by selecting OPEN WPT FILE from the FILE menu and selecting its file (in our example, *MyTree.wpt*)
- Append the new tree to the existing one by selecting APPEND FROM WPT FILE from the FILE menu and selecting its file (in our example, *Temp.wpt*).
- Copy and paste the added nodes from the appended copy of the tree to the appropriate place on the existing tree.
- Delete the appended copy of the tree.

Updated DemoR2 Files

To resolve two known issues relating to the navigation tree system distributed with the DemoR2 station, we have included updated versions of two of the files in that system.

Faster Navigation Trees (tree.js)

As the tree structure becomes larger, the time it takes for the web page containing it to load in the browser will become excessive. Users may also receive a warning from Internet Explorer indicating that the script might hang their computer and suggesting that they abort it:



We have resolved this problem by creating an updated version of the standard *tree.js* file distributed as part of the *DemoR2* station. This updated version can be found in the C:\Program Files\WorkPlace Pro Utilities\TreeLoader directory. Simply copy this version of *tree.js* to your station's local library.

This new version also replaces the need for the *Script Timeout Adjustment Pages* shipped with previous versions of WorkPlace Pro Utilities.

The new version is also required if you want to specify different targets for the various links in the tree.

Unable to run the Tree on a UNC (LeftFrIE.html)

We have been able to isolate and reproduce a problem in the DemoR2 station that prevents the system from running on certain UNC devices, including (but not necessarily limited to) the UNC 510-21. In the *LeftFrIE.html* file there is a reference to the "treeLoader.js" file. The 'L' in "treeLoader" needs to be a lowercase letter. To resolve the problem, simply make this change in your version of *LeftFrIE.html* (if you customized it) or use our updated version included in the C:\Program Files\WorkPlace Pro Utilities\TreeLoader directory.

4. The Container Duplicator Utility

The Container Duplication Utility can be used for several different scenarios:

- Duplicating a single container any number of times within a station. All local links are duplicated as well.
- Duplicating two related containers in a single station any number of times within their respective parent containers. All local links for each container are duplicated, and all internal links between the two source containers are reproduced for each set of duplicated containers.
- Duplicating two related containers in two separate stations any number of times within their respective parent containers. All local links for each container are duplicated, and all external subscription links between the two source containers are reproduced for each set of duplicated containers.
- Duplicating a single container any number of times, while duplicating internal or external subscription links to another set of existing related containers.

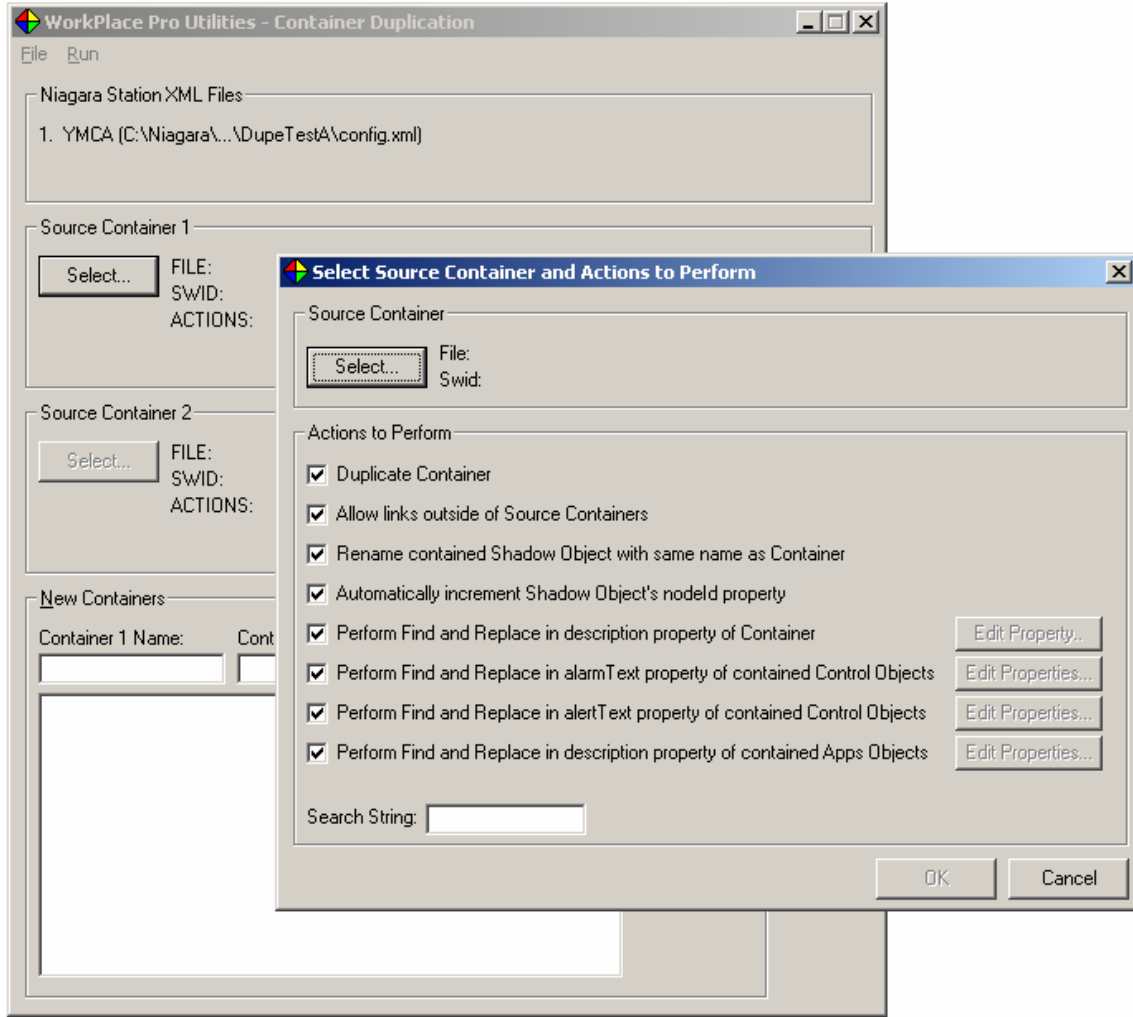
In each scenario you can select which additional actions are taken as the containers are duplicated:

- Rename a shadow object that has the same name as the container (i.e., if container C1001 has a shadow object with that name, the duplicated containers will have that shadow object renamed with the new name of the container).
- Automatically increment the value of the *nodeId* property in the shadow object.
- Perform a search and replace in the *alarmText* and *alertText* properties of the various control objects in the containers.
- Perform a search and replace in the *description* properties of the containers and various application objects in the containers.
- Duplicate or warn about links from the source container to objects outside of the source containers.

Selecting the Source Container

To begin, open the Niagara station XML file by selecting OPEN NIAGARA XML FILE... from the FILE menu. See *Chapter 2: Working with the WorkPlace Pro XML Station Files* for more details.

Next, click on the SELECT... button under SOURCE CONTAINER 1. This will bring up the following dialog:



Click on the SELECT... button in this dialog to select the source container, as described in *Chapter 2: Selecting a Container*. You can then select or unselect the actions that will be taken as the new containers are created.

Duplicate Container:

Indicates that this source container will be duplicated. If you are linking to a secondary set of containers that already exist, uncheck this action.

Allow links outside of Source Containers

If unchecked, any internal or external subscription links that exist between a source container and another container or object other than the secondary source container (if one is specified) will be considered "broken", will not be duplicated, and will be listed with a warning in the output report.

If checked, all links will be duplicated. If the link is local to a source container, or links the two source containers, it will be reproduced such that it links the duplicated containers. If the link is from a source container to another container or object other than the secondary source container, each duplicated container will have a link pointing to the exact same object.

See the later section on *Special Considerations* for more details.

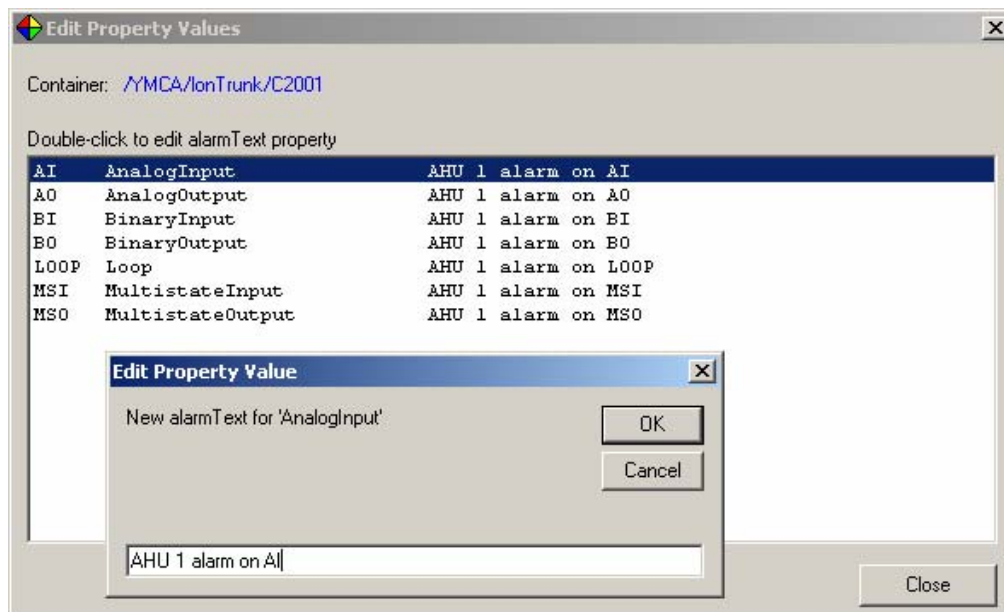
Rename contained Shadow Object with same name as Container

If checked, the Utility will look within the source container for an object with the same name as the container, assuming it is a shadow object. It will then rename the shadow object with the appropriate name of the duplicated container. See the later section on *Special Considerations* for more details.

Automatically increment Shadow Object's *nodeId* Property

If checked, the Utility will increment the *nodeId* property of the shadow object (see previous action) in each duplicated container. The *subnetId* property value will not be changed. For example, if the source container has a shadow object with subnet and node values of 3 and 4, the first three duplicated containers' shadow objects will have values of 3 and 5, 3 and 6 and 3 and 7.

For the Find and Replace actions on the various properties (i.e., container *descriptions*, *alarmText*, *alertText* and log *descriptions*), you can click the appropriate EDIT PROPERTIES... button to view the current values of these properties for the control objects in the source container:



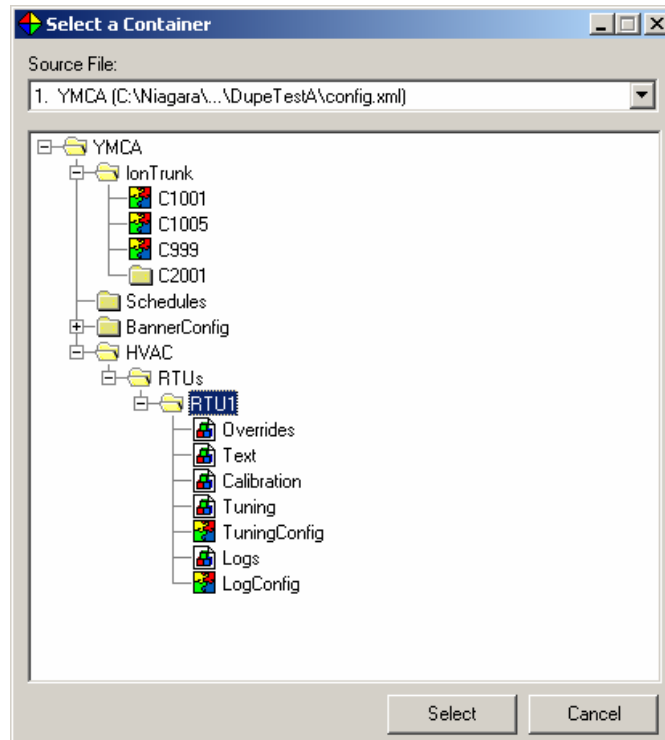
This list box shows the type, name and the property text value for each of the appropriate controls in the container. By double-clicking on an entry you can change the value of that property for the source container.

In this example, the *alarmText* for each control contains the text "AHU 1" to indicate the system in alarm. For the duplicated containers, we can specify that this text be replaced by the appropriate new system name (i.e., "AHU 2", "AHU 3", etc.). The string to search for ("AHU 1") should be specified in the SEARCH STRING text box on the Container Selection dialog. The text to replace it with for each new container ("AHU 2", "AHU 3") is specified with the list of new container names, as described later in this chapter.

Note: *For more advanced changes to the properties in the duplicated container and their objects, you can use the Property Sheets Utility to view and edit sets of properties as a group using Microsoft Excel. Refer to Chapter 7: The Property Sheets Utility for more information.*

Selecting a Secondary Source Container

There are cases where you want to duplicate two associated containers simultaneously. Consider the following example:



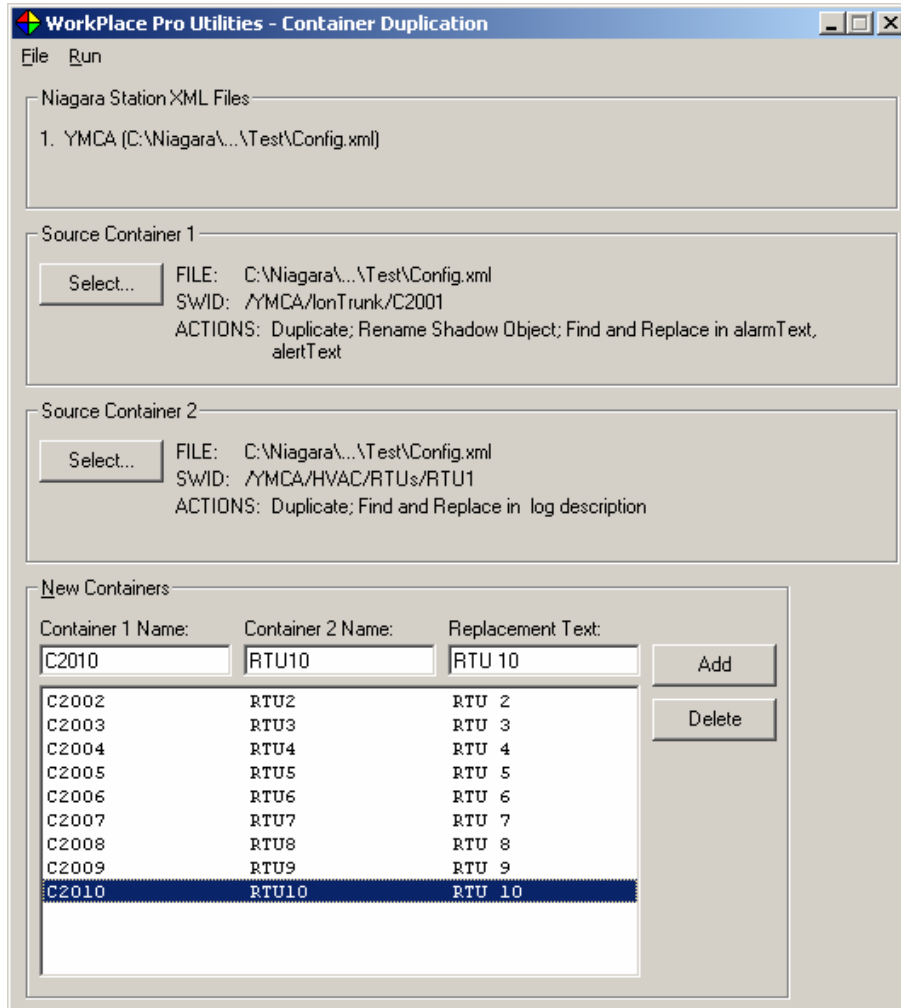
In our example, we have a logic container named *C2001* in the *lonTrunk* container, and another container named *RTU1* (in the *HVAC/RTUs* container) which consists of several graphics pages and logic bundles. There are numerous links between the *C2001* container and the pages and bundles in *RTU1*.

We want to create nine additional instances of *C2001* (named *C2002*, *C2003*, etc), and also need to duplicate the entire *RTU1* container and its contents as well (naming them *RTU2*, *RTU3*, etc.).

In this case, we will select *C2001* as Source Container 1, and *RTU1* as Source Container 2. If the logic and graphics were in separate files, we would have simply opened the second file and selected it from the SOURCE FILE list box when selecting the second source container.

Specifying the New Container Names

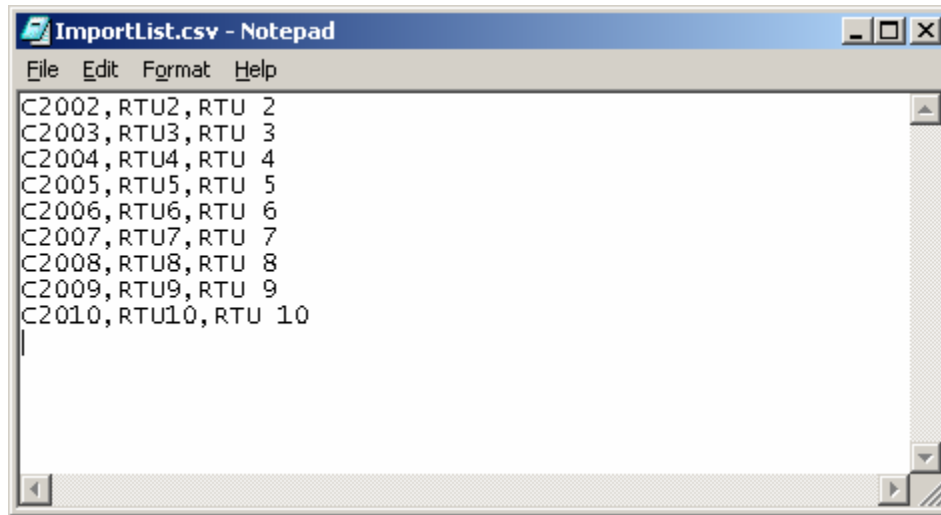
After selecting the source container or containers, you then specify the names to be given to the new instances of the containers as they are created.



Enter the name of the new container or containers in the appropriate text box. If you specified a Search String value to find and replace in the container, you should also specify the replacement text for each container. Click the ADD button or press ENTER to add the set to the list. You can edit an entry by selecting it in the list box, modifying the values in the text boxes, and again clicking ADD or pressing ENTER.

Importing the New Container List From a File

For larger systems, you can import the list of new container names and replacement text values from a comma-separated text file generated by a word processor or spreadsheet.

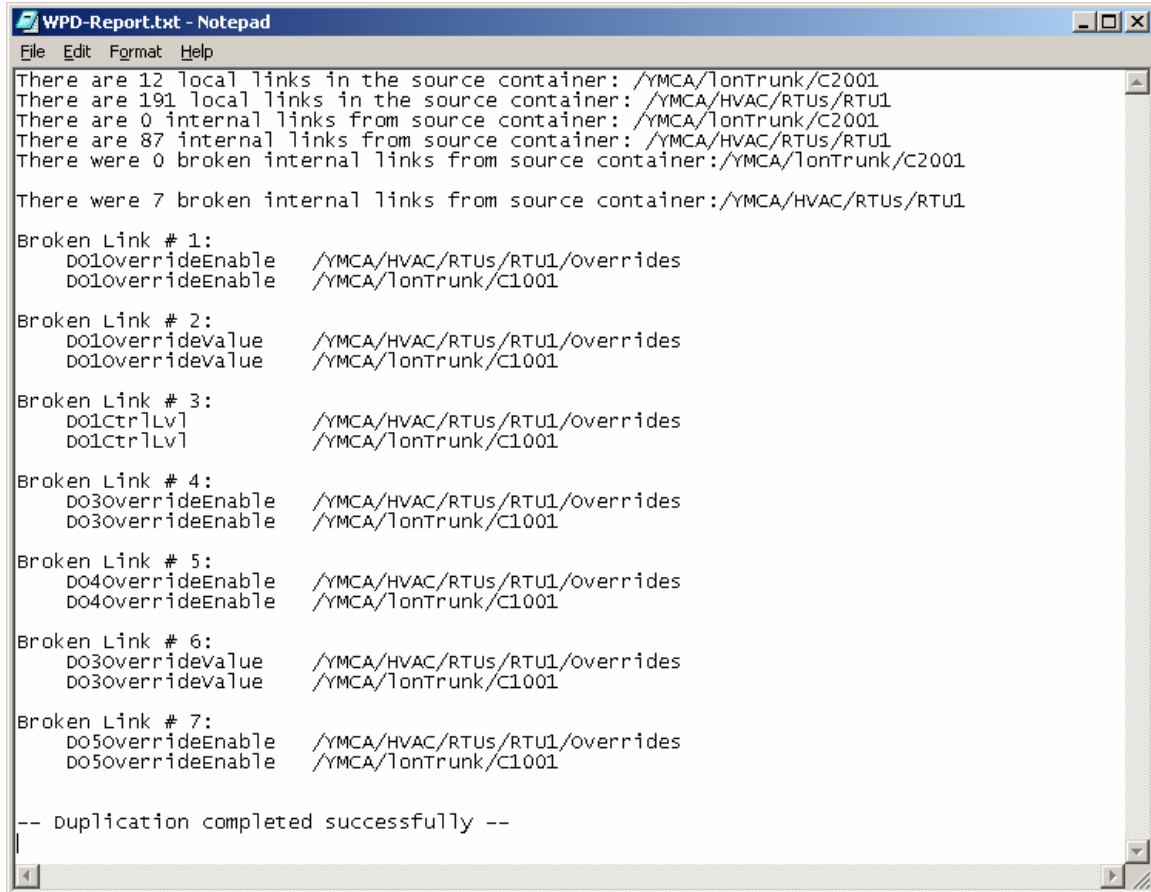


Select IMPORT NEW CONTAINER LIST... from the RUN menu to select the file to import.

Duplicating the Containers

Once the source containers and new container names have been specified, simply select DUPLICATE CONTAINERS from the RUN menu. The Utility will duplicate the nodes and prompt you to save the changes to the XML files. Click YES to save the changes immediately, or NO to preview the output report first.

The Output Report will be displayed in Notepad:



```

WPD-Report.txt - Notepad
File Edit Format Help
There are 12 local links in the source container: /YMCA/lonTrunk/C2001
There are 191 local links in the source container: /YMCA/HVAC/RTUS/RTU1
There are 0 internal links from source container: /YMCA/lonTrunk/C2001
There are 87 internal links from source container: /YMCA/HVAC/RTUS/RTU1
There were 0 broken internal links from source container:/YMCA/lonTrunk/C2001

There were 7 broken internal links from source container:/YMCA/HVAC/RTUS/RTU1

Broken Link # 1:
  DO1overrideEnable  /YMCA/HVAC/RTUS/RTU1/overrides
  DO1overrideEnable  /YMCA/lonTrunk/C1001

Broken Link # 2:
  DO1overridevalue   /YMCA/HVAC/RTUS/RTU1/overrides
  DO1overridevalue   /YMCA/lonTrunk/C1001

Broken Link # 3:
  DO1CtrlLv1         /YMCA/HVAC/RTUS/RTU1/overrides
  DO1CtrlLv1         /YMCA/lonTrunk/C1001

Broken Link # 4:
  DO3overrideEnable  /YMCA/HVAC/RTUS/RTU1/overrides
  DO3overrideEnable  /YMCA/lonTrunk/C1001

Broken Link # 5:
  DO4overrideEnable  /YMCA/HVAC/RTUS/RTU1/overrides
  DO4overrideEnable  /YMCA/lonTrunk/C1001

Broken Link # 6:
  DO3overridevalue   /YMCA/HVAC/RTUS/RTU1/overrides
  DO3overridevalue   /YMCA/lonTrunk/C1001

Broken Link # 7:
  DO5overrideEnable  /YMCA/HVAC/RTUS/RTU1/overrides
  DO5overrideEnable  /YMCA/lonTrunk/C1001

-- Duplication completed successfully --

```

The report indicates the number of local links within each container and the number of internal (or external subscription) links between the two containers. It also shows the number of internal and external subscription links between either source container and other nodes in the system. If the "Allow links outside of Source Containers" action was not selected, these links are considered "broken" and are not duplicated.

Saving the Files

After the containers have been duplicated, you can select **SAVE NIAGARA XML FILES...** from the **FILE** menu to save the changes back to the source XML files for importing back into WorkPlace Pro. The original XML files are automatically backed up in the same directory using numbers appended to the filename (i.e., **CONFIG.XML** will be copied to **CONFIG-1.XML** before being overwritten with the new version).

To save the configuration of the Container Duplicator itself (i.e., the names of the source XML files, the selected source containers, and the list of the new containers), select **SAVE WPD FILE...** from the **FILE** menu. This file can then be reloaded at a later time by selecting **OPEN WPD FILE...** from the **FILE** menu.

Special Considerations

The Container Duplicator does a lot of its work without a real regard for the specific nature of certain objects or links, allowing the Niagara Admin Tool to catch any errors when the XML file is reimported. To help avoid these errors, you should be aware of the following special cases.

Shadow Objects

The Container Duplicator makes a special case of an object with the same name as the source container, allowing you to indicate that it is a shadow object and have the Utility rename it along with the container. You can also have it increment the *nodeId* property as well.

You should avoid, however, trying to duplicate a shadow object that has already been commissioned or bound. Since essentially only the *name* and *nodeId* properties are changed (although the *neuronId* is cleared and the *status* set back to unconfigured), it is hard to predict how Niagara will react to the situation. You should try to duplicate containers that have only uncommissioned, unbound shadow objects.

