GENESIS32 Standard Training Manual

Version 9.22-Rev C

© 2013 ICONICS, Inc. All rights reserved worldwide. This document may not be reproduced or copied in any manner without written authorization from ICONICS, Inc.

The information contained within this document is subject to change without notice. ICONICS, Inc. does not guarantee the accuracy of the information.

GENESIS32, GENESIS64, Pocket GENESIS, BizViz and their respective modules, OPC-To-The-Core, and Visualize Your Enterprise are trademarks of ICONICS, Inc.

Windows, Windows XP, Windows 2000, Windows 2003 Server, Windows NT, Windows XP-64, Windows 2003-64, Windows Vista, Windows Server 2008, Windows 7, Windows Server 2008 R2, and Windows CE are registered trademarks of Microsoft Corp. in the United States and other countries.

Other product and company names mentioned herein may be trademarks or registered trademarks of their respective owners.

Document Number: TM-GEN32Standard-922

Visualize Your Enterprise™

ICONICS LICENSE AGREEMENT

YOU SHOULD CAREFULLY READ THE FOLLOWING TERMS AND CONDITIONS!

Opening and using the enclosed software for any purpose indicates your acceptance of the terms and conditions of this license agreement. If you do not agree with them you should return all software, documentation and copy protection keys within seven days of shipment unopened and your money will be refunded.

ICONICS provides this program and licenses its use in the United States, Puerto Rico, or internationally. You assume the responsibility for the selection of the program to achieve your intended results, and for the installation, use and results obtained from this program.

LICENSE

You are granted a personal license to use this program under the terms stated in this Agreement. You may: 1) Install and use the program on a single machine. 2) Make a single archival back-up copy of the program for the sole purpose of supporting your use of the single program on a single machine. 3) You may not use, copy, modify, or transfer the program, or transfer any copy, in whole or in part, except as expressly provided in this license, or with a written contractual agreement with **ICONICS,** Inc.

TERM

The license is effective until terminated. It will terminate if you fail to comply with any term or condition of this Agreement. You agree, upon such termination, to destroy the program and all copies that were made from it and to promptly return any copy protection key(s) to **ICONICS.**

LIMITED WARRANTY

ICONICS WARRANTS THE CD-ROM OR PHYSICAL DISKETTES, ON WHICH THE PROGRAM IS FURNISHED, AND PHYSICAL DOCUMENTATION TO BE FREE OF DEFECTS IN MATERIALS AND WORKMANSHIP UNDER NORMAL USE FOR A PERIOD OF NINETY (90) DAYS FROM THE DATE OF DELIVERY TO YOU AS EVIDENCED BY YOUR RECEIPT. THIS PROGRAM IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO THE IMPLIED WARRANTIES OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE ENTIRE COST OF ALL NECESSARY SERVICING, REPAIR, OR CORRECTION.

ICONICS SPECIFICALLY DISCLAIMS ALL OTHER WARRANTIES, EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO, IMPLIED WARRANTIES OF TITLE, NON-INFRINGEMENT, MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN PARTICULAR, WITH RESPECT TO ANY PARTICULAR APPLICATION, USE OR PURPOSE, IN NO EVENT SHALL **ICONICS** INC. BE LIABLE FOR ANY OTHER COMMERCIAL DAMAGE, INCLUDING BUT NOT LIMITED TO SPECIAL, INCIDENTAL, CONSEQUENTIAL OR OTHER DAMAGES. **ICONICS** DOES NOT WARRANT THAT THE FUNCTIONS CONTAINED IN THE PROGRAM WILL MEET YOUR REQUIREMENTS OR THAT THE OPERATION OF THE PROGRAM WILL BE UNINTERRUPTED OR ERROR FREE.

LIMITATION OF REMEDIES

ICONICS' entire liability and your exclusive remedy shall be the replacement of any CD-ROM or diskette not meeting **ICONICS**' "Limited Warranty" which is returned to **ICONICS** with a copy of your receipt within the warranty period. The remedy for breach of this warranty shall be limited to replacement and shall not encompass any other damages including but not limited to loss of profit, special, incidental, consequential, or other similar claims arising out of the use or inability to use such program even if **ICONICS** has been advised of the possibility of such damages, or for any claim by any other third party.

GENERAL

This Agreement will be governed by the laws of the Commonwealth of Massachusetts. Should you have any questions concerning this Agreement, you may contact **ICONICS** in writing at:

ICONICS, Inc.
100 Foxborough Boulevard
Foxborough, MA 02035

You acknowledge that you have read this agreement, understand it and agree to be bound by its terms and conditions. You further agree that it is the complete and exclusive statement of the Agreement between you and **ICONICS** which supersedes any proposal or prior Agreement, oral or written, and any other communications between you and **ICONICS** relating to the subject matter of the Agreement.

ICONICS RETURN POLICY

All sales are final (NO RETURNS) unless one of the following conditions is applicable: Program is returned in its original packaging material within 7 days after shipment. In such a case, a full refund is provided, less all incurred shipping and handling costs. Program is returned within 30 days from the original shipment date. In such a case a 20% restocking fee is applicable. No returns are allowed on OPC ToolWorX or ActiveX ToolWorX.

Contents

1. Introduction to GENESIS32

1.1. GENESIS32 INSTALLATION 1-2

LAB: OFFSITE TRAINING INSTALLATION AND CONFIRMATION L1-1

REVIEW: INTRODUCTION TO GENESIS32 ..... R1-1

2. GraphWorX32: Application Basics

2.1. GRAPHWORX32 LAYOUT 2-1

2.2. TOOLBARS 2-2

2.3. SYSTEM CONFIGURATION 2-6

2.4. DISPLAY SUMMARY 2-14

2.5. GRAPHWORX32 TEMPLATES 2-17

2.6. LAYERS 2-18

LAB: ACMESEUTICAL GRAPHICS TEMPLATES L2-1

REVIEW: GRAPHWORX32 APPLICATION BASICS R2-1

3. GraphWorX32: An Animation Platform

3.1. DRAW TOOLBAR 3-1

3.2. COLOR PALETTE TOOLBAR 3-4

3.3. ARRANGE TOOLBAR 3-5

3.4. DYNAMICS 3-6

LAB: CONTROLLING PILL PRODUCTION L3-1

REVIEW: GRAPHWORX32 AN ANIMATION PLATFORM R3-1

4. GraphWorX32: Pick Actions and ActiveX

4.1. PICK ACTIONS 4-1

4.2. ACTIVEX 4-10

LAB: CONTROLLING THE MIXERS L4-1

REVIEW: GRAPHWORX32 PICK ACTIONS AND ACTIVEX R4-1

5. Introduction to Workbench32

5.1. WORKBECH32 5-1

5.2. WORKBENCH32 LAYOUT 5-2

5.3. PACK AND GO 5-3

5.4. WORKBENCH32 RUNTIME 5-11

5.5. REMOTE ACCESS 5-11

LAB: ACMESEUTICAL PROJECT L5-1

REVIEW: INTRODUCTION TO WORKBENCH32 ..... R5-1

6. AlarmWorX32 Server

6.1. ALARM SERVER CONFIGURATOR 6-1

6.2. ALARM SERVER CONFIGURATION 6-2

6.3. ALARM TAGS 6-3

6.4. ALARM AREAS 6-7

6.5. CONFIGURING MANY SIMILAR ALARMS 6-8

LAB: ALARM NOTIFICATIONS L6-1

REVIEW: ALARMWORX32 SERVER R6-1

7. AlarmWorX32 Viewer

7.1. Alarm Viewer ActiveX 7-1
7.2. Alarm Viewer Properties 7-2
7.3. Alarm Subscription 7-2
7.4. Column Configuration 7-4
7.5. Row Configuration 7-6
7.6. Record Sorting 7-7
7.7. Filtering Configurations 7-7
7.8. ToolTips Configuration 7-9
7.9. Runtime Options 7-10
LAB: VIEW ALARM NOTIFICATIONS L7-1
REVIEW: ALARMWORX32 VIEWER R7-1

8. AlarmWorX32 Logger

8.1. The AlarmWorX32 Logger 8-1
8.2. Alarm Logger Configuration 8-2
8.3. Alarm Logger Columns 8-6
8.4. Node Configurations 8-7
8.5. Workbench32 8-7
LAB: MAINTAINING HISTORY ALARMS L8-1
REVIEW: ALARMWORX32 LOGGER R8-1

9. AlarmWorX32 Reporter

9.1. Alarm Report ActiveX 9-1
9.2. Alarm Report Properties 9-2
9.3. General Settings and Database Connection 9-3
9.4. Column Configuration 9-4
9.5. Records Sorting 9-5
9.6. Alarm Report Runtime Options 9-6
LAB: VIEWING HISTORY ALARMS L9-1
REVIEW: ALARMWORX32 REPORTER R9-1

10. TrendWorX32 Logger

10.1. TrendWorX32 Logger System Architecture 10-2
10.2. TrendWorX32 Logger Configurator 10-3
10.3. Trend Logger Configuration 10-3
10.4. Database Group Configuration 10-4
10.5. Logging Group Configuration 10-6
10.6. Logging Tag 10-9
10.7. TrendWorX32 Configuration in Workbench32 10-11
LAB: MAINTAINING OPC DA RECORDS L10-1
REVIEW: TRENDWORX32 LOGGER R10-1

11. TrendWorX32 Viewer

11.1. TrendWorX32 Viewer 11-1
11.2. Trend Viewer Properties 11-2
11.3. General Configurations 11-3
11.4. Appearance Settings 11-4
11.5. Configuring Ranges 11-5
11.6. Pen Configuration 11-6
11.7. Configuring Rates 11-7
11.8. Trend Viewer Runtime Toolbar 11-8
LAB: VIEWING HISTORICAL TRENDS L11-1
REVIEW: TRENDWORX32 VIEWER R11-1

12. TrendWorX32 Reporting

12.1. Reporting with TrendWorX32 12-1
12.2. TrendWorX32 Reporter 12-1
12.3. TrendWorX32 SQL Tool Control 12-3
12.4. TrendWorX32 SQL Query Engine 12-4
LAB: SIMPLE HISTORY TREND REPORT L12-1
REVIEW: TRENDWORX32 REPORTING R12-1

13. Aliasing

13.1. Aliasing in GENESIS32 13-1
13.2. Local Aliasing 13-2
13.3. Global Aliasing 13-6
LAB: SIMPLIFIED DEVELOPMENT WITH GLOBAL ALIASING L13-1
REVIEW: ALIASING R13-1

14. Language Aliasing

14.1. Introduction to Language Aliasing 14-1
14.2. Background 14-2
14.3. Configuring Language Aliasing 14-2
14.4. Using Language Aliases 14-7
14.5. Language Alias Switching 14-9
LAB: MULTILINGUAL DISPLAYS L14-1
REVIEW: LANGUAGE ALIASING R14-1

15. Unified Data Manager

15.1. The Unified Data Manager Configurator 15-2
15.2. Introduction to Global Expression 15-3
15.3. Introduction to Groups 15-8
15.4. Introduction to Value Sets 15-12
15.5. Introduction to Recipes in the UDM 15-18
15.6. Introduction to Global Alarm Subscriptions and Filters 15-24
REVIEW: ACMESEUTICAL PILL RECIPES R15-1

16. Security
16.1. INSTALLATION 16-1
16.2. SECURITY SYSTEM COMPONENTS 16-2
16.3. SECURITY SERVER CONFIGURATION 16-3
16.4. SECURITY LOGIN CLIENT 16-14
16.5. THE SECURITY INDICATOR ACTIVEX CONTROL 16-15
LAB: SECURITY L16-1
REVIEW: SECURITY R16-1

17. Additional Applications
17.1. GENBROKER 17-1
17.2. GENTRAY AND MONITORWORX 17-4
17.3. PROJECTWORX32 17-6
REVIEW: ADDITIONAL APPLICATIONS R17-1

A. GraphWorX32 Shortcuts A-1

B. TrendWorX32 SQL Query Engine Syntax B-1

C. Delimiters and Syntaxes C-1

D. Glossary D-1

E. Answers E-1

Introduction to GENESIS32 1

Welcome to the GENESIS32 Standard Training! GENESIS32 contains a colossal amount of applications that help you visualize and control your automation process. In this course, we aim to demonstrate the most common applications inside ICONICS' 32-bit HMI/SCADA software suite.

A typical production line may have various PLCs connected to sensors and actuators to control the line. OPC Servers are then connected to the PLCs to provide HMI/SCADA software suite, such as GENESIS32, the necessary information for operators to monitor the production line.

The OPC Foundation is the organization that makes this type of system architecture possible and easy to setup. They maintain and develop standards for communications between OPC Servers and Clients; as long as your OPC servers are compliant, GENESIS32 will be able to connect and display data from your devices without any particular drivers.

> **NOTE:** If you would like to learn more about OPC Servers and the OPC Foundation, you can find more information in the *GENESIS32 Basics Training Manual*.

At the end of this training course, you should be familiar with the following applications in the GENESIS32 suite:

- AlarmWorX32
- GraphWorX32
- TrendWorX32
- Workbench32
- Global Aliasing
- Language Aliasing
- Unified Data Manager
- Security

You should also be comfortable with the GENESIS32 architecture and be able to design a system from scratch, taking advantage of all that the product suite has to offer.

In this chapter, you will find instructions on installing GENESIS32 on your machine.

1.1. GENESIS32 Installation

It is a relatively easy procedure to install GENESIS32. Similar to how you install other software, you would insert the installation disk and follow the onscreen instructions to install it. However, since there are many optional products in GENESIS32, each may require different hardware configuration or other software; you should consider these factors when installing GENESIS32.

1.1.1. System Requirements

In the root of the installation disk, you will find a *ReadMe* file that includes information about Sizing your system, User Privileges and SupportWorx options. It is always a good practice to go through the *ReadMe* file to learn about new features in the installing version. Probably the most important information in the *Readme* is the system requirements; you should make sure that your machine meets the requirements before installing. Table 1.1 shows some of the most commonly asked requirements.

Table 1.1 - System Requirements

Operating Systems[1]	Windows 2000, Windows XP Professional, Windows Server 2003, Windows Vista[2], Windows Server 2008, Windows 7[2]
Processor Speed	1.0 GHz
RAM	1GB
Hard Disk Space	2 GB
Optical Drive	DVD Drive
Graphics Card	SVGA Video Cards with 256 or more colors
Input Devices	Microsoft compatible keyboard and mouse
Additional Software[3]	Internet Explorer 6.0 or higher

Note that the requirements listed here are the minimum requirements. Depending on the size and complexity of your system, the requirements will change. You can consult the *ReadMe* file for guidelines of the system requirement change as the system scales. In general, a faster processor and an increase in RAM will produce better performance.

[1] If the listed operating system (OS) comes in x86 and x64, GENESIS32 will work in both. Note that when running GENESIS32 in x64 OS, it will run in 32-bit compatibility mode.
[2] In general, ICONICS does not support Home edition OS. The supported editions for Windows Vista are Windows Vista Business, Ultimate, and Enterprise Editions. Similarly, the supported editions for Windows 7 are Professional, Ultimate and Enterprise.
[3] This only applies for general GENESIS32 installation; you may need other software or hardware when installing WebHMI or AlarmWorX32 Multimedia.

1.1.2. Installing GENESIS32 for the First Time

When you are ready to install, you should insert the product DVD. The installation should launch automatically. If not, double click on the Default.hta file located in the root of the installation DVD. In the splash screen shown in Figure 1.1, click on the GENESIS32 link under the Product Installation section.

Figure 1.1 - GENESIS32 Installation Welcome Screen

A window will pop up, stating that it needs to install some required software first. Click on the "Install" button to continue.

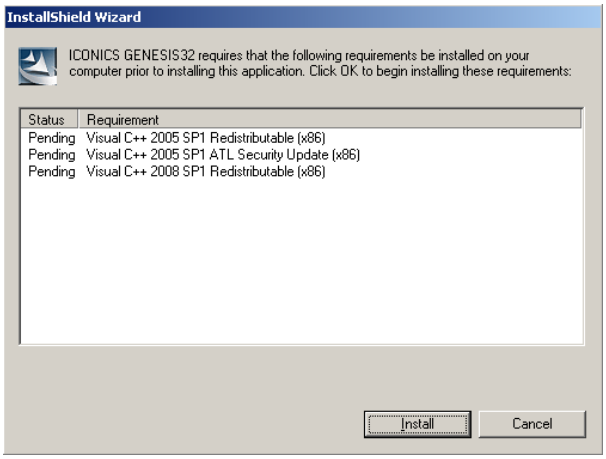


Figure 1.2 - Install Shield Wizard: Pre-installation Requirements

It may take a few minutes to install the listed files. When it's done, you will see the Welcome screen for the GENESIS32 installation as shown in Figure 1.3.

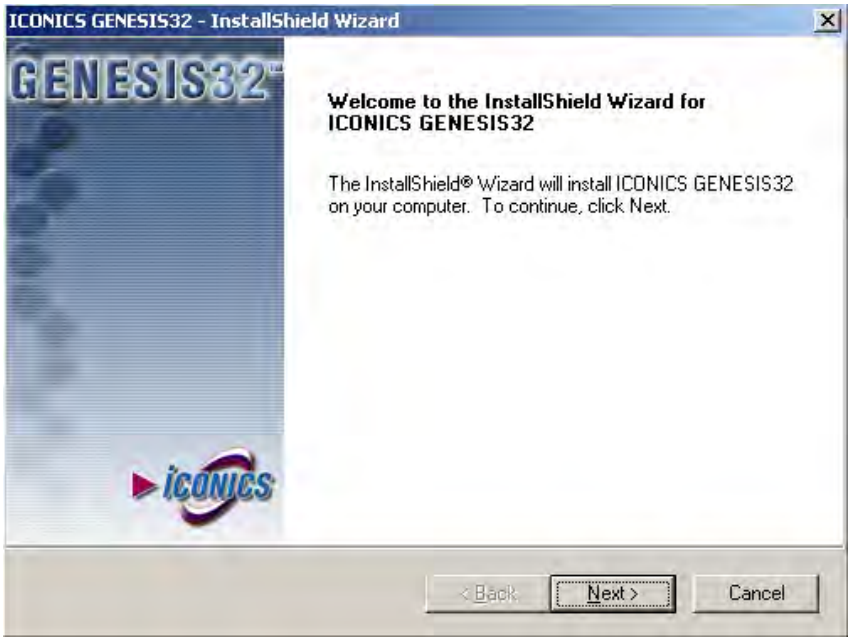


Figure 1.3 - GENESIS32 Welcome Screen

Simply click on "Next" to continue. You will see the License Agreement on the next screen. You must accept the license agreement to continue the installation. You can click on the "Print" button to print a copy of the agreement before you continue.

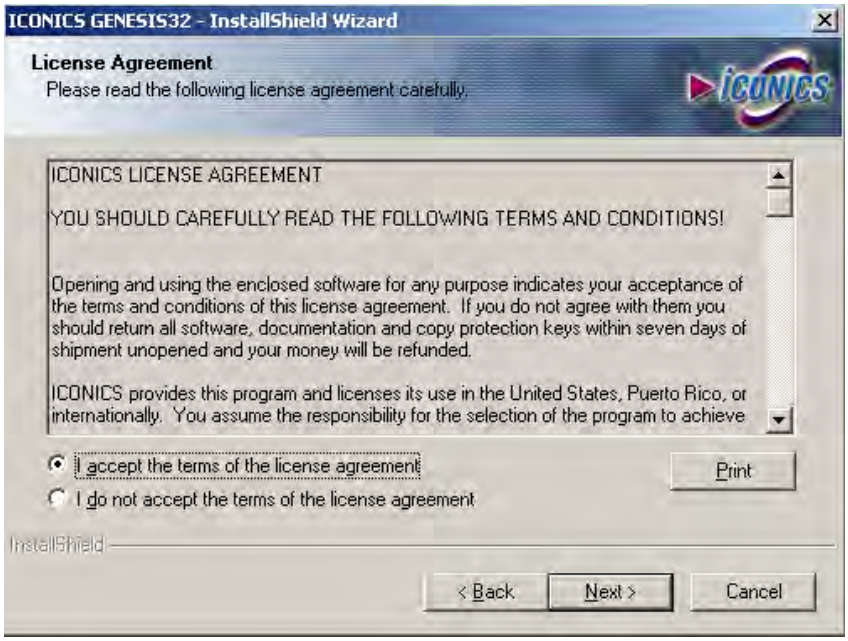


Figure 1.4 - GENESIS32 Installation: License Agreement

Once you accept the license agreement, the “Next” button will become available for you to click on and continue the installation.

The next screen allows you to choose how you want to install GENESIS32. As with most software installations, it gives you three installation options: *Typical*, *Compact*, and *Custom*.

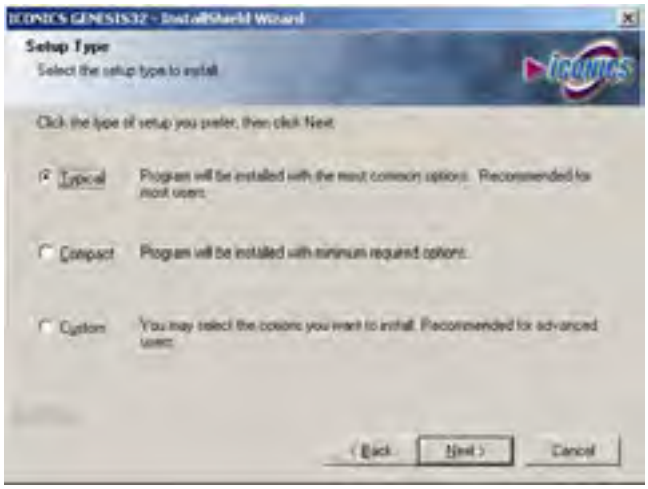


Figure 1.5 - Installation Options

The default and recommended option is the *Typical* installation. You can also choose *Compact* install if you want to save some space on the machine. This will only install the essential components to run GENESIS32; nonessential files such as help files may be eliminated from the installation. For our advanced users, we provide a *Custom* installation in which you can choose the different components to install on the computer. This option gives you the most flexibility in choosing what to install. Assuming that you choose the *Typical* option, when you can click on “Next,” you will see a screen like the one shown in Figure 1.6. Note: option selections in Figure 1.6 - GENESIS32: Options to Install will be grayed out if the installation prerequisites are not installed.

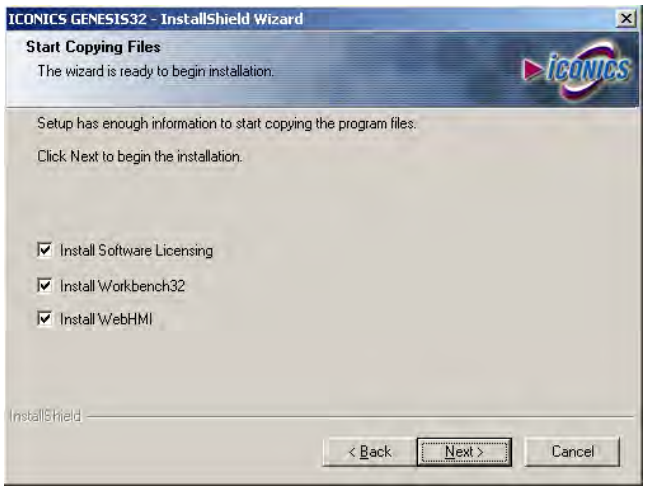


Figure 1.6 - GENESIS32: Options to Install

ICONICS Inc. now creates one DVD which includes software for both US and International customers. Make the appropriate selection as shown in Figure 1.7 – Language Selection below.

Figure 1.7 – Language Selection

On this screen, you can choose to install the License Software, Workbench32, and WebHMI. If you do not want to install any of these options at this time, you can uncheck it. If you decide to install these later, return to section 1.1.2 and select the preferred options under the Product Installation section on the installation splash screen. When you have chosen the desired options, click on “Next” to continue the installation. It may take a few minutes, but you should see the Application Setup Utility come up.

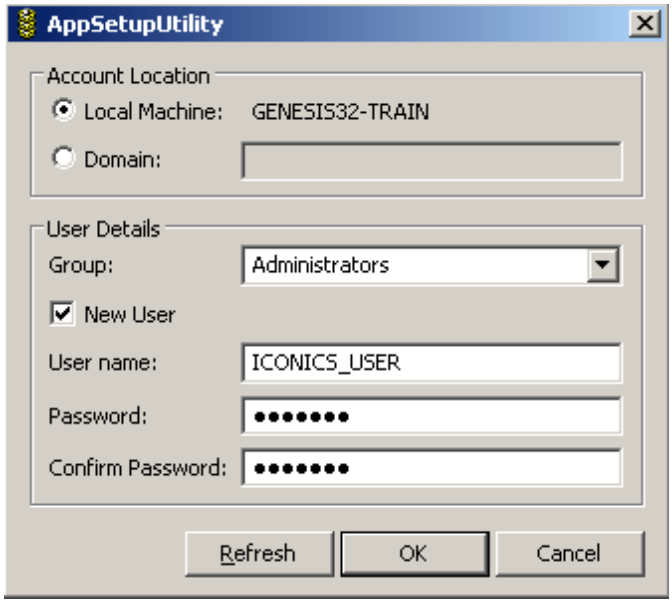


Figure 1.8 – Application Setup Utility

We will discuss the Application Setup Utility in more detail in the next section. When you see the utility come up, you should create an administrative user to run the ICONICS applications and services. The default user is ICONICS_USER; although it is highly recommended first time installers proceed with the default user. However, you can create a user of your own or select an existing user from the Administrators group. When you are done, click on "OK" to continue the installation.

The installation time will vary depending on the number of software options being installed.

NOTE: You may not see any activity as the installation is running. Please be patient, the software installation is running in the background.

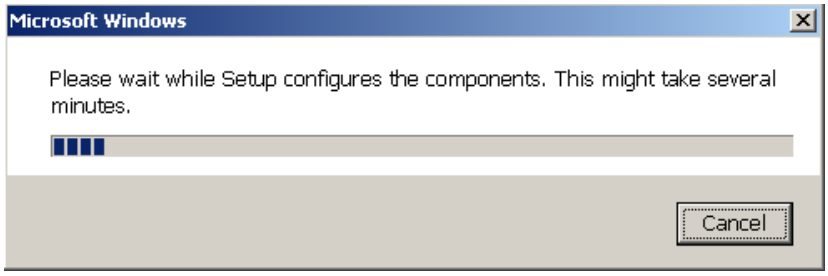


Figure 1.9 – Installation Progress

You will receive a message when the installation finishes that will prompt you to restart the machine. If you do not want to restart the machine immediately, you can choose the option to restart it later and close the dialog. However, know that you must restart your computer after software installation for some components to function properly.

1.1.3. Application Setup Utility

The Application Setup Utility is an ICONICS tool that will help set some application to run as automatic services and configure DCOM between the different ICONICS applications so that they will communicate with each other without problems.

Before there was the Application Setup Utility, when OPC Servers failed to give ICONICS applications data, you would need to edit DCOM manually. In fact, we have different application notes located on the installation DVD in the GENESIS32\Info\files\app_notes folder including information on setting up DCOM on different operating systems and network configuration. The introduction of our Application Setup Utility greatly reduces the need to edit DCOM manually after you installed GENESIS32.

During installation, you will see the Application Setup Utility come up as shown in Figure 1.8. This is where you would choose an existing user or create a new user to run your ICONICS applications and services. You can choose or create a user local to your machine or if your machine is in a domain, you can use a domain user. If you choose to use a domain user, please be sure to add local machine privileges to the domain user.

Regardless of whether you choose or create a user to run the ICONICS applications and services, we recommend that you pick a user with administrative privilege. You should know that you do not have to log into the computer with this user for our applications and services to work. We are simply specifying credentials to run an application so that regardless of who logs into the computer, the ICONICS applications and services run with the same credentials.

If you are creating a user, you should provide a password for the user that conforms to your password policy. If you decide to use an existing user, provide the password for that user in the form. You may NOT elect a user without a password to run your applications and services. Windows simply will not allow any user without a password to run services.

When you click on “OK” to run the Application Setup Utility, it does a few things behind the scenes. First, it sets up some of the most popular applications such as GenBroker, AlarmWorX32 Logger, AlarmWorX32 Server, TrendWorX32 Logger, and many others to run as a service. With this setup, you can easily change the startup type through the Windows OS Services later so that the services will start when the machine boots-up.

> **NOTE:** GenBroker is set to run as an automatic service. Therefore, it will start when the machine boots-up by default.

Furthermore, the Application Setup Utility adds the specified user into DCOM and sets all the applications and services to run as this user. Since all ICONICS services will now run as this particular user, it is unlikely that you will ever run into communication problems between ICONICS applications. Because everything is set up to run with default DCOM permissions, if you have a third-party OPC server on the system that will not communicate with ICONICS, you can simply edit the DCOM permissions for that OPC server to use default settings and run as the same user that runs all ICONICS applications. This eliminates some confusion with DCOM. Note, the aforementioned DCOM related app notes may be helpful.

If you decided not to run the Application Setup Utility during installation, you can always run it afterwards. You can find the Application Setup Utility in your Start menu under ICONICS Tools. When you run the utility, you will receive the same figure 1.7 dialog when you run it during installation. When prompted to restart the machine after running the utility, you should always do as instructed.

While the Application Setup Utility takes care of most communication problems, there are still cases where you could run into problems. For instance, if you are using a domain user and if the domain policy forces users to change passwords periodically, there is a chance that your applications and services will stop working. If that is the case, you will get a logon failure when you try to start a service from the Windows Service Manager. In such cases, you simply need to change the user's password and run the Application Setup Utility again with the new password. **The most ideal case in this situation is to use a local user where the password never expires.**

1.1.4. Upgrading from Prior Versions

If you are upgrading GENESIS32 from a prior version, you have made a very good decision. Version 9.22 contains many new features and enhancements from prior versions that would improve your HMI development experience.

To upgrade to version 9.22, you must first uninstall the prior version. You can do this from Windows' Programs and Features (or Add/Remove Programs for older operating systems) in the Control Panel.

> **NOTE:** If you have BizViz installed on the same machine, it is best that you uninstall BizViz and upgrade to 9.22 as well.

Once you have uninstalled any prior versions, you can follow the same installation process as mentioned before.

If you have done upgrades to our other versions, you would know that in prior versions, while you would uninstall all components of GENESIS32, you would leave the installed license alone and the new licensing installation would detect that you are doing an upgrade. This is NOT the case for this version.

In Version 9.22, we are trying to add better support for RAID and x64 systems. Because of these enhancements, the software licensing has changed enough that we can no longer allow for a simple upgrade process as before. You must uninstall and reinstall the licensing software when upgrading to this version.

Before you uninstall your licensing software from prior versions, you should kill the current license using the License Utility and credit the license on our web site. You can find out more about this procedure in our application notes.

Lab: Offsite Training Installation and Confirmation

L1

In this lab we will install GENESIS32. If you are attending this training course at one of our training facilities and are using one of our provided machines, the support files listed in section 1.2 should reside in a C:\Training folder. Otherwise, the instructor will deliver the files to you.

Estimated time to complete this lab: 90 minutes

1.1. Task Checklist

In this lab, you will complete the following tasks:

- Install GENESIS32
- Apply the training support files
- Create necessary database files

1.2. Support Files

You will need the following file(s) to complete this lab:

- The Installation disk
- Acmeseutical_DWXConfig.mdb
- AcmeseuticalGASConfig.mdb
- LanguageAliasingTranslation.txt
- TrainingCourse.sim
- PainreliefEuropeMap.gdf
- PainReliefUSMap.gdf
- PainReliefWorldMap.gdf
- TrainingCourseV9.22.sdf

1.3. Installation

In this section, we will walk you through to install GENESIS32 as well as list software that you should have on the system before you begin the installation procedure.

1.3.1. Installation Prerequisites

Before we install any ICONICS applications, it is generally a good idea to make sure that the required software has already been installed on the computer. To prepare for a successful installation, we recommend that you have the following already installed on the computer:

- Internet Information Services (IIS) with FTP components
  - FTP is required when you intend to publish displays for use with WebHMI clients.
- The latest .NET Framework – the GENESIS32 installation will try to detect this component and install it if it is not there. However, this takes time (up to an hour) to install. Therefore, we recommend installing this beforehand.
- SQL Server Express can be installed from the GENESIS32 installation DVD. See figure 1.1 for more information. If your application requires, a full featured SQL Server installation is also supported. We will log history data into SQL databases. It is recommended that you have the database server installed beforehand.

Depending on your operating system, the IIS installation may require your Operating System's installation disk. You should obtain this from your IT personnel. In addition, the path where IIS needs to be installed varies by Operating System. You may want to ask your instructor its required location on the Operating System you will be using.

1.3.2. Installing GENESIS32

1. Insert the GENESIS32 installation disk.

2. When the splash screen comes up, click on the "GENESIS32" link, which is circled in Figure L1.1 - Installation Splash Screen. In case the splash screen does not pop up (you may have auto-run disabled), you can go to your DVD drive and double-click on the "Default.hta" file to bring up the splash screen.

Figure L1.1 - Installation Splash Screen

3. You will then see a dialog in the lower-right corner preparing for the installation.

4. A window may popup stating that it needs to install some required software first. Click on the "Install" button to continue.

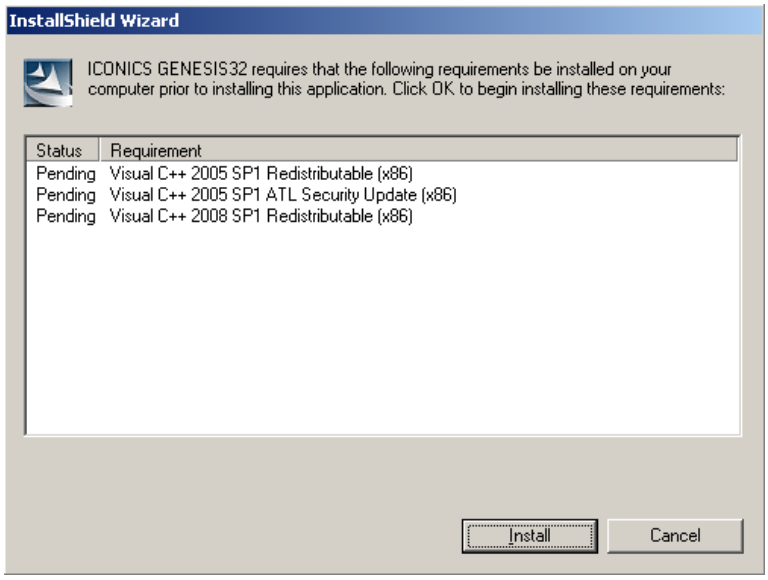


Figure L1.2 - Install Shield Wizard: Pre-installation Requirements

5. Once the prerequisites installation completes, you should see the Welcome screen as shown in Figure L1.3 - Welcome Screen.

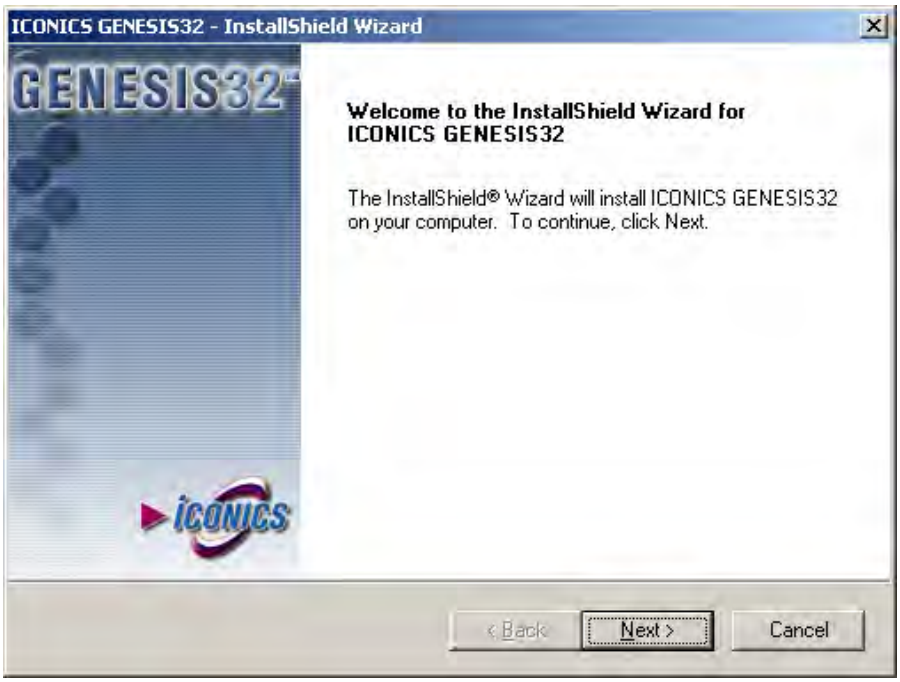


Figure L1.3 - Welcome Screen

6. Click on “Next >” to continue the installation.

7. On the **License Agreement** screen, read through the agreement and select “I Accept the terms of the license agreement” at the bottom and click on “Next >” to continue.

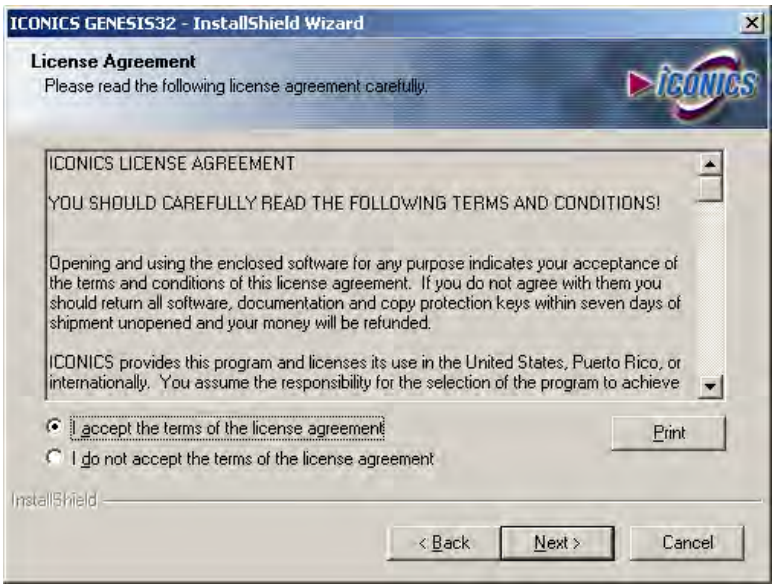


Figure L1.4 - License Agreement

8. On the Setup Type screen, leave the selection at "Typical" and click on "Next >."

9. ICONICS Inc. now creates one DVD which includes software for both US and International customers. Make the appropriate selection as shown in Figure L1.5 below.

Figure L1.5 – Language Selection

10. On the **Start Copying Files** screen, the default selection installs Licensing, Workbench32, and WebHMI. You can leave this setting as-is and click on "Next >."

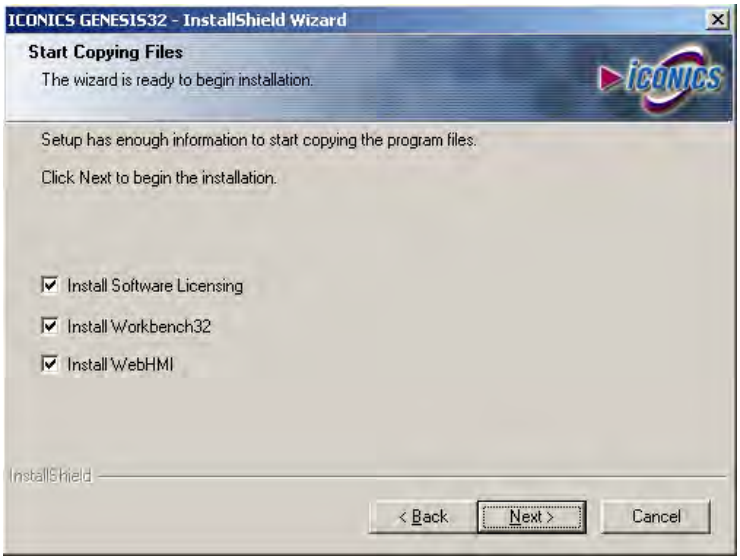


Figure L1.6 - Start Copying Files

Note: If any selection is grayed out, look closely at the text after each of the checkboxes. This information is very helpful in determining the source of any problems.

11. The Setup Status window will show you the progress of the installation. During this step, the **Application Setup Utility** will launch. You can choose to create the new "ICONICS_USER" (see Figure L1.7 - Application Setup Utility) or choose an existing user that is a member of the administrators group on the system.

12. Regardless of the user you choose, you must provide a password. Enter the password twice, in the appropriate fields, and click on "OK" to close the dialog.

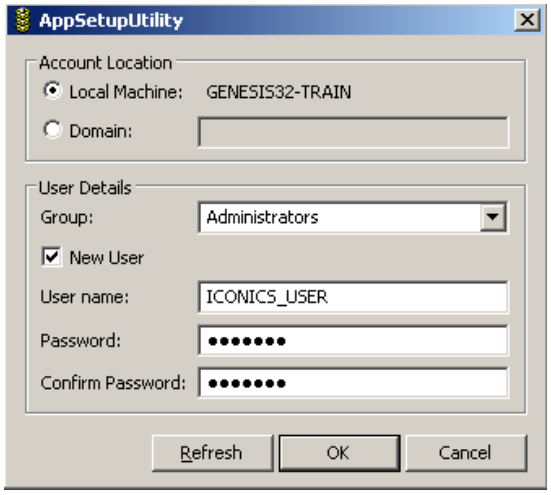


Figure L1.7 - Application Setup Utility

13. The installation will continue. Once it completes, you should see the Installation Complete dialog shown in Figure L1.8 - Installation Complete, which will ask you to restart the computer.

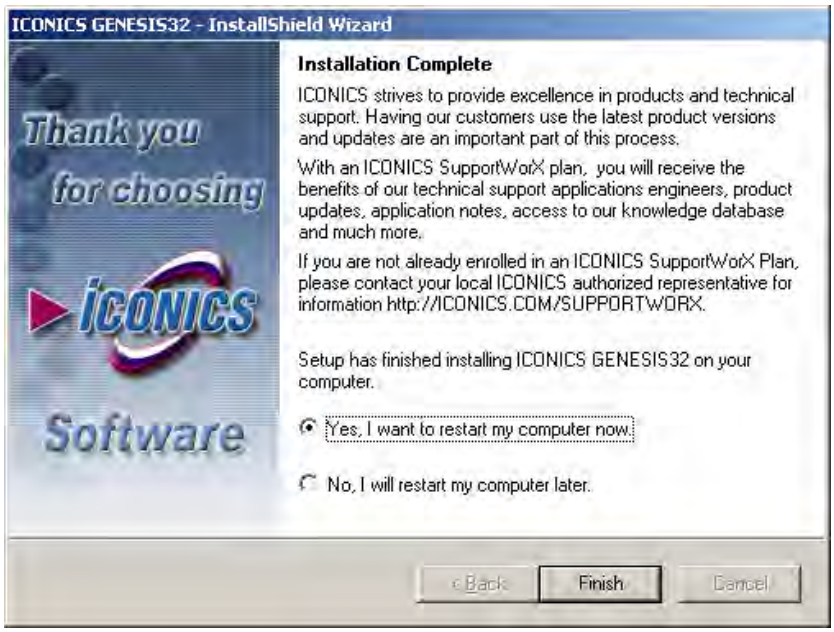


Figure L1.8 - Installation Complete

14. Leave the selection as “Yes, I want to restart my computer now” and click on “Finish” to restart the system.

15. Once the system restarts, you are ready to use GENESIS32.

1.3.3. Installing SQL Server Express

16. Insert the GENESIS32 installation disk.

17. When the splash screen comes up, click on the “SQL Server Express 2005 SP3 Wrapper” link, which is circled in Figure 1.9 - SQL Server Installation. In case the splash screen does not pop up (you may have auto-run disabled), you can go to the root folder of your DVD drive and double-click on the “Default.hta” file to bring up the splash screen.

Figure L1.9 - SQL Server Installation

18. During the installation, preserve each default selection (other than the two selections defined in the following figures: Figure L1.10 - Client Components and Figure L1.11 - SQL Administrator Role).

19. Select Client Components and select "Entire feature will be installed on local hard drive." Click Next to continue.

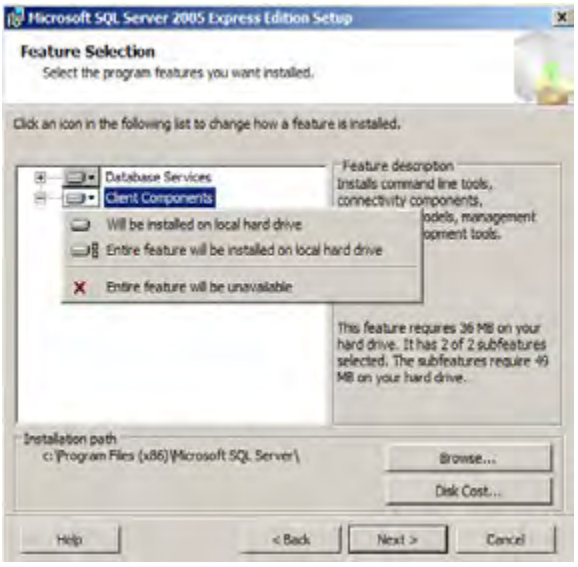


Figure L1.10 Client Components

20. From the Configuration Options dialog select "Add user to the SQL Server Administrator role." Click Next to continue.

Figure L1.11 – SQL Administrator Role

1.4. Training Support Files

There are a few support files that you will need in order for the training to run smoothly. If you are using one of the provided training computers, we will review the subsequent steps for accuracy.

1. Open Windows Explorer and navigate to C:\
2. The required files may already reside in the C:\Training folder. If not, create a folder here called "C:\Training." This folder will be your project folder and you will store all of the training files into this folder.
3. Copy the following files into the C:\Training directory:

- Acmeseutical_DWXConfig.mdb
- AcmeGASConfig.csv
- LanguageAliasingTranslation.txt
- TrainingCourse.sim
- PainReliefEuropeMap.gdf
- PainReliefUSMap.gdf
- PainReliefWorldMap.gdf

NOTE: Ask your instructor for these files if you do not have them.

- **Symbol Library**

  4. We will need to move a training course file into a system location. Depending on your operating system, the location could be different. Browse to one of the following directories, and copy the **TrainingCourseV9.22.sdf** from C:\Training into this directory.

     - *For Windows Server 2003 and prior operating systems:*
       C:\Documents and Settings\All Users\ICONICS\Symbols\2_Training
     - *For Windows Vista, Server 2008 and later operating systems:*
       C:\ProgramData\ICONICS\Symbols\2_Training –
       Note: you may have to change the Folder Option and enable Show hidden files, folders, or drives.

- **DataWorX32**

  5. Start DataWorX32 Configurator by going to Start → Programs → ICONICS GENESIS32 → DataWorX32 → DataWorX32.
  6. Once DataWorX32 is opened, click on File → Open and browse to C:\Training and select the Acmeseutical_DWXConfig.mdb and click on "Open."
  7. Select File → Make Active.
  8. Click Yes to the question "Do you want to set the edited database as active database?"
  9. Click on the traffic light to start DataWorX32. Be Patient, wait for the green light.
  10. Close the DataWorX32 configurator.

- **Simulator**

  11. Start the ICONICS Simulation Server configurator by going to Start → Programs → ICONICS Tools → Simulator OPC Server.
  12. Go to File → Open and browse to C:\Training.
  13. Select the TrainingCourse.sim file and click on "Open."
  14. Click on File → Save to save the configuration.
  15. Close the Simulator Configurator.

- **Unified Data Manager**

  16. Start the ICONICS Unified Data Manager configurator by going to Start → All Programs → ICONICS Tools → Unified Data Manager.
  17. Go to File → Open and browse to C:\Training.
  18. Select the ACME_UDMconfig.mdb file and click on "Open."
  19. Click on File → Make Active.
  20. Click on "Yes" to the question "Do you want to set the edited database as active database?"
  21. Close the UDM Configurator.

- **Global Aliasing**

  22. Start the ICONICS Global Aliasing configurator by going to Start → All Programs → ICONICS Tools → Global Aliasing Configurator.
  23. Go to File → Open and browse to C:\Training.
  24. Select the AcmeseuticalGASConfig.mdb file and click on "Open."
  25. Click on File → Make Active.
  26. Click on "Yes" to the question, "Do you want to set the edited database as active database?"
  27. Close the Global Aliasing Configurator.

1.5. Creating Necessary Logging Databases

In this section, we will create the necessary databases that you will use in the course to log Trend and Alarm data. In this document, we will assume that you are using SQL 2005. The steps for creating databases in SQL 2008 or higher should be similar.

1. Open the SQL Server Management Studio by going to Start → Programs → Microsoft SQL Server 2005 → SQL Server Management Studio.

2. Once the Management Studio opens, you will receive a dialog that asks you to connect to an SQL Server. If this is the first time you are using the SQL Management Studio you may receive a welcome and setup dialog prior to the connection dialog. You can use the default credentials that are already filled in the dialog. Take note of these credentials. Click on the "Connect" button to connect to the server.

   a. *Please note the syntax for the server name. The syntax (local) could be replaced with the hostname of the node you're running on or simply a period which represents the local computer. SQLEXPRESS is the name of the DB that is created by default during the SQL Server Express installation.*

Figure L1.12 – SQL Server Login

3. Once you are connected, expand your server instance → Databases in the Object Explorer on the left-hand side.

4. Right-click on Database in the Object Explorer and select "New Database."

5. In the **New Database** dialog, type in "AcmeAlarmData" (without the quotes) in the Database name field.

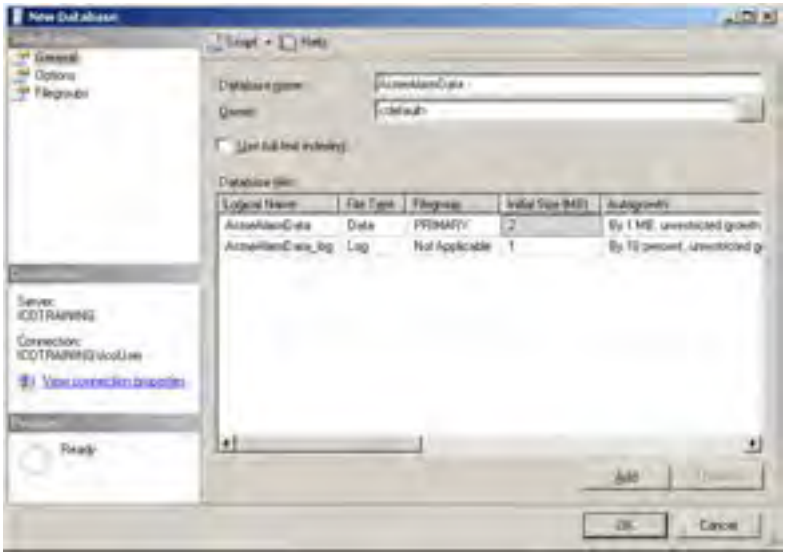


Figure L1.13 - Creating a New SQL Database

6. Click on “OK” at the bottom of the dialog to create the database. This may take a few seconds.

7. Repeat steps 4-6 to create another database called “AcmeTrendData.”

8. Once you are done with the database creation, you can close the SQL Management Studio. You are now ready to begin the training labs.

Review: Introduction to GENESIS32

R1

1.1. Chapter Objectives

At the end of this chapter, you should know and understand the following topics:

- System requirements for installing GENESIS32
- How to install GENESIS32
- What the Application Setup Utility does and why you should run it

1.2. Summary

GENESIS32 is a suite of product that includes three major components: GraphWorX32, AlarmWorX32, and TrendWorX32. Each of these components also includes different applications. Other commonly used applications within the GENESIS32 suite include Global Aliasing, Security, Unified Data Manager, and many others.

The installation of GENESIS32 is simple. You can insert the installation disk and follow the onscreen prompt to complete the installation. The installation is also flexible. You can choose to install only components you need as well as help files that are necessary for you.

During installation, the Application Setup Utility will appear. It is important that you run this utility to configure DCOM properly between the different ICONICS application. This utility also sets up your machine to run in an unattended environment.

1.3. Terms

PLC	HMI	SCADA
OPC Servers	OPC Foundation	GENESIS32
Application Setup Utility	DCOM	

1.4. Test Your Knowledge

In this chapter, you are introduced to GENESIS32. You should be able to answer the following questions.

1. What are the major components of GENESIS32?

    a. GraphWorX32, TrendWorX32, and AlarmWorX32
    b. GraphWorX32, AlarmWorX32, and Unified Data Browser
    c. GraphWorX32, TrendWorX32, and Global Aliasing
    d. GraphWorX32, TrendWorX32, and Language Aliasing

2. During installation, which other components would you have the option to install?

    a. Software Licensing
    b. Workbench32
    c. WebHMI
    d. All of the above
    e. Only A and B
    f. Only A and C
    g. None of the above

3. True or False: You must uninstall any prior version before you can install the current version.

4. True or False: You can only run the Application Setup Utility during installation.

5. True or False: Dot Net Framework is required in order to install GENESIS32.

GraphWorX32: Application Basics 2

GraphWorX32 is an OPC-based HMI graphics development tool. Using its complete set of drawing and animation tools, it is simple to create powerful live operator and visualization displays. Like many of the GENESIS32 components, GraphWorX32 is an OPC client application. It can connect to any OPC compliant server available on the market allowing you to view and control OPC data in a virtual environment that represents the real one.

Since GraphWorX32 is also an ActiveX container application, it can easily extend its functionality by inserting ActiveX controls. Using these ActiveX controls, which can be easily developed or obtained from any supplier, it becomes possible to integrate operational procedures, quality reporting, or anything else. In fact, ActiveX controls are key components in GENESIS32. They allow you to view alarms, trends, even data within a database table directly within GraphWorX32.

Furthermore, GraphWorX32 is tightly integrated with Visual Basic for Applications (VBA), which contains a rich command set, making it possible to extend the capabilities of GraphWorX32 using any application that is OLE automation compliant. In environments that do not allow VBA, GraphWorX32 can use VBScript or Jscript to fulfill your scripting needs.

In the next few chapters, we will focus our attention on GraphWorX32. By the end of this chapter, you should be familiar with the following:

- Layout of GraphWorX32
- Available Toolbars and the Symbol Library
- Configuring Display Properties and Application Preferences
- Obtaining summary information in a display
- Creating Templates and Layers

2.1. GraphWorX32 Layout

The first time you open GraphWorX32 after installation, you will probably see something similar to Figure 2.1. In many ways, this is comparable to the Windows drawing application *Paint*, yet it is quite different when you get to know it. A stark difference between the two programs is the different modes.

What you see in Figure 2.1 is GraphWorX32 in configuration mode, which is what you will use to develop your displays. When the displays are ready, you can put them into runtime mode (CTRL-M) to visualize your production floor and control necessary equipment.

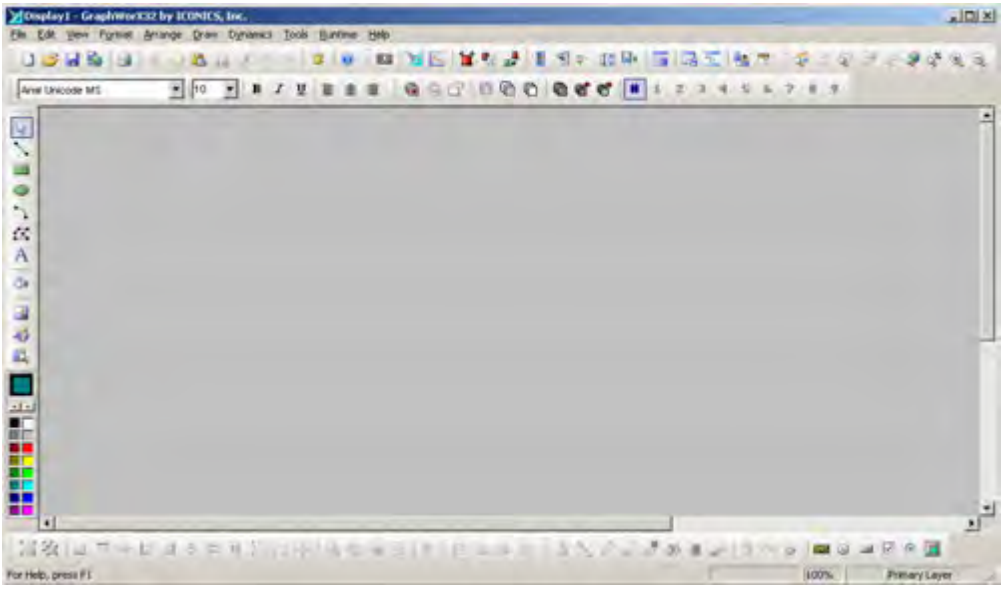


Figure 2.1 - Blank GraphWorX32 Display

Like many drawing applications, GraphWorX32 is made up of Menu options, toolbars, and a canvas or as we refer to it, the contents area. As you look around in the application, you should notice many familiar menu items and icons in various toolbars. Indeed, the menu items and available icons in toolbars are modeled after Microsoft applications to reduce the learning curve.

The toolbar arrangement that you see in Figure 2.1 is just the default. You can rearrange the toolbars to suit your work habits by dragging the header by the handle and dropping it in a new location. The handle is a vertical dotted line near the end of the toolbar. See Figure 2.2 – Toolbar Handle for more information. If you drop it outside of a docked position the toolbar will become free-floating.

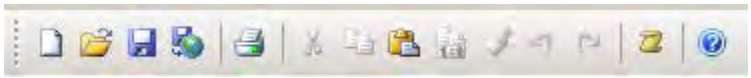


Figure 2.2 – Toolbar Handle

2.2. Toolbars

GraphWorX32 includes many tools to help you build aesthetically pleasing displays that can dynamically control your production floor. The available tools are grouped by functionality in dockable toolbars. You can also get to them by using the menu bar options. Selecting **View →Toolbars** opens the *Show Toolbars* dialog box, shown in Figure 2.3, which allows you to select the different toolbars you want to *be visible.*

The **View →Toggle Toolbars** menu hides or shows all of the toolbars at once. You can use Ctrl+T as a keyboard shortcut.

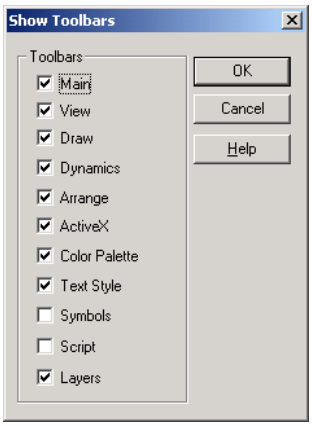


Figure 2.3 - Show Toolbar Dialog

We will take a quick look at the different available toolbars. If you need more information, you can find detail descriptions of the toolbars and menus in the GraphWorX32 Help documentation.

2.2.1. Main Toolbar

The **Main** toolbar, shown in Figure 2.4 - Main Toolbar, is displayed by default across the top of the application window, below the menu options. The toolbar provides quick mouse access to many tools used in your application. Use the **Main** toolbar for basic file and display maintenance such as creating a new display, opening a file, printing a file, and cutting, pasting and duplicating objects. It also allows you to undo and redo your last editing action, and to access help on your GraphWorX32 application.

Figure 2.4 - Main Toolbar

2.2.2. View Toolbar

Using the **View** toolbar, shown in Figure 2.5, you can change the view of the display by activating different zoom functions. All of the following functions can also be found in the first section of the **View** menu.

Figure 2.5 - View Toolbar

2.2.3. Dynamics Toolbar

You can use the **Dynamics** toolbar, shown in Figure 2.6, to make dynamic connections (e.g., action dynamics, selector dynamics, and intrinsic dynamics) to data points in OPC servers.

Figure 2.6 - Dynamics Toolbar

2.2.4. ActiveX Toolbar

The **ActiveX** toolbar, shown in Figure 2.7, enables you to insert various ActiveX objects into the GraphWorX32 display. Click on the "OLE" button to browse for a list of objects.

Figure 2.7 - ActiveX Toolbar

2.2.5. Draw Toolbar

You can use the **Draw** toolbar, shown in Figure 2.8, to draw static graphic objects; to select objects; to fill objects, including drawing objects and text objects; and to import bitmaps, metafiles, and GraphWorX32 symbols.

Figure 2.8 - Draw Toolbar

2.2.6. Arrange Toolbar

The **Arrange** toolbar, shown in Figure 2.9, is used for grouping and ungrouping objects, aligning, rotating, and shifting objects to the back or to the front inside a layer.

You should know that not all objects are created equal. While most objects can use all of the functionalities provided in the Arrange toolbar, text rotation is limited to rotating in 90-degree increments by using the “Rotate Left” and “Rotate Right” buttons.

This ability to rotate text applies to:

- Multiline text
- Text objects
- Process points
- Data entries
- State fields
- Time/date
- Buttons

Note that checkboxes and radio buttons cannot be rotated.

Figure 2.9 - Arrange Toolbar

2.2.7. Text Style Toolbar

The **Text Style** toolbar, shown in Figure 2.10, is used to change the font attributes and text alignments of text objects in a display.

Figure 2.10 - Text Style Toolbar

2.2.8. Layers Toolbar

The **Layers** toolbar, shown in Figure 2.11, gives you quick control over the layers. We will see this in more detail later in this course.

Figure 2.11 - Layers Toolbar

2.2.9. Symbol Library

GraphWorX32 comes with a Symbol Library. Although this is not necessarily a toolbar, it does contain a useful set of static and dynamic graphical objects to help you build displays.

Using the Symbol Library, you can easily find commonly used objects and quickly place them on a display. The Symbol Library includes static objects, as well as objects with previously applied dynamic actions. In many cases, you may find that the Symbol Library already contains objects that you can use inside your projects, saving you development time.

You can also build your own Symbol library. GraphWorX32 is widely used and while we continue to expand the existing Symbol Library, it could never contain objects from every industry. Once you have built objects of your own, you can simply drag-and-drop the symbols to insert your own graphical objects inside the symbol library.