Links Outside of Source Containers

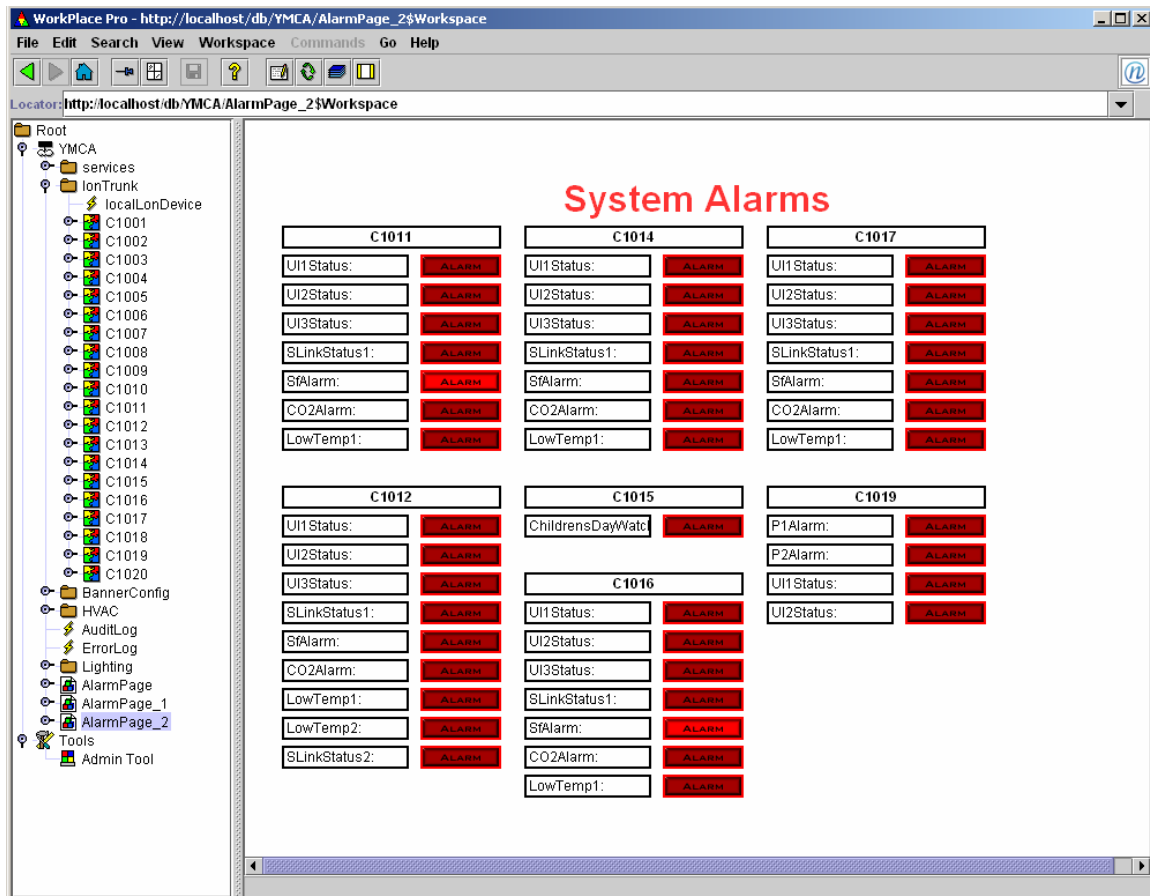
Another case that might the Container Duplicator to generate an XML file that Niagara can't reimport is when the source container has a link to an object outside itself and the other source container. The Utility allows you to either report these as warnings and not duplicate the link, or to ignore the warning and duplicate the link anyway.

In some cases, particularly an external value such as Outside Air Temperature that is being fed into an input in each of the duplicated containers, you will want the link duplicated and no errors will arise. Another valid example is an output in each of the duplicated containers being fed into a common trigger input. However, if you have a normal output value in the source container linked to an input in an outside container, this will not be accepted by Niagara.

To avoid problems, you should try running the duplicator with the "Allow Links Outside of Source Container" option turned off to have the report show you a list of the potential problems as warnings (and afterwards saying "no" when you are asked if you want to save the XML file). Once you are certain that each case will be valid, you can turn the option on and run the duplicator again.

5. The Alarm Page Builder Utility

The Alarm Page Builder Utility is used to populate a set of Gx Pages with boolean objects linked to alarmable controls throughout the system. These pages make it easy for users and technicians to locate the source of alarm conditions by showing all of the status indicators in one place.



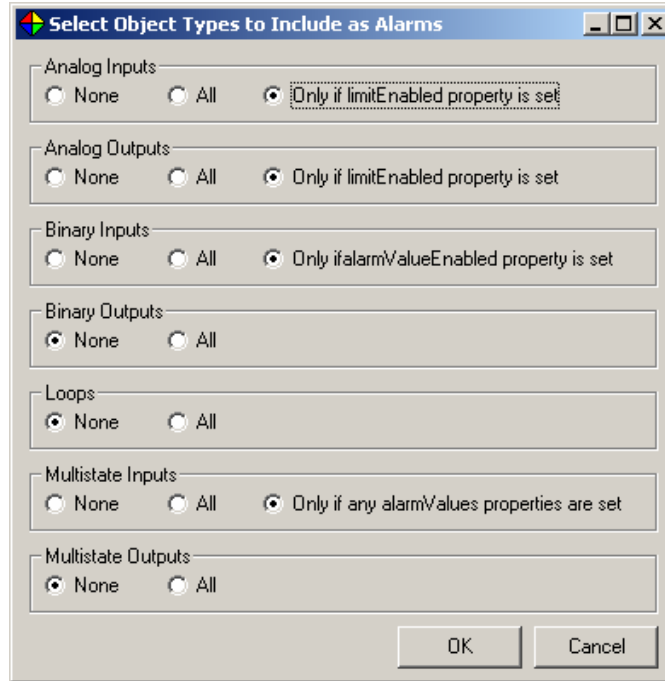
The source for the alarms are any or all of the alarmable controls (*Analog Input, Analog Output, Binary Input, Binary Output, Loop, Multistate Input and Multistate Output*) in a set of containers. In this example, we are pulling the controls from a set of logic bundles (C1001 – C1020) all located in the parent container named *lonTrunk*.

Selecting the Source Containers for Alarm Controls

To begin, open the Niagara station XML file by selecting OPEN NIAGARA XML FILE... from the FILE menu. See *Chapter 2: Working with the Workplace Pro XML Station Files* for more details.

Selecting the Control Types to Include

Before you select the containers with the control objects, you should specify which of the alarmable controls it will include. Select OBJECT TYPES TO INCLUDE... from the EDIT menu:



For each of the alarmable object types, you can choose to initially include or exclude all of those types of controls, or include only those with specific alarm properties set to values that indicate the control is configured for alarms. While you can remove alarms from the list later, you cannot add them, so it is preferable to select these options to include all possible controls you want on the alarm page.

Selecting the Parent Container

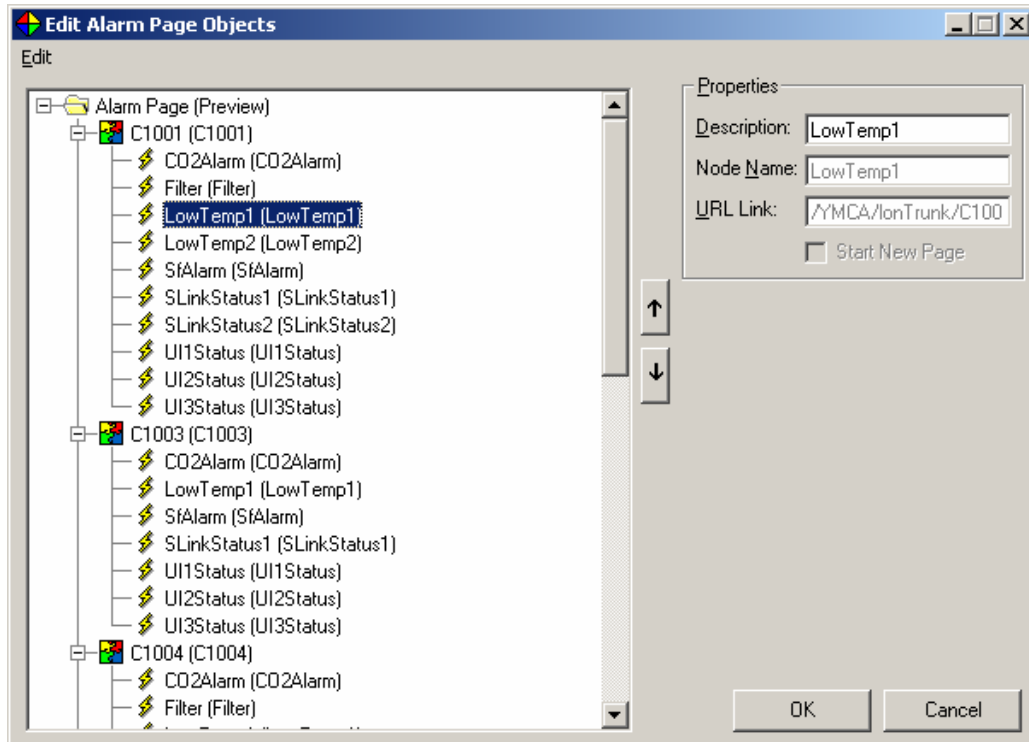
Click on the SELECT... button under SOURCE CONTAINER FOR ALARMABLE CONTROL OBJECTS. This will bring up the Container Selection dialog (for more information see *Chapter 2: Selecting a Container*). Select the *parent container* for all the containers which have alarmable controls. In our previous example, the parent container is *lonTrunk*, which contains the *C1001 – C1020* containers, which in turn contain the alarmable control objects.

After selecting the parent container, a preview tree showing the containers and their controls will be shown on the bottom of the form:



Selecting and Ordering the Alarms on the Page

Select ALARM LIST... from the EDIT menu to bring up an editable version of the alarm page structure. The dialog contains a two-level tree representing the alarms to be created on the page.



The first level shows each of the containers in the parent container that contain at least one control that meets the criteria specified in the OBJECT TYPES TO INCLUDE dialog. It also shows any sub-containers that had such controls (in the case of nested containers, the original tree structure is not reflected here). The second level shows each of the control objects in the container that were included.

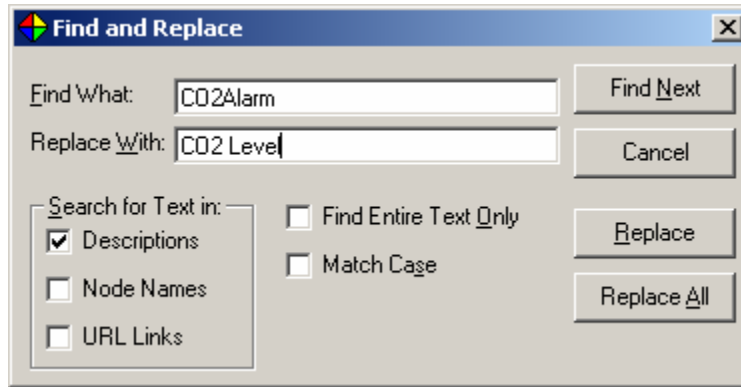
The *description* property of each container and input is shown on the tree, and will be used as the column headings and alarm captions on the generated page. You can modify these descriptions by selecting a node and editing the value in the DESCRIPTION text box, or by right-clicking on the node and selecting RENAME.

The node name, as specified in the station, is shown in parentheses and can't be changed here.

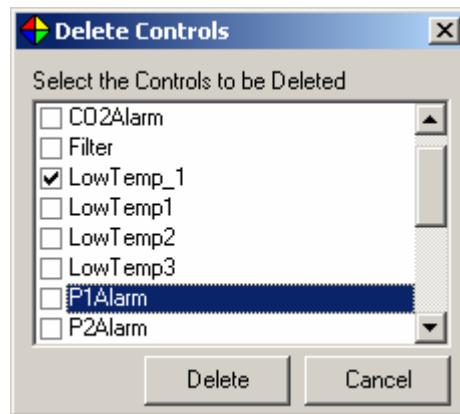
To delete a single input, right click on the node and select DELETE from the pop-up menu, or click on the node and press the DELETE key. You can prevent the delete confirmation message from being displayed by checking the DELETE WITHOUT CONFIRM item in the EDIT menu.

To change the order of the inputs within a container, or of the containers on the alarm pages, use the up and down arrows on the right side of the tree. These changes affect only the alarm page that is generated, and do not affect the position of these objects in the WorkPlace Pro file structure itself.

To change the text on more than one alarm, select FIND/REPLACE from the EDIT menu:



To remove the same control from all the containers, select DELETE CONTROLS... from the EDIT menu. Again, this affects only the list of alarms to be generated, and does not delete the actual control from the system:



Specifying the Gx Page for the Alarm Screen

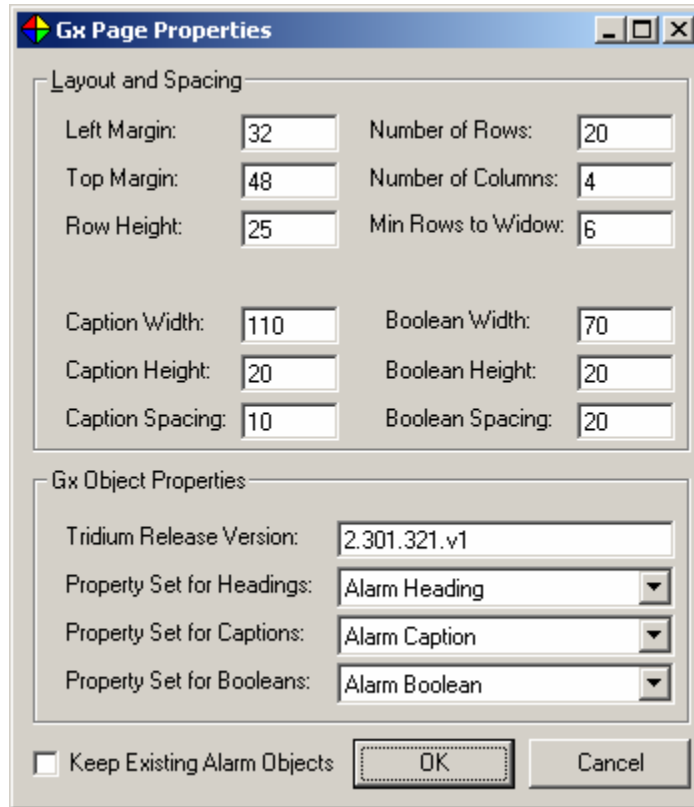
The boolean objects for the alarms and the associated captions and headings will be generated onto one or more Gx Pages. Before you run the utility, you should create a Gx Page in your system to use as a template for each of the alarm pages. This page will most likely be blank, except for heading information and background color settings. If you wish to include other static information on the top, bottom or sides of the page, be sure to set the page layout properties appropriately to avoid the alarms being placed on top of these other objects.

If the Gx Page for the alarm is in the same station file as the alarmable controls themselves, simply click on the SELECT... button under GX PAGE ON WHICH TO CREATE ALARM TABLE and choose that page. If the alarm page will be located in a separate file, open that station file first (by selecting OPEN NIAGARA XML FILE... from the FILE menu) and select that file in the Container Selection dialog to locate the page in its tree structure.

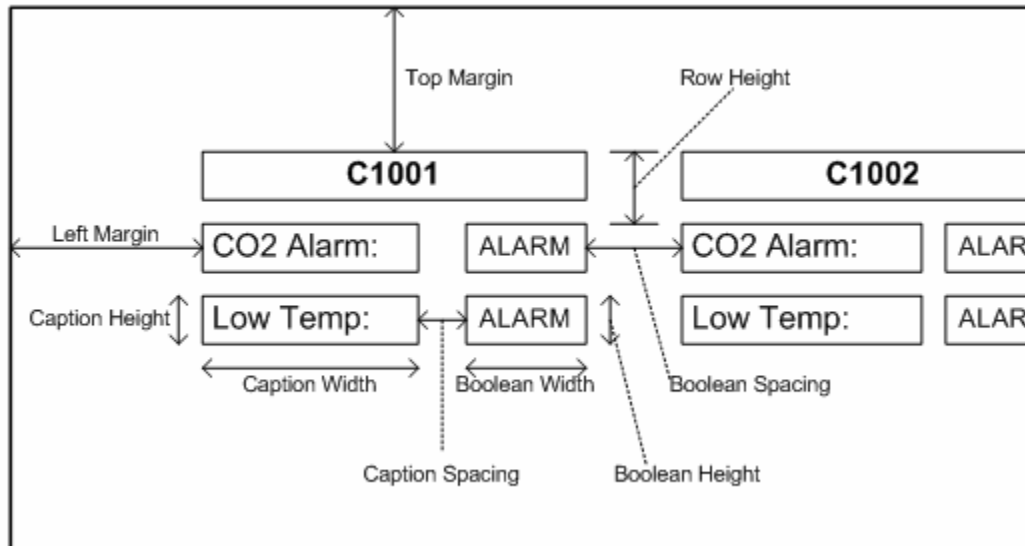
When the Gx Pages for the alarms are generated, they will be placed in the same container as the template page with "_1", "_2", etc. appended to the name of the page.

Setting the Page Layout Properties

To see the various page layout properties that can be specified, select PAGE LAYOUT PROPERTIES from the EDIT menu:



These values represent the following on the generated alarm page:



In addition, the NUMBER OF ROWS and NUMBER OF COLUMNS on a page can be specified. To help control where the column and page breaks occur, you can specify the MINIMUM ROWS TO WIDOW property. During the layout, before a new group of alarms is added to the page, the number of remaining rows in the column is checked. If there are less than the specified number of rows available at the bottom of the column, the utility will automatically break to the top of the next column.

To force a page break at the top of any container group, check the START NEW PAGE checkbox for the container when viewing the inputs as a tree, as shown in the *Selecting and Ordering the Alarms on the Page* section.

There is a template containing the default layout values, called DEFAULT.WPA, which you can change to meet your needs. It is a better idea, however, to save your own WPA files to create custom templates. Refer to the later section on *Loading and Saving the WPA File* and to *Chapter 2: Saving and Loading the Workplace Pro Utilities WPx File*.

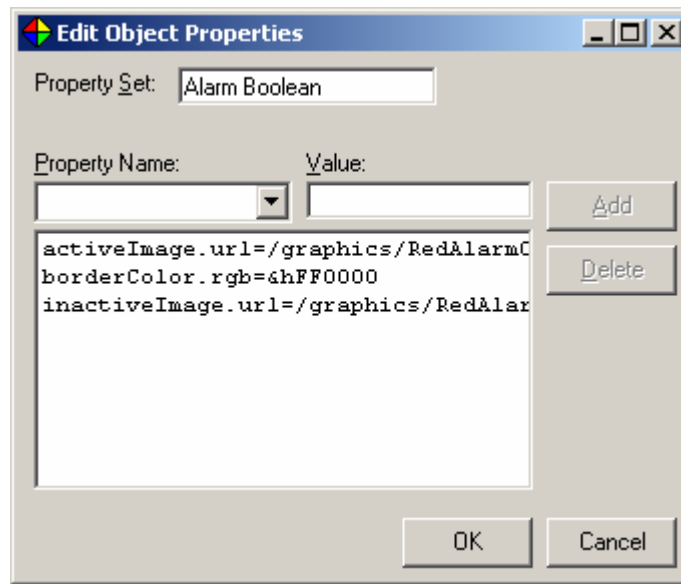
Setting the Properties of the Gx Objects

There are three type of Gx objects created on the alarm page. The description of the container becomes the heading (a GxText object). The description property of the alarmable control becomes the caption (another GxText object). The alarm itself is a GxBoolean object linked to the control in the source container.

You can update the release number to be specified for the created objects by changing the value shown for TRIDIUM RELEASE VERSION.

You can also specify additional properties for these Gx Objects by creating property sets and specifying their names as PROPERTY SET FOR HEADINGS, PROPERTY SET FOR CAPTIONS or PROPERTY SET FOR BOOLEANS. You can edit the property sets by selecting GX OBJECT PROPERTY SETS from the EDIT menu. Refer to *Chapter 2: Gx Object Property Sets* for more information.

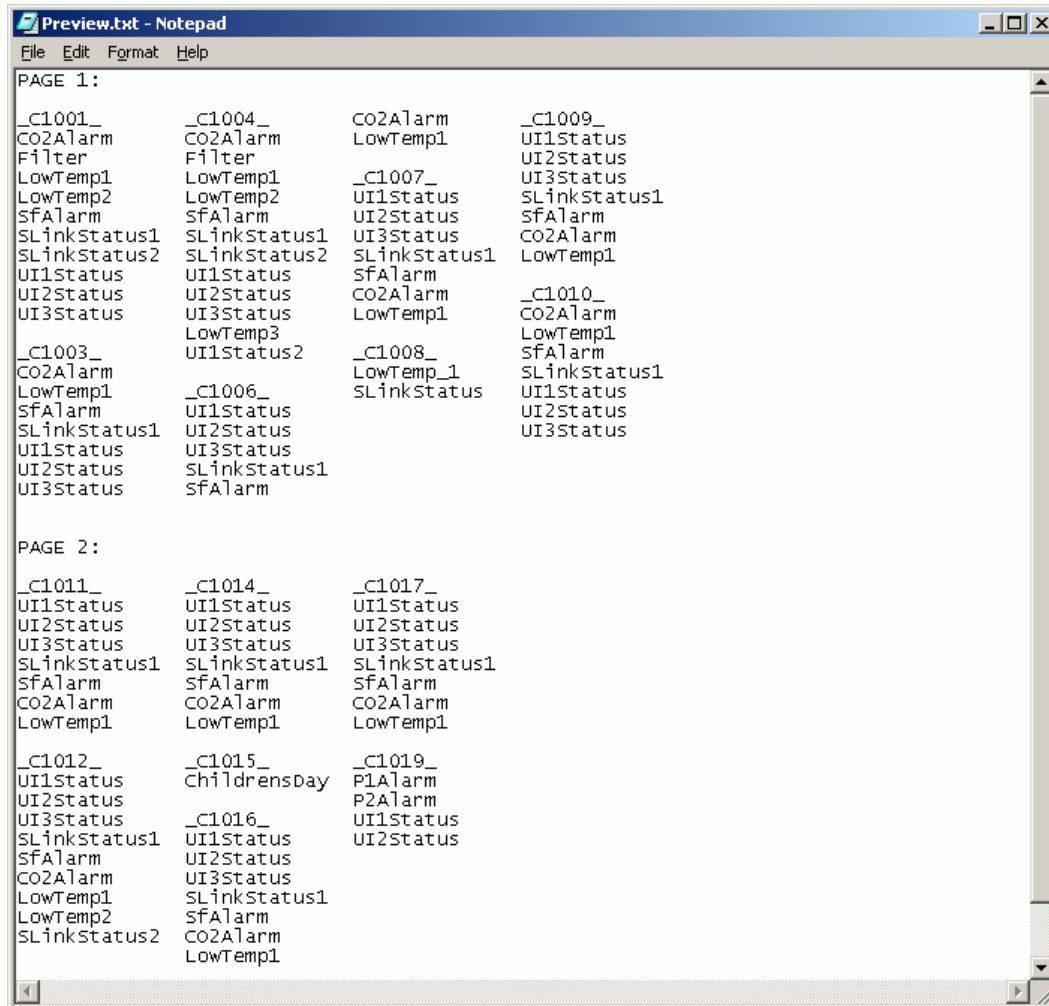
The default template contains property sets for the headings, captions and boolean objects. Two important properties you will probably want to set for the boolean objects are the `activeImage.url` and `inactiveImage.url`, which indicate the name of the graphics file to display for the alarms under active and inactive conditions. By default these properties refer to the "RedAlarmOn.gif" and "RedAlarmOff.gif" files in a directory called "graphics". This directory is installed in the Workplace Pro Utilities directory, but must be copied into the appropriate Niagara directory (i.e., "r2.301.321") on the server running the application. Alternatively, you can copy these files into a different directory and change these properties, or point to entirely different image files.



As with the layout parameters, these object property sets are stored in the WPA file, including the DEFAULT.WPA template. You can also export the property sets for importing into other files. Refer to *Chapter 2: Gx Object Property Sets* for more information.

Previewing the Page Layout

To see how the columns and pages will be arranged without generating the page and reimporting the XML file back into WorkPlace Pro, select PREVIEW PAGE LAYOUT from the RUN menu.



```

Preview.txt - Notepad
File Edit Format Help
PAGE 1:
_C1001_      _C1004_      CO2Alarm      _C1009_
CO2Alarm    CO2Alarm    LowTemp1      UI1Status
Filter      Filter      LowTemp1      UI2Status
LowTemp1    LowTemp1    _C1007_      UI3Status
LowTemp2    LowTemp2    UI1Status     SLinkStatus1
SfAlarm     SfAlarm     UI2Status     SfAlarm
SLinkStatus1 SLinkStatus1 UI3Status     CO2Alarm
SLinkStatus2 SLinkStatus2 SLinkStatus1  LowTemp1
UI1Status   UI1Status   SfAlarm
UI2Status   UI2Status   CO2Alarm
UI3Status   UI3Status   LowTemp1
              LowTemp3
              UI1Status2
_C1003_      _C1006_      _C1008_
CO2Alarm    UI1Status   LowTemp_1    SfAlarm
LowTemp1    UI2Status   SLinkStatus  UI1Status
SfAlarm     UI3Status   UI1Status    UI2Status
SLinkStatus1 SLinkStatus1 SLinkStatus1 UI3Status
UI1Status   UI2Status
UI2Status   SLinkStatus1
UI3Status   SfAlarm
PAGE 2:
_C1011_      _C1014_      _C1017_
UI1Status    UI1Status    UI1Status
UI2Status    UI2Status    UI2Status
UI3Status    UI3Status    UI3Status
SLinkStatus1 SLinkStatus1 SLinkStatus1
SfAlarm     SfAlarm     SfAlarm
CO2Alarm    CO2Alarm    CO2Alarm
LowTemp1    LowTemp1    LowTemp1
_C1012_      _C1015_      _C1019_
UI1Status    ChildrensDay P1Alarm
UI2Status    UI1Status    P2Alarm
UI3Status    UI2Status    UI1Status
SLinkStatus1 UI3Status    UI2Status
SfAlarm     SLinkStatus1
CO2Alarm    SfAlarm
LowTemp1    CO2Alarm
LowTemp2    LowTemp1
SLinkStatus2 LowTemp1
  
```

In this example, the NUMBER OF ROWS was set to 20, the NUMBER OF COLUMNS was 4, and the MINIMUM ROWS TO WIDOW was set to 5. The column breaking can be shown by looking at the second and third columns of the first page, where C1006 did not force a new column to start, even though all its alarms did not fit, because there were more than the specified 5 rows available. On the other hand, C1009 did not start at the bottom of the third column, but instead forced a new column because there were only 5 free rows remaining.

The preview is intended to show you the layout of the column and page breaks. The other page layout properties such as margins and column widths are not reflected in the preview.

Generating the Alarm Pages

To generate the alarm pages select **GENERATE ALARM PAGE(S)** from the **RUN** menu.

As long as the **KEEP EXISTING ALARM OBJECTS** check box in the **PAGE LAYOUT PROPERTIES** dialog is not selected, any text and boolean objects previously placed on the pages by the utility will be removed.

After the pages are generated you will be prompted to save the updated XML file. You can also save the XML file at any time by selecting **SAVE NIAGARA XML FILES** from the **FILE** menu.

Loading and Saving the WPA File

You can save the configuration of your alarm page (the name of the source XML file, the selected alarm controls and their ordering, the page layout properties and the Gx object property sets) as a file (with a **.WPA** extension) by selecting **SAVE WPA FILE** from the **FILE** menu. You can then reload the settings by selecting **OPEN WPA FILE** from the **FILE** menu.

There is a standard set of page layout and Gx object properties saved in a file called **DEFAULT.WPA**, which is located in the directory where the application was installed ("C:\Program Files\WorkPlace Pro Utilities" by default). This file is automatically loaded when you start the Alarm Page Utility. To modify these defaults, simply run the utility, make the desired changes, and resave the WPA file back to **DEFAULT.WPA**. It is a better idea, however to save to a different configuration file and explicitly load it as a template before creating a new alarm page.

6. The Value Table Builder Utility

The Value Table Builder Utility is used to create a table of values on a Gx Page, where each row of the table represents a container (or in some cases, multiple rows per container) and each column represents a composite value from that container. This greatly simplifies the aggregation of certain status information for a group of similar systems into a single place for users and technicians to access.

The screenshot shows the Value Table Builder Utility interface. On the left is a tree view of the project structure, including containers C1001 through C3010. The main area displays a table titled "My Value Table" with the following data:

	MSO PV	AI sOut	BI sOut	Totalizer	Duration	Max	Min
C3001	0	0.00	Alarm	451.34	1	100.00	0.00
C3002	0	0.00	Alarm	451.34	1	100.00	0.00
C3002-b					1	100.00	0.00
C3003	0	0.00	Alarm	451.34	1	100.00	0.00
C3004	0	0.00	Alarm	451.34	1	100.00	0.00
C3004-b					1	100.00	0.00
C3005	0	0.00	Alarm	451.34	1	100.00	0.00
C3006	0	0.00	Alarm	451.34	1	100.00	0.00
C3006-b					1	100.00	0.00

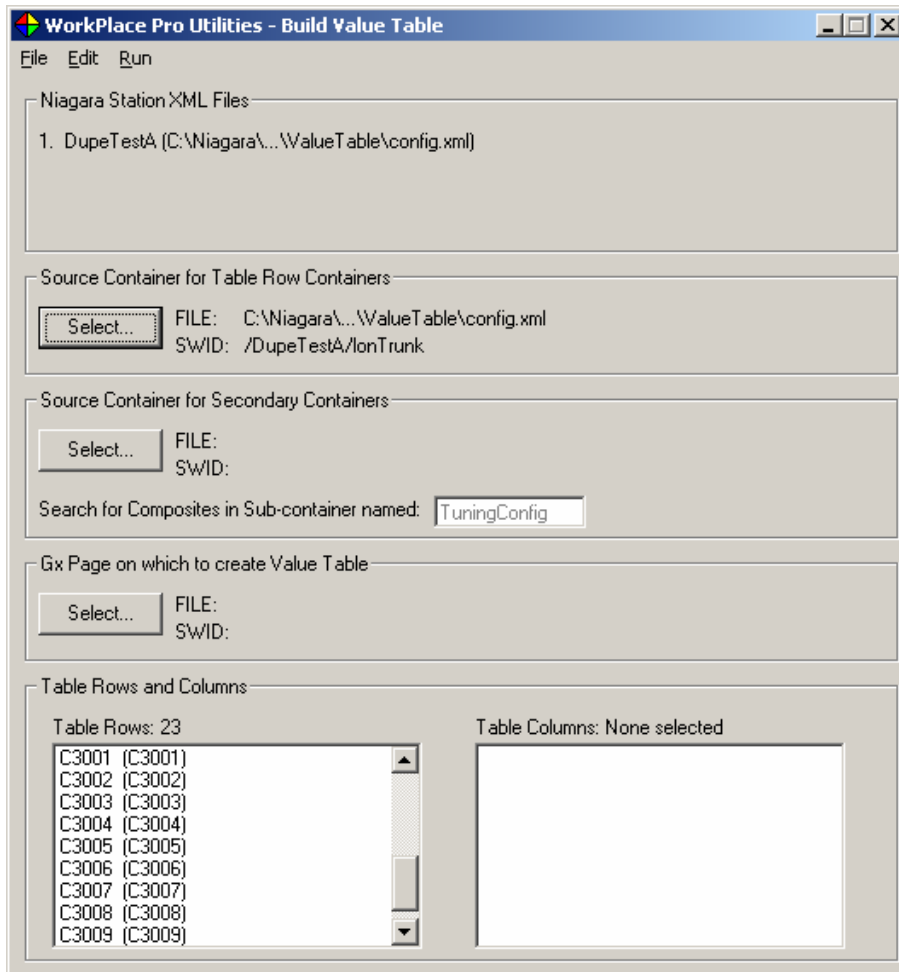
In this example we have selected 7 values from each of the logic bundles *C1001 – C1010* in the *lonTrunk* container.

Selecting the Source Containers for the Table Rows

To begin, open the Niagara station XML file by selecting **OPEN NIAGARA XML FILE...** from the **FILE** menu. See *Chapter 2: Working with the Workplace Pro XML Station Files* for more details.

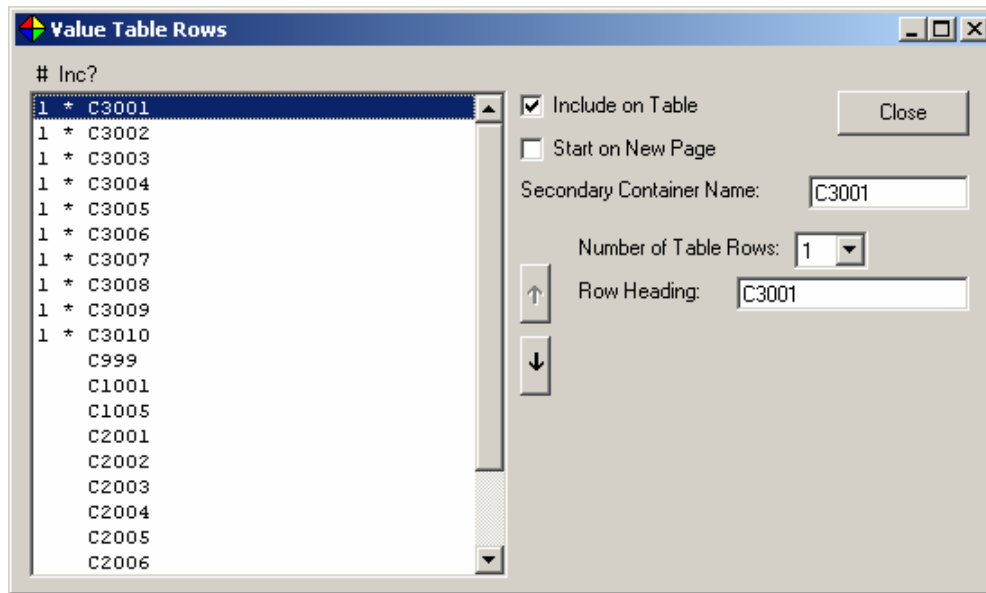
Next, click on the SELECT... button under SOURCE CONTAINER FOR TABLE ROW CONTAINERS. This will bring up the Container Selection dialog (for more information see *Chapter 2: Selecting a Container*). Select the *parent container* for all the containers which have the composites to be used as the values in the table. In our previous example, the parent container is *lonTrunk*, which contains the *C1001 – C1010* logic bundles, which in turn have the composites for the individual cells of the table.

After selecting the parent container, the list of table rows will be shown in the lower left corner of the form. By default, all of the containers in the parent container are included as rows of the table.



Selecting the Table Rows

Select TABLE ROWS... from the EDIT menu to bring up the Table Rows dialog.



By clicking on any of the containers listed on the left, you can view and edit the following properties as they pertain to the table:

INCLUDE ON TABLE If checked, composite values from the container will be used to create one (or more) rows on the table. The asterisk next to the container name in the list box indicates it is currently included. You can also double-click the name in the list box to change whether or not it is included.

START ON NEW PAGE If checked, the row for the selected container will automatically begin on a new page of the table. This overrides the Number of Rows setting described later in this chapter.

SECONDARY CONTAINER NAME: If you want each row of the table to be comprised of values from two separate, but associated containers, specify the name of the corresponding secondary container here. This method is described in the later section, *Selecting Secondary Source Containers for Table Rows*.

NUMBER OF TABLE ROWS: By default, each source container creates one row in the table. For containers that represent systems with multiple sub-systems (i.e., an Exhaust Fan system with three separate areas), you can specify more than one row. This method is described in the later section *Creating Multiple Rows per Container*.

ROW HEADING: This is the text that will be displayed as a heading on the left side of the row. By default it is the same as the container name. If you specified more than one row, you will be presented with a separate text box for each row.

You can also reorder the rows in the table by using the up and down arrows on the right side of the list box.

Selecting Secondary Source Containers for Table Rows

In some systems, you might find it useful to incorporate values from two separate containers, which may reside in different stations, for each row. For example, the system represented by *C1001* may have associated graphics pages in a container called *RTU1*, which is located in a parent container called *RTUs*. In this *RTU1* container, there may be a logic bundle called *TuningConfig*, which provides certain logic functions to drive the other Gx Pages in *RTU1*. In this scenario, we can create a table that takes some of its columns from *C1001*, and others from *RTU1/TuningConfig*.

Let us also assume that the row for *C1002* will have column values from *RTU2/TuningConfig*, that *C1003* will be associated with *RTU3*, and so on.

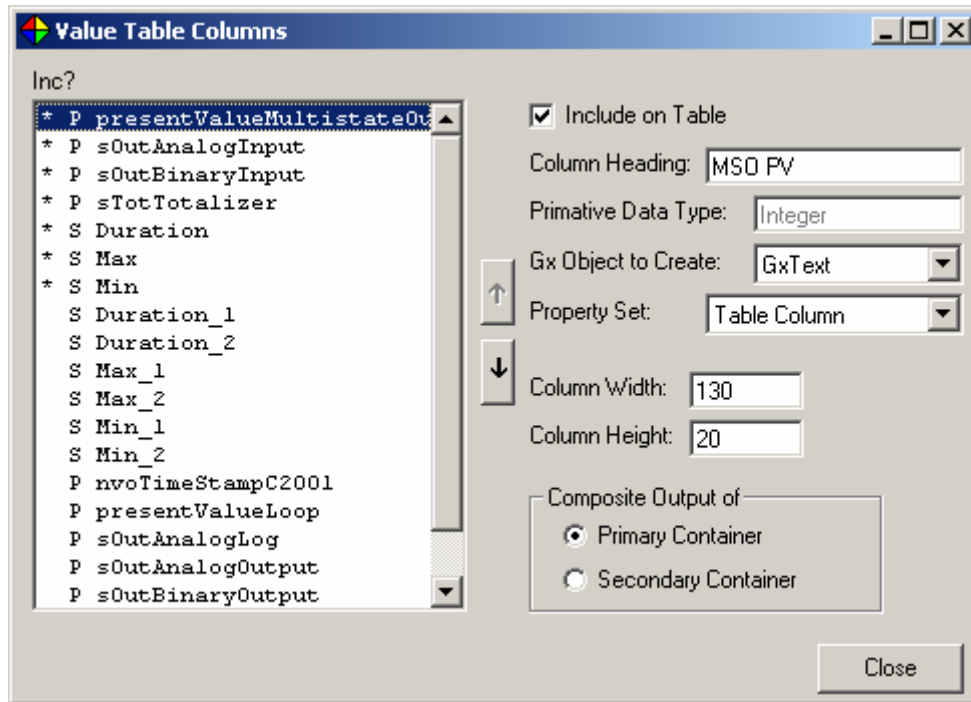
To select the secondary parent container (in this case, *RTUs*), click on the **SELECT** button under **SOURCE CONTAINERS FOR SECONDARY CONTAINERS**.

To associate each primary container with its secondary container (i.e., *C1001* with *RTU1*, *C1002* with *RTU2*), specify the secondary container name (i.e., *RTU1*) for each primary container (i.e., *C1001*) in the Table Row dialog, as described in the previous section.

Finally, to have the column values be found as a composite of each secondary container's sub-container called *TuningConfig*, specify "TuningConfig" in the **SEARCH FOR COMPOSITES IN SUB-CONTAINER NAMED** text box on the main form.

Selecting the Table Columns

After selecting the primary containers (and, optionally, the secondary containers), select TABLE COLUMNS... from the EDIT menu to select the composites to be used as columns.



The list box on the left shows a complete list of the composites from all the containers selected as table rows, as well as from their associated secondary containers (if specified, as described in the previous section). The asterisk next to the name indicates that the values is included as a column of the table. The "P" or "S" indicate whether it is located in the primary or secondary container.

By clicking on any of the composites listed on the left, you can view and edit the following properties as they pertain to the table:

- | | |
|---------------------|--|
| INCLUDE ON TABLE | If checked, the value of the composite will appear as a column of the table. The asterisk next to the composite name in the list box indicates it is currently included. You can also double-click the name in the list box to change whether or not it is included. |
| COLUMN HEADING: | This is the text that will be displayed as a heading on the top of the column. By default it is the same as the composite name. |
| PRIMITIVE DATA TYPE | This read-only field shows the data type of the composite. |

GX OBJECT TO CREATE:	In most cases the best object to create to display the values in the table will be a GxText object (except for boolean values, which will likely be shown with a GxBoolean object). You can, however, override this choice by selecting an alternative object type (such as GxInteger or GxFloat) from the list.
PROPERTY SET:	This is the name of the set of additional properties to be applied to the Gx Object created for the cell values. Refer to <i>Chapter 2: Gx Object Property Sets</i> for more information.
COLUMN WIDTH:	The width of the Gx Objects created for the column values, including the heading.
COLUMN HEIGHT:	The height of the Gx Objects created for the column values, <i>not</i> including the heading.
COMPOSITE OUTPUT OF:	Indicates if the composite is coming from the primary or secondary source container.

You can also reorder the columns in the table by using the up and down arrows on the right side of the list box.

Creating Multiple Rows Per Container

By default, each source container (or each pair of primary and secondary containers) corresponds to one row in the table. For containers that represent systems with multiple sub-systems (i.e., an Exhaust Fan system with three separate zones), you can have each container generate more than one row.

As an example, assume a logic bundle (*C3001*) has a controller shadow object that is configured to control three systems. As a result, there are output composites named *OccSp*, *OccSp_2*, *OccSp_3*, along with *Min*, *Min_2*, *Min_3* and *Max*, *Max_2* and *Max_3*. We would like this bundle to represent three rows on the value table.

To accomplish this, you would do the following:

- In the Table Rows dialog, select *C3001* in the list box and set Number OF TABLE ROWS to 3. Then specify ROW HEADING values for the three rows.
- In the Table Columns dialog, include the *OccSp*, *Min* and *Max* composites.

The second and third rows of the table will automatically be populated with composites with names the same as the selected composites, but with “_2” and “_3” appended to the names. You should not explicitly include the *OccSp_2*, *OccSp_3*, *Min_2*, *Min_3*, *Max_2* or *Max_3* composites as columns.

This method will also work if the composites for the first row were named *OccSp_1*, *Min_1*, and *Max_1* (i.e., the “_1” will be removed and replaced with “_2” and “_3”).

Specifying the Gx Page for the Value Table

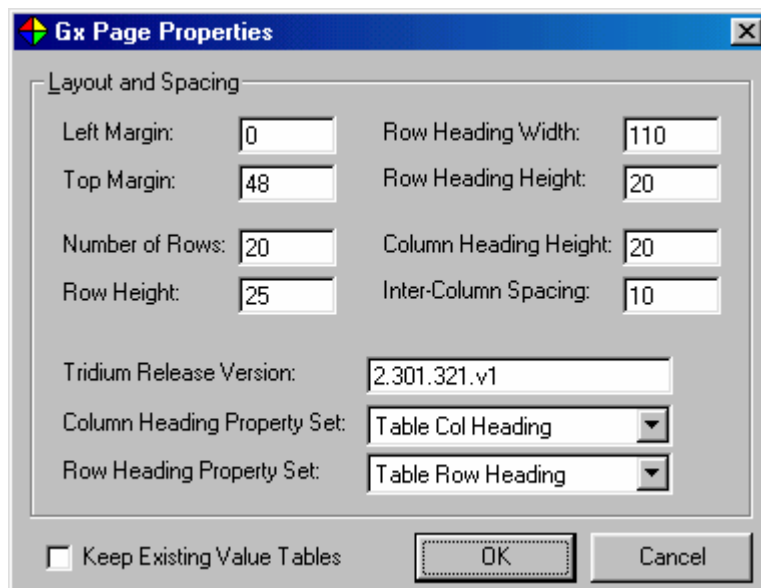
The value table will be generated onto one or more Gx Pages. Before you run the utility, you should create a Gx Page in your system to use as a template for each of the pages of the table. This page will most likely be blank, except for heading information and background color settings. If you wish to include other static information on the top, bottom or sides of the page, be sure to set the page layout properties appropriately to avoid the table being placed on top of these other objects.

To specify the template Gx Page, click on the SELECT... button under GX PAGE ON WHICH TO CREATE VALUE TABLE and choose that page. If the value table page will be located in a station file different from the source containers, open that station file first (by selecting OPEN NIAGARA XML FILE... from the FILE menu) and select that file in the Container Selection dialog to locate the page in its tree structure.

When the Gx Pages for the value tables are generated, they will be placed in the same container as the template page with “_1”, “_2”, etc. appended to the name of the page.

Setting the Page Layout Properties

To see the various page layout properties that can be specified, select PAGE LAYOUT PROPERTIES from the EDIT menu:



Here you can specify the left and top margins, the number of table rows per page, the total height of each row, the width and height of the row heading Gx Text objects, and height of the column heading Gx Text objects, and the amount of space between each table column.

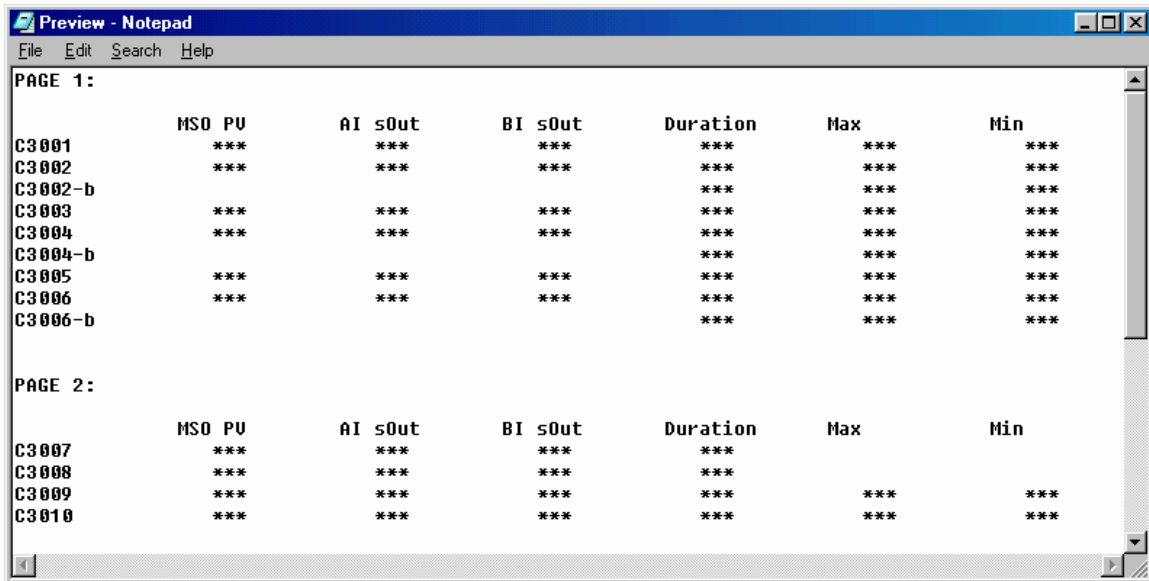
You can also update the release number to use when creating the Gx Objects.

Additionally, you can specify additional properties for the Gx Text objects created for the row and column headings by creating property sets and specifying their names as COLUMN HEADING PROPERTY SET, and ROW HEADING PROPERTY SET. You can edit the property sets themselves by selecting GX OBJECT PROPERTY SETS from the EDIT menu. Refer to *Chapter 2: Gx Object Property Sets* for more information.

There is a template containing the default layout values and property sets, called DEFAULT.WPV, which you can change to meet your needs. It is a better idea, however, to save your own WPV files to create custom templates. Refer to the later section on *Loading and Saving the WPV File* and to *Chapter 2: Saving and Loading the Workplace Pro Utilities WPx File*.

Previewing the Page Layout

To see how the rows and columns of the table will be arranged without generating the page and reimporting the XML file back into Workplace Pro, select PREVIEW PAGE LAYOUT from the RUN menu.



In this example, *C3002*, *C3004* and *C3006* were set to two rows each, and *C3007* has the **START ON NEW PAGE** checkbox selected. The asterisks in the cells of the preview indicate that the composite was found and can be linked. The missing cells (i.e., the first three columns of *C3002-b*, and the last two columns of *C3007* and *C3008*) indicate composites that were not found. In these cases, a blank spot will be left on the generated table.

Generating the Value Table

To generate the value table select **GENERATE VALUE TABLE** from the **RUN** menu.

As long as the **KEEP EXISTING VALUE TABLES** check box in the **PAGE LAYOUT PROPERTIES** dialog is not selected, any objects previously placed on the pages by the utility will be removed.

After the pages are generated you will be prompted to save the updated XML file. You can also save the XML file at any time by selecting **SAVE NIAGARA XML FILES** from the **FILE** menu.

Loading and Saving the WPV File

You can save the configuration of your value table (the name of the source XML file, the selected rows and columns and their ordering, the page layout properties and the Gx object property sets) as a file (with a .WPV extension) by selecting **SAVE WPV FILE** from the **FILE** menu. You can then reload the settings by selecting **OPEN WPV FILE** from the **FILE** menu.

There is a standard set of page layout and Gx object properties saved in a file called **DEFAULT.WPV**, which is located in the directory where the application was installed ("C:\Program Files\WorkPlace Pro Utilities" by default). This file is automatically loaded when you start the Alarm Page Utility. To modify these defaults, simply run the utility, make the desired changes, and resave the WPV file back to **DEFAULT.WPV**. It is a better idea, however to save to a different configuration file and explicitly load it as a template before creating a new value table.

7. The Property Sheets Utility

The Property Sheets Utility is designed to view and edit a specified set of property values for a specified group of objects in a Niagara application. The reports and edit screens are created using Microsoft Excel, providing you with the utmost flexibility and familiarity in manipulating the values.

Some of the possible uses include:

- Setting or changing the same property across multiple controllers or objects without going into the property page for each object individually
- Creating reports of current object values, such as set points
- Verifying consistency and completeness of descriptions, alarm text and alert text values.
- Renaming objects
- Running unattended reports as part of a nightly or weekly batch procedure

The use of Microsoft Excel provides the following capabilities:

- Cut, copy and paste capabilities within the data and from external sources
- Use of formulas to quickly complete values based on other values
- Fill Down feature to easily populate many objects with identical or similar values
- Knowledge of property types and acceptable values to validate data and create list boxes that mimic those in WorkPlace Pro's property pages
- Ability to search, sort and auto-filter reported values

Notes about Microsoft Excel

To generate a Property Sheet report, Microsoft Excel (Version 2000 or later) must already be installed on the computer.

If you are using Microsoft Excel XP, you will need to set the following security setting to create editable Property Sheets:

- From the Excel TOOLS menu, select MACRO and then SECURITY...
- On the TRUSTED SOURCES tab, make sure the TRUST ACCESS TO VISUAL BASIC PROJECT checkbox is selected.

Selecting the Source File and Container to Report

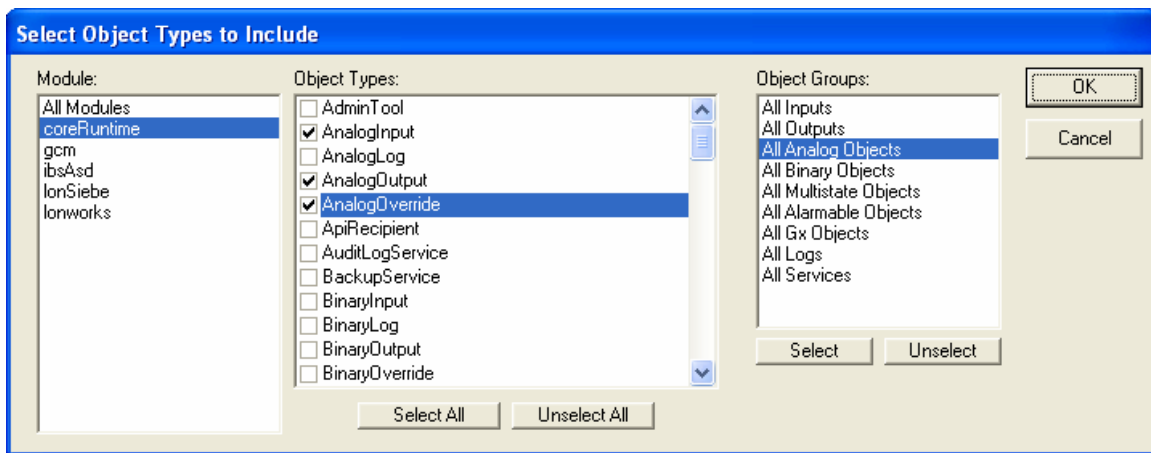
To begin, open the Niagara station XML file by selecting OPEN NIAGARA XML FILE... from the FILE menu. See *Chapter 2: Working with the WorkPlace Pro XML Station Files* for more details.