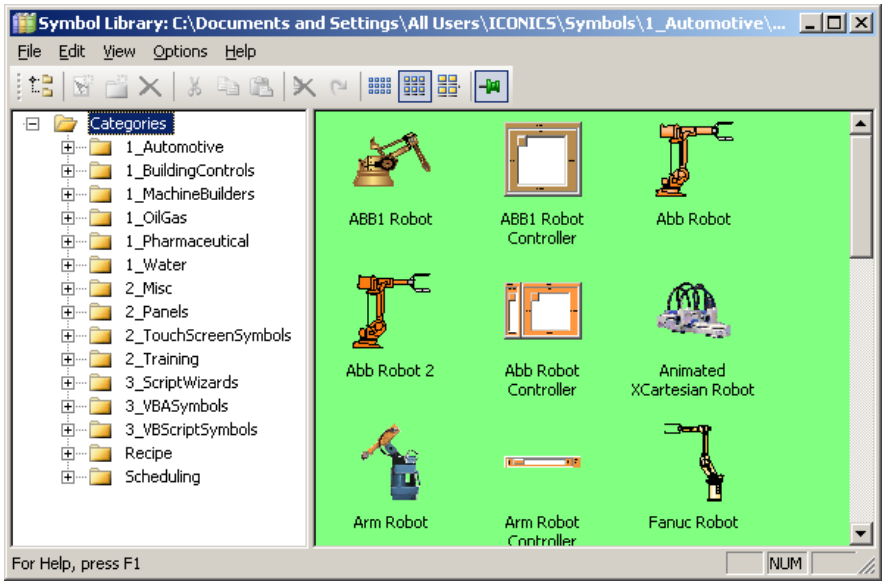


Figure 2.12 - Symbol Library

2.3. System Configuration

This section explains the different ways of configuring and customizing GraphWorX32. You can customize your GraphWorX32 system using:

- Command line options
- The Application Preferences dialog box
- The Display Properties dialog box

Options in the Application Preferences and Display Properties dialog boxes are almost identical. The difference between the two is that in Application Preferences, you will be configuring settings for the GraphWorX32 application. Any configuration set in this dialog will apply to each new display that you create.

On the other hand, you can use the Display Properties to configure settings specific to the currently opened display. Use this to overwrite properties of the displays set by the Application Preferences on occasion.

You can apply application preferences as display properties and vice-versa. To re-apply the application preferences to the currently loaded display, select **Format → Apply Preferences to Properties.** To set the current display properties as the new application preferences, select **Format → Save Properties as Preferences**.

Let us take a closer look at the two dialog boxes now.

2.3.1. Application Preferences

The **Format →Application Preferences** dialog box, shown in Figure 2.13, sets default operating parameters for your display-editing environment. These parameters apply to each new display created in GraphWorX32. You can open this dialog by selecting **Application Preferences** from the **Format** menu. Here you can set the defaults for the display. Since GraphWorX32 Preferences apply only to new displays, changes made in this dialog box will only take affect the next time you create a new display.

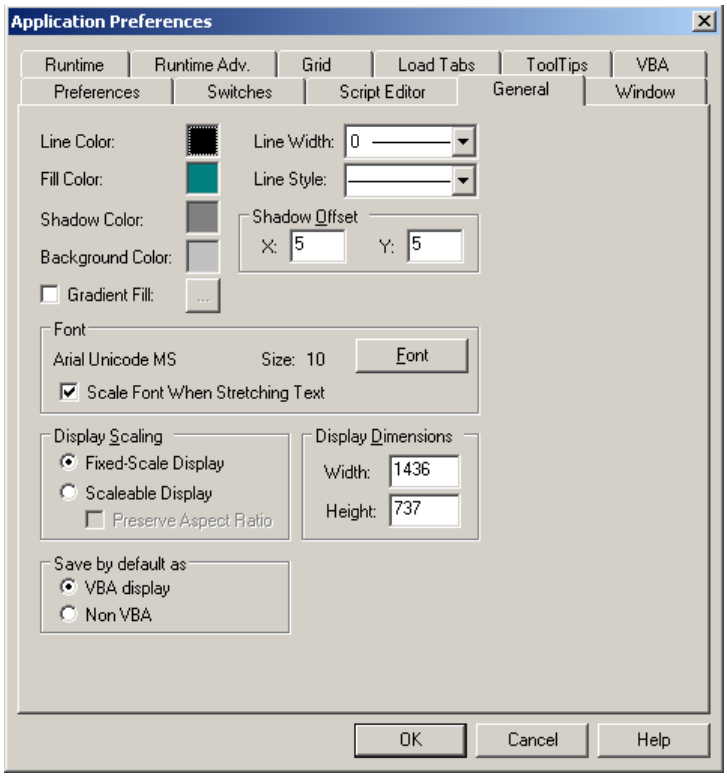


Figure 2.13 - Application Preferences

We will discuss those tabs that are specific to Application Preferences dialog in this section. We will discuss those tabs that are common to application preferences and display properties in the Display Properties section later.

2.3.1.1. Preferences Tab

On the **Preferences** tab, you can configure how the application will behave. In the Configuration Settings section, you can see that we have enabled the following by default:

- Query OPC data type and ranges during configuration
- Load previous versions without warnings
- Generate button title from tag name
- Use short tag name when creating buttons

The first option, “query OPC data type and ranges during configuration,” will allow the application to find out the low and high ranges of the OPC tag. This is useful when you are using dynamics that would depend on the range of the OPC tag. This option will also find the data type of the OPC tag and automatically set the client display type to match the data type of the OPC tag.

The second enabled option, “load previous versions without warnings,” will do exactly as it says. Without this option checked, when you load a display from a previous version, it will generate a message box saying that the display will be upgraded.

The third and fourth enabled options go hand-in-hand. The third option allows for automatic generation of the button text with the name of the tag and the fourth option will use a short tag name, usually the last bits of the OPC tag path, instead of the entire tag path.

One useful setting is to make backup copies of the displays that you are working on currently. This option is disabled by default. If you enable the “Make backup when saving displays” option, each time you save the display, it will create or update a *.bak* file in the same directory of the display using the same name. The .bak file is the display before you saved it, effectively making a version of the display prior to saving. If you find yourself needing the backup, you can always change the .bak extension back to .gdf to open the file in GraphWorX32.

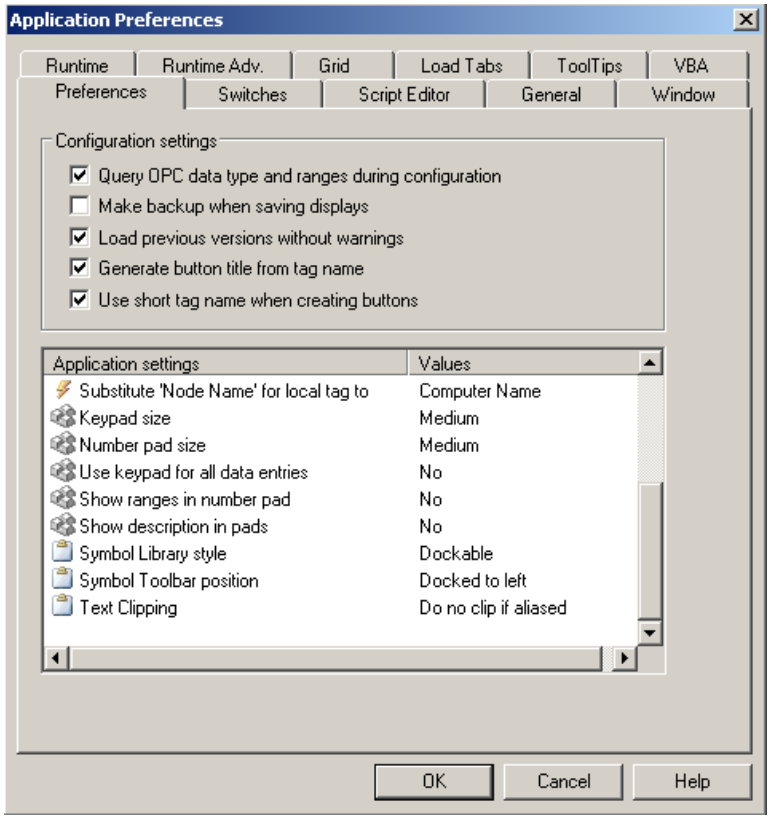


Figure 2.14 - Application Preferences: Preferences Tab

In the Preferences tab, you will also find many application settings. In general, it is unnecessary to adjust these settings. However, sometimes you may find these settings helpful.

One setting may be particularly useful when working with the Symbol Library. At the bottom of the application settings section, you will find items for the Symbol Library style and position. As you may have already known, the Symbol Library, like many other toolbars, can be a free-floating window as well as a docked window. You can configure this in the "Symbol Library style." Furthermore, if you choose to dock the Symbol Library, you can configure where to dock it using the "Symbol Toolbar position" setting.

2.3.1.2. Switches Tab

GraphWorX32 contains many switches, or optional settings, to control how the application behaves. You will find many of these switches in the **Switches** Tab of the Application Preferences dialog box. Similar to the application settings, in most cases, you will never need to configure these switches. However, you should know a few of the most commonly used switches.

Figure 2.15 - Application Preferences: Switches Tab

First, many times in a production environment, you may want a display to startup the application runtime mode as opposed to the default configuration mode. You can do this by checking the "Launch application in Runtime mode" switch. Although this setting launches the application in runtime mode and you can prevent users from going into configuration mode with security, sometimes it is just necessary to disable configuration mode completely. You can do that by checking the "Disable Configuration mode" switch.

> **NOTE:** If you disabled configuration mode, you will only be able to get back into configuration mode by editing the *icoSetup.ini* file.

Other useful switches include “Suppress security warning” and “Suppress security login.” The first switch will disable a popup message when there is insufficient security permission to open a display. The second switch will stop the login dialog from automatically coming up if the application detects that there is insufficient security to perform an action.

2.3.1.3. Script Editor Tab

GraphWorX32 is a powerful HMI design application. It supports three different scripting languages: VBA, VBScript, and JScript. In the **Script Editor** tab, you can configure keywords and constants for VBScript and Jscript. Since VBA has its own scripting environment, you will need to edit the constants and keyword for VBA in the VBA environment.

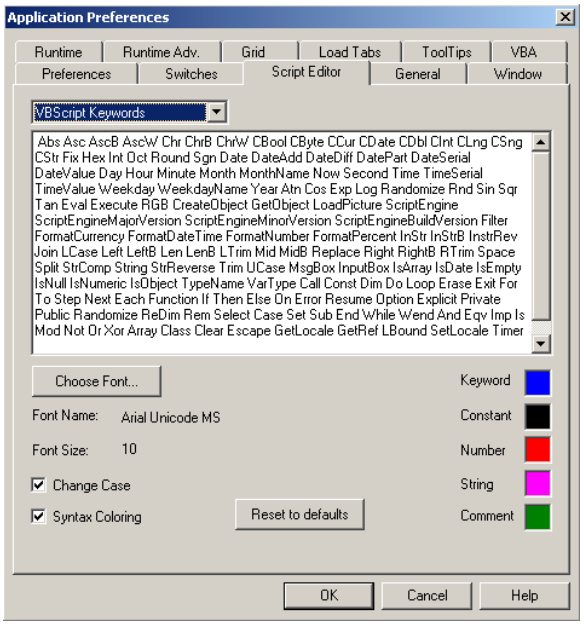


Figure 2.16 - Application Preferences: Script Editor Tab

As you can see from Figure 2.16, there are already defined keywords and constants for both VBScript and JScript. If you need to define your own, you can add them into the appropriate textbox.

On this tab, you can also define the text style in the script editor. You can choose the font face, the font size, and text colors for keywords, constants, numbers, strings, and comments. If you ever need to change anything back to default, you can use the “Reset to Default” button to revert your changes.

2.3.1.4. Set Working Directory

One of the first things you may want to do when you start a project is to set a working directory. While this is not an option in the Application Preferences dialog, it is a setting available in GraphWorX32.

To set your working directory, you can click on the "Set Working Directory…" option in the **Tools** menu. This will bring up the **Set Working Directory** dialog box, and you can specify your project directory here. With this configured, when you save a file, it will save in this directory by default. Figure 2.17 shows the default working directory.

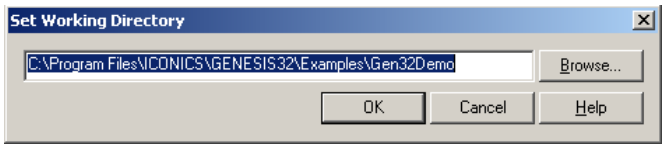


Figure 2.17 - Set Working Directory

2.3.2. Display Properties

Now that we know how to configure some of the settings for the GraphWorX32, let us look at individual display settings. You can define the properties for the current display by selecting the **Format→Display Properties** menu. This opens the Display Properties dialog box, shown in Figure 2.18. Any changes you make in this dialog box will be reflected in the current display.

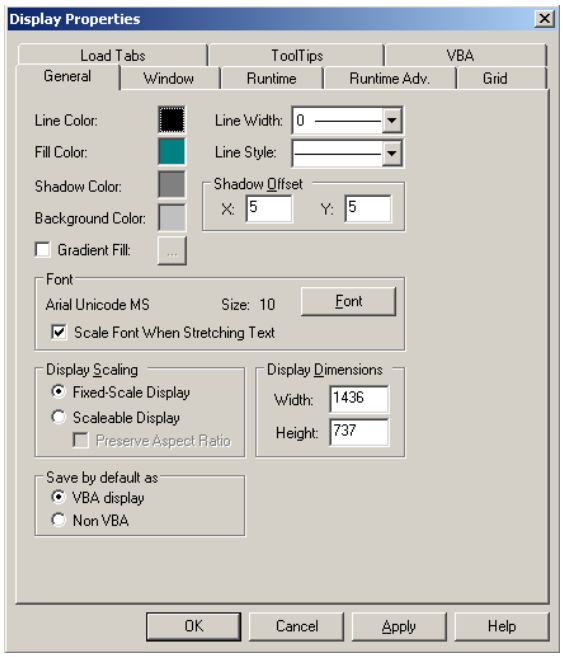


Figure 2.18 - Display Properties

As you can see, the **Display Properties** dialog box is very similar to the **Application Preferences** dialog box. Since we already discussed those tabs only available in application preferences, the remaining common tabs are:

- General
- Window
- Runtime
- Runtime Adv. (Advanced)
- Grid
- Load Tabs
- ToolTips
- VBA

In addition to the "Font" scaling option, the **General** tab of the **Display Properties** also offers options for the scalability of the display itself. The default is a "Fixed Scale Display," which will not change its size depending on the screen resolution. When desired, a display can be made scalable by selecting the "*Scalable Display*" option. The display will then resize depending on the screen resolution. When using scalable displays, you may also want to enable "*Preserve Aspect Ratio*" to avoid distortion of the screen.

The **General** tab also provides the settings for the size of the display. In the *Display Dimensions* section, you can enter the Width and Height of the display. You should not confuse the display dimensions with the *Window Dimensions* on the **Window** tab. The windows dimension can only be as big as your monitor resolution size, while the display size can be bigger than your monitor size. You can consider the window size being the size of the container for your display. You may want to check the "Scale window dimensions based on current screen resolution" setting if you have selected to scale the display.

Other options that can be modified from the **Window** tab include the visibility of the title bar and menu during runtime mode, and whether or not the display can resize using its frame. You can also define scrollbars and options to save the scrollbar state, view position, and window position.

The **Runtime** tab contains the configuration of the "*Runtime Focus Highlights*" and the "*Disabled Object Colors*." There is also one very important property called "*Maximum Update Rate*." The default value for this is 1000 milliseconds; for nearly all applications, a value of 500 or 1000 would be sufficient.

> **NOTE:** In older versions of GraphWorX32, the default Maximum Update Rate is 50 milliseconds. If you are experiencing slower updates to OPC tags in this version compared to prior versions, this could be the reason.

GraphWorX32 uses the “Maximum Update Rate” when communicating with the OPC Server. When GraphWorX32 creates its connection to the OPC server, it specifies the “Maximum Update Rate” to the OPC server, effectively requesting that the OPC server send its updates at this rate.

Lastly, on the Grid tab, you can define the style of a grid in the display. Sometimes it is useful to have a grid when aligning objects. You can show the grid by checking the “Show Grid” option. You can define the grid spacing by entering values in the Width and Height fields and specify the grid style. You can also check the “Snap to Grid” option, which can be helpful when aligning objects.

Any of the settings that are set inside the **Display Properties** dialog can also be set as application preferences. When setting them as application preferences, each newly created display will automatically inherit the settings as specified inside these application preferences. Of course, it is still possible to override the application preferences inside the Display Properties.

2.4. Display Summary

Oftentimes it is necessary to obtain a summary about the display. Whether you want to know the point count so that you will not exceed the limit, or you just want to know which OPC tags you are using in the display, GraphWorX32 has tools to help you find out such information.

2.4.1. Object Count

By selecting **View →Object Count**, you can easily find out how many static and dynamic objects are on your display as well as the number of OPC tags (or process points).

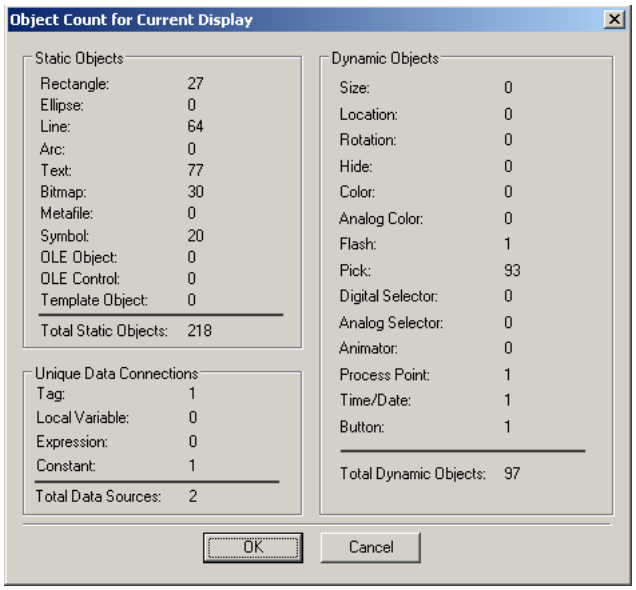


Figure 2.19 - Object Count for Tutor.gdf

Figure 2.19 shows the object count for the Learning Center homepage (Tutor.gdf) display. Using this dialog, it is effortless to see that there are 218 static objects on the display. We can also tell that the display only uses one OPC tag, and is therefore hardly ever going to exceed any licensing limits.

As far as dynamic objects go, 93 pick actions in the dialog hold the ability to perform actions such as navigating to different displays and writing values to OPC tags when the display is in runtime mode.

2.4.2. OPC Connections

While we can easily see the object count on a display, it does not tell us which OPC tags are being used. To find out which OPC tags are employed in the display, we need to use a different tool. In fact, GraphWorX32 offers multiple tools that you can use to find out which OPC tags are used in the display. You can choose between dialogs such as:

- Edit Connections
- Find/Replace
- Report

By selecting the **Dynamics →Edit Connections** menu you can bring up the *Edit Data Source Connections* dialog. If nothing was selected when you brought up the dialog, it will show you the data sources, including high and low ranges of the data source found in the display. If there was an object selected when you brought up the dialog, it will only show you data connections for the selected objects.

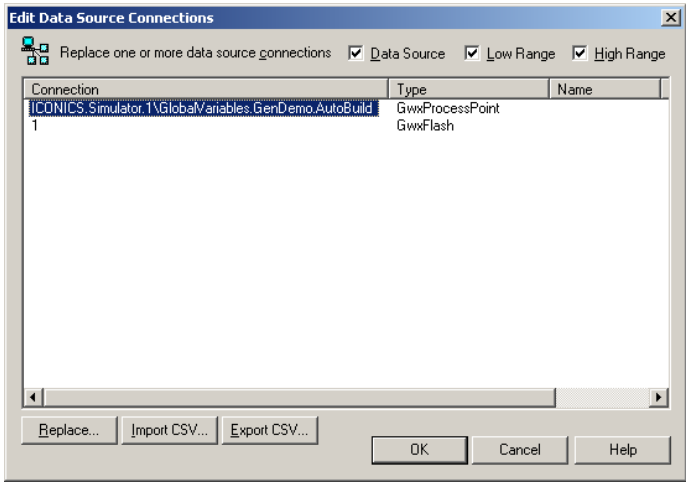


Figure 2.20 - Edit Data Source Connections

In the Edit Data Source Connections dialog, you can click on the "Replace…" button to change a data source connection. You can also export it into a comma separated value (CSV) file. If the objects in the display have object names, then you can edit the CSV file and import the changes by using the export and import buttons.

From the **Edit** menu you can access the **Find, Replace,** and **Report** dialogs. Each dialog is similar to the other and has comparable functionality. As you can see from Figure 2.21, the Report and Replace dialogs are very similar.

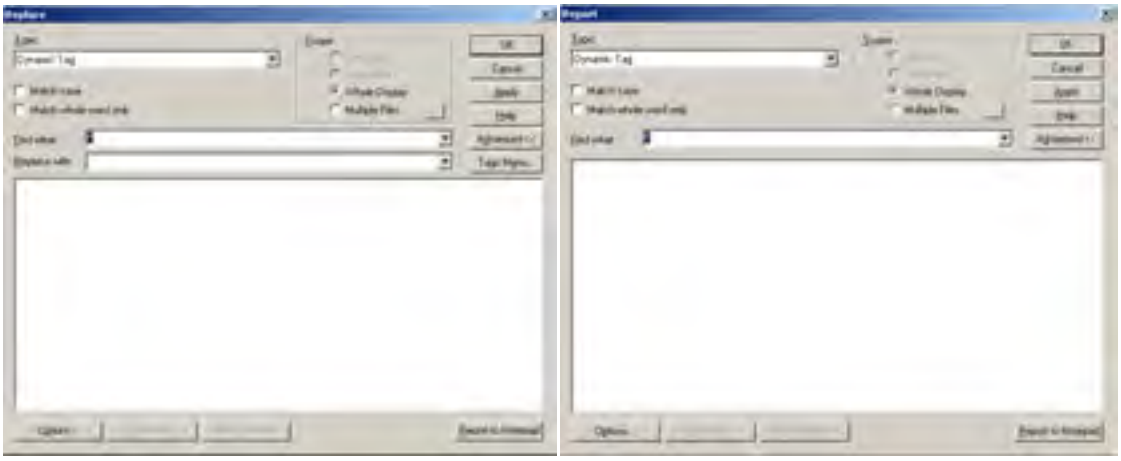


Figure 2.21 - Replace (left) and Report (right) Dialogs

The default option in these dialogs is to find dynamic tags with an asterisk (*) in the "Find what" field. This acts like a wildcard character telling the system to find everything. With this setting, when you click on the "Apply" button, it will return results similar to those shown in Figure 2.22.

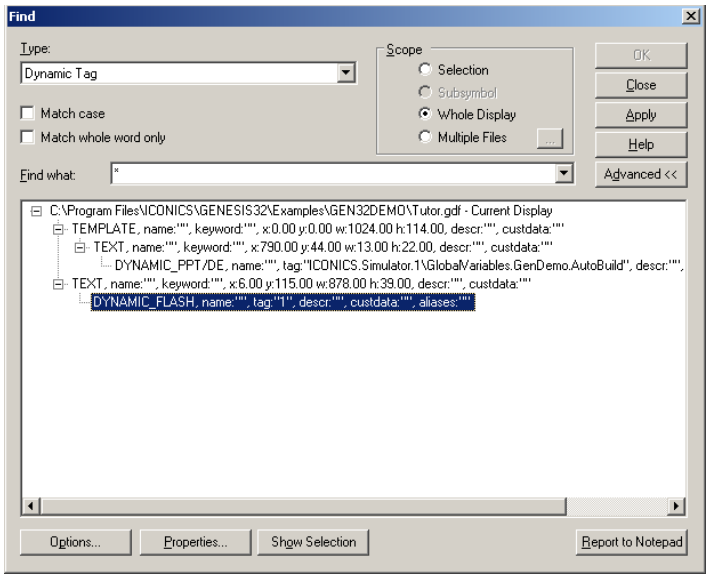


Figure 2.22 - Find Dialog with Results

Selecting a dynamic in the results list will enable the "Properties…" and "Show Selection" buttons. Clicking on the "**Properties**" button will bring up the properties inspector dialog for the selected dynamic. You can make changes directly in the properties inspector. You can also use the "**Show Selection**" button to find out the location of the dynamic on the display. An additional function allowed in this dialog is that you can find dynamic tags used in multiple displays. The Edit Data Source Connections dialog only searches for connections in the working display where you brought up the dialog.

The "**Report to Notepad**" button on the lower right-hand corner allows you to export the information from the results window into the windows text editing application *Notepad*. While this functionality is similar to the CSV export available in the Edit Data Source Connections dialog, you will not be able to import changes made in the notepad file back into GraphWorX32.

2.5. GraphWorX32 Templates

Templates are standard formats with established parameters that you can apply to a display at any time in configuration mode. A GraphWorX32 template file uses a *.tdf* (template display file) extension to differentiate from the regular display file with a *.gdf* (graphic display file) extension, which is the default file extension when saving a display.

It is simple to create a GraphWorX32 template. Start GraphWorX32 and insert all the graphic objects, as you would normally do when creating a display file.

The only difference is when you save the display; in the **Save As** dialog, you change the "**Save as type**" option to "**GraphWorX32 Templates (*.tdf)**." This way, you are telling the system that this is a template file instead of a regular display file.

Once you have created a template file, you can open a regular GraphWorX32 display and apply the template to it, or double-click on the template file to start creating a new display. Similar to Microsoft Office templates, when you double-click on a GraphWorX32 template file you are not opening the template file; you are actually creating a new display based on the template. If you need to edit the template file, you can open GraphWorX32 first and browse to open the template file.

After you apply a template to your displays, you may find that you need to make changes to the template. It is easy to reflect changes made to the template in other displays that were created from the template. Upon saving a changed template you will be prompted to "update display files." Answering yes will open a dialog box where you can select the displays to update with the current template. Alternatively, you can update display files manually by selecting the preferred option from the **Format →Template** menu. Please note that just changing the template will not update the displays containing this template.

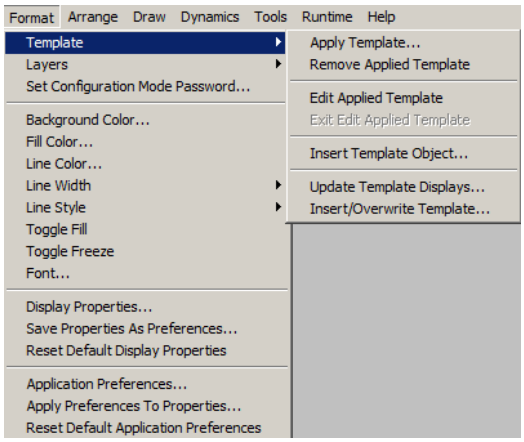


Figure 2.23 – Format Template Options

2.6. Layers

The idea of layers in GraphWorX32 is similar to other graphics editing tools like *Photoshop*® and *AutoCAD*®; you use layers for different levels of information or security levels. The bottom most layer, or layer 0, is always visible and should be the layer to put your base objects.

Unlike other software that uses layers, in GraphWorX32 you can use layers to display information selectively based on user choice or currently active security levels. Additionally, use layers to clutter and de-clutter the screen content based on zoom level.

By using the **Format →Layers** command you can add, remove, or duplicate a layer. You can also edit the layer properties as shown in Figure 2.24 - Edit Layer Properties, set the active layer, set the next layer, set the previous layer, hide layers above the current layer and hide layers below the current layer. Hiding the layers above and below the current layer is useful when configuring one of the layers and do not want to see the others.

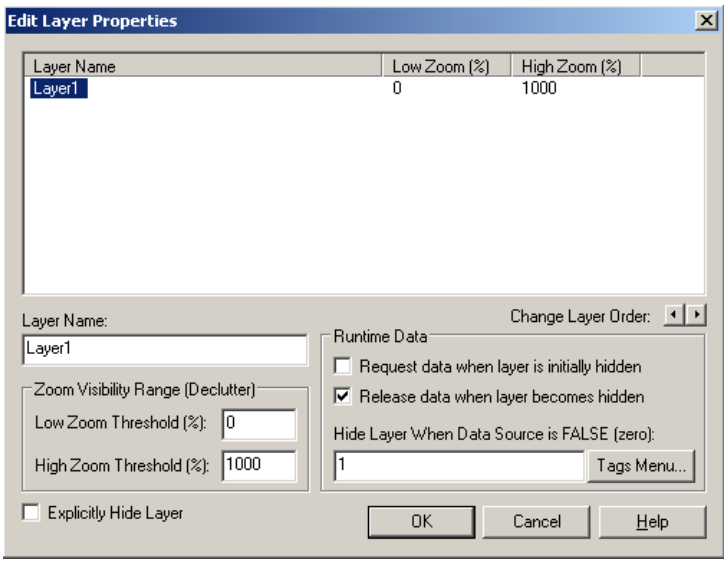


Figure 2.24 - Edit Layer Properties

You will notice in the layer configuration that the visibility of a layer in Runtime can be dependent upon the result of an expression, or explicitly hidden. There are pick actions that can control a layer's visibility as well.

It is also important to note that there is no limit to the number of layers in a display, although the **Layers** toolbar only shows buttons for nine layers.

Lab: Acmeseutical Graphics Templates *L2*

Acmeseutical, an international pharmaceutical company that specializes in pain medication has just hired you to create an HMI/SCADA system for their US facilities. The entire project will include creating displays for their operators, generating alarms, and trending their production status. The system will eventually expand to include other locations worldwide and therefore, we will eventually implement security on the system.

In the first step towards creating their system, we will generate some graphical displays in which their operators can control the production. Before we begin creating specific displays, we will create a template so that all of the displays will use a standardized format. This lab will take you through the steps of creating such a template. We will use some of the tools discussed in this chapter as well as some others that we will expose you to in depth later on.

Estimated time to complete this lab: 20 minutes

2.1. Task Checklist

In this lab, you will complete the following tasks:

- Configure display Properties
- Use the Symbols toolbar to add objects onto the display
- Save a simple GraphWorX32 template

2.2. Support Files

You do not need any support files to complete this lab.

2.3. Configuring the Working Environment

In this section, we will configure the display settings for our template such that when you use the template to create a display, it will inherit all of the settings.

1. Launch GraphWorX32 by going to Start → Programs → ICONICS GENESIS32 → GraphWorX32 → GraphWorX32.

2. Once the application opens, click on the Format menu and select the "Display Properties…" option.

3. On the General tab, select the "Scalable Display" option in the Display Scaling section. Check the "Preserve Aspect Ration" option as well. These settings are ideal for having a flexible display during configuration or when other applications are used on the workstation.

4. Set the Display Dimensions to use a width of 1024 and a Height of 768 if the settings are different.

   NOTE: The Display Dimensions and Windows Dimension are by default set to the monitor resolution.

5. Select the "Non VBA" option in the Save by default section. The General tab should look similar to Figure L2.1.

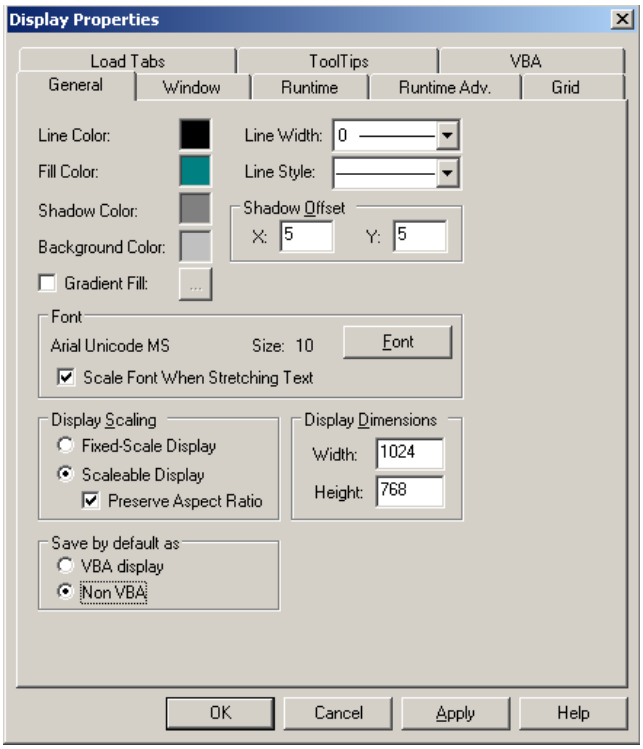


Figure L2.1 - Display Properties: General Tab

6. Go to the Window tab, and check the "Scale window dimensions based on current screen resolution" option. This way, when the display is moved to other machines with different resolution, the dimensions should look the same.