Next, click on the SELECT... button under SOURCE CONTAINER FOR OBJECTS. This will bring up the Container Selection dialog (for more information see *Chapter 2: Using the Station Building Utilities*). Select the *top-level container* for all the objects on which you wish to report. To report on all objects in the station, select the root station node.

The INCLUDE OBJECTS IN SUB-CONTAINERS checkbox is selected by default. If you uncheck this box, only the selected container and its objects are reported. Otherwise, all objects in the container and all sub-containers are included.

Selecting the Object Types to Include

To specify which object types you wish to include in the report, click on the SELECT OBJECT TYPES... button. This will bring up the Object Type selection dialog:



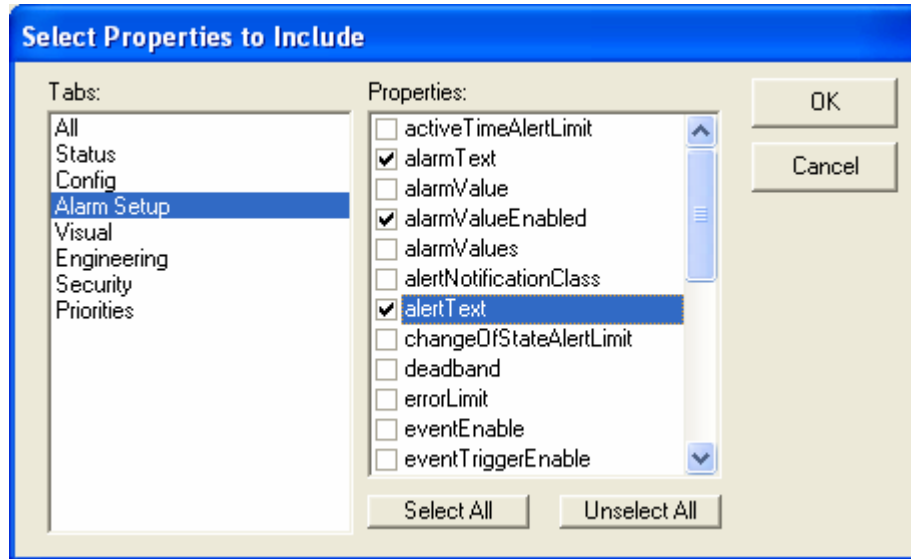
The leftmost list box shows the list of the Niagara modules that the utility has in its database (for information on updating the module database, see the *Adding to the Module and Object Database* section later in this chapter). Selecting a module in this list causes the Object Types list to display only the objects in that module. If you select All Modules, all the objects in the database are displayed, prefixed with their module name.

Select the object types you want by clicking the checkbox next to each one. You can select (or unselect) all the objects in a particular module by selecting that module and clicking the SELECT ALL (or UNSELECT ALL) button beneath the OBJECT TYPES list.

For some modules, a list of object groups have been defined to make it easier to select multiple object types. For example, the *coreRuntime* module shows groups such as All Analog Objects. You can select (or unselect) a group of objects by clicking the SELECT (or UNSELECT) button beneath the OBJECT GROUPS list.

Selecting the Properties to Include

To specify which properties of the selected object types you wish to include in the report, click on the SELECT PROPERTIES... button. This will bring up the Property Selection dialog:



The PROPERTIES list will include all the properties available on at least one of the selected object types. Click the check box next to those you wish to include as columns on the report.

The left listbox shows the list of property page tabs that the properties are normally found on when setting them with Workplace Pro. Selecting one of these tab names limits the PROPERTIES list to only those on that tab. Clicking on the SELECT ALL (or UNSELECT ALL) button selects (or unselects) the properties on that tab.

Not every property of an object will be available for reporting. Specifically, only those items listed as *Persistent* (such as *position* or *alarmText*) in the Niagara documentation are exported to the XML file. Others listed as *Transient* (such as status properties) have no meaning outside of a running station, and are not included in the XML file that the Admin Tool generates.

Filtering on Object Names

An optional method for paring down the list of reported objects to just those you are interested in is to specify one or more filter patterns to be applied to the names of the objects.

You enter the list of filter patterns, separated by commas, in the text area towards the bottom of the form:

The Property Sheet will include any objects whose name matches any of the patterns listed. If you leave the list blank, all objects of the types in the Object Types to Include list will be shown.

As shown in the example, you can include standard wildcard characters in your patterns:

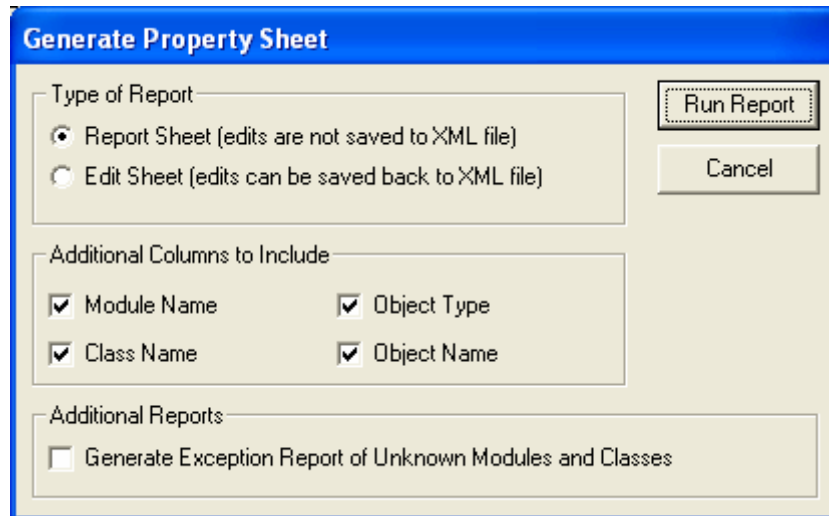
Characters in <i>pattern</i>	Matches in <i>object name</i>
?	Any single character.
*	Zero or more characters.
#	Any single digit (0–9).
[<i>charlist</i>]	Any single character in <i>charlist</i> .
[!<i>charlist</i>]	Any single character not in <i>charlist</i> .

A group of one or more characters (*charlist*) enclosed in brackets ([]) can be used to match any single character in the object name. By using a hyphen (–) to separate the upper and lower bounds of the range, *charlist* can specify a range of characters. For example, “[A-Z]” results in a match if the corresponding character position in the object name contains any uppercase letters in the range A–Z. Multiple ranges are included within the brackets without delimiters. An exclamation point (!) at the beginning of *charlist* means that a match is made if any character except the characters in *charlist* is found in the object name.

Creating the Property Sheet Report

To generate the Property Sheet report, select **GENERATE PROPERTY SHEET...** from the **RUN** menu.

The Generate Property Sheet dialog will be displayed:



For **TYPE OF REPORT**, select whether you want to create a spreadsheet for viewing, sorting and searching for property values (**REPORT SHEET**) or a spreadsheet designed for editing the property values (**EDIT SHEET**).

By default, all reports include basic object information such as **Swid**, **MODULE NAME**, **CLASS NAME**, **OBJECT TYPE** and **OBJECT NAME**. To remove some of these columns, uncheck the appropriate boxes.

To generate an additional text report showing a list of any modules or objects that the Utility did not have in its database, check the **GENERATE EXCEPTION REPORT** box. For more information on the Object Database, see the *Adding to the Module and Object Database* section later in this chapter.

After making these selections, click the RUN REPORT button to generate the spreadsheet:

	A	B	C	D	E	F	G	H	I
1									
2	/YMCA/lonTrunk/C1001/AOIOverrideValue	AnalogOutput	tridium.control.AnalogOutputNode	coreRuntime	AOIOverrideValue	percent		80	1088
3	/YMCA/lonTrunk/C1001/AO2OverrideValue	AnalogOutput	tridium.control.AnalogOutputNode	coreRuntime	AO2OverrideValue	percent		80	1152
4	/YMCA/lonTrunk/C1001/BinaryInput	BinaryInput	tridium.control.BinaryInputNode	coreRuntime	BinaryInput			1088	698
5	/YMCA/lonTrunk/C1001/BinaryInput_1	BinaryInput	tridium.control.BinaryInputNode	coreRuntime	BinaryInput_1			1089	795
6	/YMCA/lonTrunk/C1001/CO2Alarm	BinaryInput	tridium.control.BinaryInputNode	coreRuntime	CO2Alarm		RTU 1 high CO2 alarm.	512	592
7	/YMCA/lonTrunk/C1001/CO2Loop_1	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	CO2Loop_1			608	1279
8	/YMCA/lonTrunk/C1001/CO2Sp	AnalogOutput	tridium.control.AnalogOutputNode	coreRuntime	CO2Sp	parts_per_million		48	160
9	/YMCA/lonTrunk/C1001/CoolLoop_1	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	CoolLoop_1			608	1200
10	/YMCA/lonTrunk/C1001/DATemp	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	DATemp	degrees_Fahrenheit		784	272
11	/YMCA/lonTrunk/C1001/DO1OverrideValue	BinaryOutput	tridium.control.BinaryOutputNode	coreRuntime	DO1OverrideValue			16	656
12	/YMCA/lonTrunk/C1001/DO3OverrideValue	BinaryOutput	tridium.control.BinaryOutputNode	coreRuntime	DO3OverrideValue			16	720
13	/YMCA/lonTrunk/C1001/DO4OverrideValue	BinaryOutput	tridium.control.BinaryOutputNode	coreRuntime	DO4OverrideValue			16	764
14	/YMCA/lonTrunk/C1001/DO5OverrideValue	BinaryOutput	tridium.control.BinaryOutputNode	coreRuntime	DO5OverrideValue			16	848
15	/YMCA/lonTrunk/C1001/Filter	BinaryInput	tridium.control.BinaryInputNode	coreRuntime	Filter		RTU 1 filter alarm.	816	1072
16	/YMCA/lonTrunk/C1001/HeatLoop_1	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	HeatLoop_1			608	1359
17	/YMCA/lonTrunk/C1001/HVValve	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	HVValve	percent		640	174
18	/YMCA/lonTrunk/C1001/LowTemp1	BinaryInput	tridium.control.BinaryInputNode	coreRuntime	LowTemp1		RTU 1 low space Temp.	1072	608
19	/YMCA/lonTrunk/C1001/LowTemp2	BinaryInput	tridium.control.BinaryInputNode	coreRuntime	LowTemp2		RTU 1 low space Temp.	816	1120
20	/YMCA/lonTrunk/C1001/MaDampers	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	MaDampers	percent		640	127
21	/YMCA/lonTrunk/C1001/MaTemp	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	MaTemp	degrees_Fahrenheit		784	224
22	/YMCA/lonTrunk/C1001/CO2Sp	AnalogOutput	tridium.control.AnalogOutputNode	coreRuntime	CO2Sp	degrees_Fahrenheit		48	32
23	/YMCA/lonTrunk/C1001/CO2	AnalogOutput	tridium.control.AnalogOutputNode	coreRuntime	CO2	parts_per_million		784	336
24	/YMCA/lonTrunk/C1001/SfAlarm	BinaryInput	tridium.control.BinaryInputNode	coreRuntime	SfAlarm		RTU 1 supply fan failure.	800	528
25	/YMCA/lonTrunk/C1001/SLinkStatus1	BinaryInput	tridium.control.BinaryInputNode	coreRuntime	SLinkStatus1		RTU 1 space sensor alarm.	768	448
26	/YMCA/lonTrunk/C1001/SLinkStatus2	BinaryInput	tridium.control.BinaryInputNode	coreRuntime	SLinkStatus2		RTU 1 space sensor alarm.	816	1168
27	/YMCA/lonTrunk/C1001/SpaceAlarmSp1	AnalogOutput	tridium.control.AnalogOutputNode	coreRuntime	SpaceAlarmSp1	degrees_Fahrenheit		48	224
28	/YMCA/lonTrunk/C1001/SpaceAlarmSp2	AnalogOutput	tridium.control.AnalogOutputNode	coreRuntime	SpaceAlarmSp2	degrees_Fahrenheit		1040	1088
29	/YMCA/lonTrunk/C1001/SpaceTemp1	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	SpaceTemp1	degrees_Fahrenheit		784	176
30	/YMCA/lonTrunk/C1001/SpaceTemp2	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	SpaceTemp2	degrees_Fahrenheit		816	1216
31	/YMCA/lonTrunk/C1001/UH1Status	BinaryInput	tridium.control.BinaryInputNode	coreRuntime	UH1Status		RTU 1 CO2 sensor alarm.	768	400
32	/YMCA/lonTrunk/C1001/UI2Status	BinaryInput	tridium.control.BinaryInputNode	coreRuntime	UI2Status		RTU 1 mixed air sensor alarm.	896	416
33	/YMCA/lonTrunk/C1001/UI3Status	BinaryInput	tridium.control.BinaryInputNode	coreRuntime	UI3Status		RTU 1 discharge air sensor alarm.	1040	432
34	/YMCA/lonTrunk/C1001/UhoccSp	AnalogOutput	tridium.control.AnalogOutputNode	coreRuntime	UhoccSp	degrees_Fahrenheit		48	96
35	/YMCA/lonTrunk/C1001/SpaceTemp_1	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	SpaceTemp_1	degrees_Fahrenheit		1312	930
36	/YMCA/lonTrunk/C1001/OccSp_1	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	OccSp_1	degrees_Fahrenheit		192	80
37	/YMCA/lonTrunk/C1001/OccSp_2	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	OccSp_2	degrees_Fahrenheit		192	32
38	/YMCA/lonTrunk/C1001/OccSp_3	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	OccSp_3	degrees_Fahrenheit		192	176
39	/YMCA/lonTrunk/C1001/SpaceTemp_2	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	SpaceTemp_2	degrees_Fahrenheit		1312	978
40	/YMCA/lonTrunk/C1001/SpaceTemp_3	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	SpaceTemp_3	degrees_Fahrenheit		1312	1026
41	/YMCA/lonTrunk/C1001/DATemp_1	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	DATemp_1	degrees_Fahrenheit		944	272
42	/YMCA/lonTrunk/C1001/REnable	AnalogOutput	tridium.control.AnalogOutputNode	coreRuntime	REnable	percent		48	288
43	/YMCA/lonTrunk/C1001/DO2OverrideValue	BinaryOutput	tridium.control.BinaryOutputNode	coreRuntime	DO2OverrideValue			16	912
44	/YMCA/lonTrunk/C1001/CO2_1	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	CO2_1	parts_per_million		944	320
45	/YMCA/lonTrunk/C1001/MaDampers_1	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	MaDampers_1	percent		784	128
46	/YMCA/lonTrunk/C1001/CO2Sp_1	AnalogOutput	tridium.control.AnalogOutputNode	coreRuntime	CO2Sp_1	degrees_Fahrenheit		192	128
47	/YMCA/lonTrunk/C1001/MensLockers	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	MensLockers	degrees_Fahrenheit		944	176

Property values that are not set (i.e., left as their default values in WorkPlace Pro) or properties that do not exist on a specific object type will be left blank. Some properties are *compound* types (such as *position*), and will be displayed as two or more columns (*position.x* and *position.y*).

Special Property Values

While most of the property values are strings or numbers, there are some that will be displayed differently:

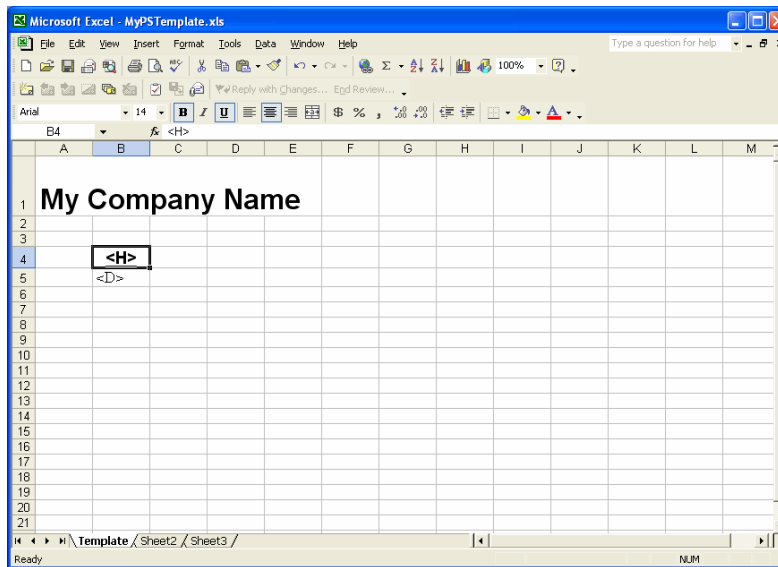
- Properties that represent colors (e.g. *borderColor*) or bit fields are stored as hexadecimal values, and will appear as such in the property sheet.
- The *securityGroups* property will be shown as a string of letters representing the groups that are selected ('G' for General, 'H' for HVAC, 'S' for Security, 'L' for Life Safety, and '4', '5', '6' and '7' for the other generic groups).

Using a Report Template

For Report Sheets, you can generate a more formatted report by using a Report Template. You specify the name of the template file by clicking on the SELECT button in the Report Template section on the bottom of the Property Sheets form.

The Report Template is a regular Excel file containing the prototype report form as the first worksheet. This template can include heading information above the data, as well as other formatting information for the header row and data in the report.

To indicate the upper-left corner of the report (i.e., the heading for the *Swid* column), put an “<H>” into the appropriate cell:



Besides indicating the location of the output, the format characteristics (such as font, color, alignment, etc.) for the column headers in the report are taken from this cell. In the example above, the column headings will now be 14 point and underlined.

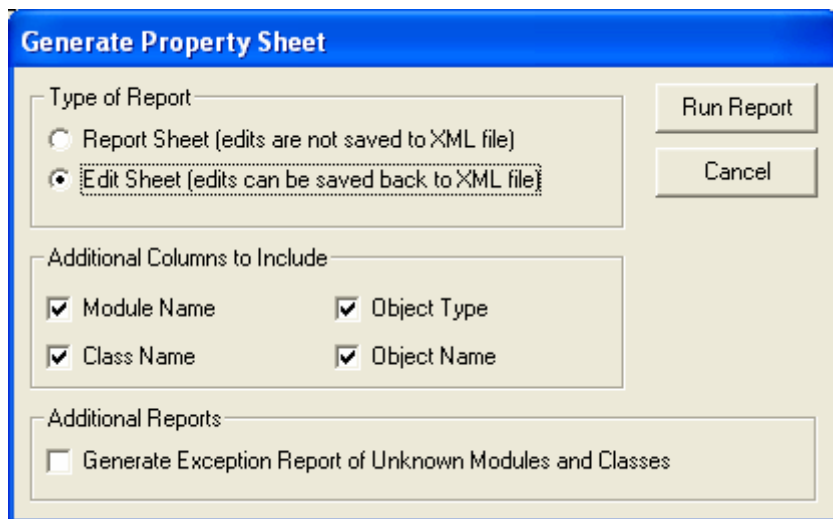
The cell directly below the “<H>” will be used for the formatting characteristics of the data in the table.

Swid	Object Type	Class	Module	Name
/YMCA/lonTrunk/C1001/AO1OverrideValue	AnalogOutput	tridium.control.AnalogOutputNode	coreRuntime	AO1OverrideValue
/YMCA/lonTrunk/C1001/AO2OverrideValue	AnalogOutput	tridium.control.AnalogOutputNode	coreRuntime	AO2OverrideValue
/YMCA/lonTrunk/C1001/BinaryInput	BinaryInput	tridium.control.BinaryInputNode	coreRuntime	BinaryInput
/YMCA/lonTrunk/C1001/BinaryInput_1	BinaryInput	tridium.control.BinaryInputNode	coreRuntime	BinaryInput_1
/YMCA/lonTrunk/C1001/CO2Alarm	BinaryInput	tridium.control.BinaryInputNode	coreRuntime	CO2Alarm
/YMCA/lonTrunk/C1001/CO2Loop_1	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	CO2Loop_1
/YMCA/lonTrunk/C1001/CO2Sp	BinaryOutput	tridium.control.AnalogOutputNode	coreRuntime	CO2Sp
/YMCA/lonTrunk/C1001/CO2Loop_1	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	CO2Loop_1
/YMCA/lonTrunk/C1001/DaTemp	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	DaTemp
/YMCA/lonTrunk/C1001/DO1OverrideValue	BinaryOutput	tridium.control.BinaryOutputNode	coreRuntime	DO1OverrideValue
/YMCA/lonTrunk/C1001/DO3OverrideValue	BinaryOutput	tridium.control.BinaryOutputNode	coreRuntime	DO3OverrideValue
/YMCA/lonTrunk/C1001/DO4OverrideValue	BinaryOutput	tridium.control.BinaryOutputNode	coreRuntime	DO4OverrideValue
/YMCA/lonTrunk/C1001/DO5OverrideValue	BinaryOutput	tridium.control.BinaryOutputNode	coreRuntime	DO5OverrideValue
/YMCA/lonTrunk/C1001/Filter	BinaryInput	tridium.control.BinaryInputNode	coreRuntime	Filter
/YMCA/lonTrunk/C1001/HeatLoop_1	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	HeatLoop_1
/YMCA/lonTrunk/C1001/HtValve	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	HtValve
/YMCA/lonTrunk/C1001/LowTemp1	BinaryInput	tridium.control.BinaryInputNode	coreRuntime	LowTemp1
/YMCA/lonTrunk/C1001/LowTemp2	BinaryInput	tridium.control.BinaryInputNode	coreRuntime	LowTemp2
/YMCA/lonTrunk/C1001/MaDampers	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	MaDampers
/YMCA/lonTrunk/C1001/MaTemp	AnalogInput	tridium.control.AnalogInputNode	coreRuntime	MaTemp

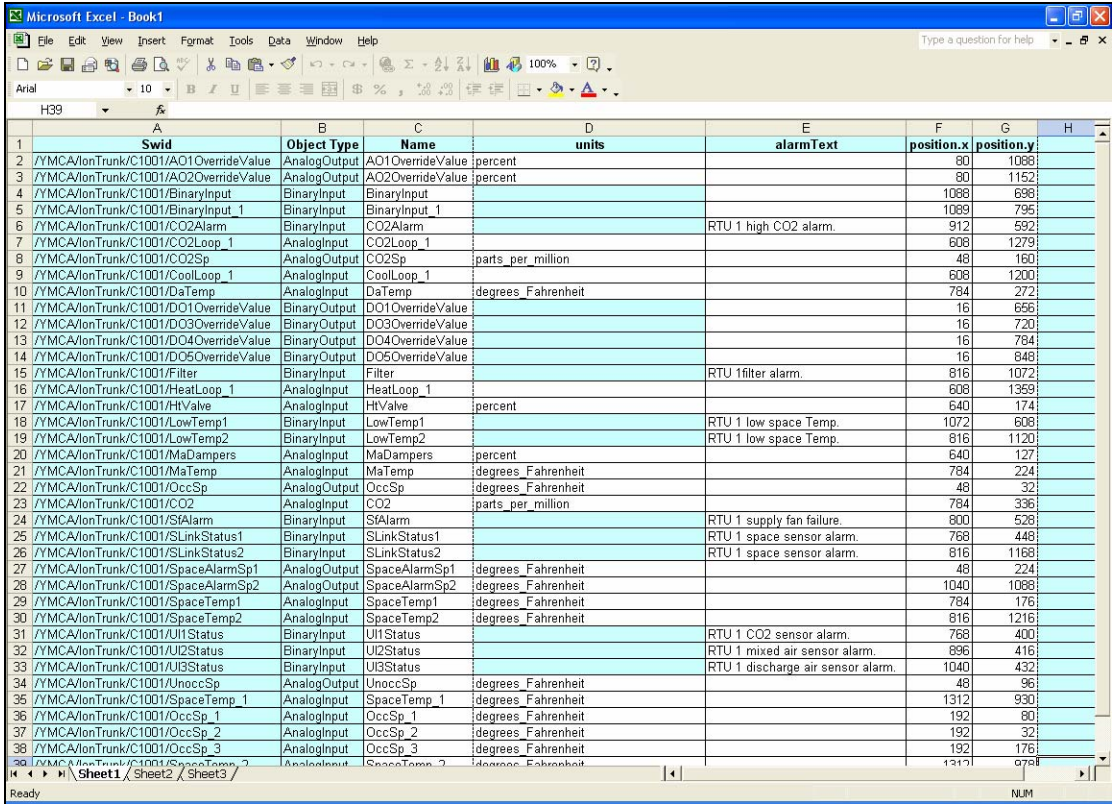
Report files are usually stored in the Reports subdirectory of the WorkPlace Pro Utilities directory (“C:\Program Files\WorkPlace Pro Utilities” by default). You can indicate that you wish to use the default reporting formats by selecting the *Default.xls* file.

Editing Property Values with an Edit Report

One of the most powerful features of the Property Sheets is the ability to edit the properties of multiple objects at the same time. To generate an editable version of the Property Sheet, select the EDIT SHEET radio button when generating the report:

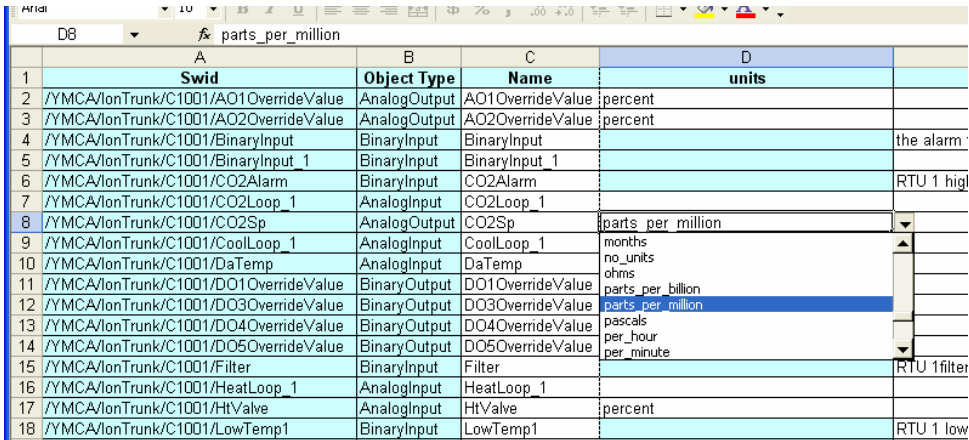


The resulting spreadsheet will appear somewhat different:

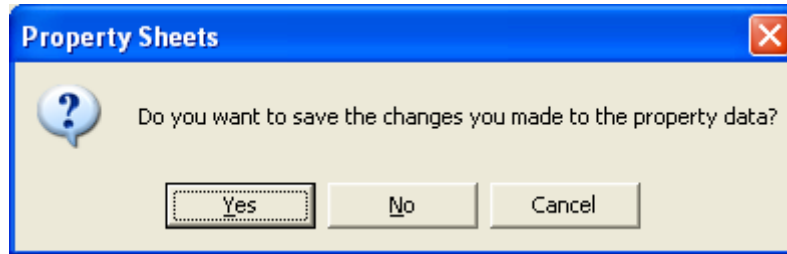


Shaded cells indicate the values that do not exist (i.e., the *units* property for a Binary Input object) or those that are read-only. Cells with a white background can be modified. If you specified a Report Template file for your Property Sheets, it will not be used for the Edit Sheet.

For property values with a specific type (i.e., *boolean*, *int*, or an enumerated type such as *EngUnitsEnum*), you will be limited to entering valid values. Non-numeric values entered into an integer property cell will be rejected. Boolean and enumerated types will have a drop-down list box to select from the appropriate values:



When you have completed your changes, simply close Excel or select RETURN TO WORKPLACE PRO UTILITIES from Excel's FILE menu. The Utility will ask you if you want to save your changes:



If you say YES, the changes you made will be read from Excel and updated in the station's XML file. These changes will not be saved to disk until you select SAVE NIAGARA XML FILE from the FILE menu. You will also be prompted to save the XML file when you exit the Utility.

Unprotecting the Edit Report

When the Property Sheet Utility generates an Edit Sheet in Excel, the sheet is protected to prevent accidental changes that may prevent it from reading the information back in. This protection has the side effect of disabling certain editing features, such as Replace, and makes others somewhat more limited. To remove this restriction, you can simply unprotect the sheet by selecting PROTECTION from the TOOLS menu and then selecting UNPROTECT SHEET.

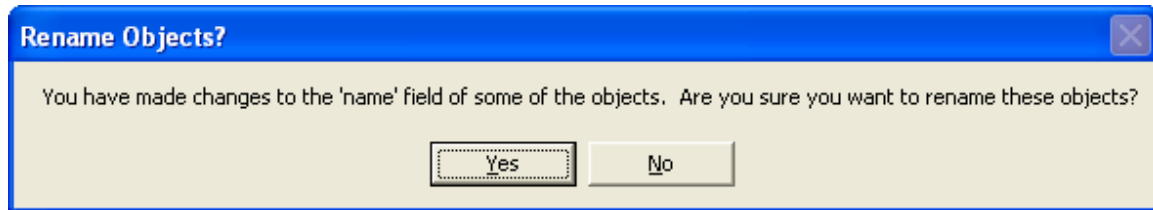
If you unprotect the sheet, you must not delete, reorder or insert between any of the rows or columns generated by the utility. *Doing so will cause the station file to be corrupted when the values are read back in.* You can, however, use the rows or columns outside the generated area of the sheet for temporary calculations, as those cells are ignored.

Checking for Invalid Property Values

While the Utility will limit you to valid values based on the defined data species and types of the object properties, it will not validate other restrictions such as valid data ranges or other rules that may exist between two or more property values. These errors will be reported by WorkPlace Pro's Admin Tool when you reimport the XML file. The error messages it presents should be sufficient to diagnose the problem. You can then reopen the file in the Property Sheet Utility, make the appropriate change, and reattempt the import.

Renaming Objects with an Edit Sheet

If you include the Name property on your Edit Sheet, it will be displayed as an editable column. You can then rename your objects by simply changing these values. When you close the Edit Sheet and agree to save your changes, you will be prompted with a separate confirmation regarding the name changes:



The name changes will be applied to the objects and any links within the station will be updated as well. However:

- External links from other stations to this object will not be changed.
- Log and audit records regarding this object will not be changed.

Changes to the names of higher level objects such as shadow objects and the station object could have unexpected results, especially on commissioned shadow objects and stations that are active. Caution is recommended in these cases.

Loading and Saving the WPS File

You can save the configuration of your report (the name of the source XML file, the selected container, the list of object types, the list of properties and the name of the Report Template) as a file (with a .WPS extension) by selecting **SAVE WPS FILE** from the **FILE** menu. You can then reload the settings by selecting **OPEN WPS FILE** from the **FILE** menu.

If there is a file called **DEFAULT.WPS** located in the directory where the application was installed ("C:\Program Files\WorkPlace Pro Utilities" by default), this file will automatically be loaded when you start the Property Sheet Utility. To create this file, simply run the utility, make the desired selections and save the WPS file to **DEFAULT.WPS**. It is a better idea, however to save to a different configuration file and explicitly load it as a template before creating a new Property Sheet report.

You do not have to specify all the settings to create a WPS file. For example, you can create a template that has the list of objects and properties you use frequently, but do not select the XML file or container. When you load this template, simply select the XML file and run the report.

Running a Batch Report

There may be situations where you want to run a specific report on property values from a stations on a regular, automated basis. WorkPlace Pro Utilities accepts command line arguments that allow you to have it launch the program, open a WPS file, generate a Report Sheet, save it to a file and then automatically shut itself down. This is particularly useful when specified as part of a batch file.

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:

```
WPUutilities.exe Report.wpt OutputFile.xls
```

The first argument is the name of the WorkPlace Pro Utilities executable file. The second argument should be the full path of the Property Sheets WPT file that contains the report parameters. The third argument is the full name of the file to which the report will be saved (existing data will be overwritten).

For example:

```
C:\Program Files\WorkPlace Pro Utilities\WPUutilities.exe  
"C:\My Reports\NightlyReport.wpt" "F:\Public\Reports\Station Report Data.xls"
```

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 and output file names, including drive letters. If either path contains a space anywhere within it, you must enclose the entire path in double quotes.

Adding to the Module and Object Database

To properly report and edit the properties of an object, the Utility needs to know certain information about the objects. WorkPlace Pro organizes its objects into modules that are stored in *jar files*. Examples include *coreRuntime*, *lonworks* and *lonSiebe*.

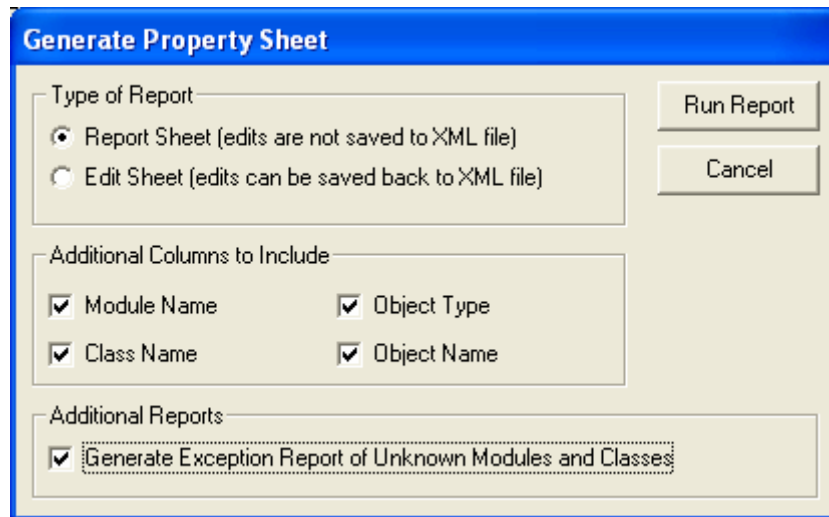
In WorkPlace Pro Utilities, the relevant information from each module is stored in a corresponding Jar File Map, such as *coreRuntime.wpj*. This file is a straightforward XML file containing the details of the module's enumerations, data types, objects and their properties. These map files are created by Smartware Technologies, although they can be edited and expanded on by advanced users.

The Jar File Map files are stored in the JarFileMaps subdirectory of the WorkPlace Pro Utilities directory ("C:\Program Files\WorkPlace Pro Utilities" by default). Updated and additional .WPJ files should be placed in this directory, from where they will be loaded automatically when the Utility is launched.

Issues with Missing Modules and Objects

If an object or its module is unknown, it will not be presented in the list of Object Types to report. Besides the obvious case where you try to report or edit properties for one of these unknown object types, a report that tries to include all object types will be incorrect and the omission harder to notice.

To check to see if any of the objects in your station file are unknown, simply check the GENERATE EXCEPTION REPORT OF UNKNOWN MODULES AND CLASSES checkbox when generating the report:



Adding a Jar File Map for a New Module

The Property Sheet Utility includes a few basic modules that represent the vast majority of objects found in a typical station. There are, however, many other modules, most of which are specific to a manufacturer or controller line. If you need additional module files, please contact us at techsupport@smartwaretech.com, and we will be happy to generate them for you. Updated releases of the Utility will include additional module files as well.

It is also possible for advanced users to create the mapping file themselves. The .WPJ file is a simple XML file that contains the following information for a module taken from its documentation:

For each enumeration:

- The name of the enumeration (e.g. *ExecutionOrderEnum*)
- For each possible value, the text description (e.g. *input*, *processor* or *output*) and the corresponding integer value (which is not used by the Property Sheet Utility)

For each compound data type:

- The name of the type (e.g. *SnvtObjRequestType*)
- A list of the fields that make up the type (e.g. *statusFlags*, *objectId* and *objectRequest*)
- For each field, an indication if its access level (read-only or read/write) and its data type species (a primitive type, such as *boolean* or *int*, or the name of an enumeration or compound type).

For each object:

- The name of the object (e.g. *GxPage*)
- The fully qualified class of the object (e.g., *tridium.gx.GxPage*)
- A list of the object's properties

For each object property:

- The tab name on which the property is found (e.g. *AlarmSetup*)
- The name of the property (e.g. *alarmText*)
- The attributes of the property, as described in Appendix A of the *Niagara Standard Programming Reference*. (e.g. {rwP:Ad})
- The data type species of the property.

Refer to the .WPJ files included with WorkPlace Pro Utilities as an example of how to create your own. It is not necessary to include all the objects in a module's map file, especially if you are attempting to report on just a few. Simply include only the relevant objects and properties in the .WPJ file.

For more information on issues relating to these module maps, please contact our technical support team.

8. The Profile Upgrade Utility

The Profile Upgrade Utility helps automate the process of swapping out shadow objects and recreating links in a Niagara station when it becomes necessary to replace one or more controllers using the Version 2 profile with comparable Version 3 controllers.

Specifically, the following upgrades can be made:

<u>Existing Profile</u>	<u>Converted To</u>
MNLRF2	MNLRF3
MNLRH2	MLNRH3
MNLRR2	MNLRR3
MNLRS1	MNLRS3
MNLRS2	MNLRS4
MNLRV2	MNLRV3

The Upgrade Process

To upgrade a controller, follow these general steps:

1. Update the *lonSiebe* jar files on the UNC or Enterprise Server containing the controller.
2. Export the station to an XML file using the *Backup and Transport Utility* or the WorkPlace Pro Admin Tool.
3. Use the Profile Upgrade Utility to convert the shadow object within the XML file, as described in the section *Using the Utility*.
4. Replace the physical controller.
5. Download the controller application from WorkPlace Tech 4.0 into the new controller.
- 6a. For UNC-600/610s:
 - Stop the running station..
 - Using the WorkPlace Pro Admin Tool, do a “Put” of the updated XML station file.
 - Restart the station.
- 6b. For all other UNC's:
 - Install the new station using the WorkPlace Pro Admin Tool.

7. Open up the LON Device Manager.
8. Find the new, unconfigured controller, select it, and click the COMMISSION button. You will be prompted for the Neuron ID of the new controller.

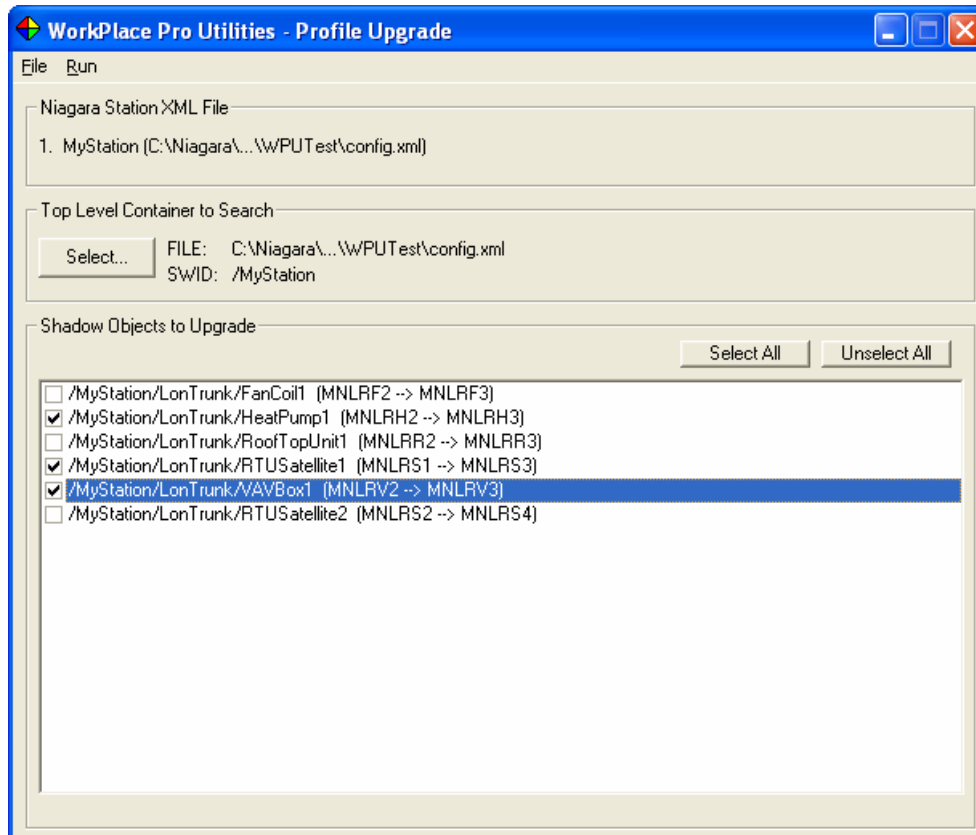
NOTE: *After commissioning, DO NOT perform an Upload of the new controller until after rebinding. Doing so will delete all the current bindings to the controller in the station.*

9. Open up the LON Link Manager and click the BIND button.
10. Verify that the new controller is working properly

Using the Utility

The Profile Upgrade Utility is simple to use:

- From the FILE menu, select OPEN NIAGARA XML FILE. Select the appropriate *config.xml* file.
- By default the utility will automatically scan the entire station looking for controllers that can be upgraded and present them in the SHADOW OBJECTS TO UPGRADE list. You can narrow down this list by clicking the SELECT... button and selecting a different TOP LEVEL CONTAINER TO SEARCH.
- In the SHADOW OBJECTS TO UPGRADE list, check the controller or controllers you wish to upgrade.
- From the RUN menu, select CONVERT SHADOW OBJECTS.
- You will automatically be prompted to resave the XML file when the conversion is complete.



The Upgrade Process

The Utility will perform all the necessary actions to effect the upgrade, including:

- Replacing the old shadow object class object with the new class object
- Copying, updating and replacing the appropriate SNVTs and their properties
- Updating the addressing information from the old shadow object to the new shadow object
- Moving the links from the old shadow object to the new shadow object
- Warning about any possible unexpected consequences of the upgrade and how to resolve them

9. Using the Station Monitoring Utilities

Where the first half of this manual has discussed the various utilities designed to reduce the time it takes to design and build Niagara stations, this second part will cover the utilities that are dedicated to the monitoring and maintenance of Niagara stations that are already running.

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.

The Station Address List

WorkPlace Pro Utilities 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.

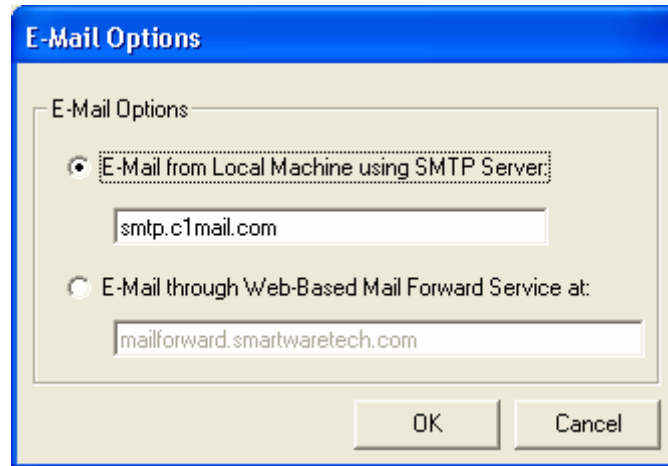
You can change the station addresses as often as you like. The limitations reflect only the number of simultaneous addresses you can have stored for automated use. If you need to support additional stations concurrently, please contact Smartware Technologies directly about upgrading your account.

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 Station Monitoring 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. Workplace Pro Utilities 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

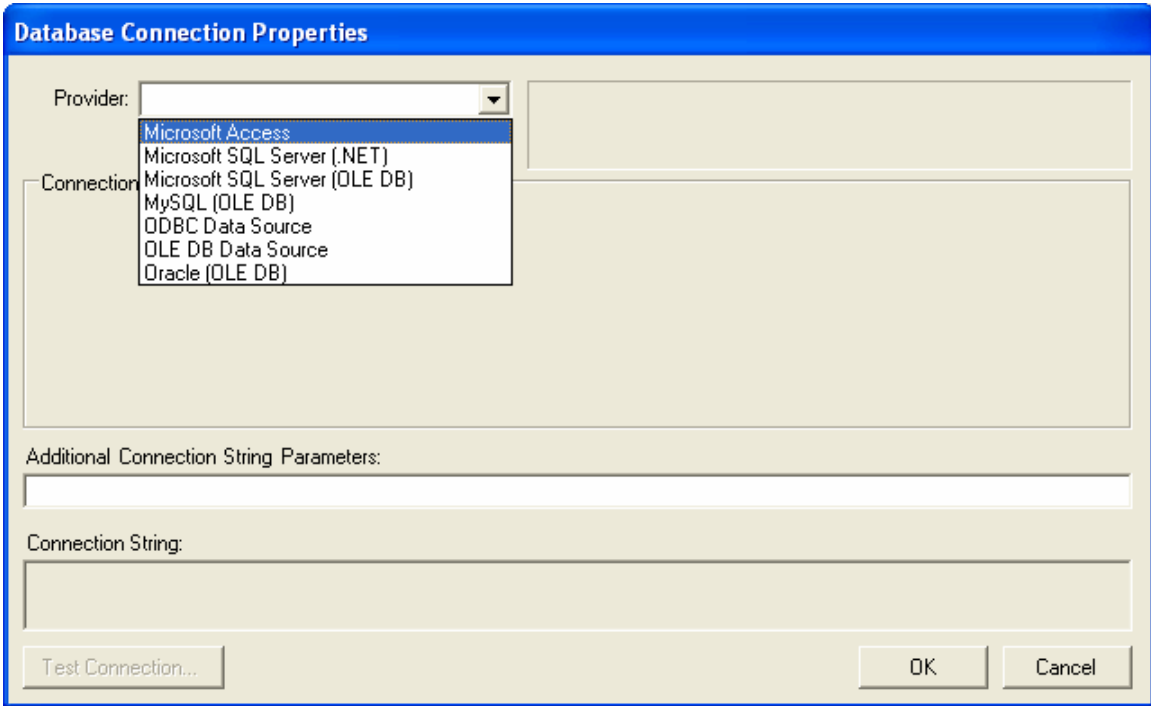
The software used to connect to and interface with a database system is commonly referred to as a provider. All the providers eventually connect to the database with parameters specified in a *Connection String*. There are currently three categories of database providers in use with Windows systems:

- 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. WorkPlace Pro Utilities 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 WorkPlace Pro Utilities, 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 WorkPlace Pro Utilities.
- 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

WorkPlace Pro Utilities 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.

The screenshot shows the 'Database Connection Properties' dialog box. At the top, the 'Provider' is set to 'Microsoft Access'. Below this, there is a text box for 'The name of a Microsoft Access Database (.MDB) File.' The 'Connection Parameters' section includes a 'File Name' field with the path 'C:\MyData\CollectedData.mdb', a 'Browse...' button, and a 'New...' button. There is also a 'Workgroup Information Filename' field with a 'Browse...' button. Below these are 'User ID' and 'Password' fields. The 'Additional Connection String Parameters' section is empty. The 'Connection String' field contains the text 'Provider=Microsoft.Jet.OLEDB.4.0;Data Source=C:\MyData\CollectedData.mdb'. At the bottom, there are 'Test Connection...', 'OK', and 'Cancel' buttons.

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.

Database Connection Properties

Provider: Microsoft SQL Server (.NET) A connection to Microsoft SQL Server (Version 2000, 7.0 or later) through native .NET providers.

Connection Parameters

Data Source: MyServer Specify either (a) The name of the SQL Server (use 'local' for this machine) or (b) the IP Address and Port Number (probably 1433) for a remote server.

Server Name: MyServer

IP Address: Port #: 1433 Network Library: DBMSSOCN

Initial Catalog: myDatabaseName The name of the database

Use Integrated Security User ID: Password:

Additional Connection String Parameters:

Connection String:
Data Source=MyServer;Initial Catalog=myDatabaseName;Integrated Security=SSPI

Test Connection... OK Cancel

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:

Database Connection Properties

Provider: A connection to a MySQL Database through an OLE DB connection.

Connection Parameters

Data Source: The name of the MySQL Database

Initial Catalog: (Not Generally Used)

User ID: Password: Provider:

Additional Connection String Parameters:

Connection String:

Data Source This is usually the name of the database. The naming conventions vary from database system to system.

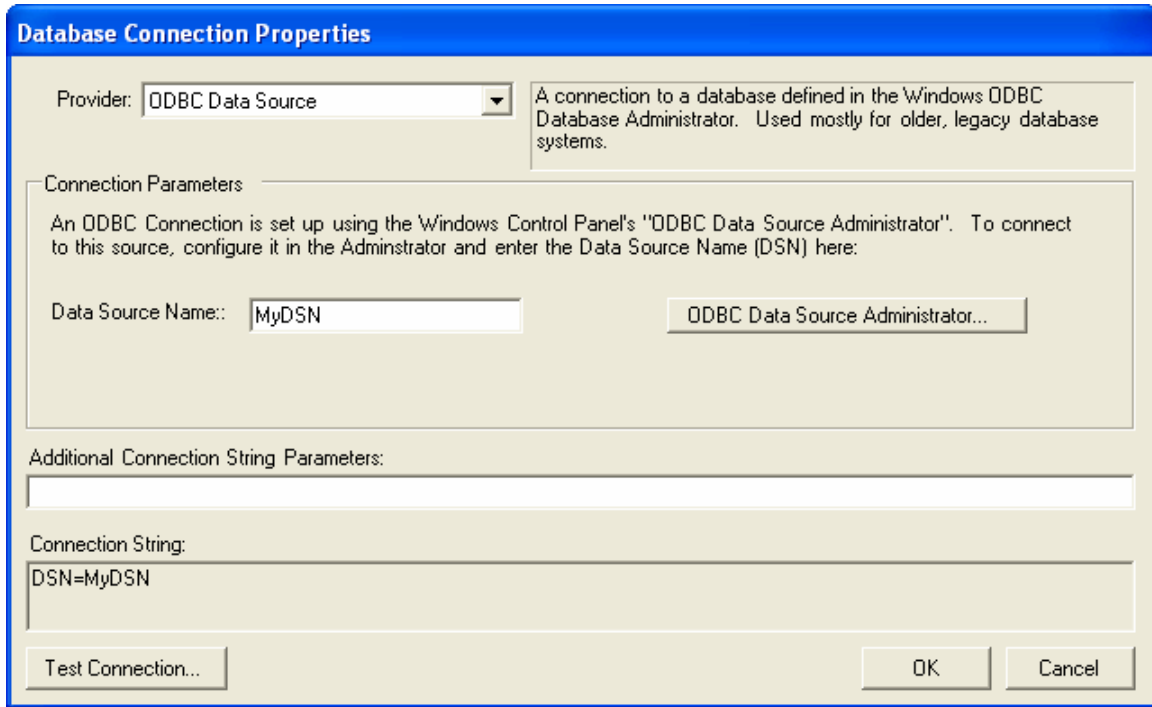
Initial Catalog For some systems, such as Microsoft SQL Server, this is the actual name of the database (with the *Data Source* field being used to identify the server machine instead). For many other providers, this field does not need to be provided.