7. Also, check the “Start Maximized” button so that the display will be maximized to the screen when we begin.

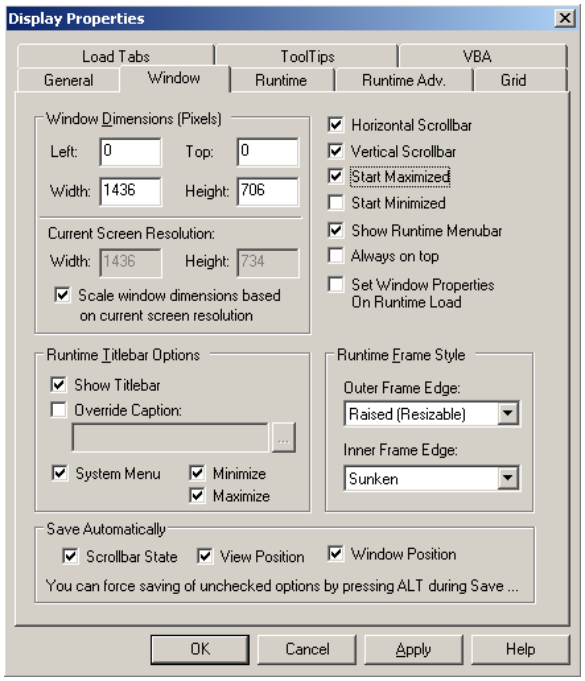


Figure L2.2 - Display Properties: Window Tab

8. Select the Runtime tab. Click on “Fill Color” in the Runtime Focus Highlights section and change the color of the focus rectangle to blue. (Red: 0; Green: 0; Blue: 255) as shown in Figure L2.3.

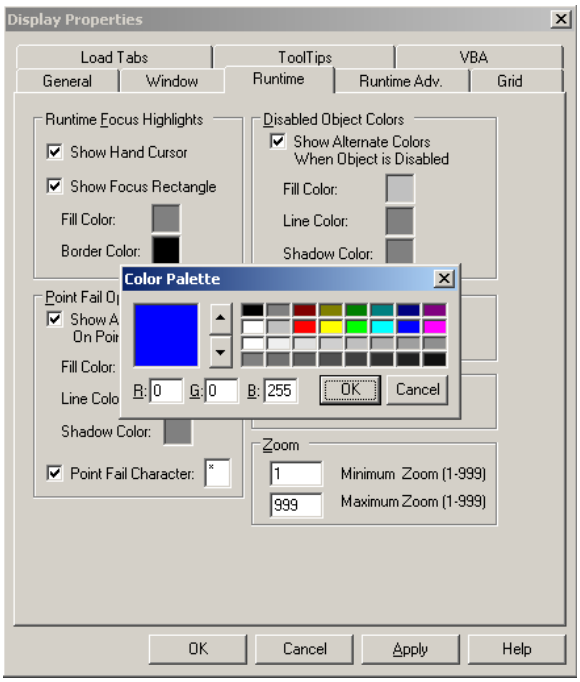


Figure L2.3 - Display Properties: Runtime Tab

9. Click on “OK” to close the Color Palette dialog. Click on the “Border Color” and change the color to yellow (Red: 255; Green: 255; Blue 0:). Click on “OK” to close the Color Palette dialog.

10. Select the Grid tab and change the “Grid Width” and “Grid Height” to 25.

11. Enable the “Show Grid” options as shown in Figure L2.4.

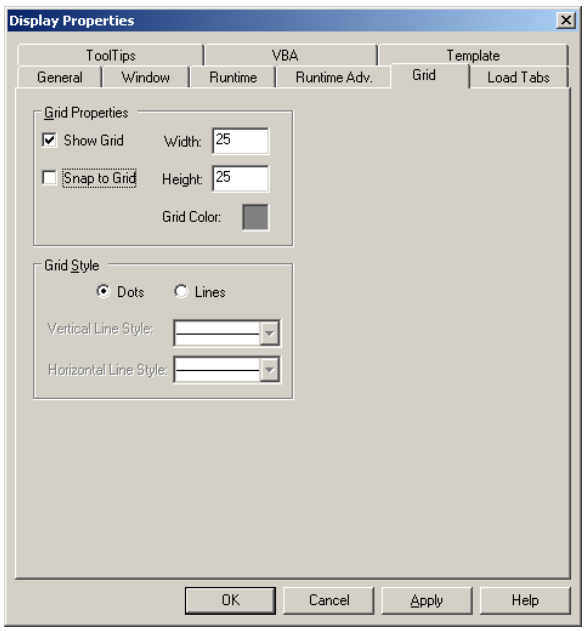


Figure L2.4 - Display Properties: Grid Tab

12. Click on “OK” to apply the changes and close the “Display Properties” dialog.

> **NOTE:** If you want to use these settings throughout the project, you can go to Format →Save Properties as Preferences to force these settings for any new GraphWorX32 displays.

13. Save the display to C:\Training and name it “AcmeseuticalTemplate.” Make sure that the Save as Type is “GraphWorX32 Templates without VBA (*.tdf)”. Do NOT close this display.

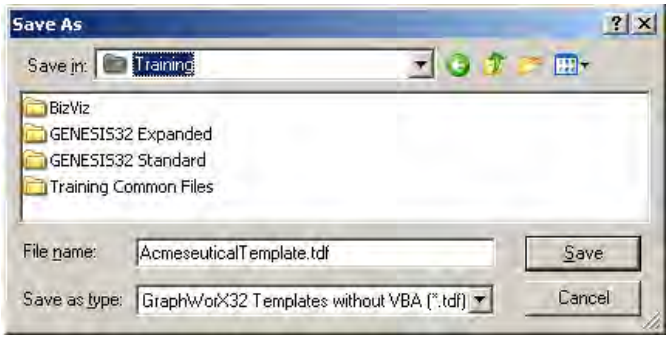


Figure L2.5 - Save as Dialog

14. We will save our displays to the Training folder from here on. Let's set that as our working directory.

15. Click on the **Tools →Set Working Directory** menu.

16. In the Set Working Directory dialog, browse to the Training folder and click on "OK" to save the setting and close the dialog.

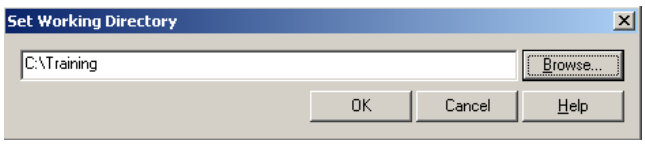


Figure L2.6 - Set Working Directory

2.4. Creating a Template for Acmeseutical

After having set up the environment of GraphWorX32, we will create a template for all Acmeseutical displays to standardize the look and feel.

Now that we have saved a template display, there is one more Display setting that we need to configure.

1. Select **Format →Display Properties** again and notice that there is now a Template tab.

2. Go to the Template tab and check the "All Properties" section. This will allow any new display created using the template to inherit all of its properties.

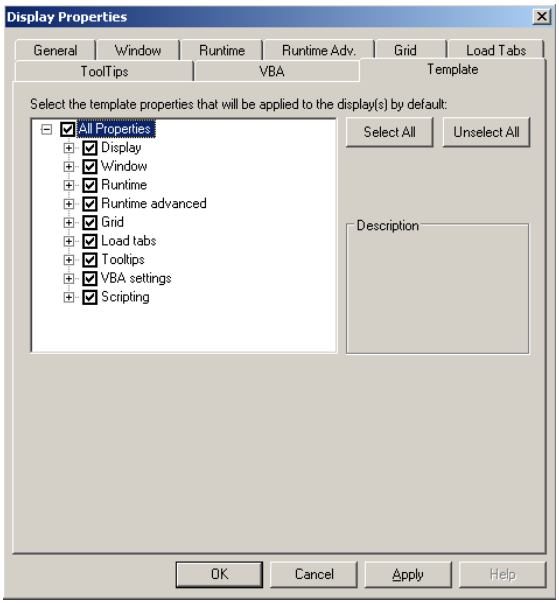


Figure L2.7 - Display Properties: Template Tab

3. Click on “OK” to apply the settings and close the Display Properties dialog.

4. If you do not see the Symbols Library, you should open it now by clicking on the “Symbols” button in the Draw Toolbar. You can also open or close the symbol toolbar with ctrl-alt-s.

5. Open the “2_Training” folder and double-click on the “TrainingCourseV9.22.sdf” file. Drag the “Acmeseutical Logo” onto the display and place it in the upper left-hand corner.

6. Also, drag the “Home,” “Next,” and “Previous” icons onto the display and place them in the upper right-hand corner.

7. Add a Text Label by clicking on the “Text” icon in the Draw toolbar and then drawing the size of the label on the display.

8. Double click on the selected Label to launch the Property Inspector. Set the fill color to white (Red: 255; Green: 255; Blue: 255). Also, set the 3D edge to Sunken.

9. The text of this label should read “Global Remote Engineering” and set the text color to a dark blue (Red: 0; Green: 0; Blue: 128).

10. Uncheck the “Stretch Text on Resize” option.

11. Click in the Font button. Change Size: to 14 and Font Style: to Bold. Click OK.

12. Below the Font button, change the text alignment to Text: Center and Bound: Center.

13. Set the description of the label to “This is a G.R.E.A.T. System.” The finish configuration for the label should be similar to what is shown in Figure L2.8.

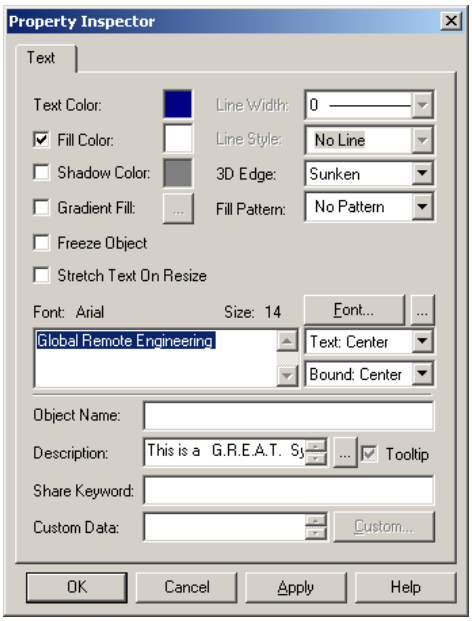


Figure L2.8 - Text Label Property Inspector

14. Add another text label to be as wide as the display and place it below the back and forward buttons leaving enough space for date and time. Change the "Fill Color:" to Gray (Red: 192; Green: 192; Blue: 192).

15. Check the "Gradient Fill:" option and click on the ellipses (…) button. In the Configure Gradient dialog, choose the third Quick Set.

16. Click on "OK" to close the Configure Gradient dialog.

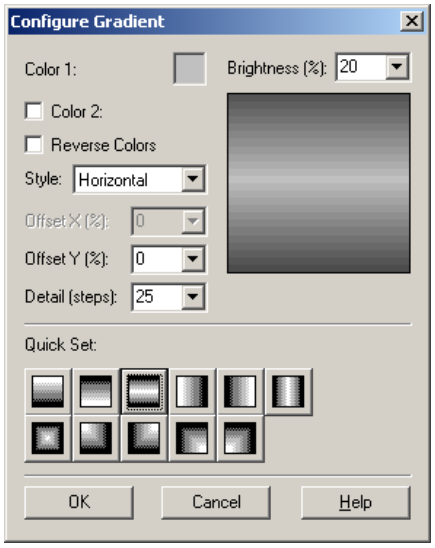


Figure L2.9 - Configure Gradient

17. Uncheck the "Stretch Text On Resize" option.

18. Click on the Font button. Set the Size: to 14. We will bring in some text here in a later lab, leave it blank for now.

19. Lastly, change the Text Color: to a dark purple (Red: 128; Green: 0; Blue: 128). Your finished configuration for this label should look similar to Figure L2.10. Click "OK."

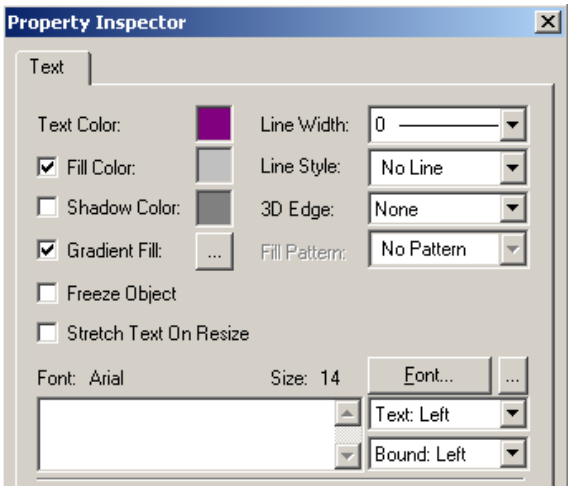


Figure L2.10 - Configuration for Text Label 2

20. On the Dynamics toolbar, click on the "Time/Date" button and then click in the display. Set the Display to show Date and Time. Specify the Time Format: as HH:mm:ss and the Date Format: as dddd, MMMM dd, yyyy.

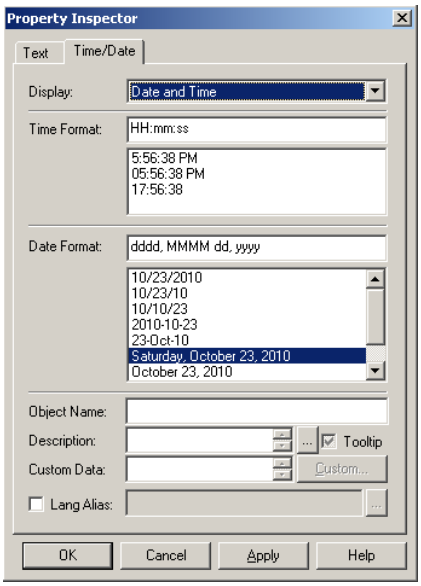


Figure L2.11 - Date Time Object Configuration

21. Select the Text tab. Below the Font button, change the text alignment to Text:Center and Bound:Center. Click on “OK” to save the configuration and close the dialog.

22. Arrange the objects you have created so that it looks similar to Figure L2.12.

Figure L2.12 - Finished Template

23. Save the template but do not close it. We will continue to work on it in the next lab.

Review: GraphWorX32 Application Basics

R2

2.1. Chapter Review

You should know and understand the following topics:

- GraphWorX32 Application Layout
- Available Toolbars
- Symbols Library
- Application Preferences and Display Properties
- Templates
- Layers

2.2. Summary

GraphWorX32 is similar to many Windows applications; it contains menu options, toolbars, and a contents area where you can create your graphical display. The menu items and the toolbars are logically grouped by functionality. You can change the locations of where the toolbars dock to best suite your work habits. The toolbars can become free-floating by dragging them out of the dock locations.

One of the most power toolbars in GraphWorX32 is the Symbols Library. It contains a set of static and dynamic objects that can help you build your display quickly. If you do not find the objects you need in the Symbols Library, you can create it once and save it in the library for later use with a simple drag-and drop option.

Before you start creating displays, it is best if you configure the Application Preferences first. Any new display that you create is based on the Application Preferences. You can set the display size, background color, and many other default settings in the Application Preferences dialog. The difference is that the configured settings in the Display Properties dialog only apply to the currently open display while the Application Preferences will apply to any newly created displays.

When you are creating displays, you may need to find summary information on the display. The information may include the number of objects and OPC tags on the display. You can easily obtain this information by using the Object Count item in the View menu and the Edit Connections item in the Dynamics menu. You will find information about how many static and dynamic objects are on your display in the Object Count for Current Display dialog. In the Edit Data Source Connections dialog, you will find the different OPC tags used on the display. You can also find out the different OPC tags used in the display by using the Find, Replace, or the Report options under the View menu.

Templates are standard formats with established parameters that can be applied to a display at any time in configuration mode. GraphWorX32 supports two file extensions: .gdf (graphic display file), the default file extension, and .tdf (template display file). Once you insert a template into a display, it is easy to reflect changes made to the template in the display. Whenever you make changes to the template, simply apply them to all affected displays, and the changes will be visible throughout all screens.

Layers allow you to display different information based on user selection or currently active security levels. Additionally, you can use layers to clutter and de-clutter the screen content based on zoom level. You can create layers with the Add Layer item in the Format menu. Although the Layer toolbar only shows nine layers, you are allowed to create as many layers as you wish.

2.3. Terms

Application Preferences	Configuration Mode	Contents Area
Display Properties	Dynamic Objects	GraphWorX32
GraphWorX32 Layers	GraphWorX32 Template	Process Point
Runtime mode	Symbols Library	

2.4. Test Your Knowledge

In this chapter, we discussed many settings for GraphWorX32. Before you continue to the next chapter, please make sure that you can answer the following questions to gauge your understanding of the chapter.

1. True or False: You can rearrange the location of the docked toolbars in GraphWorX32.

2. True or False: The Symbols Library is static; you cannot add your own symbols into the library.

3. Which of the following can you use to see the different OPC tags you have in your display?

   a. Find Dialog
   b. Edit Data Source Connections Dialog
   c. Report Dialog
   d. All of the Above
   e. None of the Above

4. Which of the following is the file extension for a GraphWorX32 template file?

   a. .gdf
   b. .tdf
   c. .gdfx
   d. .tdfx

5. How many layers can you have in a single GraphWorX32 display?

   a. 1
   b. 9
   c. 10
   d. unlimited

6. True or False: Once you apply a template to a display, it is a complicated process to change the applied template.

7. What is the difference between Application Preferences and Display Properties?

    a. They are the same, just named differently.
    b. Settings in Display Properties apply to all newly created displays while Application Preferences settings only apply to the currently opened display.
    c. Settings in Application Preferences apply to all newly created displays while Display Properties settings only apply to the currently opened display.
    d. Settings in Application Preferences apply to all ICONICS applications while settings in Display Properties only apply to GraphWorX32.

8. True or False: Display Dimensions and Window Dimensions refer to the same thing.

9. True or False: Once you disabled configuration mode in GraphWorX32, the only way to re-enable it is to edit the setting in the icoSetup.ini file.

10. Where is the default working directory?

    a. C:\Program Files\ICONICS\GENESIS32\Examples\Gen32Demo
    b. C:\Documents and Settings\All Users\ICONICS\Examples32
    c. C:\Program Files\ICONICS\GENESIS32
    d. C:\Documents and Settings\All Users\Application Data\ICONICS

GraphWorX32: An Animation Platform 3

In the previous chapter, we learned how to configure GraphWorX32 settings so that it would use your customized settings when creating new displays. In this chapter, we will discuss individual toolbars and some of their functionalities in detail. By the end of the chapter, you should be familiar with the following:

- Draw toolbar
- Color Palette
- Arrange toolbar
- Various ways to use a Process Point
- Size, location, hide/disable, and color dynamics

At this point, you should have personalized your GraphWorX32 environment in such a way that it suits your work habits. If you have not done that, it may be beneficial to do that before you begin the chapter.

If Genesis32 is installed you may find it very useful to reference the Tutor.gdf file through Gen32Demo. Select Start →All Programs →ICONICS GENESIS32 →Gen32Demo. Click on the **Learning Center** icon in the upper right. Note the tool tips as you fly by the icons.

3.1. Draw Toolbar

The Draw toolbar provides many drawing functions. You can create lines, segments, rectangles, ellipse, arcs, and text objects using the icons in the toolbar. Since most of these tools are very basic and similar to those in *Paint*, we will not take too much time to discuss these functions. If you find yourself uncertain about a property of these objects, you can find more information in the related help files as well as examples in the Learning Center.

3.1.1. Rectangle

One of the most primitive objects you can have on a display is a rectangle. To insert a rectangle, simply click on the "Rectangle/Square" icon in the draw toolbar, your mouse would change to a cross hair with a square in the center.

You can then click anywhere in the contents area, while holding down the left mouse button, drag the mouse to create a rectangle with a desired size. If you want to create a square, simply hold down the Shift key while you drag your mouse to create the object.

> ***Tip!*** Holding down the Shift key when you are creating an object makes the object use the same width and height value. When you are resizing an object, holding down the Shift key will preserve the aspect ratio.

Once you create the desired object, you can continue to create more rectangles. When you are done creating rectangles, you can escape the tool a few ways. First, you can choose to draw different objects by clicking on the respective icon in the draw toolbar. You can also click on the "Select" icon in the draw toolbar to return to the regular mouse pointer; double-clicking in the contents area or hitting the escape (Esc) key has the same effect.

Once you have the object in the display, you can double-click on it to bring up the Property Inspector. Here you can change the appearance of the object. You can change the line width, style, and color as well as the fill color and pattern for the rectangle. If you wish, you can add a shadow to the object to make the scene look more three dimensional.

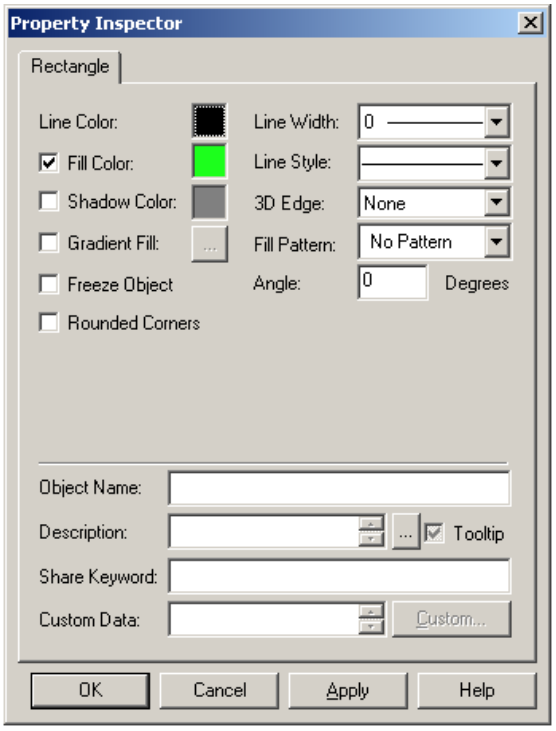


Figure 3.1 - Property Inspector

A notable option in the Property Inspector is the "Gradient Fill" option. Clicking on the ellipses (…) button next to the option brings up the **Configure Gradient** dialog. In this dialog, you can set a second color for the object to use. This is very useful when you want to add light and shadow effects to an object.

In the dialog, you can configure a second color, the direction of the gradient, and the amount of steps the two configured colors will fade into each other.

Figure 3.2 shows two rectangles. The one on the left contains default properties (the same ones you saw in Figure 3.1). The one on the right is a duplicate of the same rectangle with extra properties enabled. The rectangle on the right includes a shadow and a gradient, as well as rounded corners. As you can see, the two objects are quite different. By enabling extra properties, we were able to create a more sophisticated looking object.

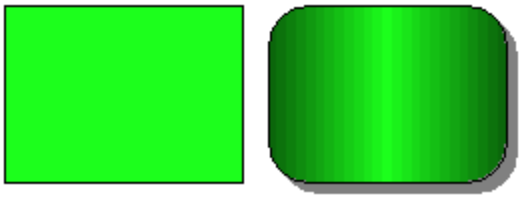


Figure 3.2 - Regular Rectangle (left) and a Rectangle with Extra Properties (right)

All objects that you can create using the Draw toolbar will contain similar properties to help you create life-like objects. You can find some examples of how these properties can change an object by looking at the respective Learning Center display.

3.1.2. Segmented Line

While most properties are the same for the different objects that you can create from the Draw toolbar, we will discuss one other function. The "Segmented Line" icon in the Draw toolbar allows you to draw anything in freehand.

This tool is somewhat a combination of the pencil and polygon tool in *Paint*. You can draw straight lines by clicking on the screen where you want the line segments to be and you can draw curvy lines by holding down the mouse button and dragging your mouse to where you want the curvy line to go.

3.1.3. Import Images and Meta Files

In the Draw toolbar, you will also find icons such as “Import Image” and “Import Metafile”. Using the “Import Image” function, you can insert any image file into GraphWorX32. There is a range of image file formats that we support. Just to name a few, you can import simple images created in *Paint* (.bmp) or very complex image files created in *Adobe Photoshop* (.psd). You can even import animated image (.gif) files into GraphWorX32 and retain the animation. There are many more supported image file formats. You can refer to the help files for a complete list.

GraphWorX32 also supports importing metafiles for your graphics. The metafiles can be either Windows Metafile (.wmf) or Extended Windows Metafile (.emf).

3.2. Color Palette Toolbar

The Color Palette is located below the Draw toolbar. At the top you will see the last used fill color and line color for an object. There is a palette at the bottom showing different default available colors. You can click on the arrows above the list of colors to scroll through the different available ones.

You do not always need to go into the Properties inspector to change the fill color or line color of an object. With the object selected, simply left-click on the color that you wish to set as the object fill color. If you right-click on the color, you will set the corresponding line color for the object.

> ***Tip!*** You can select multiple objects by holding down the Shift key while you select the objects or enclose the objects in the selection square and set them all to use the same fill color or line color at the same time.

If you do not find a color that you need in the color palette, but you see it in an image on your display, you can use the **Eyedropper** tool to add the color into the palette. First, using the arrow keys, scroll to an empty square in the list of colors in preparation to add it into the palette. Then right-click in the blank space just above the two arrow keys and choose the “Eyedropper” option in the context menu. See Figure 3.3 - Right-click above the arrow keys. Your cursor will resemble an eyedropper. Click on the color that you wish to save. The eyedropper will fill with that color. Click on the empty square you selected in step one. The square should change to the chosen color. You can now use this color in any object by clicking on it. When you are done adding colors to your color palette you can right-click anywhere in the display to escape the eyedropper function.

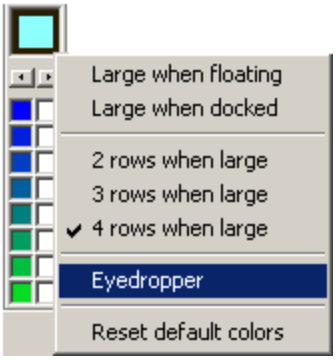


Figure 3.3 - Right-click above the arrow keys

3.3. Arrange Toolbar

Now that you have an idea of how to add objects and customize them in GraphWorX32, let us discuss the different ways you can orient and arrange these objects. The Arrange toolbar provides many functions to aid in organizing the objects the way you need.

Near the end of the toolbar there are icons to help you flip and rotate an object. You can flip the object along the x- and y-axis as well as rotate it ninety degrees clockwise and counterclockwise. If you need to rotate the object in degrees other than right angle turns, you can use the "Free Rotate" function. After selecting the icon in the toolbar your mouse pointer will include a circle with two arrows to indicate that the tool is in use. When you move the cursor over the target object for rotation, the cursor changes to a circle with two arrows. You can now hold down the left mouse button and drag the mouse until the object is at your desired angle.

When you have multiple objects on the display it may be necessary to define the order that the objects show on screen. You can do this easily with the "Bring to Front", "Send to Back", "Bring Forward", and "Send Backward" options in the Arrange toolbar. The first two options bring the object to the top of the stack or send it to be the last one respectively. The last two options perform similar functions except it brings the object forward and backward one step at a time.

After you have ordered your objects, you may want to align the objects. You can do this easily with the different aligning tools that GraphWorX32 offers. To align objects select all of the objects you need to arrange. As you select the different objects the referenced object will have an active (colored) selection squares.

This is usually the last object you select in the group. When you click on an option to align all other selected objects will align to the referenced object.

> ***Tip!*** After selecting, if you need to change the reference object simply click on the selected object that you wish to become the referenced object.

When you are ready, click on one of the align options in the Arrange toolbar. You can choose to align the objects at the "Bottom", "Top", "Center", "Left", "Right", and "Middle". Once you have aligned the objects you may also want to space them evenly. You can use the "Even Across" and "Even Down" options to help you with this task.

When you have finished with arranging the items, you may want to group the objects together. You can do that with the "Symbol" icon. Select all of the objects that you want to group together and simply click on the icon to make a symbol out of the group. You can easily ungroup the object by clicking on the "Ungroup Symbol" icon. Note that if you ungroup an object with applied dynamics you will lose the dynamics applied to that symbol.

When you double-click on a grouped object you will get the Property Inspector of the object as one entity. If you need to edit individual objects that made up the group you need to drill into the object. You can do this by right-clicking on the object and selecting the **Edit Symbol** option. A rectangle will surround the object to indicate which element you are editing inside the grouped object. From here, you can edit the individual objects normally.

> ***Tip!*** You can hold the Shift key and right-click on the object to gain access to individual objects inside a group. This is the same as right-clicking and selecting the Edit Symbol.

3.4. Dynamics

Static objects are the basis of the representation you are about to create in the display. However, the ability to add dynamics to objects separates GraphWorX32 from image creation tools available on the market. Dynamics in GraphWorX32 allows you to add animations to your display as well as attach visual alerts when something in the production floor requires extra attention.

The different types of dynamics that you can add to an object in a GraphWorX32 display are available in the Dynamics toolbar. Here is where you will find the following dynamics, separated into three groups:

- Size
- Location
- Rotation
- Hide/Disable
- Color
- Analog Color
- Flash
- Pick

- Digital Selector
- Analog Selector
- Animator

- Process Point
- Time/Date
- Button
- Checkbox
- Radio Button

In the first column are dynamics that can be applied to any single object or symbol. The second column contains dynamics that can only be applied to multiple objects.

The last column contains miscellaneous dynamics. You may not consider a Process Point (PPT) or a Time/Date object dynamic, but since the values in these object change dependent upon external factors they are considered dynamics in GraphWorX32. The other items in the final column are objects that you can add to the display and associate pick actions.

Let us now take a closer look at some of these dynamics.

3.4.1. Process Points

A Process Point (PPT) is one of the simplest yet crucial dynamics available in GraphWorX32. If you need to read the numerical value of an OPC tag in GraphWorX32 you will use this dynamic.

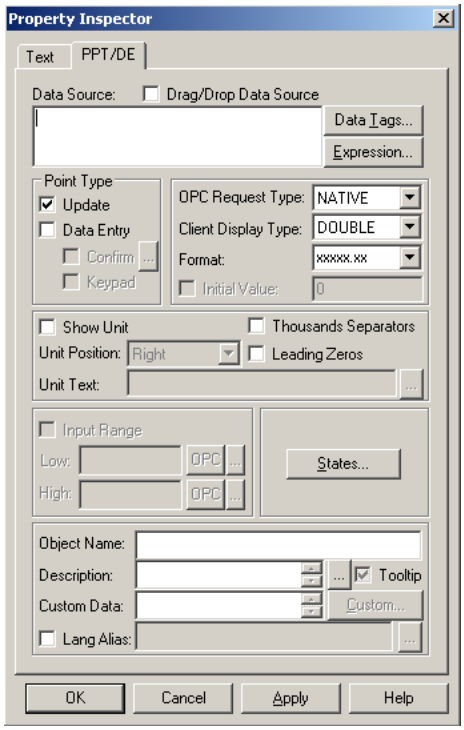


Figure 3.4 - Process Point Property Inspector

To insert a PPT on your display you can click on the "Process Point" icon in the Dynamics toolbar. Then click anywhere on the display where you want the PPT to reside. The Property Inspector dialog comes up as shown in Figure 3.4.

When the dialog first comes up the Data Source field is empty. This is where you provide information on what the PPT shall show in runtime. Usually the Data Source field houses the connection to your OPC point. To add an OPC point into the Data Source field, click on the "Data Tags…" button, this will bring up the Unified Data Browser dialog.