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:

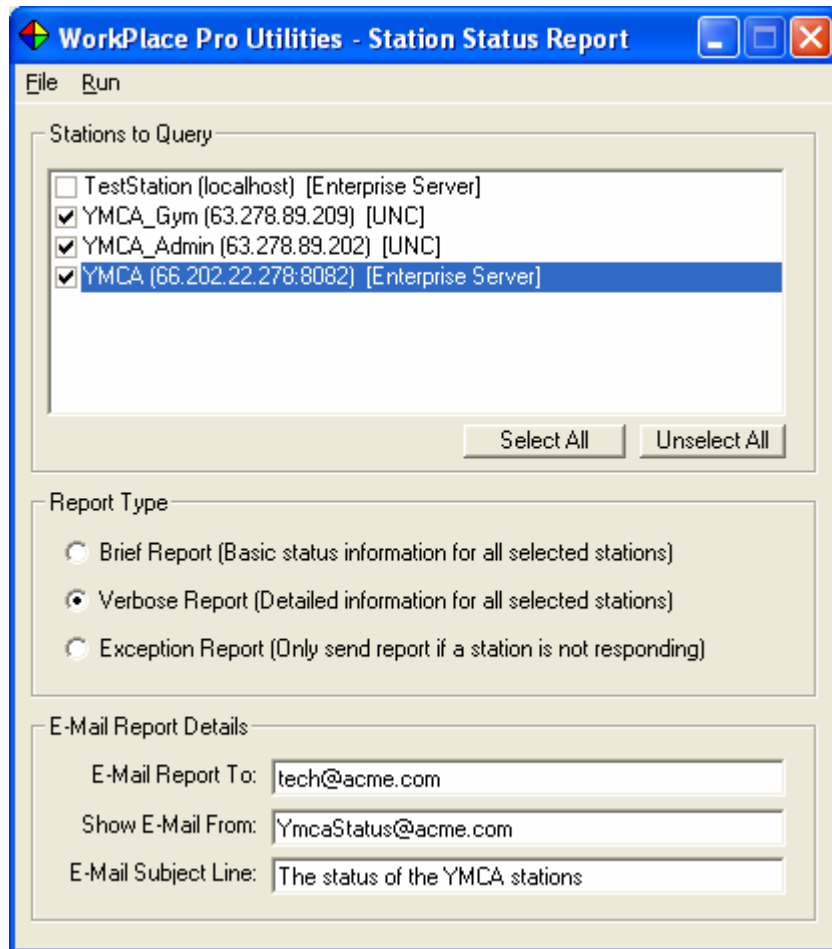
```
Pooling=true;Min Pool Size=0;Max Pool Size=100
```

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.

10. 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.



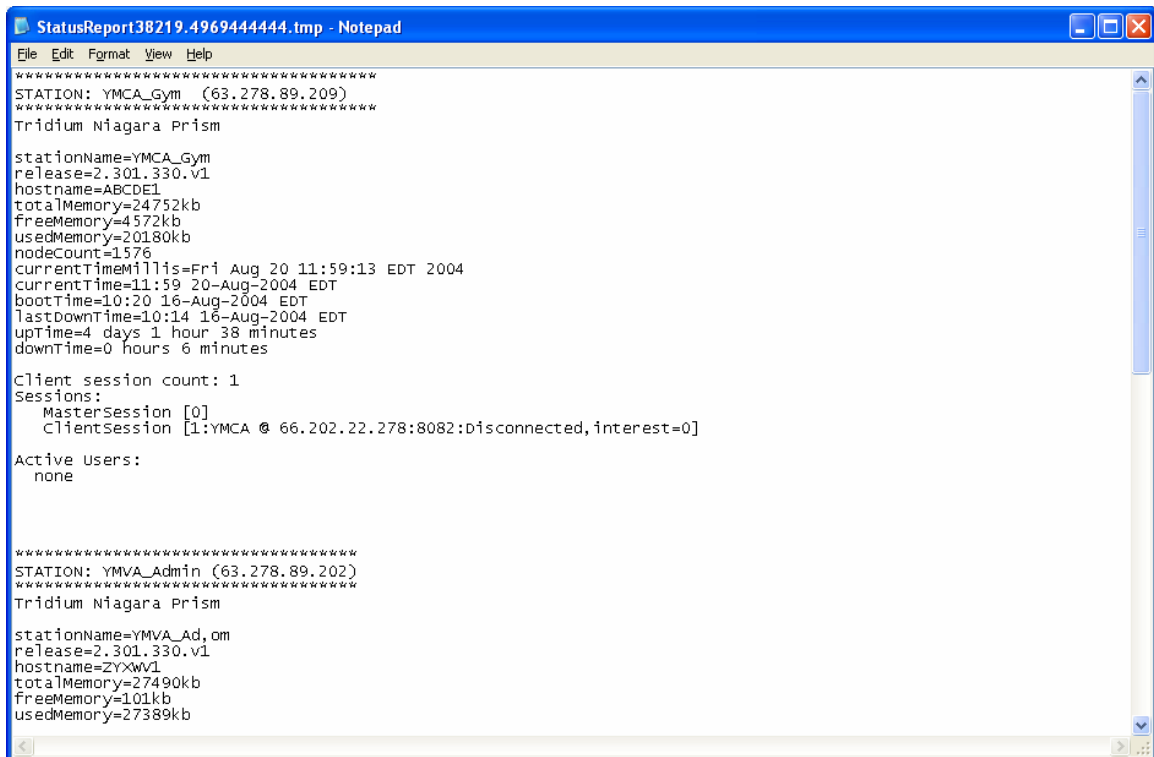
Defining a Report

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 <i>Brief</i> and <i>Verbose</i> reports, which provide response data for each station queried, or an <i>Exception</i> 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 <i>Using the Station Monitoring Utilities</i> . |

Running a Status Report

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



```
File Edit Format View Help
*****
STATION: YMCA_Gym (63.278.89.209)
*****
Tridium Niagara Prism

stationName=YMCA_Gym
release=2.301.330.v1
hostname=ABCDE1
totalMemory=24752kb
freeMemory=4572kb
usedMemory=20180kb
nodeCount=1576
currentTimeMillis=Fri Aug 20 11:59:13 EDT 2004
currentTime=11:59 20-Aug-2004 EDT
bootTime=10:20 16-Aug-2004 EDT
lastDownTime=10:14 16-Aug-2004 EDT
uptime=4 days 1 hour 38 minutes
downtime=0 hours 6 minutes

Client session count: 1
Sessions:
  MasterSession [0]
  ClientSession [1:YMCA @ 66.202.22.278:8082:Disconnected, interest=0]

Active Users:
  none

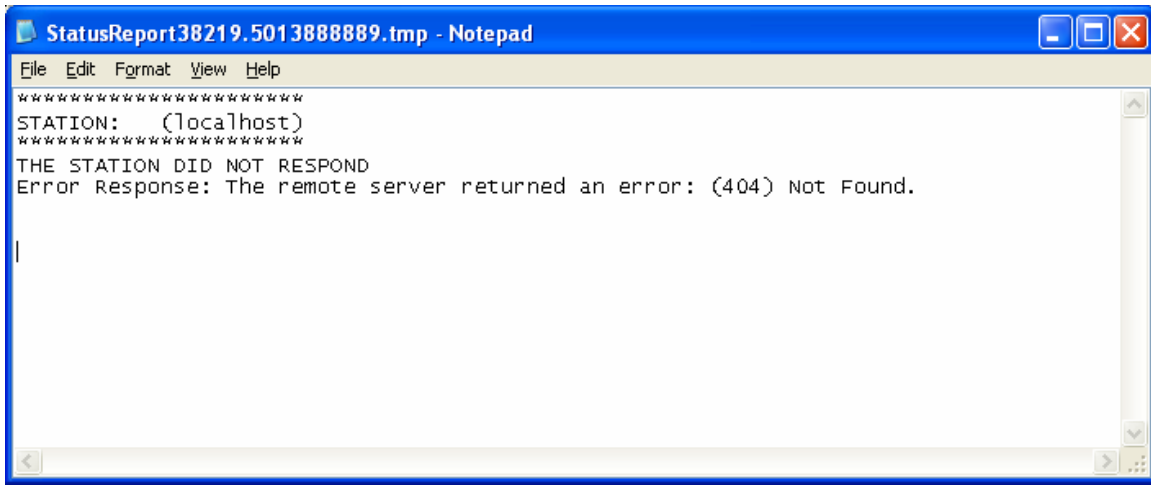
*****
STATION: YMVA_Admin (63.278.89.202)
*****
Tridium Niagara Prism

stationName=YMVA_Ad,om
release=2.301.330.v1
hostname=ZYXWV1
totalMemory=27490kb
freeMemory=101kb
usedMemory=27389kb
```

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.



```

StatusReport38219.501388889.tmp - Notepad
File Edit Format View Help
*****
STATION: (localhost)
*****
THE STATION DID NOT RESPOND
Error Response: The remote server returned an error: (404) Not Found.
|

```

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. WorkPlace Pro Utilities 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:

```
WPUutilities.exe Report.wpq RunReport
```

The first argument is the name of the WorkPlace Pro Utilities 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 be the single word "RunReport".

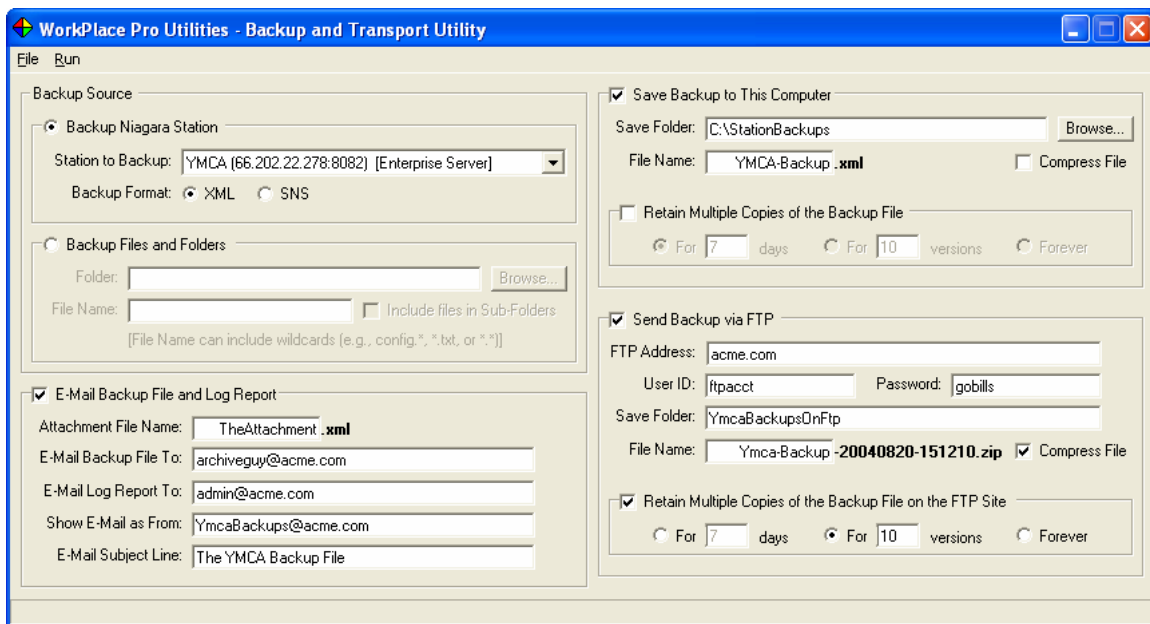
For example:

```
C:\Program Files\WorkPlace Pro Utilities\WPUutilities.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.

11. 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 Niagara 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 Niagara Station

To backup a Niagara 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 Station Monitoring 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 WorkPlace Pro Utilities, refer to the earlier chapter *Using the Station Monitoring 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. WorkPlace Pro Utilities 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:

```
WPUutilities.exe BackupInfo.wpb RunBackup
```

The first argument is the name of the WorkPlace Pro Utilities 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 be the single word “RunBackup”.

For example:

```
C:\Program Files\WorkPlace Pro Utilities\WPUutilities.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.

12. 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.

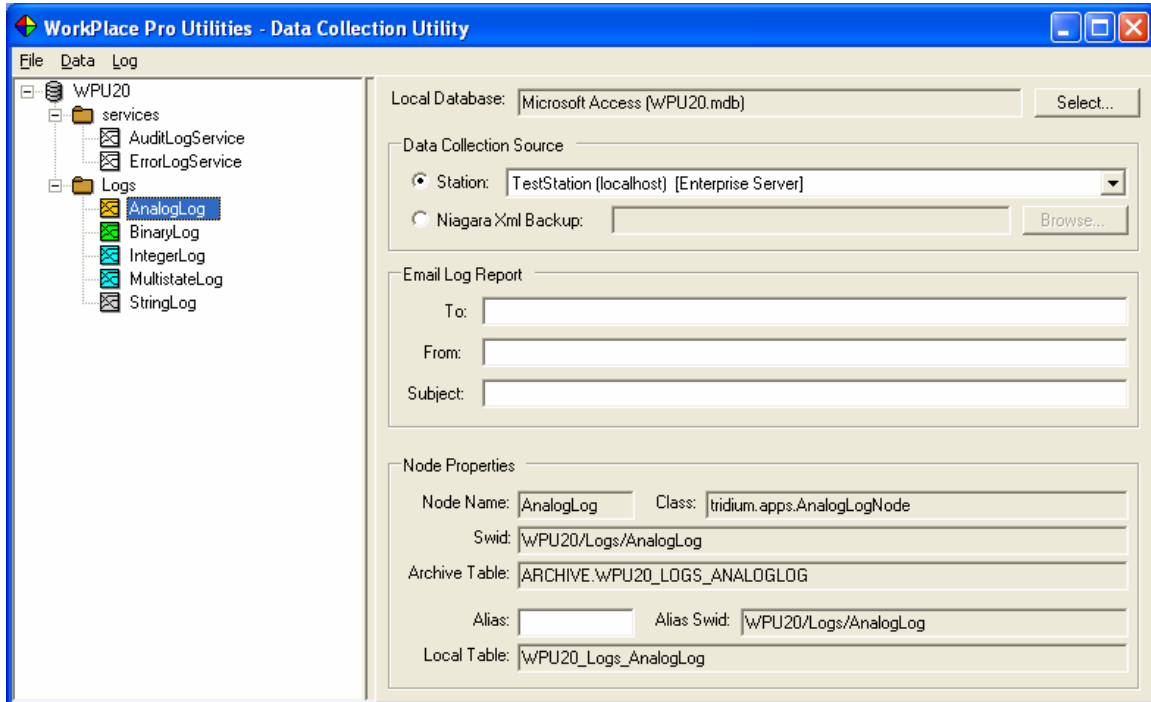
The *Data Collection Utility* can also serve to collect data from stations that are not being archived by WorkPlace Pro by scanning the log data directly from station backup files. This can provide the same archiving capabilities as an Enterprise Server in situations where none exists.

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 Station Monitoring 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 screenshot shows the "Database Connection Properties" dialog box. The "Provider" dropdown is set to "Microsoft Access". The "File Name" field contains "C:\Data\MyLocalData.mdb" and has "Browse..." and "New..." buttons. The "Workgroup Information Filename" field has a "Browse..." button. The "User ID" and "Password" fields are empty. The "Additional Connection String Parameters" field is empty. The "Connection String" field contains "Provider=Microsoft.Jet.OLEDB.4.0;Data Source=C:\Data\MyLocalData.mdb". At the bottom, there are "Test Connection...", "OK", and "Cancel" buttons.

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.

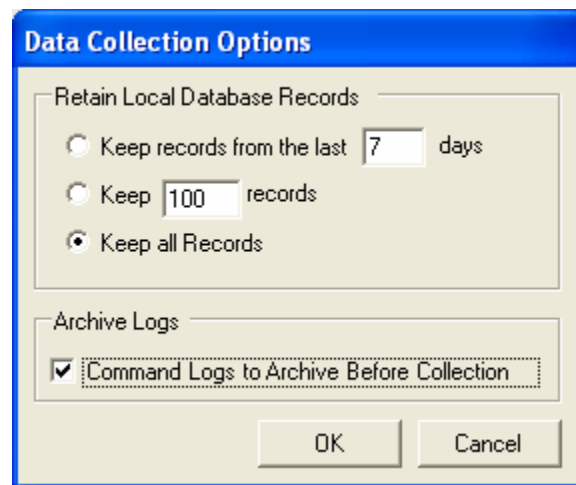
Selecting the Data Collection Source

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 Station Monitoring 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*).

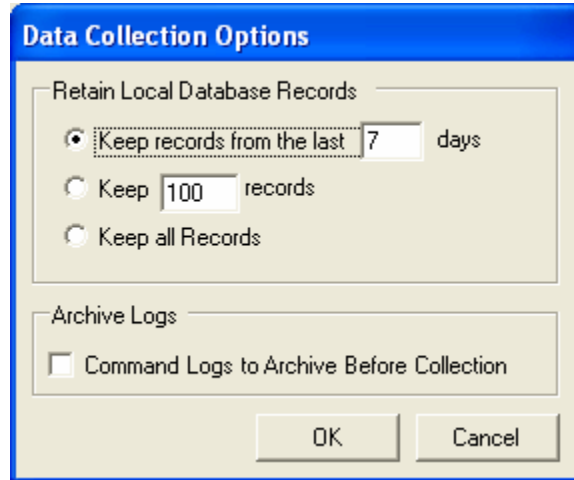
This has a few important implications. Since the station backup files contain the most recent and unarchived records, if you collect from this XML file first and then from the archive database, you will not receive any of the archived records (since they are by definition older than those in the log). 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 Utility* 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:

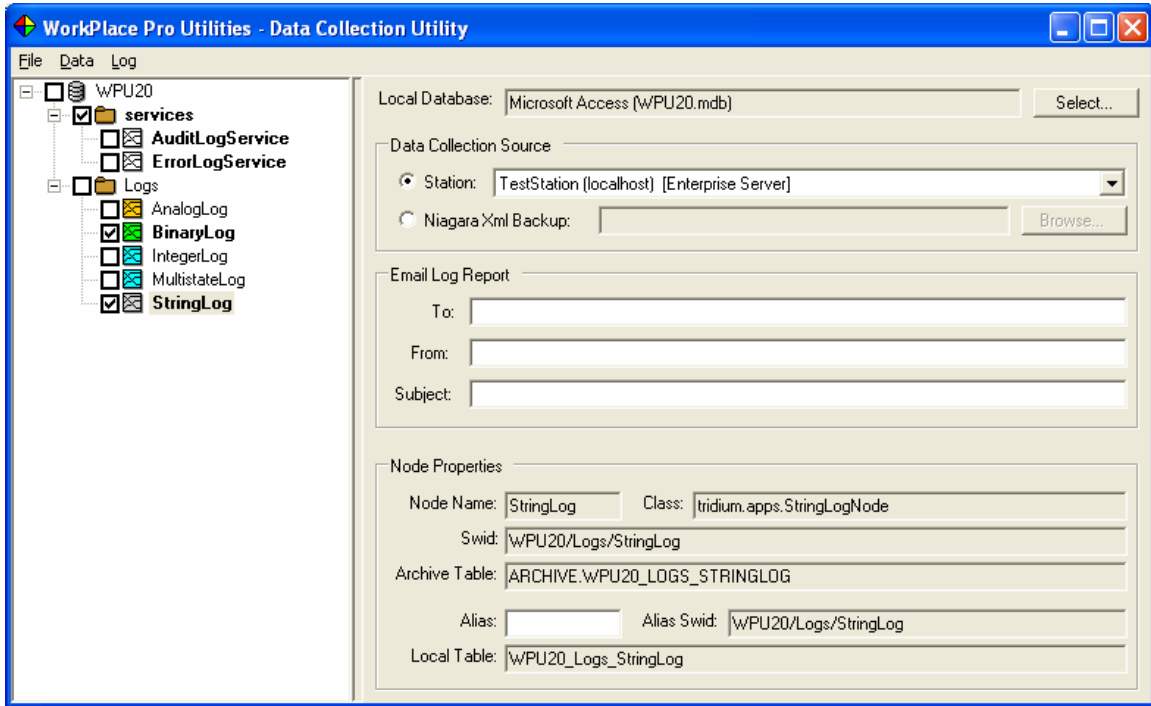


You can choose to keep records for a specific number of days, keep a specific number of records or simply retain all records indefinitely. This setting applies to all tables in the local database, and will be applied to each log the next time the data is collected.

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:



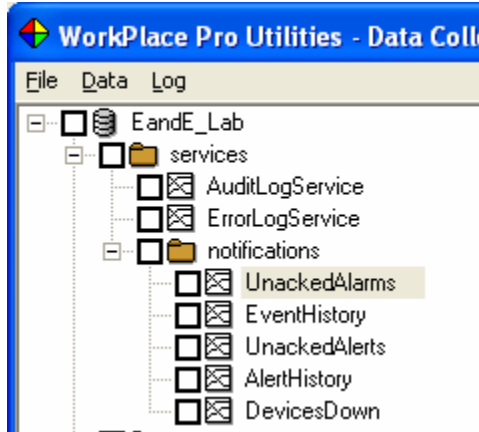
You can choose to collect from a specific log by checking its check box. The name of the log will appear in bold to indicate that it is selected. You can also choose to select all the logs within a container by checking the box next to the container name. In this case the container and all the logs will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



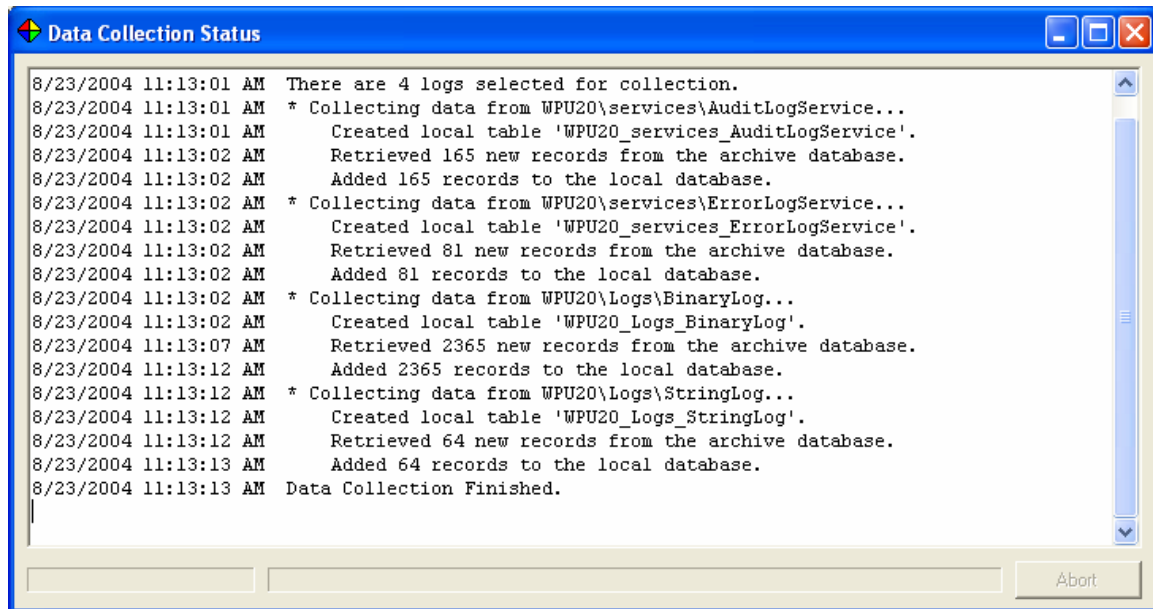
There are five tables that may exist within the station and the archive database:

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

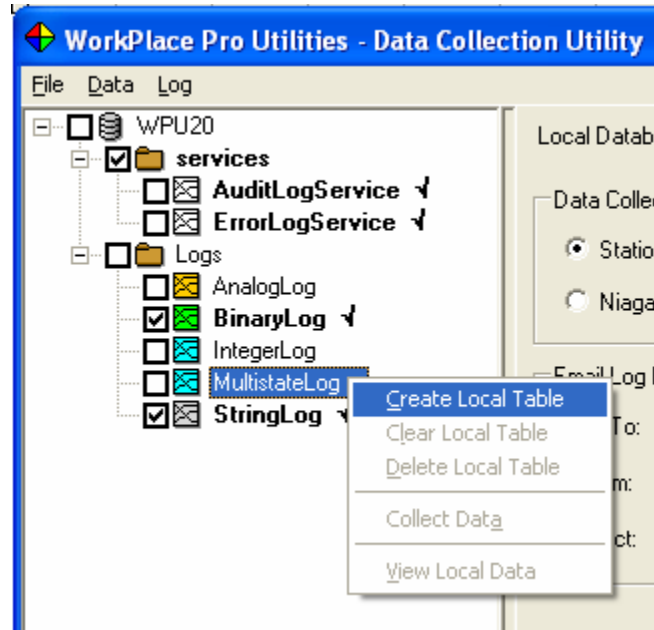
Once you have begun collecting data, you can use the *Data Collection Utility* to view and manipulate the data in the local database directly on a log-by-log basis. These commands are available by selecting a log in the tree and using the commands in the LOG menu. You can also right-click on the log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

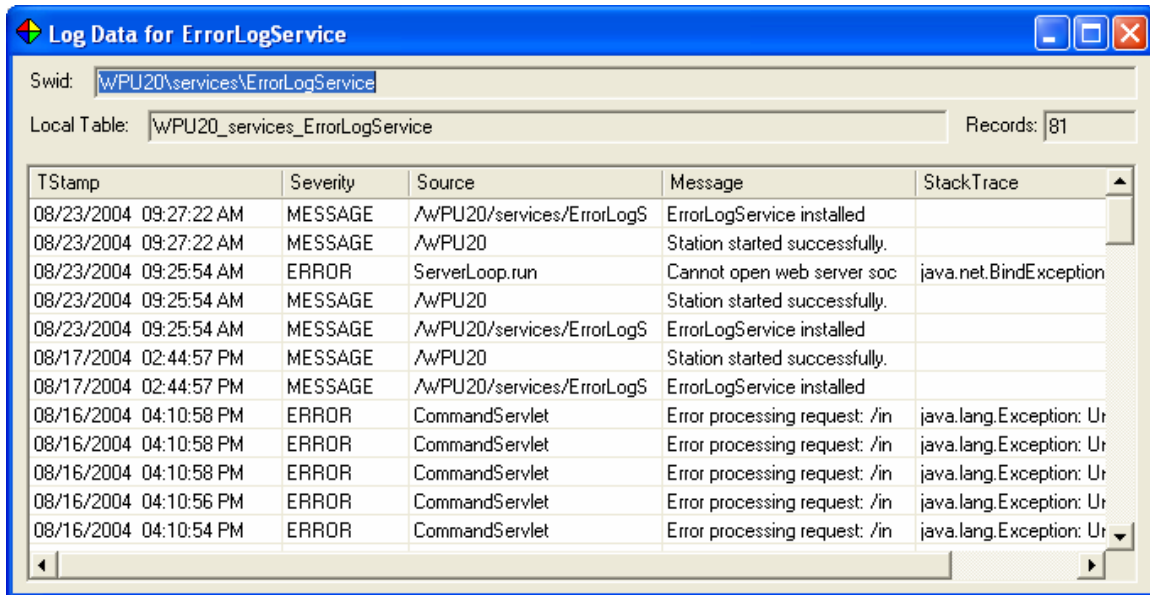
The LOG menu (or pop-up menu) contains commands to manipulate the data in the local database table. If the table does not yet exist (or if the local database hasn't been scanned for existing tables), the only option will be to CREATE LOCAL TABLE.



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:



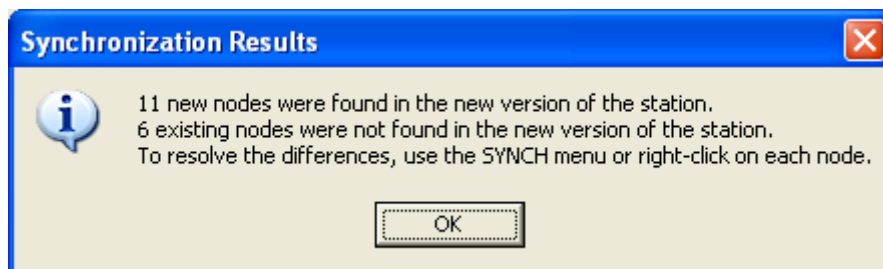
The screenshot shows a window titled "Log Data for ErrorLogService" with a table of log entries. The table has five columns: TStamp, Severity, Source, Message, and StackTrace. The entries include messages about the service being installed and started, as well as errors related to opening a web server and processing requests.

TStamp	Severity	Source	Message	StackTrace
08/23/2004 09:27:22 AM	MESSAGE	/WPU20/services/ErrorLogS	ErrorLogService installed	
08/23/2004 09:27:22 AM	MESSAGE	/WPU20	Station started successfully.	
08/23/2004 09:25:54 AM	ERROR	ServerLoop.run	Cannot open web server soc	java.net.BindException
08/23/2004 09:25:54 AM	MESSAGE	/WPU20	Station started successfully.	
08/23/2004 09:25:54 AM	MESSAGE	/WPU20/services/ErrorLogS	ErrorLogService installed	
08/17/2004 02:44:57 PM	MESSAGE	/WPU20	Station started successfully.	
08/17/2004 02:44:57 PM	MESSAGE	/WPU20/services/ErrorLogS	ErrorLogService installed	
08/16/2004 04:10:58 PM	ERROR	CommandServlet	Error processing request: /in	java.lang.Exception: Ur
08/16/2004 04:10:58 PM	ERROR	CommandServlet	Error processing request: /in	java.lang.Exception: Ur
08/16/2004 04:10:58 PM	ERROR	CommandServlet	Error processing request: /in	java.lang.Exception: Ur
08/16/2004 04:10:56 PM	ERROR	CommandServlet	Error processing request: /in	java.lang.Exception: Ur
08/16/2004 04:10:54 PM	ERROR	CommandServlet	Error processing request: /in	java.lang.Exception: Ur

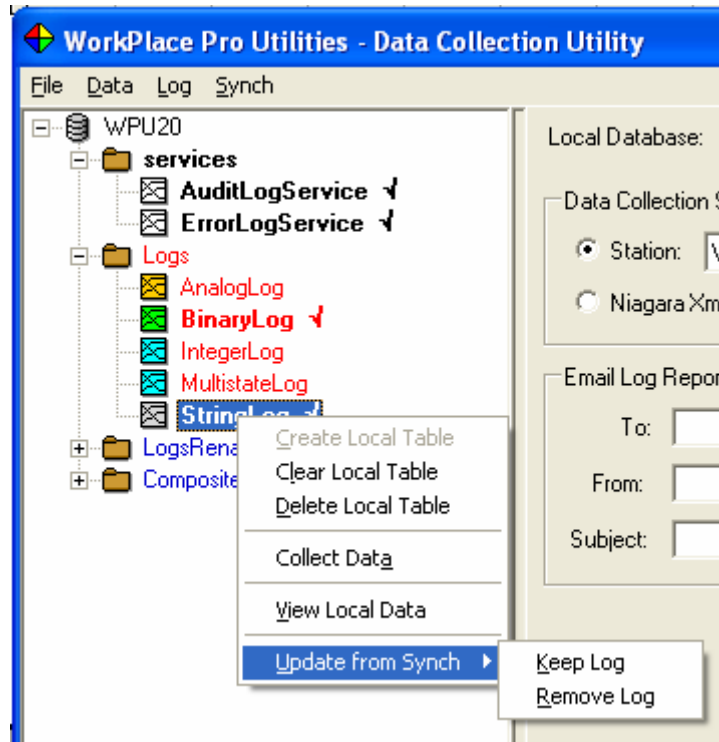
Synchronizing to an Updated Station Tree

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

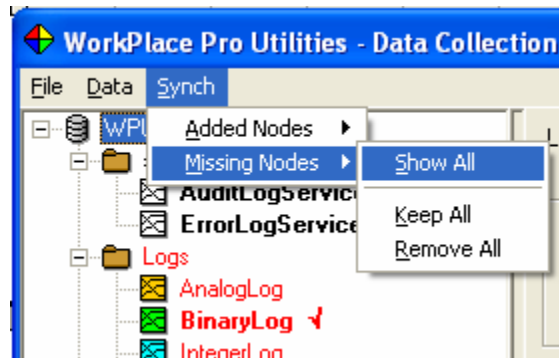
You can choose to **INCLUDE THE CONTAINER** itself, **INCLUDE THE CONTAINER AND ALL LOGS** within it (including sub-containers), or **REMOVE THE CONTAINER** (and therefore all sub-containers and logs)

Missing Containers

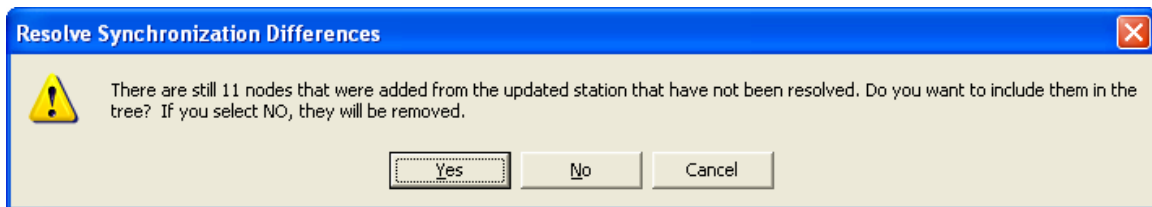
You can choose to **KEEP THE CONTAINER** itself, **KEEP THE CONTAINER AND ALL LOGS** within it (including sub-containers), or **REMOVE THE CONTAINER** (and therefore all sub-containers and logs)

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. WorkPlace Pro Utilities 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:

```
WPUutilities.exe CollectionInfo.wpc CollectData
```


The first argument is the name of the WorkPlace Pro Utilities 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 be the single word "CollectData".

For example:

```
C:\Program Files\WorkPlace Pro Utilities\WPUutilities.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.

13. 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

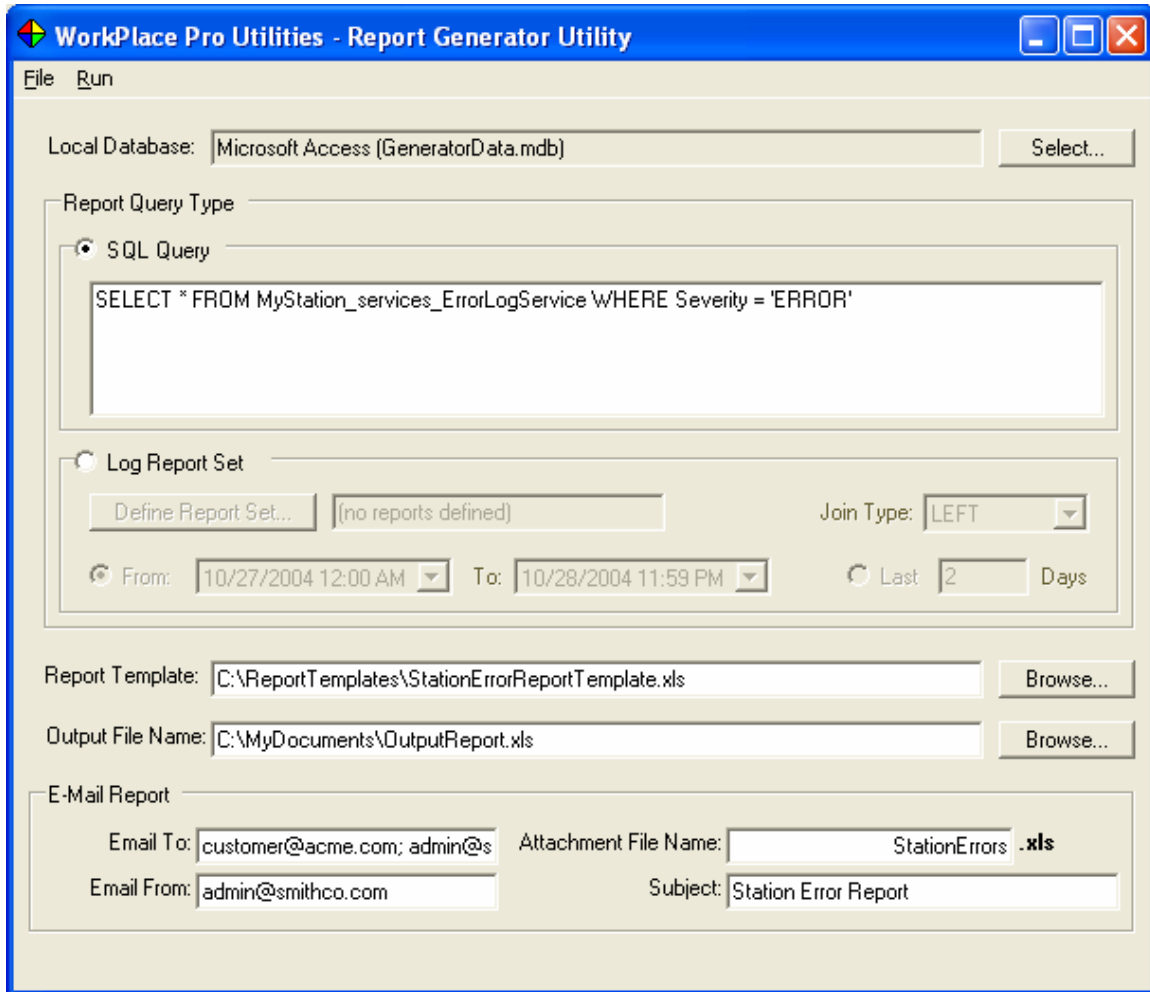
There are two types of reports that can be generated:

- | | |
|----------------|---|
| 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 Station Monitoring 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:

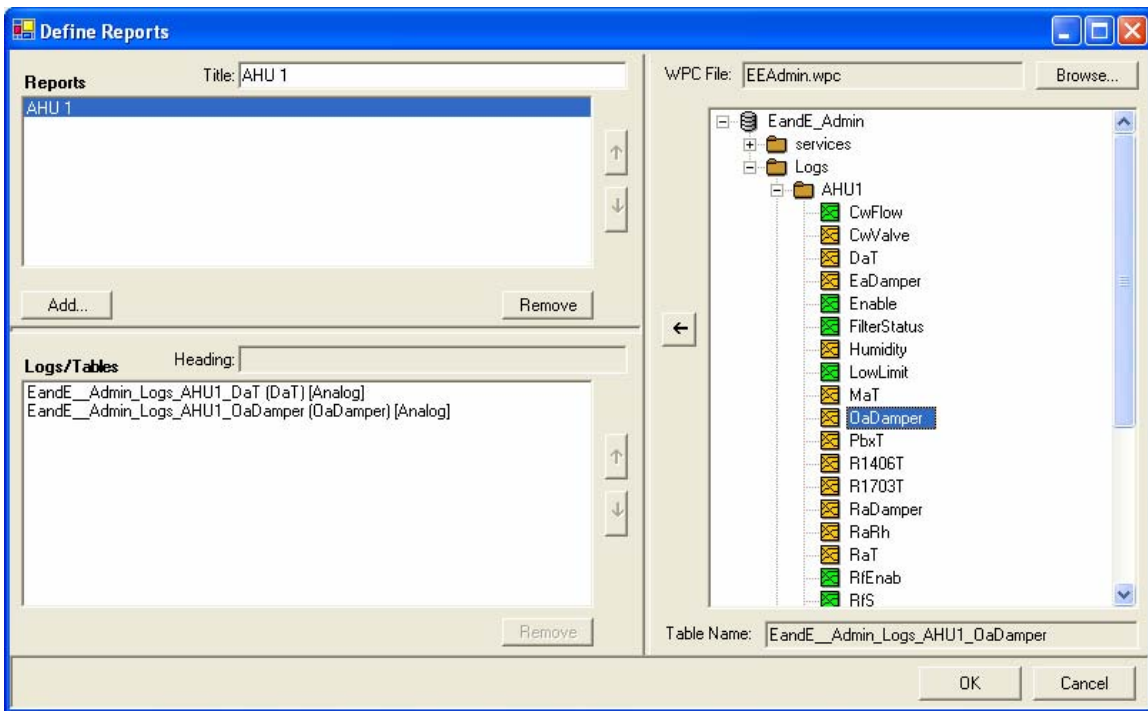
- FROM / TO** To specify a specific date and time range on which to report, enter the values in the FROM and TO boxes.
- LAST N DAYS** To generate a report on the most recent N days, check the appropriate radio button and enter the number of days.
- JOIN TYPE** The Log Reports will combine log records 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.

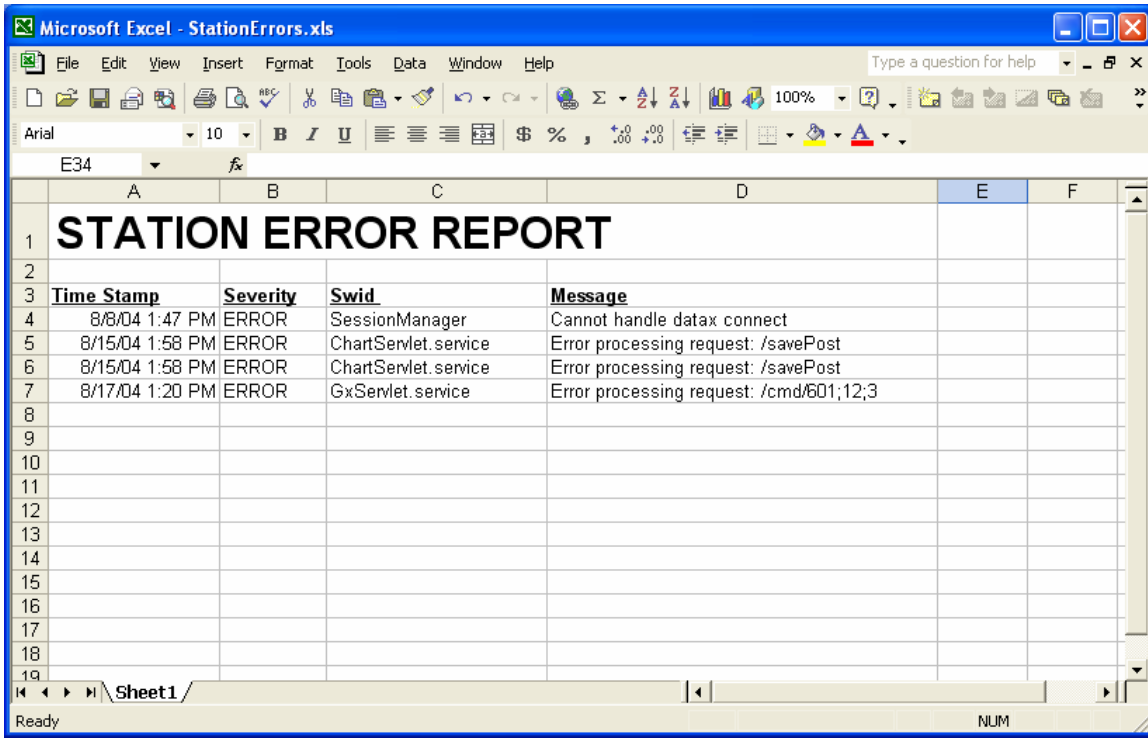
Using the Log Report Set Template

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.



	A	B	C	D	E	F
1	STATION ERROR REPORT					
2						
3	Time Stamp	Severity	Swid	Message		
4	8/8/04 1:47 PM	ERROR	SessionManager	Cannot handle datax connect		
5	8/15/04 1:58 PM	ERROR	ChartServlet.service	Error processing request: /savePost		
6	8/15/04 1:58 PM	ERROR	ChartServlet.service	Error processing request: /savePost		
7	8/17/04 1:20 PM	ERROR	GxServlet.service	Error processing request: /cmd/601;12;3		
8						
9						
10						
11						
12						
13						
14						
15						
16						
17						
18						
19						

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\WorkPlace Pro Utilities” 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 delimiters:

```
#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).

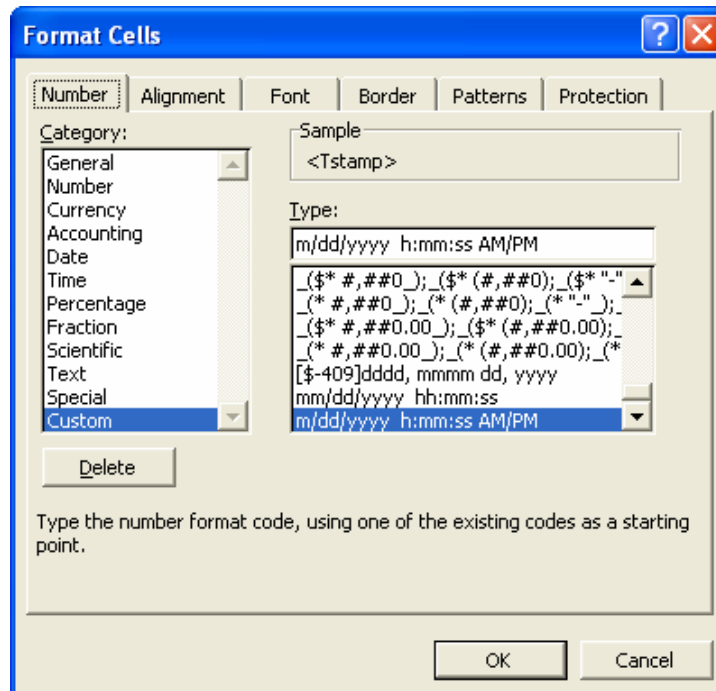
When field codes are used in the header section, the value for the field is taken from the first record generated by the query. The simplest way to get a constant value into the query results is to add it to the select statement:

```
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.



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.

Stock/Vendor	Quantity	Part Number	Price	Part Description	Price
<D>	<VendorShort>	<Qty>	<PartNumber>	<Price>	<Description>
					#/VALUE!
					#NAME?

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. WorkPlace Pro Utilities 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:

```
WPUutilities.exe ReportDefinition.wpr RunReport
```

The first argument is the name of the WorkPlace Pro Utilities 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 be the single word "RunReport".

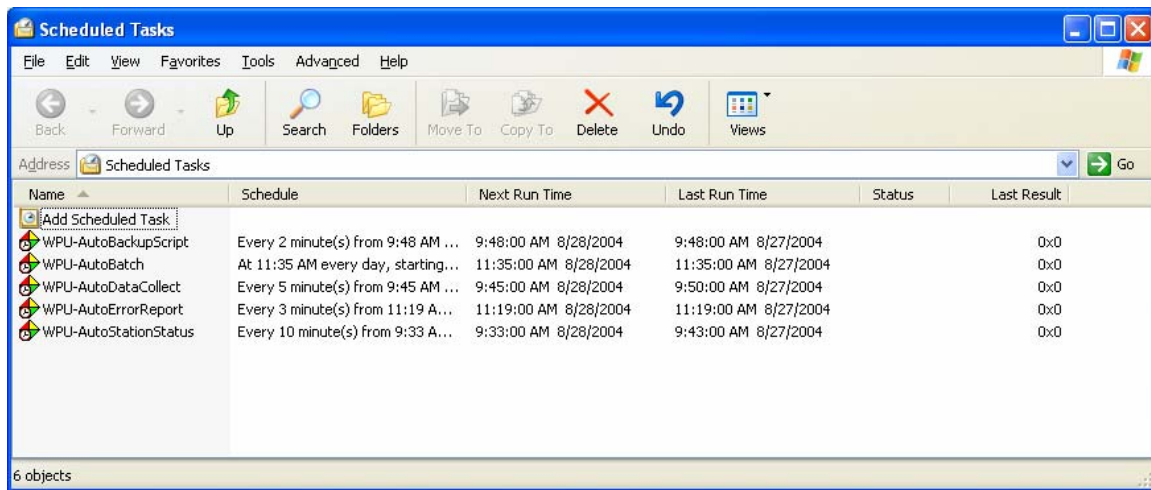
For example:

```
C:\Program Files\WorkPlace Pro Utilities\WPUutilities.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.

14. 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.



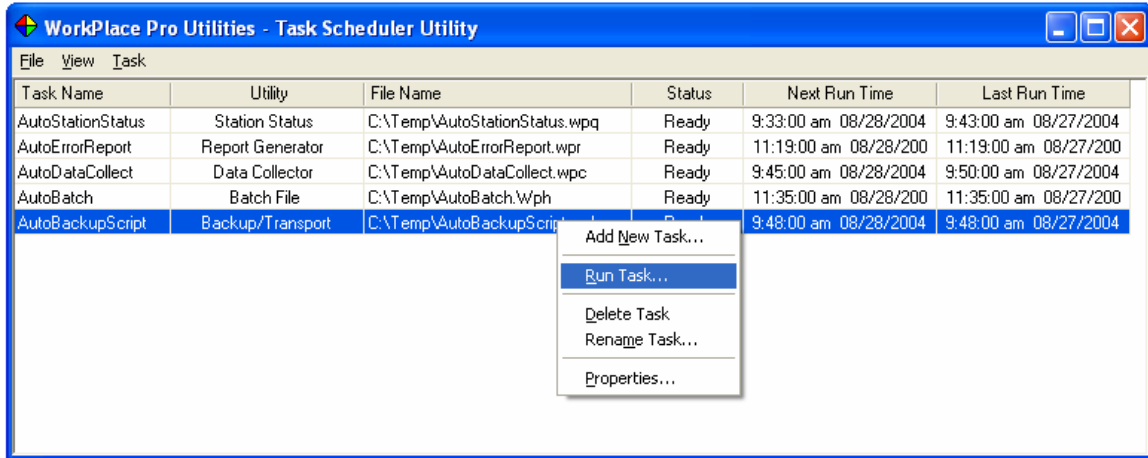
The Scheduled Tasks window, which is normally accessed from the Windows Control Panel, provides a way to set up different *events*, each of which represents a Windows program to run and a schedule of when to run it. The scheduling capabilities are very flexible, and include the ability to run an event 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 it 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 (*WPUUtilities.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:



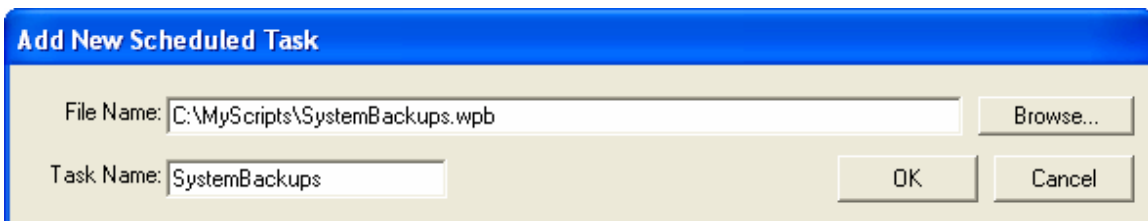
You can do most of the same actions in this window as you can from the Windows Scheduled Tasks Window, such as adding, deleting or running a task, by right-clicking on an item or choosing the command from the TASK menu. You can also access the PROPERTIES window, which allows you to specify the various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

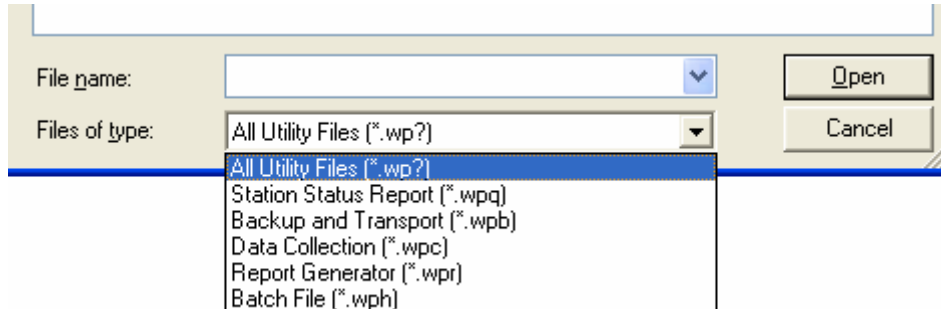
- The TASK NAME, which is used to uniquely identify a task, is actually prefixed with the string “WPU-“ in the Windows Scheduled Tasks. This allows the *Task Scheduler Utility* to show only those tasks related to one of the utilities.
- The *Task Scheduler Utility* window shows the name of the UTILITY, 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.