Here you choose the desired OPC tag. In fact, wherever it allows you to insert OPC tags you will be able to bring up this dialog and select it.

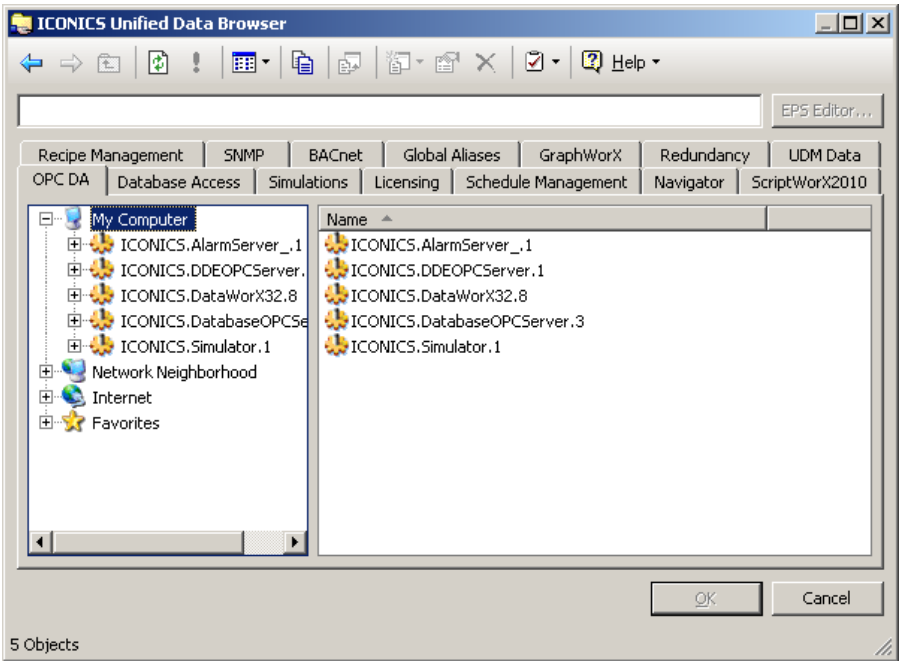


Figure 3.5 - Unified Data Browser

In the Unified Data Browser find the OPC server and drill into it to find the desired OPC tag. Once you find the tag, select it so that you can see the tag's path in the textbox at the top, then click on "OK" to close the dialog. In the Property Inspector of the PPT you should now see the selected OPC tag path. You can click on "OK" to apply the changes and close the Property Inspector. In runtime mode when the tag value changes it will automatically update in GraphWorX32 as well.

In the Property Inspector dialog of a PPT you can also find options such as Data Entry, OPC Request Type, Client Display Type, and Format. If the selected OPC tag is writable you can check the "Data Entry" checkbox in the Point Type section. This will allow you to write values to the tag during runtime. Once you check this option you can also check the "Keypad" option to use an on-screen keypad when entering a value for the PPT. In the "OPC Request Type" dropdown list you can select the data type that GraphWorX32 should request. The default value is "**Native**," which means it will request data in the type of your selected OPC tag. In the "Client Display Type" dropdown list you can choose the data type in which GraphWorX32 will use to show the value of the tag. Be aware that if the client display type does not match your OPC tag you may not see data in runtime mode. In the "Format" dropdown list you can choose the number of characters or the number of decimal places to show in the PPT.

Showing values of an OPC tag is the native use for a PPT but there is much more that you can do with a PPT. For example, you can use a PPT to show the results of an expression. By clicking on the "Expression…" button, the **Expression Editor** is brought up.

From here you can create mathematical expressions with your OPC tags and static values using available relational and logical functions. The possibilities are almost endless in the expressions editor.

One of the most common expressions used is to find out the quality of an OPC tag. FYI, a return value of 192 represents a good quality. You can do this by clicking on the "Functions" button and select OPC → OPC Quality. The system inserts the quality function into the Expression Editor and you can then browse for your OPC tag using the "Tags" button in the editor. The final expression may resemble the one in Figure 3.6. Notice the curly brackets '{{ }}' around the OPC tag. When you are using an OPC tag in an expression, you will need to enclose it with double curly brackets to indicate it is an OPC tag. Click on OK to close the dialog boxes. In runtime you will now see the quality value of the OPC tag.

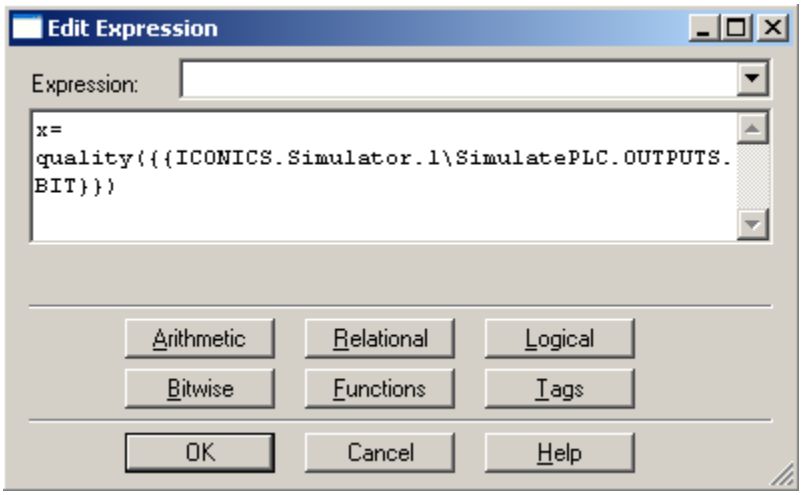


Figure 3.6 - Quality Expression in Expression Editor

Aside from reading data from OPC tags directly you can also use Local Variables in a PPT. Local variables in GraphWorX32 are temporary value holders. They do not connect to any real OPC server and are only present within the GraphWorX32 display where you defined them. On changing displays, they will lose their values and revert to their configured initial value (0 by default).

To create a local variable in any data source field, type in *~~localVariableName~~* where *localVariableName* is the name you assign to the variable. Once you have created a local variable you can select it from the Unified Data Browser by going to the GraphWorX tab and choosing Local Variables in the navigation tree.

You can edit local variable properties by selecting the **Dynamics →Edit Local Variables** menu. This will bring up the **Edit Local Variables** dialog where you can configure the data type, initial value, and the low and high range of the variable.

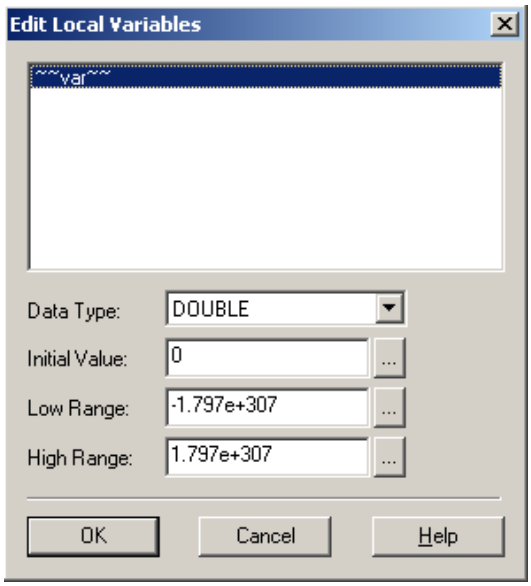


Figure 3.7 - Edit Local Variables

Another useful property in a PPT is its ability to represent numerical values in textual strings. This makes it very easy to translate machine code into human understandable texts so that operators know the state of the machine without having to look up the meaning of the code. You can define textual meaning to values by clicking on the "States…" button in the Property Inspector of a PPT. This brings up the **State Field Configuration** editor. In the State Value field you can enter the numeric value (e.g. a machine code for example) and in the State String field you can enter the corresponding text (i.e. a text string that explains the machine code). When you have defined the desired text click on the "Add" button to add the definition.

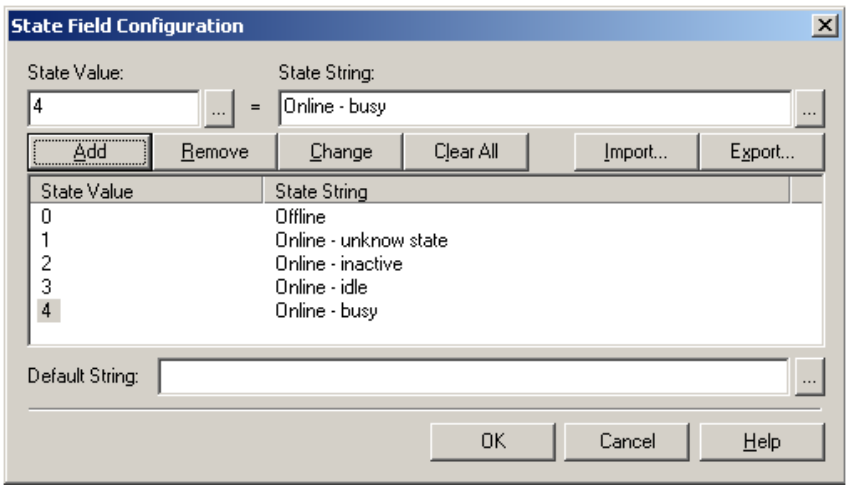


Figure 3.8 - State Field Configuration

You can add as many states as you need. If you need to change a definition at any point, simply select it in the list, change the State String field and click on the "Change" button to update it. When you are done, simply click on "OK" to close the dialog and return to the Property Inspector of the PPT. When you go in runtime, you'll notice that the PPT no longer shows a numeric value. Instead it will show the text you defined based on the current value of the OPC tag. If your tag is writable, you will notice that instead of having to type in values you will have a dropdown list to select the defined values as shown in Figure 3.9.

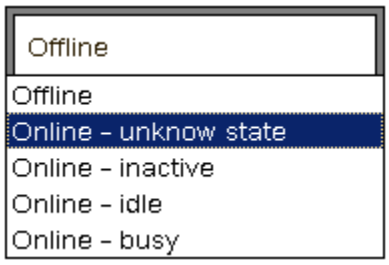


Figure 3.9 - PPT with States Defined

Now that we understand how to connect to OPC tags and use local variables, there is one more type of data available in ICONICS called **GenBroker plug-in values**. GenBroker Plug-in is an ICONICS technology that can do multiple things. It can provide system information regarding different ICONICS applications as well as provide data configured in the Unified Data Manager. For example, if you are running a time-limited license you can find out how many days are remaining until your license expires. When you use a GenBroker plug-in tag, this tag will precede with the type of tag it is (I.E. lic), followed by a colon and the remaining tag path. In our example the tag syntax looks like the one shown in Figure 3.10. You can find out more about the different available plug-ins and their associated syntax in the appropriate help files.

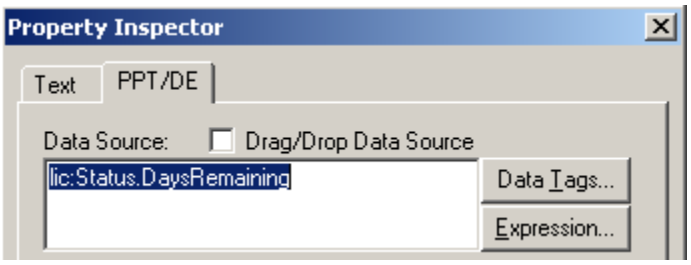


Figure 3.10 - Tag Syntax for a GenBroker Plug-in Tag

3.4.2. Size Dynamic

Using the "**Size**" icon on the Dynamics toolbar, you can define an object that changes size based on the real-time value of its connected data tag.

The object automatically sizes to scale for incoming data tags such that a 100 percent scale equals the full size of the drawn object. You can override the size by defining range override values.

During runtime mode the Size connection resizes the object in proportion to the percentage of the high and low ranges of the connected data point. The object can be resized in many different directions as indicated by the options shown in the Property Inspector depicted in Figure 3.11.

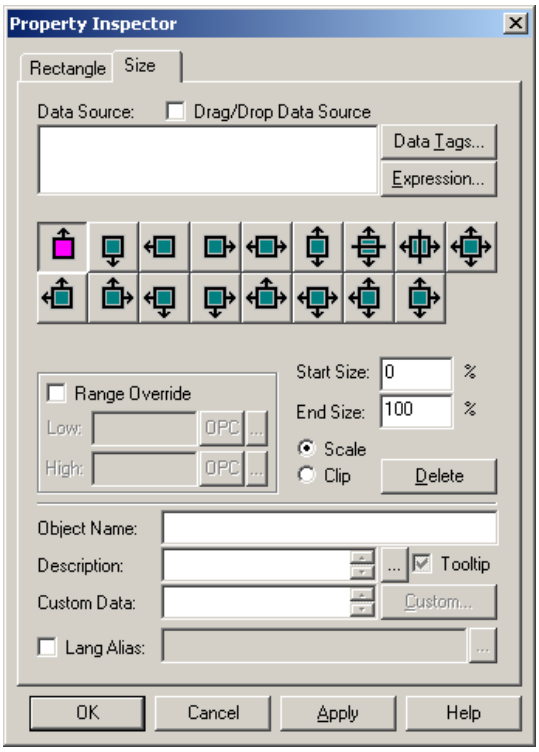


Figure 3.11 - Size Dynamic Property Inspector

While you can resize the entire object using the Size dynamic using the "Scale" option, you can also choose to show portions of an object with the "Clip" option. When you choose to clip an object, the dynamic will show you a portion of the object based on the percentage of the data source value. For example, if the OPC tag value is at fifty and the range of the tag is between 0-100, then you see only half the object in runtime mode. Figure 3.12 shows three trucks. The one on the left has no dynamic applied to it.

The one in the middle has a size dynamic attached to it using the scale option and the one on the right uses the same size dynamic except it uses the clip option. As you can see, using the scale option the entire truck in the middle is scaled down to 50% of the original one on the left. The truck on the right, however, shows only the left half of the truck compared to its original version. That is because of the clip option.

Figure 3.12 - Size Dynamic in Runtime

3.4.3. Location Dynamic

The location dynamic allows you to change the whereabouts of an object on the display based on the provided data source. This is useful when creating displays where you need to show an object moving along the production line.

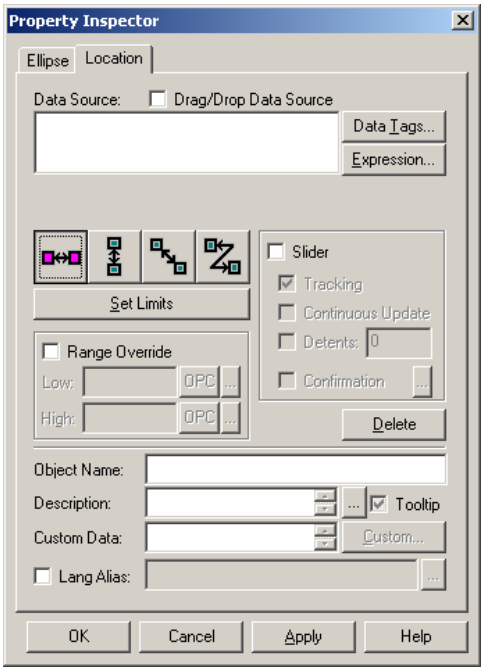


Figure 3.13 - Location Dynamic Property Inspector

In the Property Inspector of a location dynamic, you can choose to move the object in four different ways: a straight horizontal line, a straight vertical line, a straight diagonal line, and a zigzagged line. Once you choose how the object should move you can click on the “Set Limits” button to draw out the path that the object will take.

If you select to create a horizontal, vertical, or diagonal line, you will only be able to drag the shadow copy of the object to its end location without adding more stopping points in the middle of the path. If you used the zigzagged line option you can add destination points along the path to the end location making the object twist and turn before it reaches the final destination.

Using the location dynamic, you can also make a slider. You can use a slider as a method for writing a value to a set point similar to setting the temperature set point on a heater. To create a slider, simply check the "Slider" checkbox in the properties inspector. If you want to see the value update as you drag the slider you should also check the "Continuous Update" option.

3.4.4. Rotation Dynamics

The rotation dynamic is similar to the locations dynamic in some ways. Like the location dynamic it can make an object move along a path based on the value of an OPC tag. Except, unlike the location dynamic, the object will be moving in a circular pattern. You can also use the rotation dynamic to spin an object in place.

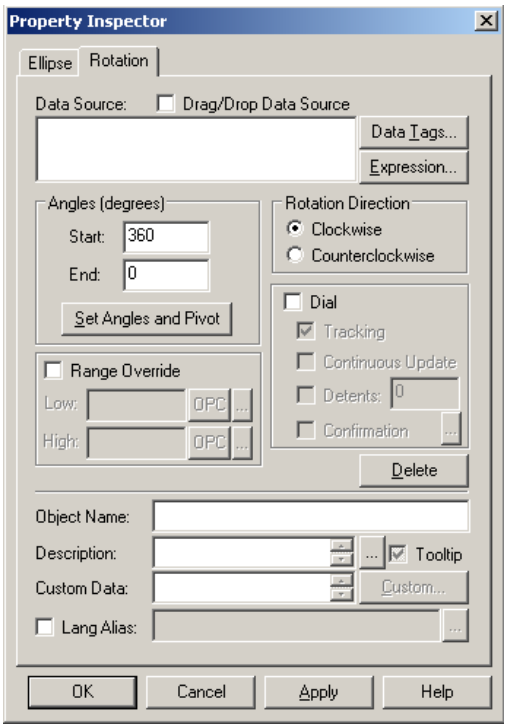


Figure 3.14 - Rotation Dynamic Property Inspector

How the object rotates depends on the pivot or center point it rotates around. When you click on the "Set Angles and Pivot" button you will see a crosshair with a circle centered on the object with the rotation dynamic. That is the pivot, or the point around which the object will rotate. When this point is inside an object, the object will spin in place. When you put this point outside of the object the object will orbit around the pivot.

The rotation dynamic has a "**Dial**" property. This property allows you to simulate functions performed by a knob in real-life (e.g. the volume control in your radio). If you want the associated tag value to increase as you rotate the dial you should also check the "Continuous Update" option.

3.4.5. Hide/Disable Dynamic

The Hide/Disable dynamic allows you to hide or disable an object based on the data source value. A value of one represents true and a value of zero represents false. You can choose to hide or disable the object on true or on false.

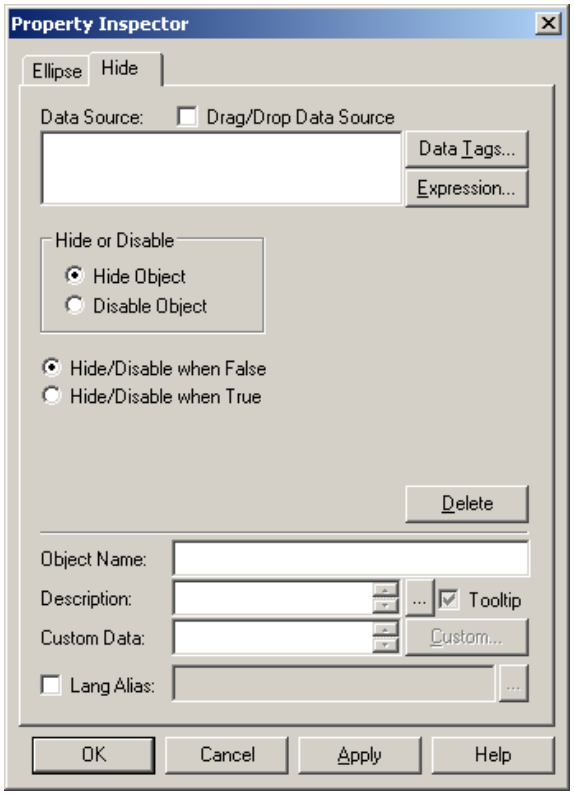


Figure 3.15 - Hide/Disable Dynamic Property Inspector

3.4.6. Color Dynamics

The **Digital Color** dynamic on the Dynamics toolbar makes the color of the object change based on a digital event. The color connections are prioritized in the order in which they are created. Therefore, if two data points are true at the same time the higher-priority color connection takes precedence. If none of the connected data points are true the object displays in its original color.

The color connection is commonly used to indicate states such as alarm conditions, temperature changes or to notify operators of the flow in a pipe.

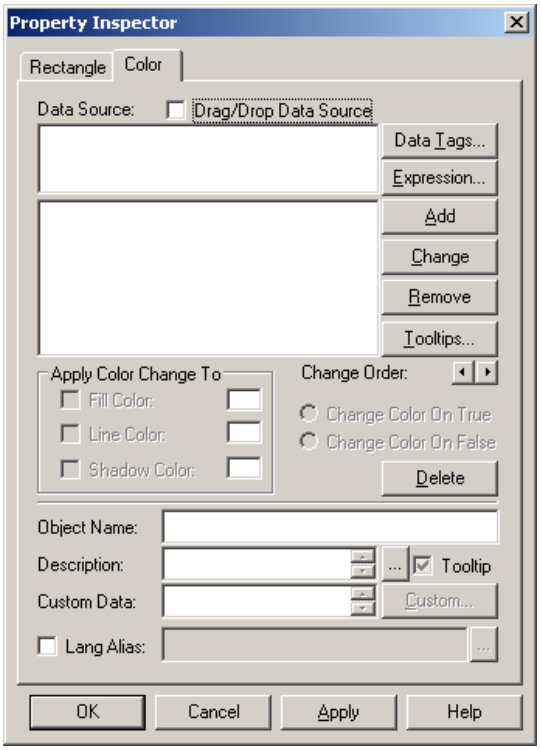


Figure 3.16 - Color Dynamic Property Inspector

The Color Dynamic Property Inspector allows you to configure several data sources, each changing to a different color. For this reason, each time you set a data source you must add it to the data source list.

There is a second color dynamic; the **Analog Color** dynamic is similar to the Digital Color dynamic, however; instead of changing the color based on true and false (digital) conditions, it changes the color based on the current value and its percentage compared to the range of the OPC tag.

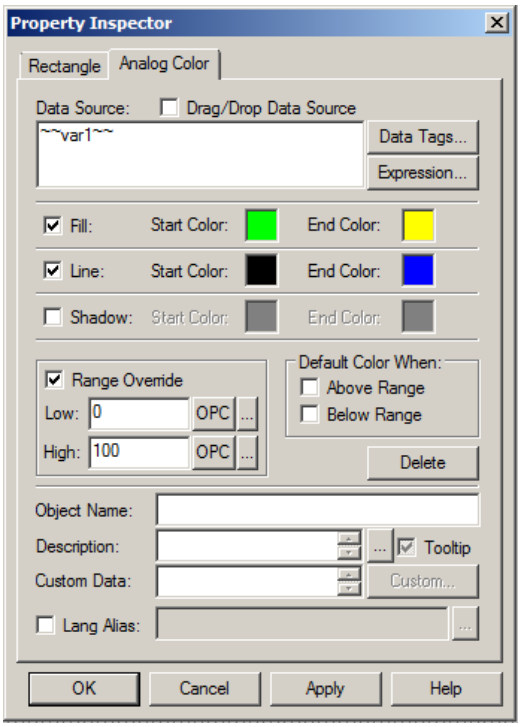


Figure 3.17 Analog Color Property Inspector

In the Property Inspector, you can set the start and end color for the fill, line, and shadow. The Start Color is the color of the object when the value of the OPC tag is at its lower limit and the End Color is the color of the object when the value of the OPC tag is at its highest limit. When the OPC tag value is between its low and high limits, the color is somewhere in between as well. Figure 3.18 shows three squares. The square on either end are static objects. The square in the middle has the start color set to the same color as the square on the left and the end color set to the color of the object on the right. The OPC tag value is at 50%. As you can see, the color of the square in the middle is halfway between the start and end colors.

Figure 3.18 - Analog Color Dynamic

3.4.7. Flash Dynamic

A flash dynamic is similar to the color dynamics. Instead of changing colors based on a data source value, a flash dynamic would go back and forth between two colors when the data source turns the dynamic on. Alternatively, it can also go between hiding and showing the object. How fast the colors switch back and forth depends on the "Flash Rate" set in milliseconds.

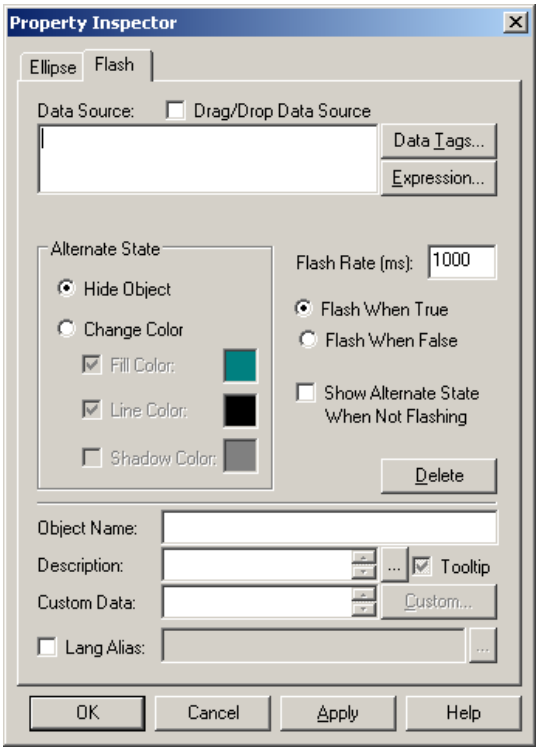


Figure 3.19 - Flash Dynamic Property Inspector

Lab: Controlling Pill Production

L3

In the previous lab, we started to configure a template file for Acmeseutical. We will put some finish touches on that template file in this lab and start using it to create display files to control the pill making process.

In this lab, you will use GraphWorX32 dynamics and pick actions to generate an interactive screen for the operators to use. Although we have not discussed pick actions in this chapter, they are very similar to a dynamic and allow the user to perform different tasks with a click of a button.

Estimated time to complete this lab: 90 minutes

3.1. Task Checklist

In this lab, you will complete the following tasks:

- Add navigation actions to the buttons on the template
- Create a display to make pain killer
- Use hide/disable dynamic
- Add process points
- Use Symbols from the Symbols Library
- Use Pick Actions to load different displays

3.2. Support Files

You will need the following file(s) to complete this lab:

- AcmesueticalTemplate.tdf
- PainReliefWorldMap.gdf

3.3. Acmeseutical Template Continued

We will add some final changes to the template in this section. Although we will not be completely done with the template by the end of this section we will be very close to having it completed and will begin using it.

1. If you closed the template file you can start GraphWorX32 from the Windows All Programs menu and then use the **File →Open** command in GraphWorX32 to reopen the template. Do NOT double-click on the .tdf file in Windows Explorer to open it for editing. When you do have the template open the title bar of GraphWorX32 should indicate the name of your template file. Namely, "AcmeseuticalTemplate.tdf".

2. Select the "home" icon on the display and click on the "Pick" button in the Dynamics toolbar.

3. Leave the Action as "Load Display" in the Property Inspector that comes up and type in the name of the file "PainReliefWorldMap.gdf".

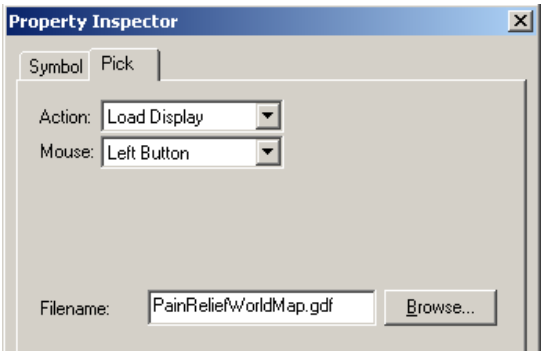


Figure L3.1 - Load Display Pick Action Properties Inspector

NOTE: Since we are only providing the name of the display in the Filename field, all displays generated from this template will need to be saved in the same directory as the "PainReliefWorldMap.gdf". Otherwise, the pick action will not be able to locate the file.

4. Click on "OK" to apply the settings and close the dialog. Notice that now you have a Pick Action associated with the icon, instead of blue selection squares, you now have red selection squares to indicate that the object has a dynamic associated with it.

5. Select the "Previous" icon on the display and click on the "Pick" button in the Dynamics toolbar again.

6. Change the Action to "Display Back" and click on "OK" to save the setting and close the dialog.

7. Select the "Next" icon on the display and click on the "Pick" button in the Dynamics toolbar again.

8. Change the Action to "Display Forward" and click on "OK" to save the setting and close the dialog.

9. Save the display. Click on "No" since for now we don't have associated displays.

3.4. Acmeseutical Pill Machine

In this section, we are going to make a display that resembles the pill making process for Acmeseutical. The medicine that they make usually consists of varying amounts of four different ingredients. We will reflect that in our display. By the end of the lab, you should have a display similar to the one shown in Figure L3.2.

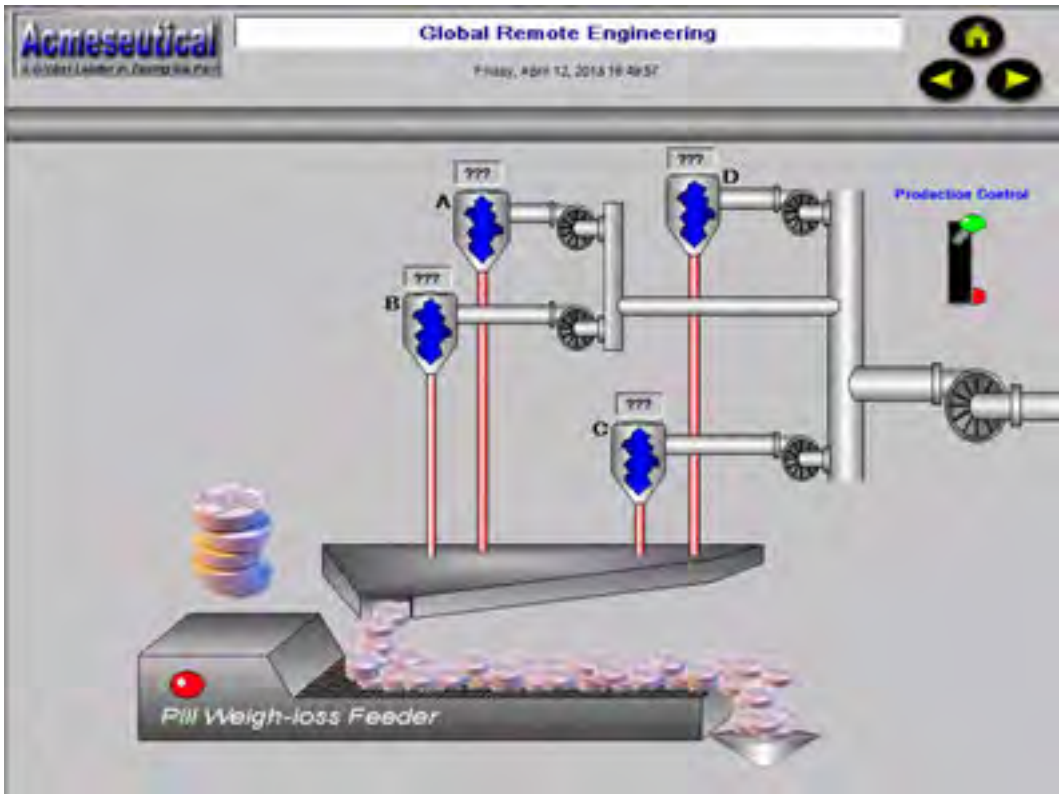


Figure L3.2 - Acmeseutical Pill Machine Display

1. Double-click on the AmeseuticalTemplate.tdf that we just finished editing. Once the display opens, notice the title bar of the display shows "Display1" and not your template file. This is the intended behavior modeled after Microsoft Office templates. Also, notice that you cannot edit the objects in the template without going into the "Edit Template" mode.