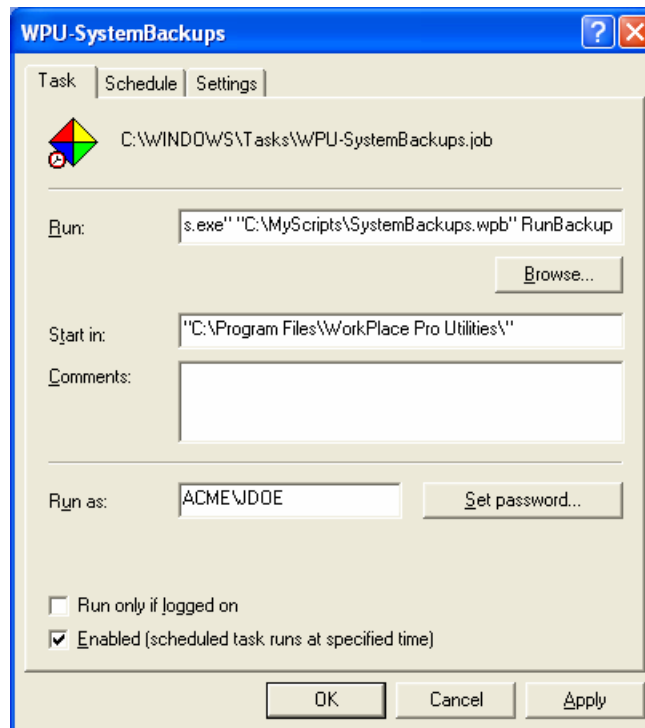


Click the BROWSE... button to select any .WPx file representing the task to be scheduled. The FILES OF TYPE filter at the bottom can be used to make it easier to find a file for a specific utility:



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.

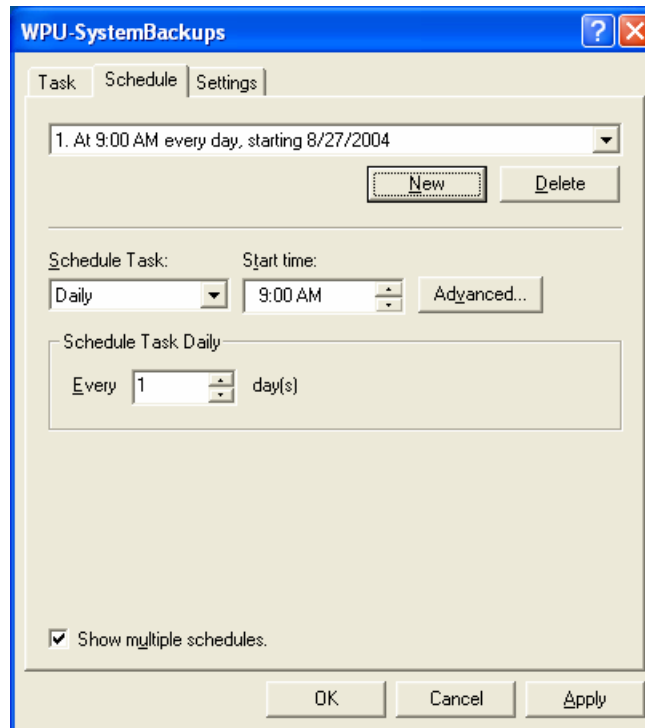
After you click OK, you will automatically be shown the Scheduled Tasks properties window:



You can see that the command line (under RUN) includes the full path to the WorkPlace Pro Utilities executable file (*WPUUtilities.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”.

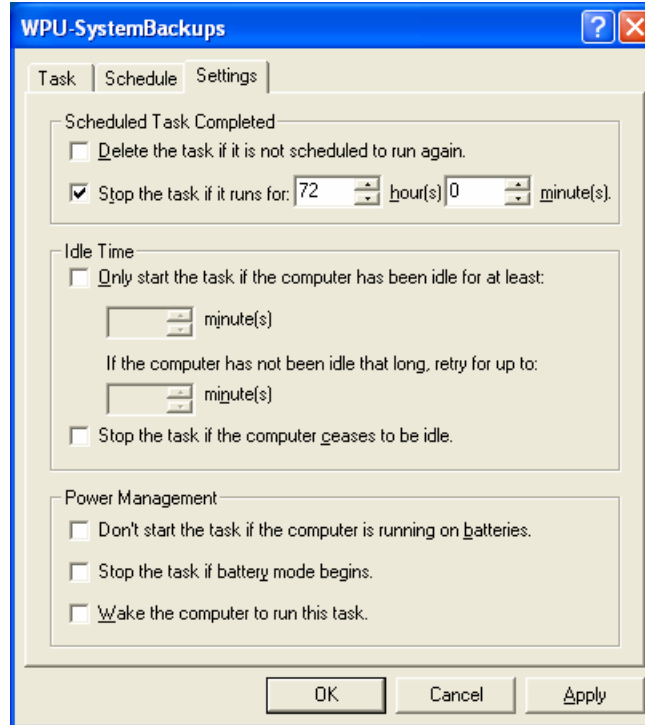
On this TASK tab, you can set the login account information (under RUN AS) and the password (by clicking SET PASSWORD...). On Windows XP, you could instead check the RUN ONLY IF LOGGED ON checkbox and omit the account information. Also note the ENABLED checkbox, which is the easiest way to enable and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

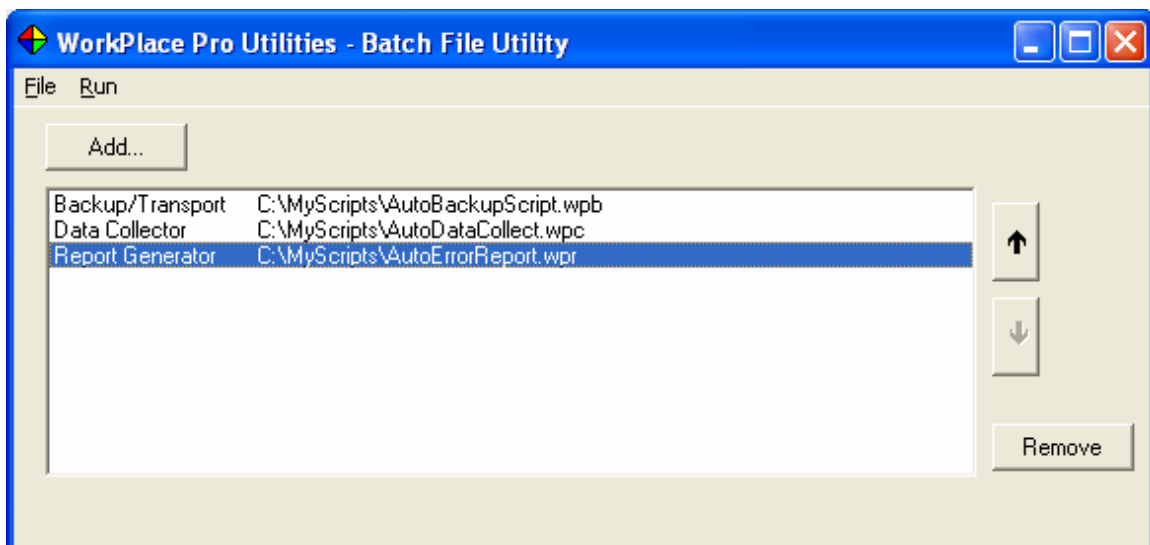
If you need to run multiple tasks in sequence, such as collecting data from a station and then generating a report from that data, you should combine the events into a single batch file using the *Batch File Utility*, and then schedule the Batch File (.WPH File) to run at a specified time. This ensures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

15. The Batch File Utility

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

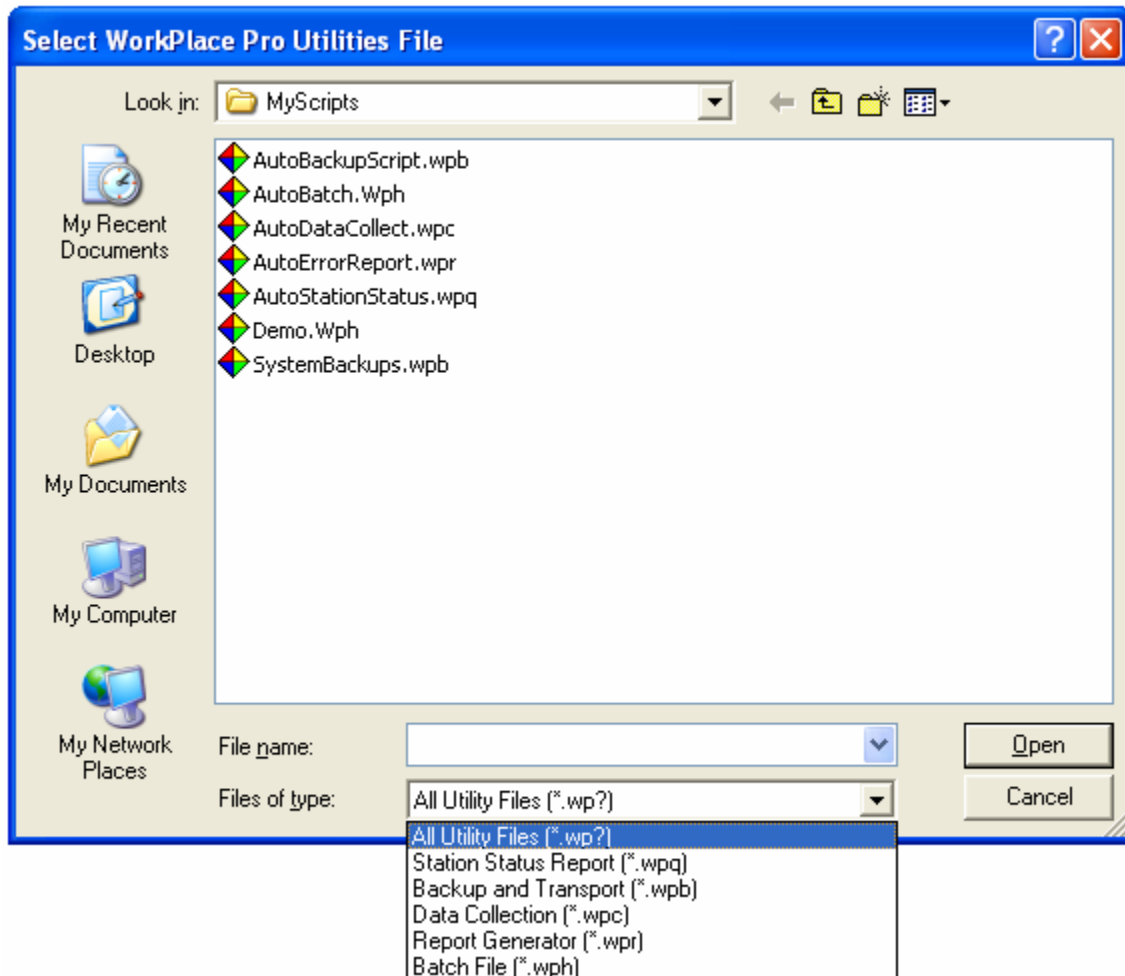
- The *Backup and Transport Utility* downloads an XML backup of a station.
- Next, the *Data Collection Utility* gathers records from the logs in that station backup file and stores them in a local database.
- Then, the *Report Generator Utility* queries the database and creates a report, e-mailing it 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 WorkPlace Pro Utilities (.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 WorkPlace Pro Utilities will be launched. The specified file will then be opened automatically, and the action executed. When the action is complete, the instance of WorkPlace Pro Utilities 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

- a) *BackupTransport1.wpb*
- b) *Batch2.wph*
- c) *BackupTransport2.wpb*

Batch2.wph

- d) *DataCollect1.wpc*
- e) *ReportGenerator1.wpr*

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. WorkPlace Pro Utilities 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:

```
WPUutilities.exe BatchFile.wph RunBatch
```

The first argument is the name of the WorkPlace Pro Utilities executable file. The second argument should be the full path of WPH batch file. The third argument is the command to execute, which must be the single word “RunBatch”.

For example:

```
C:\Program Files\WorkPlace Pro Utilities\WPUutilities.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.

16. The WorkPlace Tech 4.0 Snvt Utilities

The WorkPlace Tech 4.0 Snvt Utilities is a stencil containing shapes that you can add to your WorkPlace Tech 4.0 drawings to provide summary Snvt information. These shapes include:

- A *Data String Generator*, providing I/O counts and the data string needed for an LNC.
- A *Snvt Table Generator*, showing tabular information about the Snvt values in the string.

Installing the Smartware Utilities Stencil

The Snvt Utilities shapes can be found on the Smartware Utilities stencil (*Smartware Utilities.vss*). This stencil can be copied to and opened from any directory. For convenience, it is automatically copied to the default WorkPlace Tech 4.0 stencil directory during installation of WorkPlace Pro Utilities:

C:\Program Files\Invensys\WorkPlaceTech\Solutions

If WorkPlace Tech 4.0 is not installed on the machine at the time you install WorkPlace Pro Utilities, you can copy the stencil to the WorkPlace Tech directory (or any other directory) from:

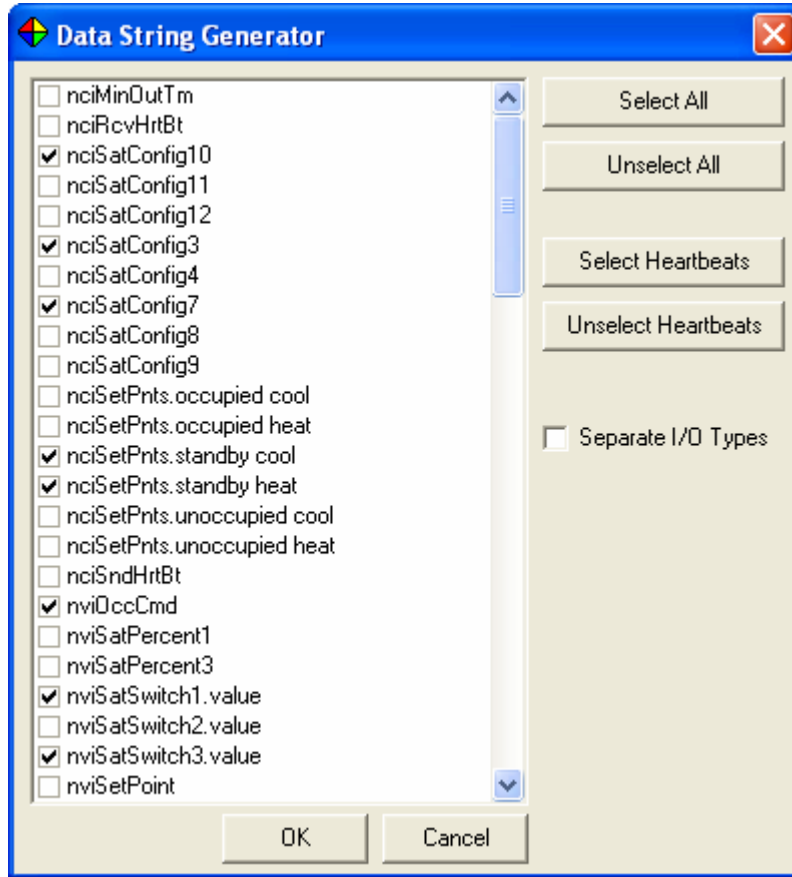
C:\Program Files\WorkPlace Pro Utilities\WorkPlaceTech

Generating an LNC Data String

To generate the data string for use in programming an LNC:

- Open the WorkPlace Tech 4.0 drawing file in Visio.
- Open the *Smartware Utilities* stencil.
- Drag a *Data String Generator* shape from the stencil onto your drawing.
- Right-click on the shape and select GENERATE DATA STRING from the pop-up menu.

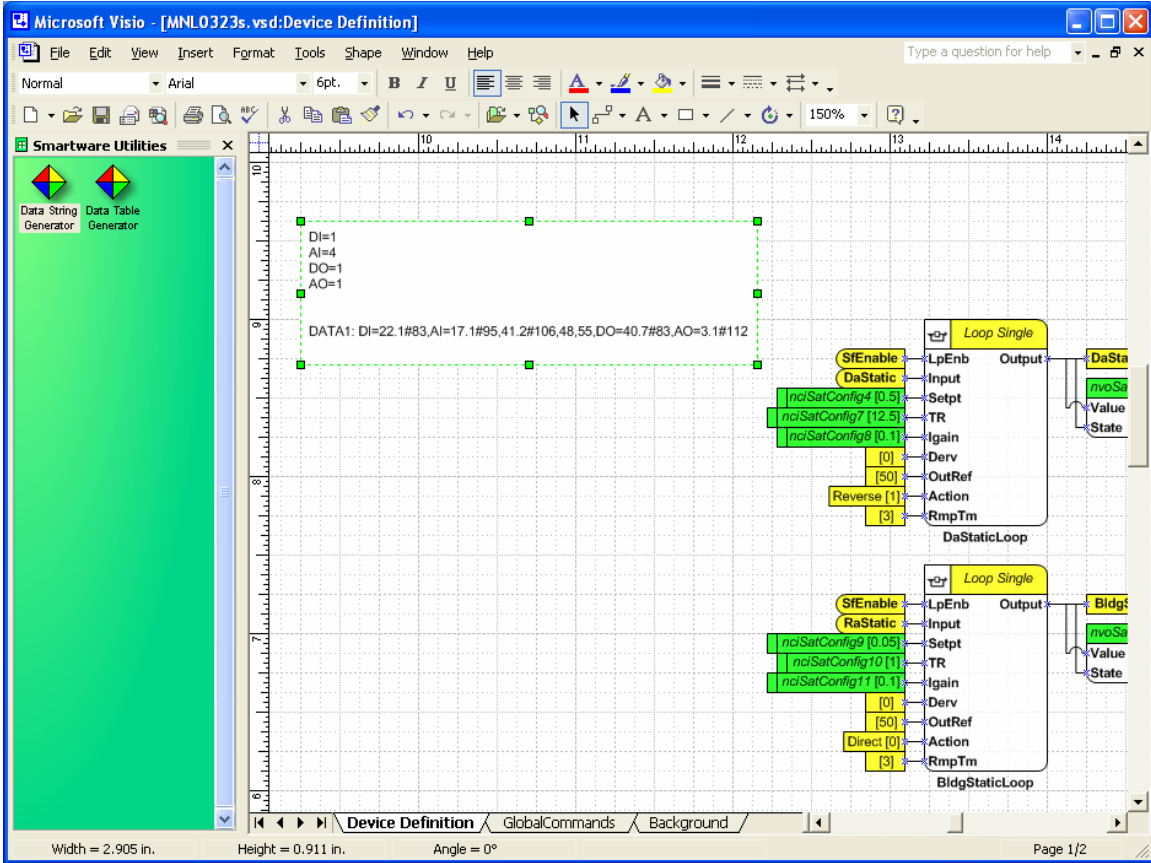
The drawing will be scanned and you will be presented with a list of all the available Snvt objects:



Select the Snvt objects to include. You can quickly select or unselect all the objects or all the heartbeats by using the buttons on the right.

The data string will consist of up to 4 lines of up to 80 characters each. To force each type of input to appear on its own data line, check the SEPARATE I/O TYPES checkbox. There can be a maximum of 16 of each type of I/O.

Click OK and the shape on the drawing will update with the appropriate information, including the I/O count for each type and the LNC data string.



If you run the Data String Generator again, you will have the option of creating a new string, or appending any new Snvt values to the end of the previously generated string.

The Data String Generator currently works on the RF2, RH2, RR2, RV2, RS1 and RS2 profiles, as well as the corresponding Version 3 profiles (RF3, RH3, RR3, RV3, RS3 and RS4). It does not, at the moment, work on MNL-800 controllers.

Generating a Snvt Table

The Snvt Table Generator is a separate shape that automatically reads the information from the Data String and displays the information in a table. After you have created the Data String, simply drag the Data Table Generator shape onto your drawing, right click it, and select CREATE SNVT LIST TABLE from the menu. If you update the Data String, you will need to select this menu item again to refresh the table:

The screenshot shows the Microsoft Visio interface with the Smartware Utilities panel on the left. The Data Table Generator shape is active, displaying a table of SNVT data. Below the table, there are data string definitions and a diagram of a Watchdog timer connected to an nviSatState1 block.

SNVT	SNVT #	Attribute
nviSatState1.bit 3	22.4#83	DIN1
nviSatState1.bit 4	22.5#83	DIN2
nviSetPoint	1	AIN1
nviOccCmd	5	AIN2
nviUnitStatus.primary heat out	8.2#112	AIN3
nviSatTemp1	9	AIN4
nvoUnitStatus.alarm	3.7#112	DV1
nvoSpaceTemp	2	AV1

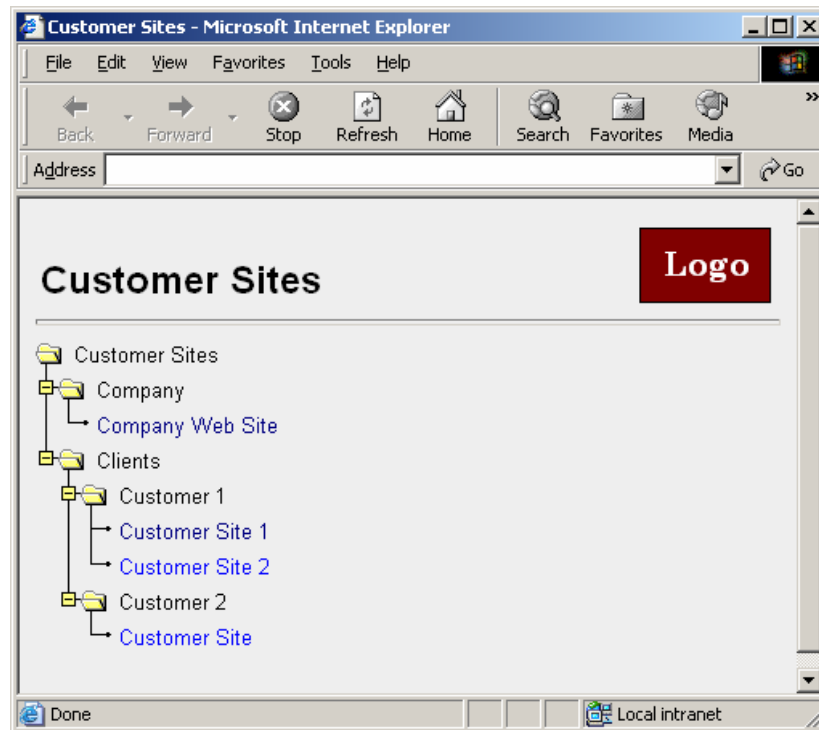
DI=2
AI=4
DO=1
AO=1

DATA1: DI=22.4#83,22.5#83,
DATA2: AI=1,5,8.2#112,9,
DATA3: DO=3.7#112,
DATA4: AO=2

The diagram shows a Watchdog timer block with inputs TmEnb, Input, and OffDly, and outputs Output and TmRem. The Watchdog is connected to the nviSatState1 block, which has 16 bits (Bit0 to Bit15) and a DO output. The bits are labeled as NviSatState1.0 through NviSatState1.15.

17. The Customer Navigation Web Site

As a practical example of the power of the Tree Builder Utility, the WorkPlace Pro Utilities includes a pre-built web site designed to serve as a launch pad to all of your customer's Niagara sites. This allows your engineers and technicians to access any of those sites from one central location without having to remember the various IP addresses. Furthermore, since the navigation trees generated by the Tree Builder Utility can link to any address, this web site can actually be used to create a navigation page that links to any web sites.



This web site works with the navigation system distributed by Tridium as part of WorkPlace Pro. You can find more information about the system in the *Niagara Web Solutions Guide (Chapter 5, DemoR2 Navigation)*

Installing the Web Site

The web site application consists of one main page (default.asp) and a few support files, some of which are optional. These files can be copied into the root directory of a new web site, or as a subdirectory of an existing web site. Be sure that the site is set to look for the file "default.asp" (as opposed to "default.html") as its default starting page.

The files, which can be found in the "Navigation Web Site" subdirectory of your WorkPlace Pro Utilities directory ("C:\Program Files\WorkPlace Pro Utilities" by default), are:

default.asp	The main page for the web site
Customer Sites.wpt	The Tree Builder Utility file containing the navigation tree structure
treeloader.js	The JavaScript file generated by the Tree Builder Utility
logo.bmp	A sample logo file displayed on the page
tree.js	The JavaScript code for creating the tree.
ftv2*.gif	The graphics files for the tree structure

Adding Your Customer Sites to the Navigation Tree

You can add and maintain the sites on the navigation tree without any knowledge of HTML or web-page design whatsoever because the JavaScript file for creating the tree is built using the *Tree Builder Utility*, described earlier in this document. This rest of this section will assume that you are now familiar with that utility.

To edit the navigation tree:

- Step 1: Launch WorkPlace Pro Utilities and select the TREE BUILDER UTILITY.
- Step 2: From the FILE menu, Select OPEN WPT FILE
- Step 3: Navigate to and select the "Customer Sites.wpt" file copied into your site from the install files.
- Step 4: Add, delete or modify the existing tree as desired with additional nodes and sub-nodes as described in the chapter on the *Tree Builder Utility*. For each node:
 - Enter the text you want displayed as the DESCRIPTION field.
 - Enter the IP address for the customer's Niagara site (or other internet site) as the URL LINK field. Be sure to include the "<http://>" in the address.
 - Check the OPEN NEW WINDOW box if you want the page to be opened in its own instance of Internet Explorer.
 - The NODE NAME field is ignored.
- Step 5: When the tree is complete, select GENERATE JAVASCRIPT from the FILE menu. This will generate an updated version of the "treeloader.js" file. You will be prompted for the location in which to save the file.
- Step 6: Copy the updated "treeloader.js" file into the web site directory

Customizing the Web Page

The tree is displayed in the web page created by the "default.asp" file. Although this file is an Active Server (ASP) Page, it is for the most part plain HTML that can be modified to look any way you wish.

The two most simple modifications that anyone can make are to the company name and the company logo file. If you open the "default.asp" file, the first few lines will look something like:

```
<%  
    'USER DEFINED VALUES  
    strPageTitle = "Customer Sites"  
    strLogoFile = "Logo.bmp"  
%>  
  
<html>  
<head>  
.  
.  
.
```

By changing the values in the USER DEFINED VALUES section, the company name and logo can be changed to your own. If the LOGO FILE value is omitted, no logo will be displayed.

Experienced HTML developers can easily integrate this navigation tree into existing sites or build more sophisticated sites around it.