2. Save the file as "ACMEPillMachine.gdf" in the Training directory. If this file name already exists, you may replace it.

3. You can start building the display according to . Most of the objects you see in the display are in the Symbols library located in the "TrainingCourseV9.20.sdf".

> ***Tip!*** As you build your display feel free to experiment with some of the shortcuts that GraphWorX32 provides. For example, to duplicate an object you can hold down the Ctrl key and drag the object or just hit Ctrl + D to duplicate it. You can also manipulate a copied object with options on the Arrange toolbar.

4. Click on the "Process Point" button in the Dynamics toolbar and click somewhere near tank A in the display.

5. In the Property Inspector's "PPT/DE" tab, click on the "Data Tags…" button to bring up the Unified Data Browser (UDB).

6. On the OPC DA tab, expand ICONICS.DataWorX32.8 → Registers → USA → Foxborough → Machine1 → Mixers → MixerA and select the Level_Setpoint tag.

7. Select the Data Entry check box to allow users to enter data into this field and set the Format field to 3 to limit the string to display only 3 characters.

8. Select the "Text" tab and change the 3D Edge filed to "Sunken."

9. Set the text alignment and bounding to Center. Click on "OK" to save settings and close the Property Inspector.

Note: Instead of repeating the steps below, you could copy the PPT/DE or Pipe (3 times) that you created, see the Tip above for the copy procedure, and just edit the data source from Mixers → MixerA to Mixers → MixerB, MixerC and MixerD. Ask your instructor for assistance if needed.

10. If you used the copy technique from the previous Note, you can skip this step. Or you can repeat steps 1-9 for Mixers B, C and D, changing the associated data source accordingly.

11. We're going to use each tank's drain pipe as a valve by adding a pick action and a color change dynamic. Red pipe = valve Closed and the tank can fill, Green pipe = valve Open and the tank will drain.

12. With Tank A's pipe selected click on the Red square in the color pallet on the left. The pipe should turn Red.

13. Click on the Pick Action icon to launch the property inspector. Set the Action: to Toggle Value.

14. Set the Toggle Value1: 0 and Toggle Value2: 1.

15. Click on the Tags Menu button. From the OPC DA tab browse to select ICONICS.DataWorX32.8 → Registers → USA → Foxborough → Machine1 → Mixers → MixerA and select the ValveStatus tag. Click OK.

16. Add a Color Change as a second dynamic to the pipe.

17. Click on the Color Change icon. Click the Data Tags button. From the OPC DA tab browse to select ICONICS.DataWorX32.8 → Registers → USA → Foxborough → Machine1 → Mixers → MixerA and select the ValveStatus tag.

18. With the Data Source string selected in the upper window click on the Add button to insert the string into the lower window.

19. Check the Fill Color box and double click on the colored square to the right. Select Green from the color pallet.

20. Select the Data Source string in the upper window again and click on the Add button to insert the string into the lower window

21. Change the Fill Color to Red.

22. Set the "Change Color on False" radio button.

23. The finish configuration should look like the Figure below.

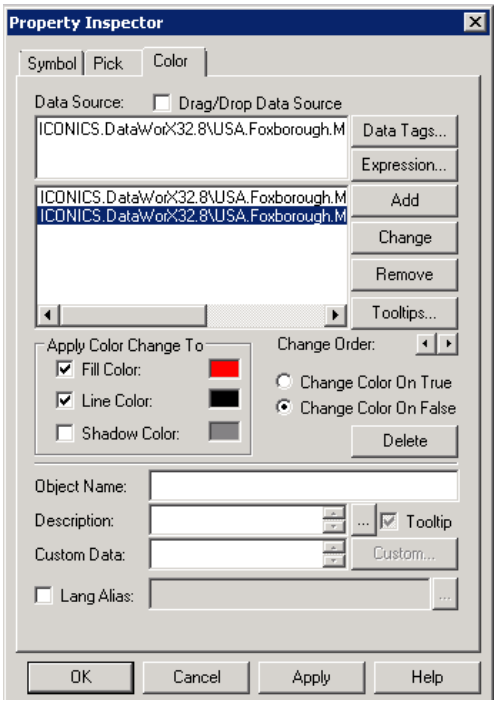


Figure L3.3 - Property Inspector Color Change

24. Repeat steps 21- 29 for Mixers B, C and D, changing the associated data source to the appropriate Mixer.

25. Next we will redefine the dynamics on the pump. Right click on the pump associated with MixerA.

26. Select **Edit Connections**. You should see a dialog that resembles the figure below.

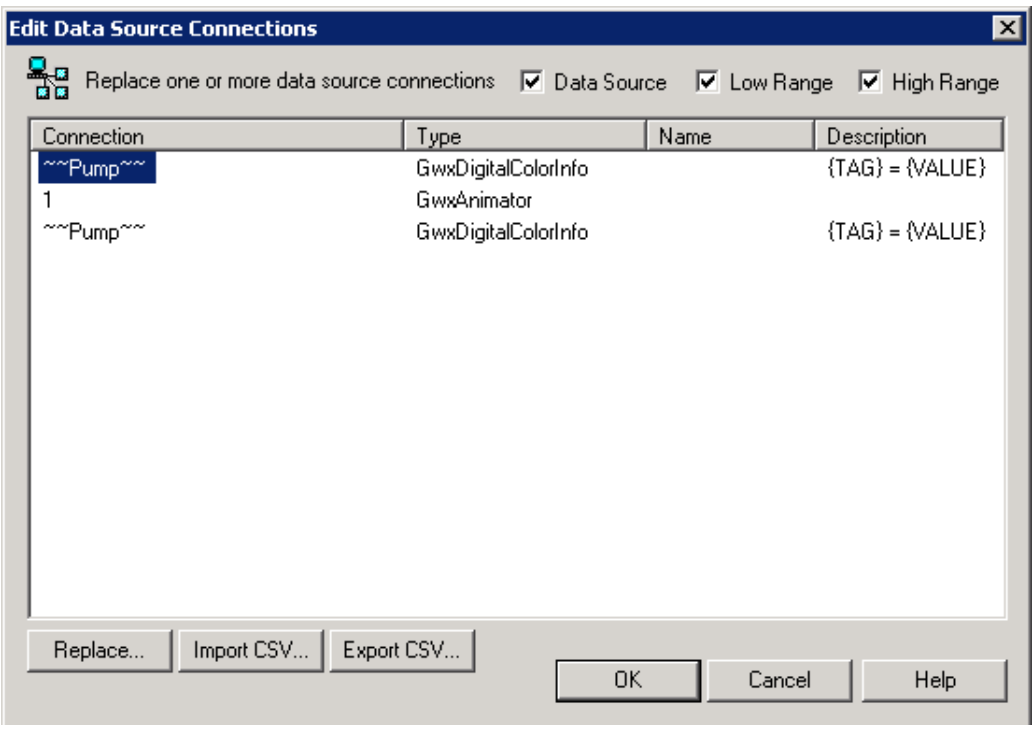


Figure L3.4 – Replace Existing Pump Dynamics

27. Select the top ~~Pump~~ entry. Hold the Shift key and select the bottom entry. All three entries should be selected and highlighted. Select **Replace...**.

28. As shown with the circle in the figure below, select the **Ellipse** to launch the **Unified Data Browser**.

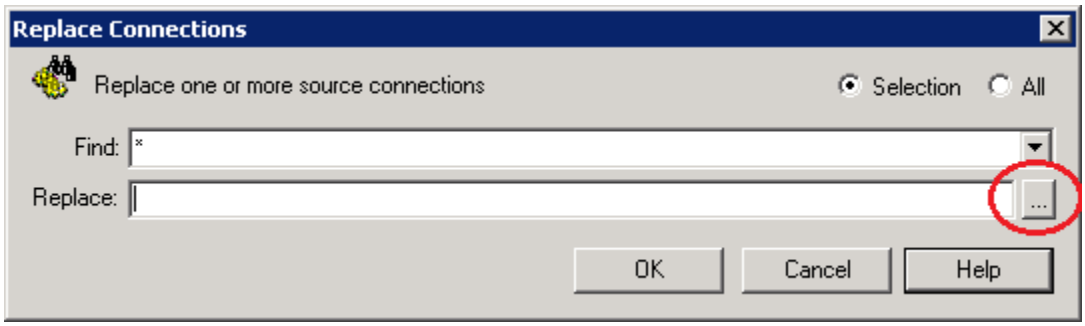


Figure L3.5 – Replace Connections Dialog

29. From the OPC DA tab browse to select ICONICS.DataWorX32.8 → Registers → USA → Foxborough → Machine1 → Mixers → MixerA and select the PumpStatus tag. Click OK.

30. As shown in the figure below, in the Replace text change **MixerA** to **Mixer<<MixerID>>**. In future steps you will see how we replace the Local Alias <<MixerID>> with the appropriate mixer letter.

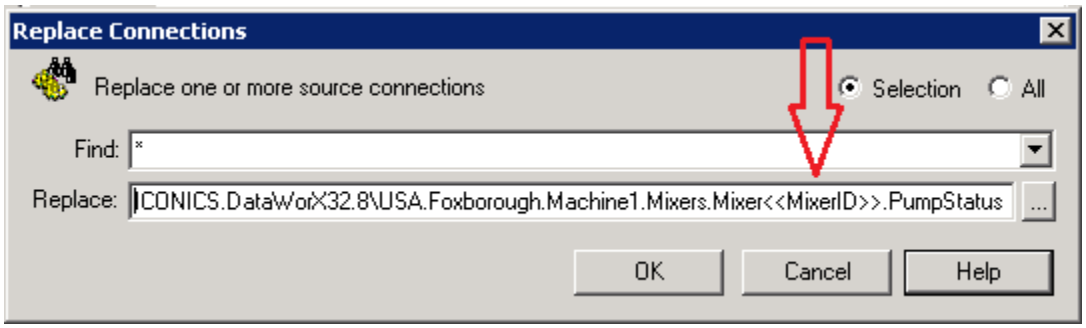


Figure L3.6 – Adding the Local Alias <<MixerID>>

31. Click **OK** when your edits are complete.

32. Double-click on the Pump for MixerA. With the Symbol tab selected enter **PumpA** in the **Object Name** field.

33. Repeat steps 26-32 for the pumps associated to Mixer B, C, and D. You should leave the local alias the way it is throughout the process. Change the name of each pump accordingly.

34. Make sure nothing is selected on the display and go to the **Dynamics** menu, and then choose the "**Edit Local Aliases**…" option.

35. In the **Edit Local Aliases** dialog, you will see a list of objects with local aliases.

36. Select each **Pump** object and set the alias definition according to .

Object Name	Alias Name	Alias Definition
PumpA	MixerID	A
PumpB	MixerID	B
PumpC	MixerID	C
PumpD	MixerID	D

Table L3.1 - Local Alias Definitions

37. When you are done configuring, your Edit Local Aliases dialog should look similar to . Click on “OK” to close the dialog.

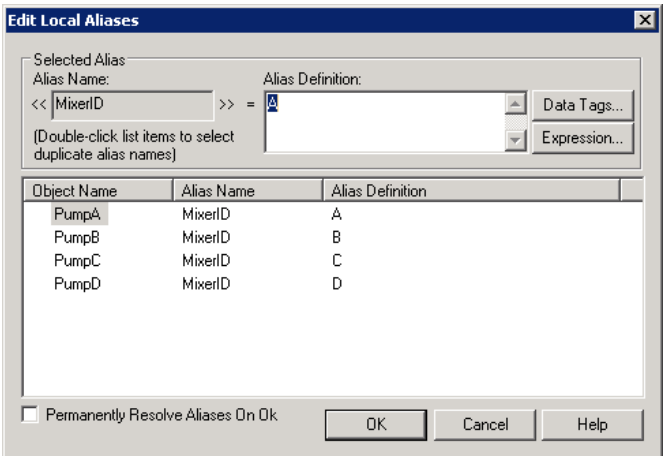


Figure L3.7 - Edit Local Aliases

38. Let’s add a pick action to the pumps. Clicking on the pump in runtime will turn the pump on or off. When the valves are closed and the pumps are running the mixers will fill.

39. Add date source: Pump Status. It will apepar as "ICONICS.DataWorX32.8\USA.Foxborough.Machine1.Mixers.MixerA.PumpStatus".

40. Select MixerA’s pump and click on the **Pick Action** icon.

41. Set the **Action**: to **Toggle Value**.

42. Set the **Toggle Value1: 0** and **Toggle Value2: 1**.

43. animate the Jagged Cut-Away symbol to simulate a mixer level.

44. Select the “**Jagged Cut-Away**” on **MixerA**.

Note: The jagged cut-away is layered object. In the next few steps we will select only the top layer and add a size dynamic.

45. Hold the Shift key and Right click on the Jagged Cut-Away.

46. Click on the blue section of the jagged cut-away..

> ***Tip!*** You can also hold down the Shift key and Right Click on a symbol to drill into it. You can double-click anywhere outside the symbol to exit it.

47. Click on the "**Size**" button in the Dynamics toolbar.

48. In the Property Inspector of the dynamic click on the "**Data Tags**..." button. In the UDB, notice that it opens to the last browsed location. Simply select the **Level** tag in **MixerA** and click on "**OK**" to close the UDB.

49. The data source field should contain the full tag path, which should read "ICONICS.DataWorX32.8\USA.Foxborough.Machine1.Mixers.MixerA.Level".

50. Change the Sizing mode from **Scale to Clip**. The finished configuration should look like Figure L3.7. Click on "**OK**" to apply the settings and close the dialog.

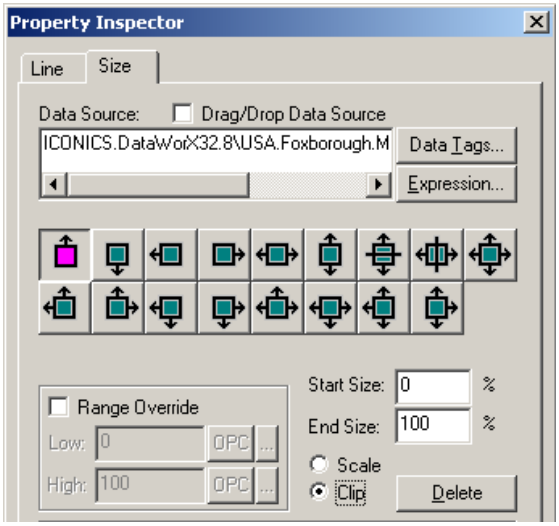


Figure L3.8 - Property Inspector: Size Dynamic

51. Repear steps 39-50 for Mixers B, C and D.

52. Click **Esc** to deselect the object.

53. Select the line of pills you have coming off the production line and double-click on it to bring up the Property Inspector. Notice that there is already a dynamic (Animator) associated to it. An animator allows you to create animations similar to a movie. We will change the data source so that when the machine is not running, the pills will not be moving.

54. Click on the "Data Tag…" button and choose the Machine1's status tag. After you click OK on the UDB the data source field should read "ICONICS.DataWorX32.8\USA.Foxborough.Machine1.Status".

55. We will also change the data source on the production control to use the same tag.

56. Double-click on the Production Control switch to bring up the Property Inspector. Notice that there is a dynamic and a pick action associated with it. The Digital Selector dynamic allows us to show a different picture base on the value of the data source and the Pick action toggles the data source value for us. Again, we will only change the data source here.

57. Select the first line you see in the Digital Selector and click on the "Data Tags…" button to bring up the UDB.

58. Choose the Machine1.Status tag again. When you are back in the Property Inspector, click on "Change" so that the new data tag takes place of the local variable "Switch."

59. Do the same for the second line. Your Digital Selector tab should look similar to Figure L3.8.

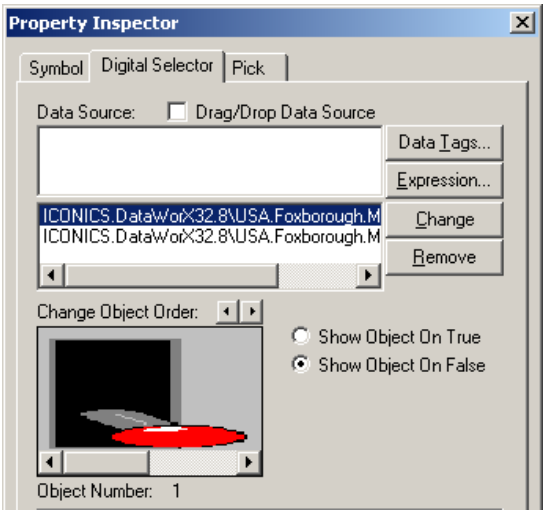


Figure L3.9 - Property Inspector: Digital Selector

60. Go to the Pick tab and change the data source to Machine1.Status. Click on "OK" to apply the settings and close the property inspector.

61. Right-click on the Feeder and select "Edit Symbol."

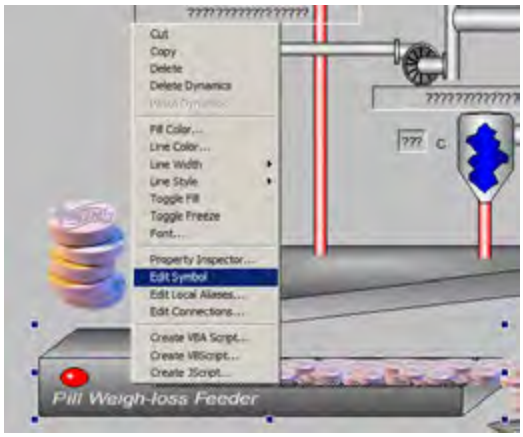


Figure L3.10 - Editing a Symbol

62. Select the button on the feeder. Click on the Red Circle object. Notice the handles are red because this object has no dynamics.

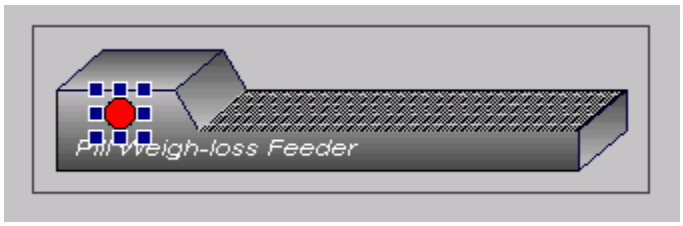


Figure L3.11 – Feeder On/Off Status Indicator

63. Double-click on the feeder **Button** to bring up the **Property Inspector**.

64. Select the **Pick** dynamic tab and connect the **Data Source to Machine1.Status**.

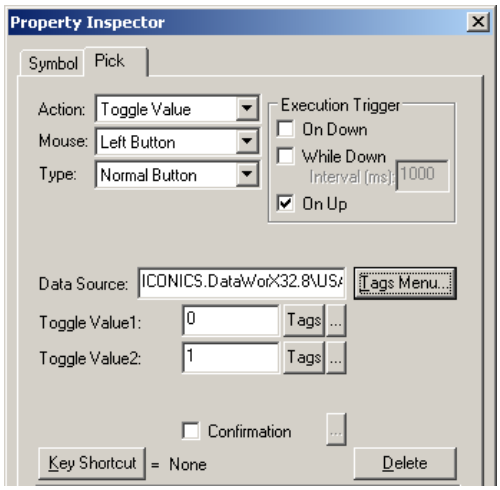


Figure L3.12 - Feeder Status Indicator

65. Click on "**OK**" to close the Property Inspector.

66. Select the pump on the far right in the display, double-click on it to bring up the Property Inspector. Change the **Object Name** to **MachinePump**.

67. Click **OK**.

68. Right-click on the **MachinePump** and select the "**Edit Connections**…" option.

69. In the Edit Data Source Connections dialog, select all of the connections and click on the "Replace…" button.

70. Click on the ellipse button next to the **Replace** field to bring up the UDB. Select the **Machine1.Status** tag and click on "OK."

71. Click on "OK" again to close the **Edit Data Source Connections** dialog.

72. If you have not been saving your display, you should save your display now.

73. Let's test our display. Go into Runtime.

74. Click on your production control switch. The MachinePump and the Pills should animate. The Feeder button should change color.

75. While in Runtime, input a Level_Setpoint value for each of the Mixers.

76. Be sure to hit the Enter key to submit the Level_Setpoint value.

77. Click on Each Pump to start the Mixer filling.

78. The Mixer should fill to the Level_Setpoint value.

79. Click on the pump to turn it off.

80. Open the drain valve by clicking on the Pipe.

81. The tank should drain.

Review: GraphWorX32 an Animation Platform

R3

3.1. Chapter Objectives

In this chapter, we discussed the basic tools available in GraphWorX32. You can create static objects in GraphWorX32 to represent your production environment as well as add animations to the static objects on a display. You should now know and understand:

- How to create static objects using the Draw toolbar
- How to arrange objects on a display
- How to edit the properties of a static object
- How to add a Process Point on a display
- How to add dynamics to a static object

3.2. Summary

Using the Draw toolbar in GraphWorX32, you can create basic objects such as lines, rectangles, ellipse, and many other primitive objects. In the Property Inspector of these objects, you can change the line and fill color, the line width, and other physical properties of the object. When you add an object such as a rectangle or ellipse in the display, you can hold down the Shift key while drawing it to create a square or circle respectively. Pressing the same key while trying to resize the object would preserve the aspect ratio of the object. Other functions of the Draw toolbar include importing image files and Meta files into GraphWorX32.

The arrange toolbar contains actions you can use to reposition and reorient objects in the display. There are three different major types of functions in this toolbar. You can use functions to align objects, order objects, and change the angle of an object. Functions on this toolbar include Free Rotate, Align Bottom, Space Evenly Across, and many others.

While the static objects make up the basis of the graphical representation of your production line, dynamics are the driving force that provides animation in runtime mode. GraphWorX32 provides many different dynamics. The simplest and most often used dynamic is a Process Point (PPT).

Using a PPT, you can read an OPC tag's value, write to an OPC tag, create a variable local to GraphWorX32, as well as create states to show textual representation of predefined numeric values. Other basic dynamics in GraphWorX32 include Size, Location, Rotation, Hide/Disable, Color, and Flash.

3.3. Terms

Data Source	Expression Editor	Local Variable
Referenced Object	Rotation Pivot	Shadow Copy
Unified Data Browser	Unified Data Manager	

3.4. Case Study

Using dynamics in GraphWorX32, you can create interactive displays with vivid animation. Let us take a look at how others have used GraphWorX32. If you open Gen32Demo from Start → Programs → ICONICS GENESIS32 → Gen32Demo, you can click on the "Industry Case Study" icon in the GenDemo title bar. When you see the Industry Case Studies display, go ahead and click on the second button under the "Automotive" column. This will bring up a display that resembles the washing process of a train. Indeed, this is what the German Railway (Deutsche Bahn) used to control the process of cleaning their trains.

At a first glance, the display may seem overwhelming; there is a lot of information on this display. There are many changing process points to indicate different values. There are flash animations to indicate status, show/hide animations to simulate water spraying on the train, and location dynamics to control where the train is in the washing process.

Go ahead and click on the "Start" button at the bottom of the display. Animation will start. You will notice some changes on the display. First, you will see that there are some water droplets to the left of the train as it simulates the train being sprayed with water. Then you will notice the washing rollers moving to where the train is and start the wash. Can you pick out the different animations used in this short sequence? There is the show/hide dynamic for the water spray, a location dynamic to move the washing rollers and a rotation dynamic on each of the rollers to simulate the cleaning.

You can feel free to click on other buttons to see what they do. For starters, try the slider under the start and stop button to see what happens to the train. If you want to learn more about the dynamics you saw in the display, simply open it in configuration mode and you can see how the dynamics are configured.

3.5. Test Your Knowledge

In this chapter, we discussed the draw, arrange, and color palette toolbars as well as basic dynamics in GraphWorX32. You should make sure that you can answer the following questions before continuing to the next chapter.

1. What are some basic dynamics?

    a. Size
    b. Location
    c. Rotation
    d. Flash
    e. All of the above
    f. None of the above

2. What can you do using the Arrange toolbar?

    a. Rotate an object
    b. Group objects
    c. Align objects
    d. Order objects
    e. All of the above
    f. None of the above

3. True or False: You can use the eyedropper to pick out colors from an image to add it to the color palette and use it at will.

4. Which key(s) should you hold down if you want to preserve the aspect ratio while resizing an object?

    a. Control (Ctrl)
    b. Shift
    c. Alternate (Alt)
    d. Ctrl + Alt
    e. Ctrl + Shift

5. True or False: A Time/Date object is considered a dynamic object in GraphWorX32.

6. True or False: Once you have made a group out of different objects, you cannot ungroup the object.

7. True or False: When you ungroup an object, you will lose the applied dynamics.

8. True or False: When selecting many objects you want to align, the first selected object is the one all other will align to.

9. What can you do with a Process Point?

    a. Read an OPC tag
    b. Write to an OPC tag
    c. Display values from an expression
    d. Display text strings based on OPC tag value
    e. All of the Above
    f. Only A and B
    g. None of the Above

10. True or False: You can only use Local Variable within GraphWorX32.

GraphWorX32: Pick Actions and ActiveX 4

In the last chapter we discussed some basic dynamics available in GraphWorX32. Besides visible animations on an object you can also use dynamics to perform other functions. In this chapter we will concentrate on the GraphWorX32 Pick Action, which is a dynamic triggered by user interaction. We will also discuss ActiveX controls that are available natively in GraphWorX32. By the end of the chapter, you should have a basic understanding of the following:

- GraphWorX32 Pick Actions
- Different ways to trigger a Pick Action
- How to configure some basic Pick Actions
- The different ActiveX natively available in GraphWorX32
- How to configure the Vessel ActiveX

4.1. Pick Actions

Pick Actions are dynamic actions that an operator can perform. Unlike other dynamic actions which animates based on OPC data, a Pick Action requires some interaction from the operator to perform the dynamic. Usually, to trigger a pick action, an operator would need to click on something in the display. It can be a button, a checkbox, or even a simple object.

Using a Pick Action, you can write a value to an OPC tag, open a new GraphWorX32 display, launch third party applications and much more. Let us now look at how to create a pick action.

4.1.1. The Pick Action Dynamic

There are a few ways to add a Pick Action Dynamic. First, let us discuss how to add a pick action to a static object on a GraphWorX32 display. Select the object that you want to add the dynamic to and click on the “Pick” icon in the Dynamics toolbar. This will bring up the Property Inspector for the pick action. Choose the action to perform in the Action dropdown list. Figure 4.1 shows the list of available actions.

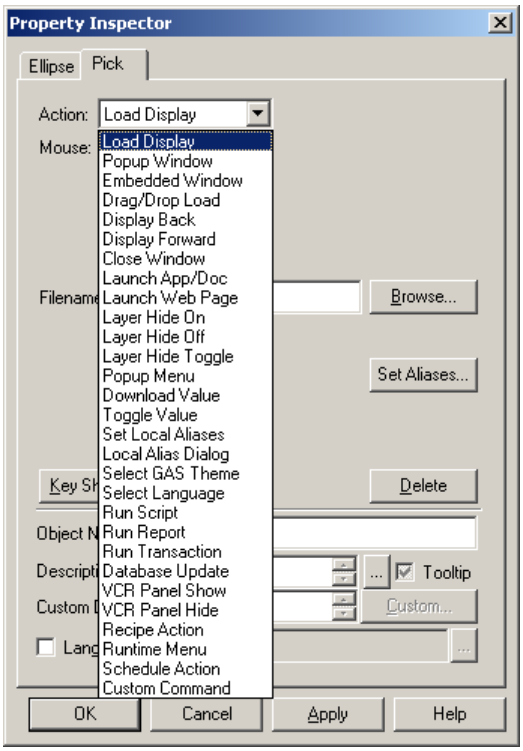


Figure 4.1 - Pick Action Property Inspector

Depending on the action you choose, the Property Inspector will show you different configurable parameters. For example, Figure 4.1 shows the parameters for a Load Display action. Since this is a load display action, it contains a field for the filename of the display that you wish to open. As we continue to discuss the different actions available you will see other parameters show up in the Property Inspector for configuration.

There are other ways to add a Pick Action besides creating it with a static object on the display. Towards the end of the Dynamics toolbar, you will see options to add a button, checkbox, and radio buttons. When you add any of these objects on the display, the Pick Action is automatically inserted with the object. Once you insert any of these dynamic objects, you can work with it like any other object on the display.

Now that you know how to add a Pick Action to an object, let us take a closer look at the different available actions. As you can see from Figure 4.1, there are many different native actions and it would be a very long chapter if we tried to cover all of them. As an alternative, we will only describe some of the most often used pick actions in this training. Throughout the next chapters, we will discuss other Pick Actions as they pertain to the subject being discussed. If there is a Pick Action that you would like more information on, feel free to look it up in the help files or in the Learning Center from the GenDemo.

4.1.2. Manipulating Displays with a Pick Action

You can manipulate displays and objects in runtime through Pick Actions in a few ways. The actions that you can perform separate naturally into three general categories: Opening Displays, Closing Displays, and Navigating between displays.

4.1.2.1. Opening Displays

In most projects, you will need to open a different display to obtain related information. When you need to open a different display in runtime, you should use a Pick Action to do the job. There are three Pick Actions to consider in achieving similar results:

- Load Display
- Popup Window
- Embedded Window

First, the Load Display Pick Action allows you to open a display in the same window as the parent display or the display where you click to perform the pick action. In essence, the load display action will load the targeted display into the currently running GraphWorX32 application window, effectively unloading the parent display. If you made changes to the parent display and did not save the changes before performing this action, your changes will be lost.

> ***Tip!*** You should always save your display before going into runtime to prevent loss of information. You should also save the display you are working on often. This way if anything unexpected happens (like a power failure) you will not lose too much work.

Because the Load Display action changes the display in the current window to the one specified in the “Filename” field, the only thing you need to configure is the target display to open. Later on, as we discuss aliasing, as you may also want to set aliases when opening the display.

The Popup Window action, in contrast to the Load Display action, will open the desired display in a new window. This is the simplest way to achieve similar effects of having a popup form; the alternative solution would be to create the form in VBA Scripts. Unlike the Load Display action, the Property Inspector of a Popup Window action consists of a few more configurable parameters.

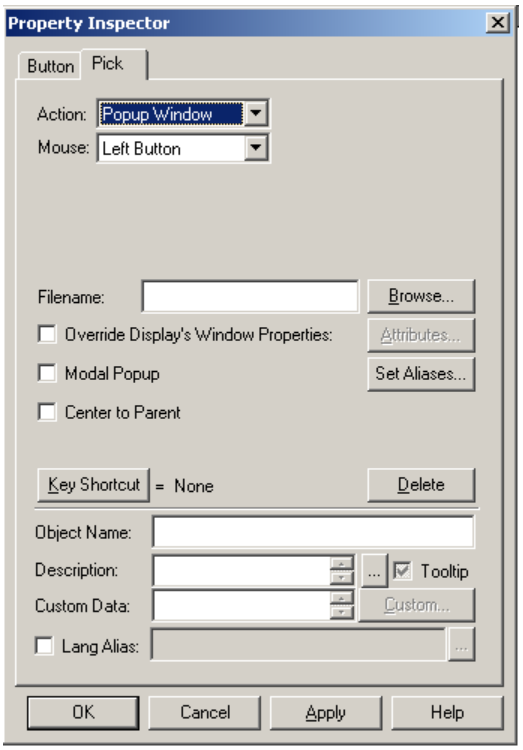


Figure 4.2 - Pick Action Property Inspector: Popup Window

Like the Load Display action, you will need to specify the display to open when triggering the Pick Action. You can also specify other parameters such as Override Display's Window Properties, Modal Popup, and Center to Parent.

> ***Tip!*** When you click on the "Browse" button to look for a display it will fill the "Filename" field with the full path that points to the display. The preferred way to reference a file is using relative paths. Using relative paths or only putting the name of the file in the field, will allow you to move the entire project directory to a different location without the need to change the parameter in each pick action. When using relative paths the parent display and the target file should reside in the same folder.

While some of the parameters are intuitive and the text describes the resulting behavior of the popup window when the parameter is checked, it may not be apparent what a Modal Popup will do. By default, this option is unchecked. Checking the option means that you will not be able to access the parent display when the popup child is still open. Normally, without the option checked, once the popup comes up, you can move it away from the parent display and you can still interact with the parent display; with the option enabled, you will not be able to interact with the parent display until you close the popup child display. This is helpful when you need to create a confirmation form.

If you check the "Override Display's Window Properties" option the "Attributes…" button will become active. When you click on the button, it will show you the **Popup Windows Properties** dialog. This dialog is very similar to the Window tab in the Display Properties dialog. Anything you configure here will only apply to the popup window; the settings you saved for the targeted display in the Display Settings stays unaffected.

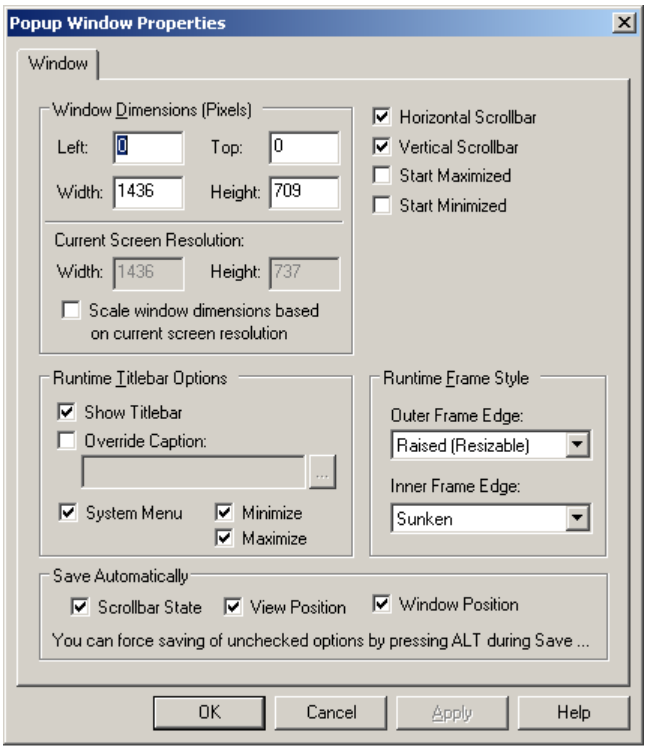


Figure 4.3 - Popup Window Properties

The last action that you can use to open a display is the Embedded Window Pick Action. As the name of the action suggests, it will open the targeted display in a window embedded into the parent display. While you can interact with an embedded window the same way you can interact with a popup window, the title bar of the window will never show it being active since its part of the parent display that called up the window.

4.1.2.2. Closing Displays

While there are many ways to open a display there are only a few ways to close a display. Aside from the usual Windows command and system menu options you can use the Close Window pick action to close the display. There is no extra parameter in the Property Inspector for this action. It will simply close the display from which you trigger the action.

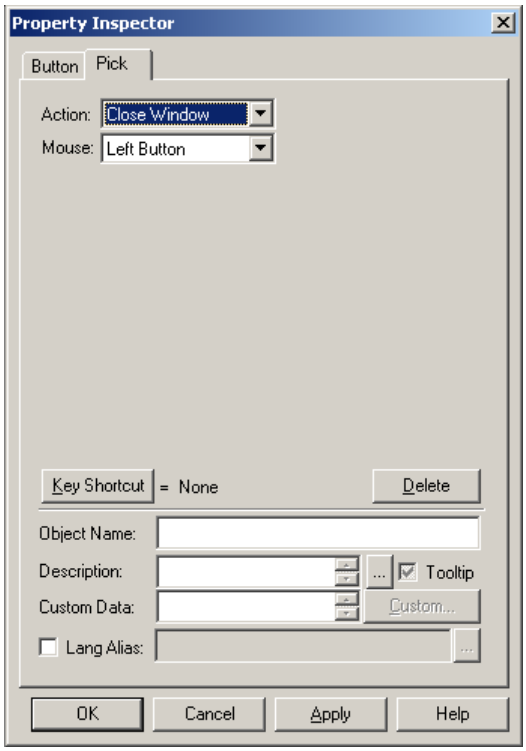


Figure 4.4 - Pick Action Property Inspector: Close Window

4.1.2.3. Navigating Between Displays

Once you have used the Load Display action a few times you now have a list of viewed displays. GraphWorX32 keeps track of this and you can browse back and forth in this list. The respective pick actions are Display Back and Display Forward. Like the Close Window Pick Action there is no extra parameter to set in these actions. It simply brings you one display back or one display forward in the order that you have viewed your display.

If you have not navigated to any other display after going into runtime or if you have reached the last display that you viewed, clicking these buttons will not result in any action. It will simply continue to display your current graphic file. In such case, you can still use the different ways to open a display to navigate to them.

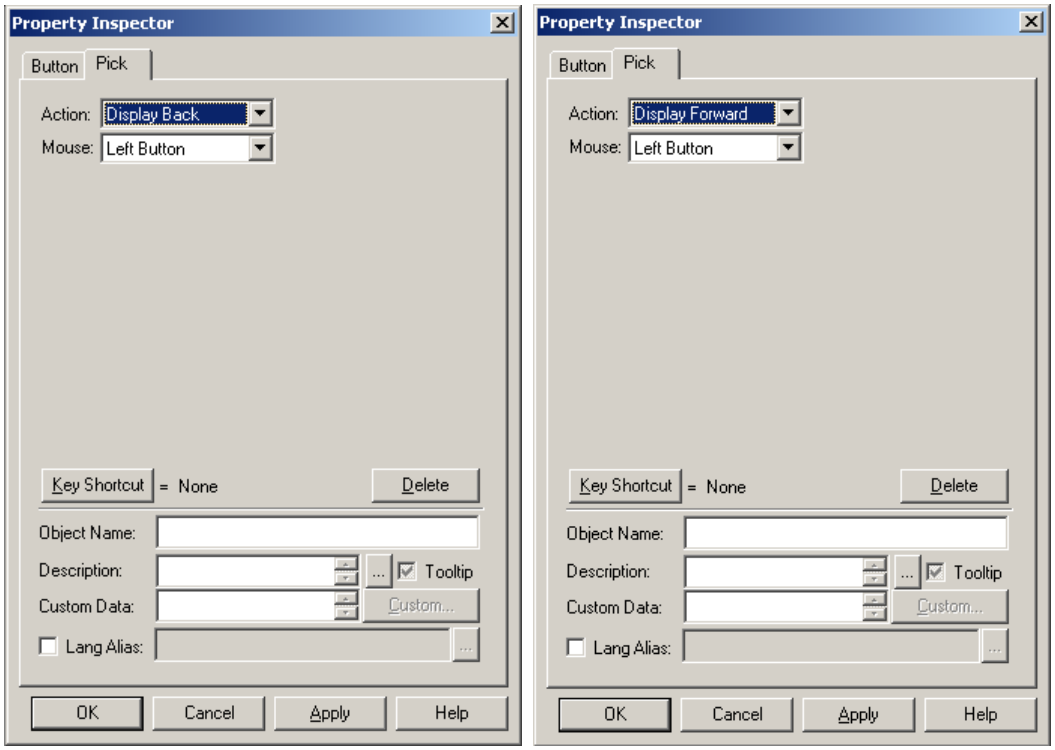


Figure 4.5 - Pick Action Property Inspector: Display Back (left) and Display Forward (right)

4.1.3. Writing OPC Values

In the previous chapter, we discussed how you can write to an OPC tag using a PPT; you can also use a Pick Action to write values to an OPC tag as well. There are two Pick Actions available: Download Value and Toggle Value.

In the Download Value Pick Action, you will need to configure, at the very least, the Data Source and the Value (On Up) parameter fields. The Data Source field should contain the OPC tag path that you would like to write the value to and the Value field should contain the information to write to the OPC tag. This can be a static value or you can use an expression.

Notice that in the upper-right hand corner there are three Execution Triggers: On Down, While Down, and On Up. By default, On Up is the checked. This means once you click on the button, it will perform the write to the OPC tag when you release the button or when the button comes back up after being pushed down. This brings up an interesting button behavior in GraphWorX32. With a normal Windows button, once you click on it, if you move the mouse away from the button and release the mouse, the action on the button will not trigger. This is not default behavior for GraphWorX32. Once you click on the button, regardless of where your mouse is when you release it the system will perform the associated action in the button.

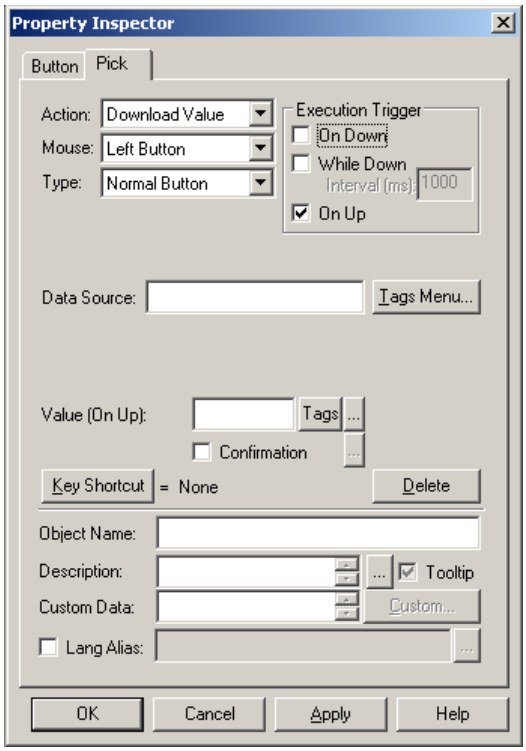


Figure 4.6 - Pick Action Property Inspector: Download Value

The other two options in the Execution Trigger have a similar function. The On Down trigger will write the value when you press the button and the While Down trigger will perform the write based on the Interval configured while the button is pressed. The While Down trigger is infrequently used for a button, but it is very useful when used with a checkbox.

Consider the scenario where you have to increase a value by one every second. Instead of having the operator write the value every second (they may not be fast enough to do such writes), you can use a checkbox to trigger a Download Value action in conjunction with the While Down trigger. You can set the While Down Interval to 1000 milliseconds (one second) and use an expression to add one to the OPC tag in the Data Source. With this setup, the OPC tag will increase by one every second while the checkbox is checked.

> ***Tip!*** You can use a checkbox with the While Down execution trigger whenever you need to do something periodically. Simply set the "While Down Interval" to specify how often to perform the action.

Before we finish the discussion on Download Value, let us consider one more scenario. Many times, there are buttons on the production floor where an operator would press a button for a few seconds to have the system perform a different action.

When the operator releases the button, the system goes back to its normal routine. You can simulate such a button in GraphWorX32. Consider the configuration of a Download Value action where you enabled both the "On Down" and "On Up" execution triggers, where the "On Down" trigger writes the value to the system to trigger a diversion and the "On Up" trigger writes the value to the system to indicate everything should run in the normal state. In such a case, when you press the button, and while the button is pressed, the OPC tag will have a value that indicates the system should run in the alternate state. When you let go of the button, the system will return to normal.

The second action available to write to an OPC tag is the Toggle Value Pick Action. This pick action will allow you to write two values to the OPC tag, alternating between the values to write each time you click on it.

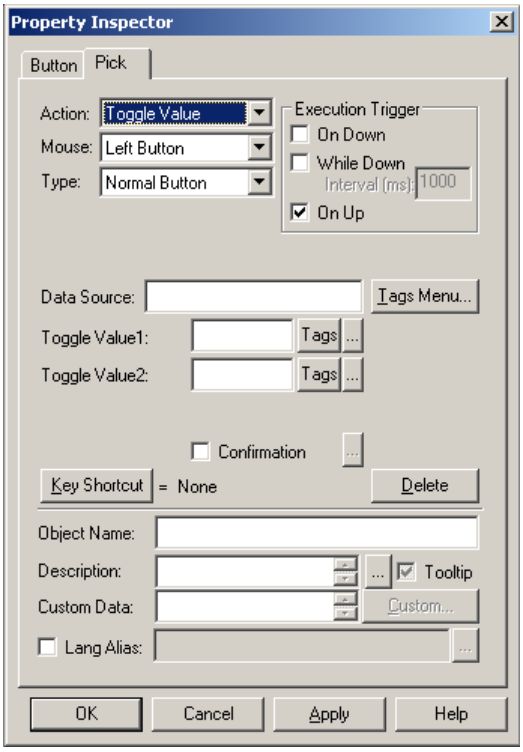


Figure 4.7 - Pick Action Property Inspector: Toggle Value

In the "Toggle Value1" and "Toggle Value2" fields, you can enter the value (static or through an expression) to switch between. The first time you click on the button, if the OPC tag does not have the same value as the one defined for "Toggle Value1," then the value in "Toggle Value1" will be written to the OPC tag. The next time you click on the button, the value will switch to the one specified in "Toggle Value2". If the value in the OPC tag is already the value defined in "Toggle Value1" the first time you click on it, then it will write the value defined in "Toggle Value2" to the tag.

The next time you click on the tag, it will write the value defined in “Toggle Value1”. Each subsequent time you click on the button, it will alternate writing the two values.

4.1.4. Popup Menu

The Popup Menu Pick Action allows you to create a runtime menu of your own. At a first glance, the Popup Menu Pick Action does not offer too many options in terms of configuration. However, once you click on the “Menu Editor…” button in the Property Inspector you will get a dialog to build your menu.

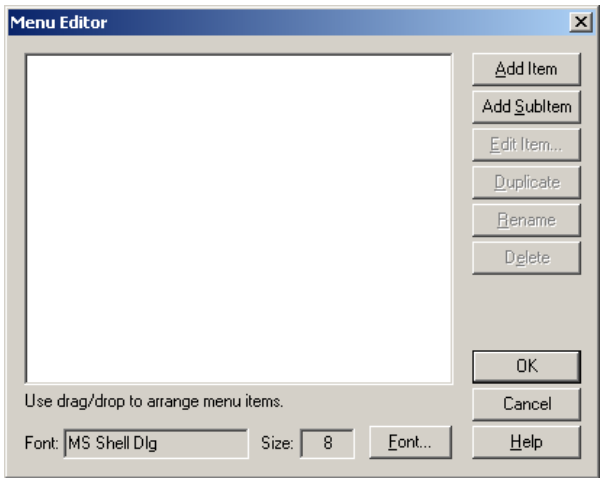


Figure 4.8 - Menu Editor for Popup Menu Pick Action

You can click on the “Add Item” button to add a Pick Action on the menu item. If you need to create an item as a child or a submenu item, click on the “Add SubItem” button with the parent selected. This will convert the parent item to a Sub Menu icon and strip from it any Pick Action you have configured.

Once you have inserted an item into the Popup Menu, you can click on the “Edit Item” button to configure an action when someone clicks on the item. The action list for a menu item is similar to the list available when adding a normal Pick Action. The only exception is that you no longer have the option to create a Popup Menu.

Once you choose the Pick Action for the menu item, be sure to configure the Description of the item. The text in the description will be the text shown for the menu item in runtime. If you forgot to configure it in the Property Inspector while you are configuring the pick action, you can click on the “Rename” button to set the description in the Menu Editor.

4.2. ActiveX

ActiveX controls are reusable software components that can perform a particular function. The idea of ActiveX controls has been introduced by Microsoft; emerging from the Component Object Model (COM) and Object Linking and Embedding (OLE) technologies, ActiveX controls quickly become a popular way to embed functionality within programs. An advantage to using ActiveX controls is that you can embed the control within another application, such as Internet Explorer and therefore, the container program would gain the necessary functionality in the ActiveX control.

Because of the vast capabilities that ActiveX controls promised, many components of GENESIS32 are ActiveX controls, allowing these components to work inside different container programs such as GraphWorX32 and Internet Explorer. The key ActiveX controls available in GENESIS32 are the Alarm Viewer, Alarm Reporter, and Trend Viewer, which we will discuss in later chapters as we discuss the respective topic.

Aside from the main ActiveX controls, ICONICS also offers other ActiveX controls such as the Data Mining Grid, the Tree Explorer, the Security Indicator, and many more. In this chapter, we will concentrate on the free ICONICS ActiveX controls. You can find these ActiveX controls in the ActiveX toolbar. The free controls include the Gauge, Switch, and Vessel ActiveX controls. You can download these controls on the ICONICS website at http://www.iconics.com/support/free_tools.asp, where many other OPC tools are available.

The configuration for these controls is intuitive. Once you learn how to configure one of the controls it is easy to figure out how the others work. Since there are many of these controls and their configuration is similar, we will only show one here as an example.

You can click on the last icon in the ActiveX toolbar to add the Vessel ActiveX control. Double-clicking on the control will bring up the properties dialog for the control. There are five tabs available for configuration.

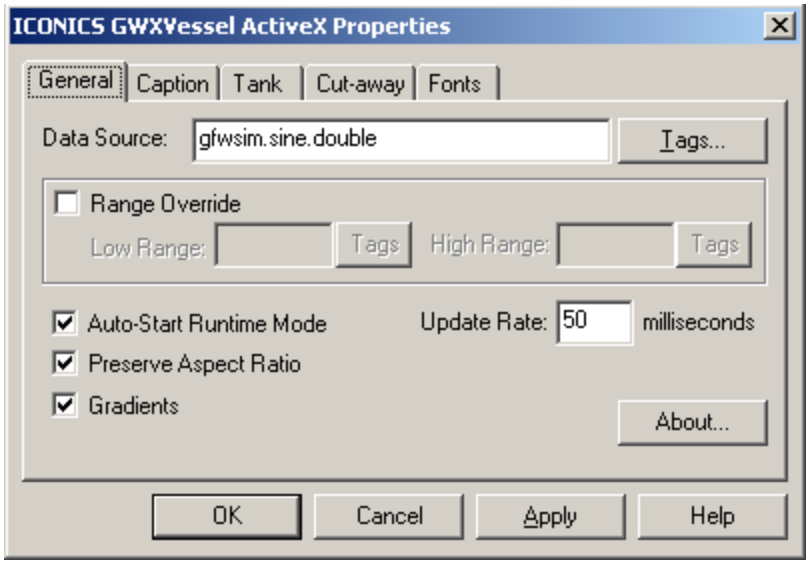


Figure 4.9 - Vessel ActiveX Properties

In the General Tab, you can configure the Data Source that drives the animation of the ActiveX. The default data source uses a GraphWorX32 simulation sine tag. You can also configure other settings such as overriding the range of the data source and changing the OPC polling rate.

On the Caption tab you can change the title of the ActiveX control. By default the title says “ICONICS.” You can change it to other text that’s more appropriate for your project. In this tab you can also set the text color of the title as well as the tooltip text when the mouse lingers over the control.

On the Tank tab you can set the different predefined tanks available. The default shape of the tank is “Barrel” but you can configure it to use other presets. If you have an image of the tank that you would like to use you can import the image file. The file must be in meta file format (.wmf or .emf).

The Cut-away tab allows you to change the shape of the cut away on the tank. This gives the illusion of looking inside the tank. You can change the color in the cut away to match with the color of the actual material stored in the tank. Figure 4.10 shows a Vessel ActiveX control with customized parameters.

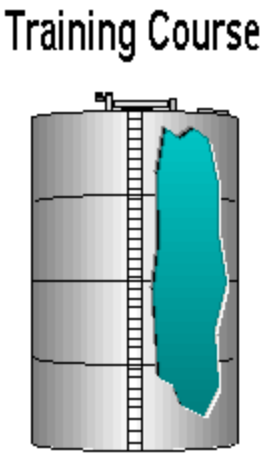


Figure 4.10 - Customized Vessel ActiveX Control

As we continue to discuss the different components of GENESIS32 we will gradually expose you to more of the available ActiveX controls. If you wish to learn more about an ActiveX control you can refer to the respective help files.

Lab: Controlling the Mixers L4

In the previous lab we developed a display to view and control the pill making process. The display allows us to see the amount of raw material in each mixer but does not allow us to refill it. In this lab we will create a popup display that allows us to refill the mixer.

Estimated time to complete this lab: 30 minutes

4.1. Task Checklist

In this lab, you will complete the following tasks:

- Create a popup display to refill mixers
- Try to pass a local alias from one display to another
- Use Pick Actions to download values, open popup displays, and close them

4.2. Support Files

You will need the following file(s) to complete this lab:

- ACMEPillMachine.gdf

4.3. Mixer Popup Display

Let us create a popup menu to refill our mixers. Recall that in the previous lab we used local aliases to simplify some of our configuration. We are going to use the same idea in the mixer popup display. We will take it one-step further by passing the local alias values from the ACMEPillMahcine.gdf to this one but first, let us build the mixer popup display.

1. Start GraphWorX32 by going to Start → Programs → ICONICS GENESIS32 → GraphWorX32 → GraphWorX32.

2. Make sure the GraphWorX32 application window is maximized.

3. In **Format →Display Properties** select the General tab. Set the display size (this could be referred to the GraphWorx32 container size) to a width of 500 and a height of 300.

> ***Tip!*** You can do this in the Display Properties. Although you set the window size now you will not actually see it changing because the application window is maximized.

4. Select the Windows tab. Under the "**Runtime Titlebar Options**" section, check the "Override Caption" option and type in "Mixer Valve Status". Also uncheck the "System Menu."

5. Under "Window Dimensions (Pixels), set the dimensions as indicated in Figure L4.1 - Display Properties: Window Tab.

6. Uncheck "Horizontal Scrollbar," "Vertical Scrollbar," "Start Maximized" and "Show Runtime Menubar" checkboxes.

7. Click on "OK" to save the settings and close the Display Properties dialog.

Figure L4.1 - Display Properties: Window Tab

8. Add a process point to the display and type in *$"Mixer <<MixerID>>"$* in the Data Source field.

9. Set the Client Display Type to "String" and change the Format field to show 10 characters.

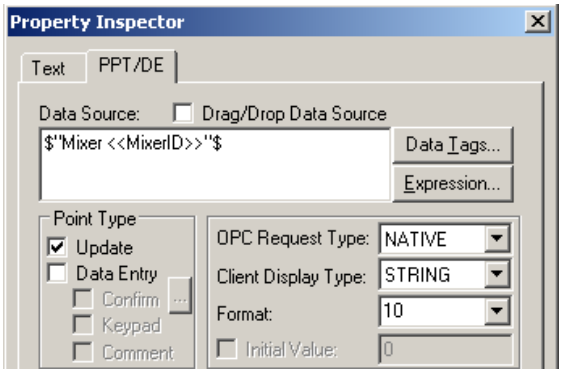


Figure L4.2 - Data Source for a Process Point

10. Click on "OK" to close Property Inspector.

11. Resize the process point on the display so that it looks like heading text and move it to the upper left-hand corner. See Figure L4.6 - ACMEMixers Display for details.

12. Find the Ball Valve in the Symbols library and drag it onto the display.

13. Select the valve symbol .

14. Click on the "Color Change" icon in the Dynamics toolbar to add a color dynamic to it.

15. In the Color tab of the Property Inspector, click on the "Data Tags…" button to bring up the UDB.

16. After you select ValveStatus tag for Mixer A. You should see the full tag path in the text box that is near the top of the UDB (Unified Data Browser). ICONICS.DataWorX32.8\USA.Foxborough.Machine1.Mixers.MixerA.ValveStatus

17. Select the "A" in MixerA and type in "<<MixerID>>" without the quotes, then click on "OK" to close the UDB.

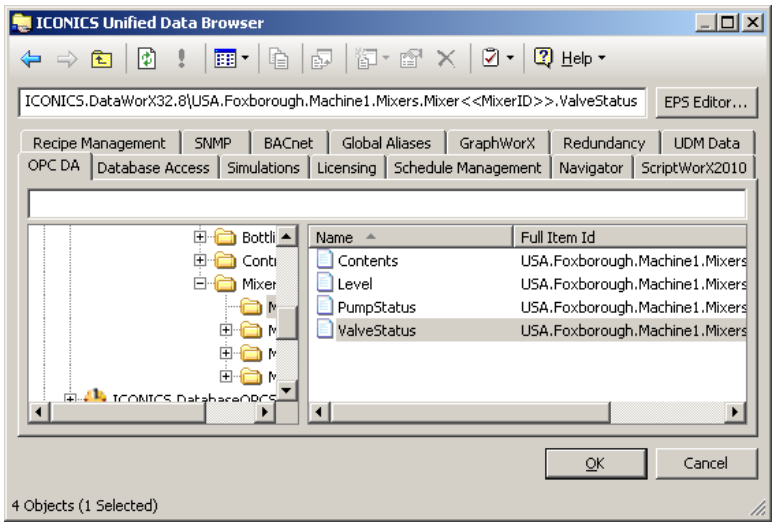


Figure L4.3 - Unified Data Browser with Tag Selection

18. In the Property Inspector, click on “Add” to add the tag.

19. In the “Apply Color Change To” section, uncheck “Line Color” and click in the color block for “Fill Color.”

20. Change the color to red (Red: 255; Green: 0; Blue: 0) using the Color Palette dialog.

21. Choose the “Change Color on False” option. Your configuration should look similar to Figure L4.4.

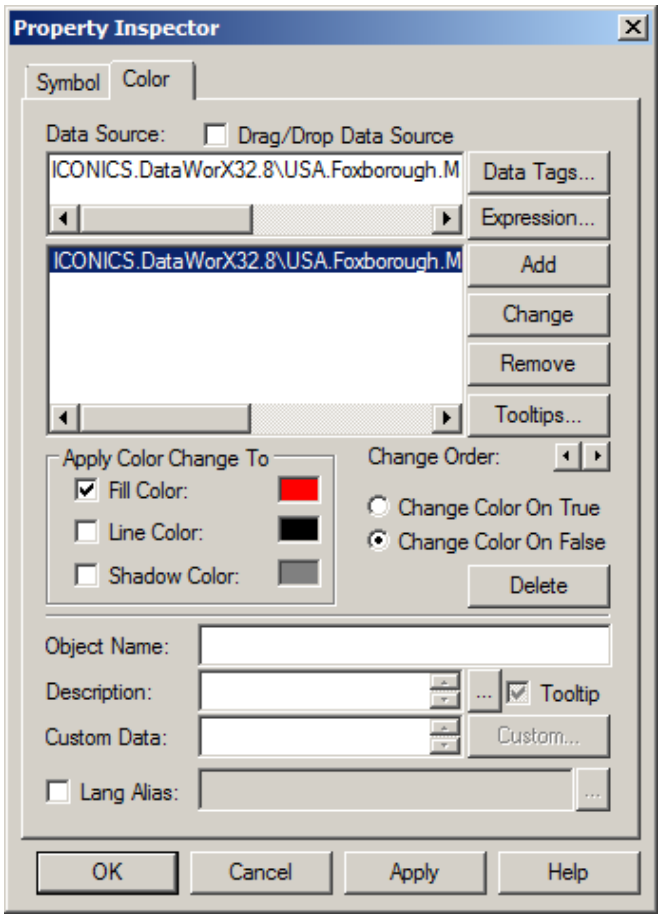


Figure L4.4 - Properties Inspector: Color Dynamic

22. Click the Add button again, and uncheck the "Line Color" option in the "Apply Color Change To" section.

23. This time change the "Fill Color" to green (Red: 0; Green 255; Blue: 0) using the Color Palette. Under Change Order, leave as "Change Color on True."

24. Click on "OK" to save the settings and close the Property Inspector and hit Esc until the valve is no longer selected.

25. Click on the "Button" icon in the Dynamics toolbar and click toward the lower left-hand corner of the display.

26. In the Property Inspector, change the Action to **Download Value**.

27. Click on the "Tags Menu…" button and browse for the MixerA.ValveStatus tag again.

28. Change the "A" in MixerA to our local alias "<<MixerID>>" like you did before. Click OK.

29. In the “Value (On Up)” field enter a value of “1.”

30. Your Property Inspector should look similar to Figure L4.5.

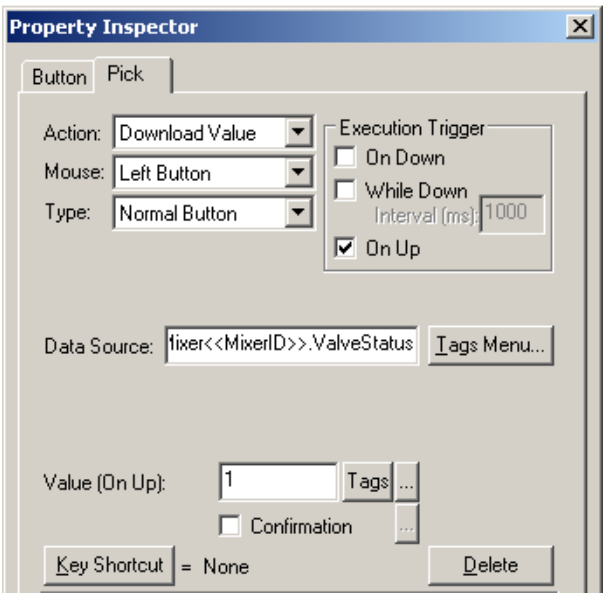


Figure L4.5 - Property Inspector: Download Value

31. Click on the Button tab and change the text below the word Font: to read “Open Valve.” Click on “OK” to save the settings and close the dialog.

32. Using ctrl-d with the button selected, duplicate the button that you have just created. Select the duplicate and double-click on it to open the Property Inspector.

33. Change the button text to read “Close Valve”.

34. Select the Pick tab and change the “Value (On Up)” field to 0.

35. Click “OK.”

36. Click on the “Button” icon in the Dynamics toolbar and click toward the lower right-hand corner of the display.

37. Select “Close Window” as the Action.

38. Go to the Button tab and change the button text to read “Dismiss Display” then click on “OK” to save the settings and close the dialog.

39. Resize the buttons to fit across the display.

> ***Tip!*** You can try resizing one button. Then hold the shift key to select multiple objects and use the “Same Size” icon in the Arrange toolbar to make the other buttons inherit the dimensions.

40. Once you have the buttons sized the way you like, select all three buttons and click on the "Even Across" icon.

41. Align the tops of the buttons by clicking on the "Tops" icon.

42. Your finished display should look similar to Figure L4.6.

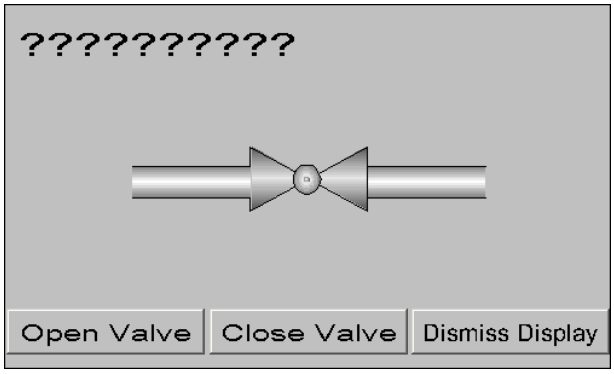


Figure L4.6 - ACMEMixers Display

1. Minimize the window so that it shows you the display in the dimensions we set before. Save the display in the current size in the Training directory and name the display ACMEMixers.gdf. If the file name already exists, you may replace it.

2. You can hit Ctrl + R to enter the "Runtime Window Properties Mode" to make sure the display looks OK. If not, you can make your edits in this mode. You can hit Ctrl + R again to return it to the normal configuration mode.

4.4. Connecting Displays

1. Open the ACMEPillMachine.gdf display. We will now add Pick Actions to this display to open our ACMEMixers display.

2. Select MixerA (Tank A) and click on the "Pick" icon in the dynamics toolbar.

3. Change the Action to "Popup Window."

4. In the Filename field, type in "ACMEMixers.gdf".

   Note: as previously mentioned, selecting a file using the Browse button will result is a fully qualified path being defined. For portability reasons you're typically better off just entering the filename without the path.

5. Check the "Center to Parent" checkbox.

6. Click on the “Set Aliases…” button to bring up the Local Aliases Configuration dialog.

7. Select the “Set Aliases Directly” option near the bottom of the dialog.

8. In the Alias Name textbox, type in “MixerID” without quotes and in the Alias Definition textbox type in “A.”

9. Click on the “Add” button to add the definition.

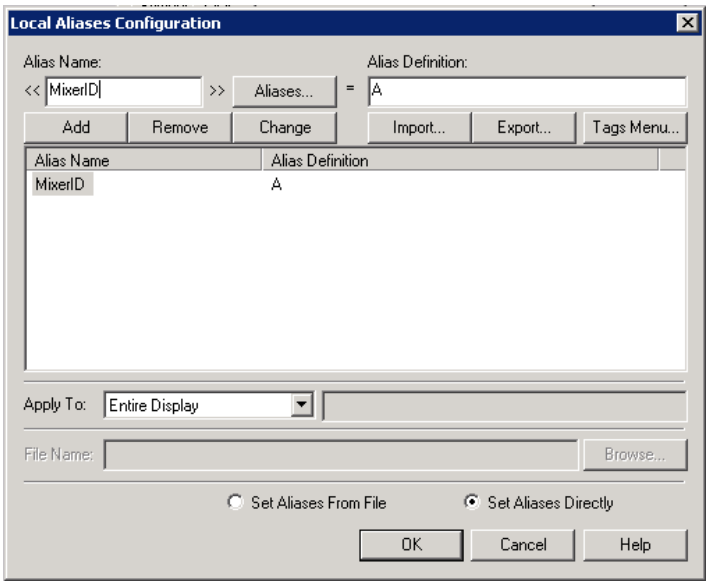


Figure L4.7 – Partial Local Aliases Configuration

10. Click on “OK” to save the changes and close the dialog.

11. Click on OK again in the Property Inspector of the Pick action and close it.

12. Repeat steps 2-11 for Mixers B, C, and D; changing the Local Alias definition accordingly.

    a. As an alternative, to speed things up, you can repeat steps 8-12 if you select the 1st tank and then select “Copy” from the “Edit” menu or Ctrl+C. Select Mixer B, do a right mouse click and select “Paste Dynamics.”

13. Save the display and go into runtime. Try to fill up the different mixers. Most probably nothing will happen but you can get an idea of how the user will interact with the process.

Review: GraphWorX32 Pick Actions and ActiveX *R4*

4.1. Chapter Objectives

In this chapter, we discuss some basic pick actions available in GraphWorX32 as well as ActiveX Controls. By now, you should have a basic understanding of the following topics:

- How to add a Pick Action into a display's object
- The different objects that contains a pick action by default
- How to configure the following pick actions:
    - Load Display
    - Popup Window
    - Embedded Window
    - Close Window
    - Display Back
    - Display Forward
    - Download Value
    - Toggle Value
    - Popup Menu
- How to configure the Vessel ActiveX

4.2. Summary

There are many tools available in GraphWorX32; two of the tools that we discussed in this chapter are Pick Actions and ActiveX controls. Similar to other dynamics discussed in the previous chapter, you can add a Pick Action to any object on the display. Unlike other dynamics, a Pick Action is only triggered based on user interaction as opposed to OPC values. There are objects available in GraphWorX32 that come with a pick action natively. The objects are: button, checkbox, and radio button. You can use a Pick Action to open new displays, navigate between displays, create your own user menu, and much more.

Another tool available in GraphWorX32 is ActiveX controls. These are considered self-contained programs. Each ActiveX control performs a specific function.

Since ActiveX controls can be embedded into other containers, such as Internet Explorer, it becomes more powerful than other programs that can only run in their own containers. ICONICS offers many different ActiveX Controls. The most widely used are the Alarm Viewer, Alarm Reporter, and TrendWorX32 Viewer ActiveX controls. We will discuss these controls in later chapters.

4.3. Terms

ActiveX Controls	Close Window	Display Back
Display Forward	Download Value	Embedded Window
Load Display	Modal Popup	Pick Action
Popup Menu	Popup Window	Toggle Value

4.4. Test Your Knowledge

In this chapter, we discussed many basic Pick Actions and a few ActiveX controls. Before you continue to the next chapter, you should make sure that you are able to answer the following questions.

1. What happens if you did not configure the “Filename” field of a Load Display pick action?

    a. It would do nothing
    b. It would load the currently open display again
    c. It would load a random display
    d. It would load a blank display

2. What happens when you click on the Display Back action when you have not browsed to any other display since you went into runtime mode?

    a. It would do nothing
    b. It would load the last open display
    c. It would load a random display
    d. It would load the first GenDemo display

3. True or False: You can use relative paths to reference a display in the Load Display pick action.

4. True or False: You can change the popup window properties in the Popup Window pick action.

5. What happens when you only define one value in a Toggle Value pick action associated to a button?

   a. It would act as a Download Value pick action and write the same value each time you click on the button.
   b. It would alternate writing the defined value and zero to the OPC tag
   c. It would be grayed out in runtime and you would not be able to click on the button.
   d. It is not possible to have such a configuration; you cannot close the Pick Action Properties Inspector without providing a second value.

6. What happens in runtime with the following configuration?

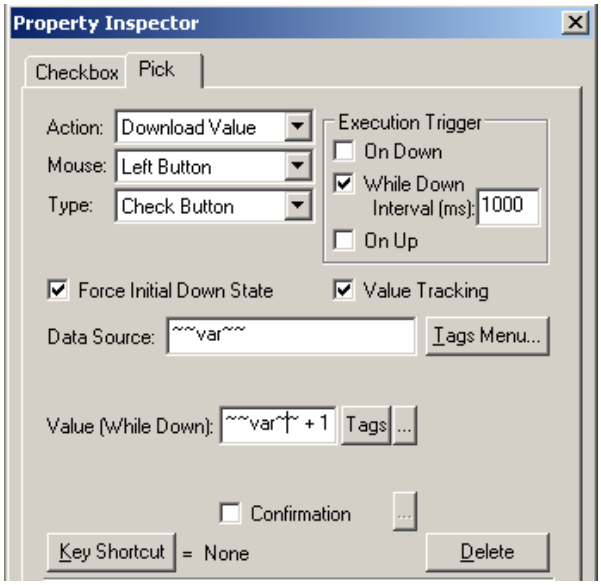


Figure R4.1 - Question 6 Pick Action Configuration

   a. It would immediately start adding one to the local variable ~~var~~ every second. Effectively keeping track of how long the display is in runtime in seconds as long as the checkbox is kept checked.
   b. It would do nothing at first. When the operator checks the checkbox, it would start adding one to the local variable ~~var~~ every seconds
   c. It would do nothing at first. When the operator clicks on the button and holds it there, it would start adding one to the local variable ~~var~~ every seconds. Effectively tracking how long the operator held the button for in seconds.
   d. It would add one to the local variable ~~var~~ one second after each time the operator pushes the button, allowing for a delay in the write.

7. True or False: GraphWorX32 has its own set of simulation tags.

8. True or False: You can insert ActiveX controls developed by a third-party company into GraphWorX32 and use the available functions in the controls.

9. Which of the following are ICONICS ActiveX controls?

    a. Alarm Viewer
    b. Alarm Reporter
    c. System Monitor Control
    d. Trend Viewer
    e. All of the above
    f. Only A, B, and C
    g. Only A, B, and D
    h. Only B, C, and D
    i. None of the above

10. Where do you define the name of an item in a Popup Menu?

    a. In the Appearance tab of the Popup Menu properties
    b. In the Description field of the Pick Action Property Inspector for the item
    c. In the ItemName field of the Pick Action Property Inspector
    d. You cannot configure this; it is smart enough to detect the pick action and use that in conjunction with some other properties of the pick action to create a name.

Introduction to Workbench32 5

GENESIS32 comes with a centralized project management tool: Workbench32. This tool is the single application that you need to learn to set up your project structure. Furthermore, this application is based on Microsoft .NET technology allowing you to make changes to your project over the Internet. In this chapter we are going to introduce you to Workbench32. If you prefer, you can use this application throughout the course.

By the end of this chapter you should be familiar with the following:

- Workbench32 Layout
- Available Providers in Workbench32
- Creating and Opening Projects in Workbench32
- Packing and Unpacking Projects
- Workbench32 Runtime Mode

5.1. Workbench32

Workbench32, sometimes simply referred to as the 'Workbench,' is the centralized Web-based environment for GENESIS32 product configuration and runtime operations. It provides a framework for security, project management, and for multimedia technology to configure alarms. Workbench32 can load various providers or plug-ins that enable the user to configure, monitor and control a project. The key elements of Workbench32 are its browsing capability, a user customizable control graphic, and a command set through the toolbar and menus that are context-sensitive to each provider.

The power of Workbench32 is to act as a single interface for a wide variety of needs. You can use Workbench32 as an advanced operator interface for distributed service management or you can configure remotely with Workbench32 by connecting to remote Project Servers with either the HTTP or HTTPS protocols.

5.2. Workbench32 Layout

You can launch Workbench32 by going to Start → ICONICS Workbench32 → ICONICS Workbench32. When the application opens, notice that there are three major sections: the toolbars and menus at the top, the Tree Explorer to the left, and the contents area to the right, as shown in Figure 5.1.

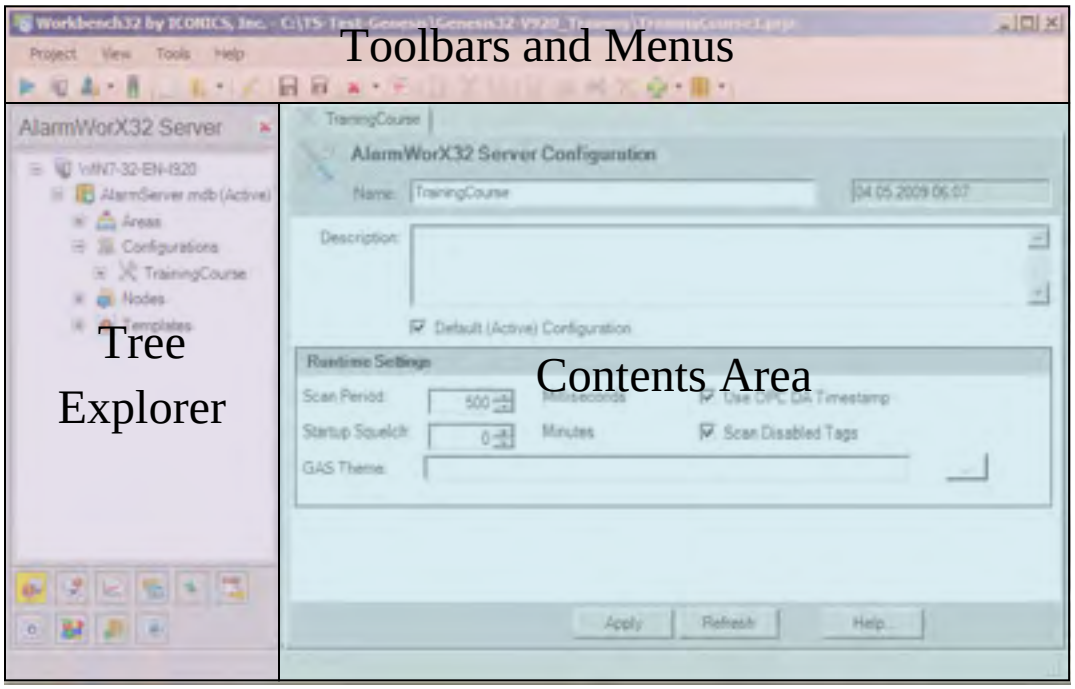


Figure 5.1 - Workbench32 and the Different Sections

What you will see in the contents area depends on the object that you have selected to open in the Tree Explorer. Double-clicking on an item in the Tree Explorer normally opens the configuration form for that item in the currently active tab in the contents area. If there are no tabs in the contents area already open, it will create a new tab. You can also open new tabs for different items by middle clicking on the item or right clicking on the item and selecting the “Edit On New Tab” option.

The Tree Explorer is divided into three parts (from top to bottom): the provider name, the navigation tree for the provider, and a button bar showing all available providers. The top portion indicates which provider you are currently viewing, while the middle portion provides the structure for that provider. At the bottom of the Tree Explorer are many different buttons, which, when clicked, show the respective provider. This structure allows you to configure many different parts of GENESIS32 within Workbench32.

You can add and remove providers using the toolbars and menu at the top of the application. To do this, you can click on the Project menu item and choose the “Add/Remove Providers” option. This will bring up the Select Applications in the Project dialog as shown in Figure 5.2. You can check and uncheck the providers as necessary. You can close the dialog by clicking on “OK” when you are done selecting the providers.

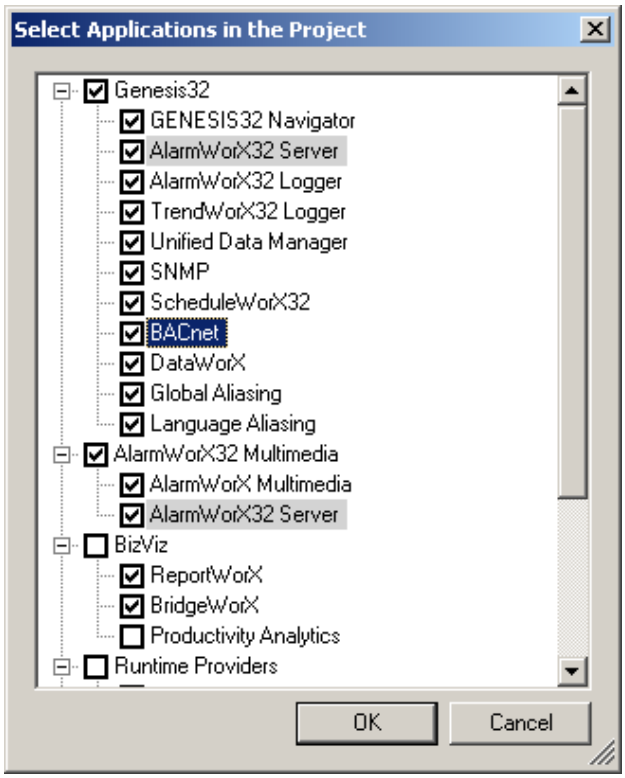


Figure 5.2 - Select Applications in the Project Dialog

Using the menu and toolbar options you can also create a new project, save a project, and much more. You can explore the different options in the menu and toolbars. If you move your mouse over a button on the toolbar a tooltip will come up giving you a description of what the button does.

5.3. Pack and Go

One of the most useful tools that Workbench32 provides is the Pack and Go feature. From the Workbench32 Project menu you will find two options: “Pack” and “Unpack.” These options allow you to pack up your entire project from the development machine, take it to the deployment site and unpack the project at the end user facility. With a few clicks you can move your project from your development machine to where it needs to run.

5.3.1. Packing a Project

When you select the option to pack a project a wizard will come up to guide you through the process starting with a Welcome screen. If you do not want to see this message again, you can check the option to "Never show this welcome page again" at the bottom. If you need to make more changes to the project before packing it, you can click on the "Cancel" button to close the wizard. Otherwise, clicking on "Next" will let the wizard proceed to the next step.

Figure 5.3 - Pack and Go Wizard: Welcome screen

In order to pack a project you will need to tell the wizard which project to pack. There are two options: the "Current project," which is the project you were editing in Workbench32 when you launched the wizard, or you can choose to provide a separate project file by selecting the "File" option and browsing for the "*.prjx" file that you wish to pack.

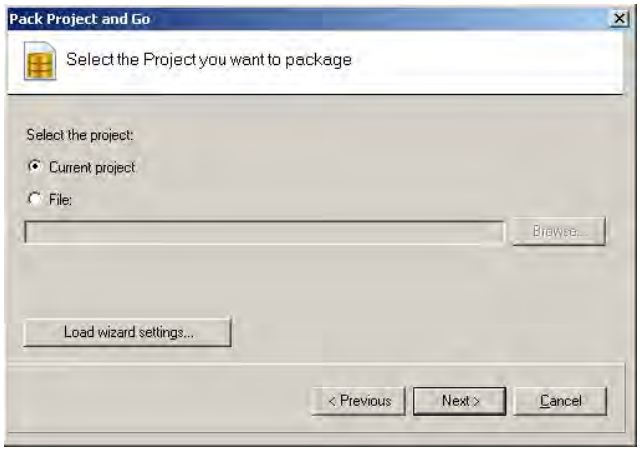


Figure 5.4 - Pack and Go Wizard: Select Project

Once you have selected the project to pack you can click on “Next” to proceed. On the next screen you will need to provide the package file name and location. Click on the “Browse” button to find a location on the computer where you want to put the package file and give it a name.

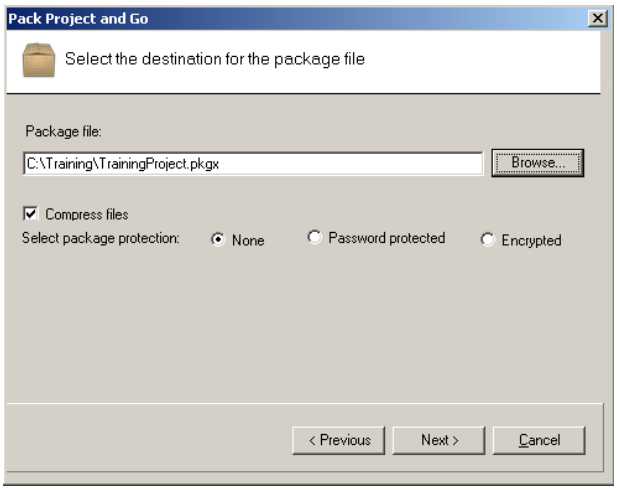


Figure 5.5 - Pack and Go Wizard: Select Package File

There are two additional options in the dialog: “Compress files” and “Package protection.” The first option is checked by default and we recommend that you leave this option checked to ensure the smallest package file. Should you decide to protect your package, you can choose to use a password or encrypt the project. Two password fields will appear asking you to provide a password with which to protect it or encrypt the file depending on the option you choose. You will have to type the password twice. The password is case sensitive and unrecoverable. You should keep the password for the corresponding project package in a safe place. NOTE: If you choose “Encrypted,” ICONICS will NOT be able to recover your project package should you lose the password. If you choose “Password protected” (without encryption) alone, ICONICS can assist in recovering your project package, should you lose the password. Keep in mind that “Encrypted” is more secure than “Password protected.” And both are more secure than “None.”

On the next screen, you will be able to choose the different components to put into the package file. Clicking on a provider in the provider list on the left-hand side will show you the different components of the provider that you can pack on the right-hand side. If you want to pack everything in the provider, simply check “all” at the top of the tree.

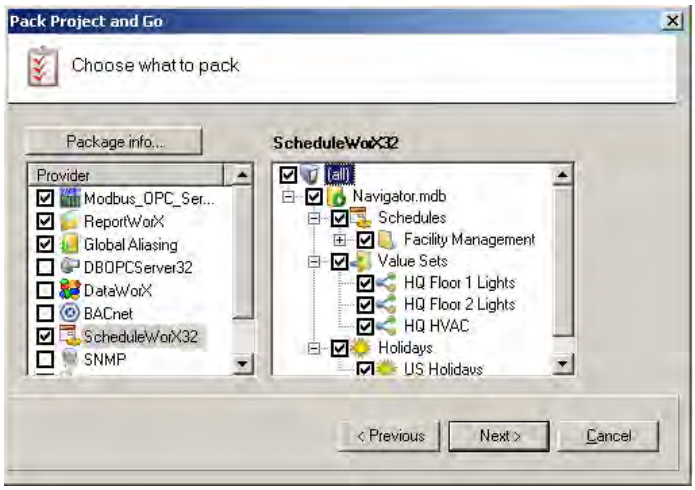


Figure 5.6 - Pack and Go Wizard: Select Components to Pack

When you have chosen all the components you wish to pack, click on “Next” to continue. On the Next screen, you can choose to “Enable Find & Replace.” This option is useful if you already know the computer name of the deployment machine. If you do not know the name of that computer when you pack the project, replace the computer name with a place holder, e.g. NodeName. When unpacking, you can replace this placeholder with the target computer name. This is optional.

Clicking the “Next” button will bring you to the final screen of the wizard. The option to “Show the log file when the operation finishes” is checked by default. We recommend leaving this option checked so that you can check the pack process and make sure all of your files are packed successfully.

You can also save the settings of the Pack and Go process you used this time by clicking on the “Save settings for later use” button at the bottom of the screen. Saving the settings may help you pack a project quicker next time.

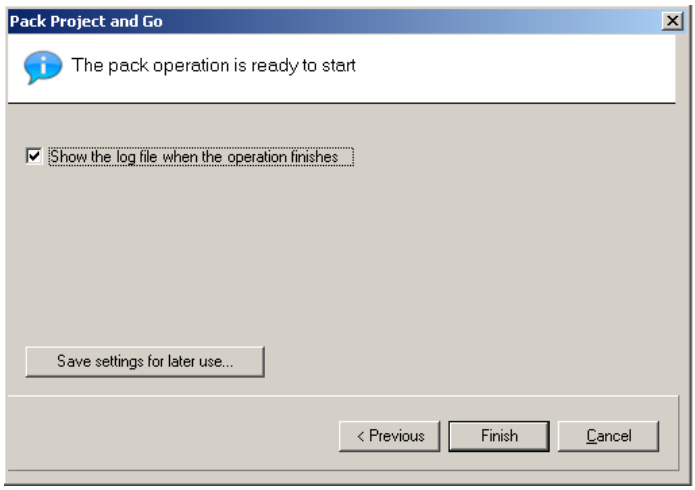


Figure 5.7 - Pack and Go Wizard: Final Screen

When you are ready, click on the “Finish” button to start the packing process. The wizard will show you the progress of the process. Depending on the components you have selected to pack, the duration of the packing process may vary.

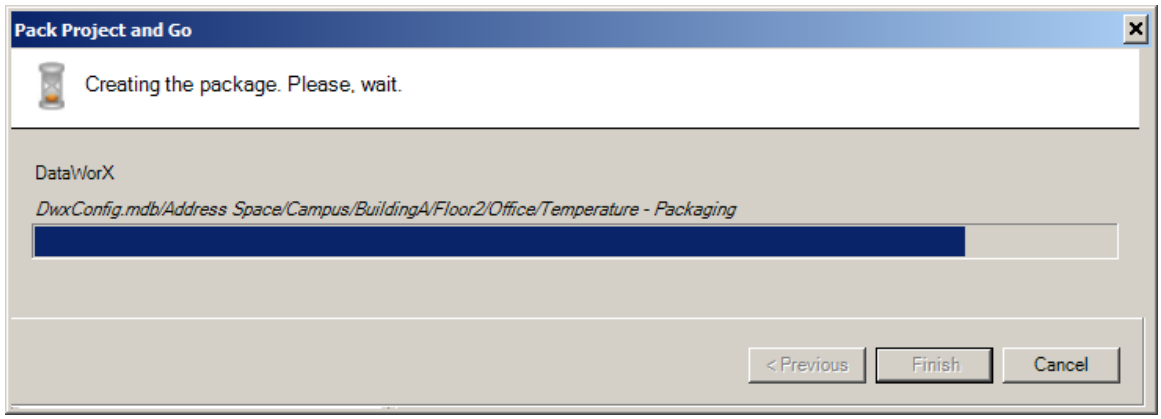


Figure 5.8 - Pack and Go Wizard: Packing in Process

When the wizard is done packing the project you will see the project log file in Workbench32 if you have chosen to show it. The project log will tell you which project file was packed, when it was packed, where the package file is located, how big the packaged file is, and all the different components that were packed (separated by providers). A green checkmark in the “Status” column indicates that the file was packed successfully and a red ‘X’ in the same column indicates the file failed to pack. In this case, you should check the “Status Info” column for more information on why the file failed to pack.

Figure 5.9 – Pack and Go Report

5.3.2. Deploying a Project

It is easy to deploy a project from a packed project file. You can choose the "Unpack" option from the Project menu. The same Pack and Go Wizard will come up, but it will take you through different steps to deploy the project.

The first screen you see after the welcome screen is one that asks you to provide the Package file. You can choose the file by clicking on the "Browse" button. If you have saved settings for the Pack and Go Wizard before, you can also click on the "Load wizard settings" button to provide the configuration file.

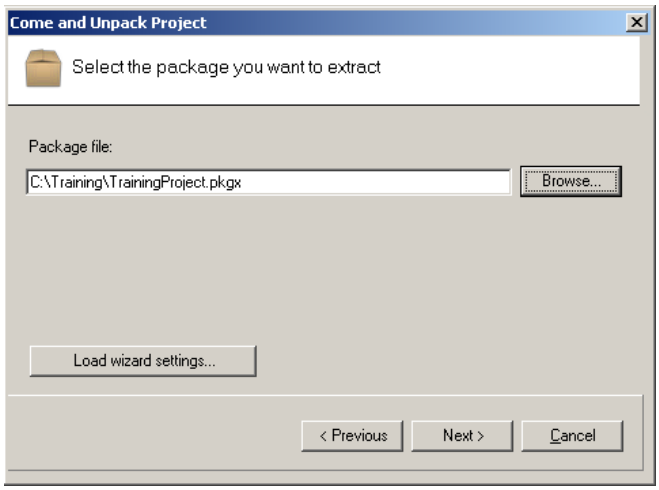


Figure 5.10 - Pack and Go Wizard: Select Package File

Once you have chosen the file to unpack, the next screen will ask you to pick a project destination folder for the unpacked files. You can choose the current project or specify a project file of your own. Click on "Next" when you are ready.

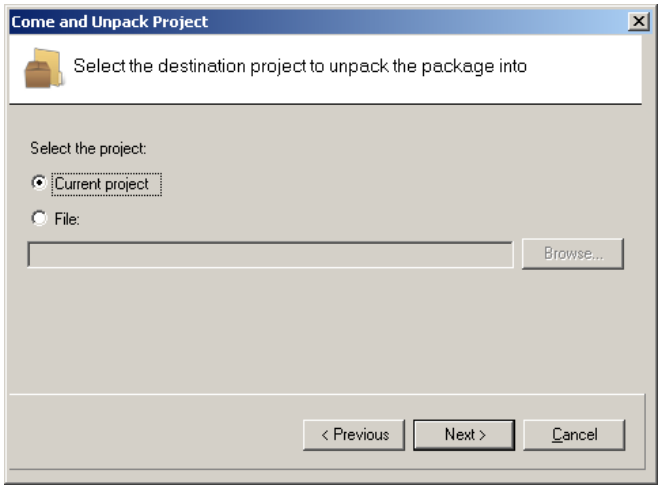


Figure 5.11 - Pack and Go Wizard: Select Project File

The next screen allows you to choose what to do with the different components that are in the project. By default, all of your packed components are checked in the providers list. You can uncheck any providers that you do not need to deploy.

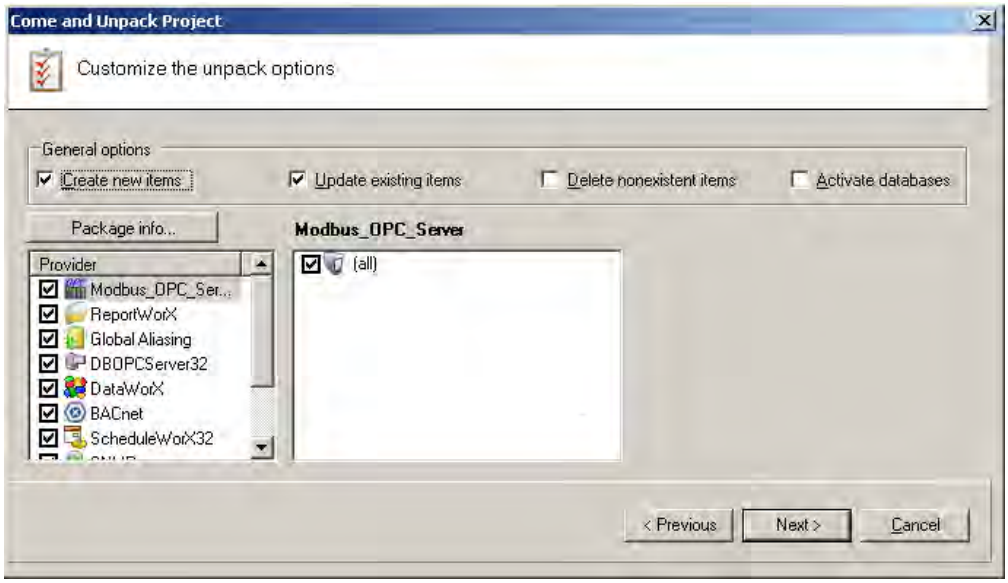


Figure 5.12 - Pack and Go Wizard: Customize Unpack Options

There are four general options at the top:

- **Create new items:** when checked, will create a new item in the project if the item did not exist in the chosen project file.
- **Update existing items:** will update the exiting item in the project with the packed project item with new information.
- **Delete nonexistent items:** will delete the items in the existing project that do not exist in the package.
- **Activate databases:** will make the deployed configuration databases active.

When you have chosen the different components to deploy and set the options to your liking, click on “Next” to reach the last screen of the wizard. On this screen, you can choose to see the log file. We recommend that you leave this option checked so that you can see which files are deployed successfully. When you are ready, click on the “Finish” button to start the unpacking process. The Wizard will show you the progress of the unpacking process.

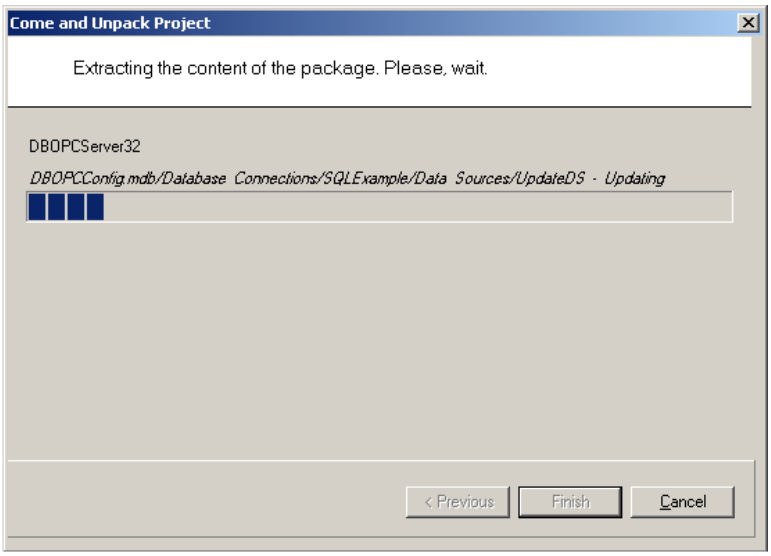


Figure 5.13 - Pack and Go Wizard: Unpacking Project

When the unpacking finishes, the dialog will close and you will see the unpack report in Workbench32, if you have chosen to show it. The report will tell you which package file you used, and how successfully each file unpacked into the project. The status indications of the unpacked file are the same as when you packed them.

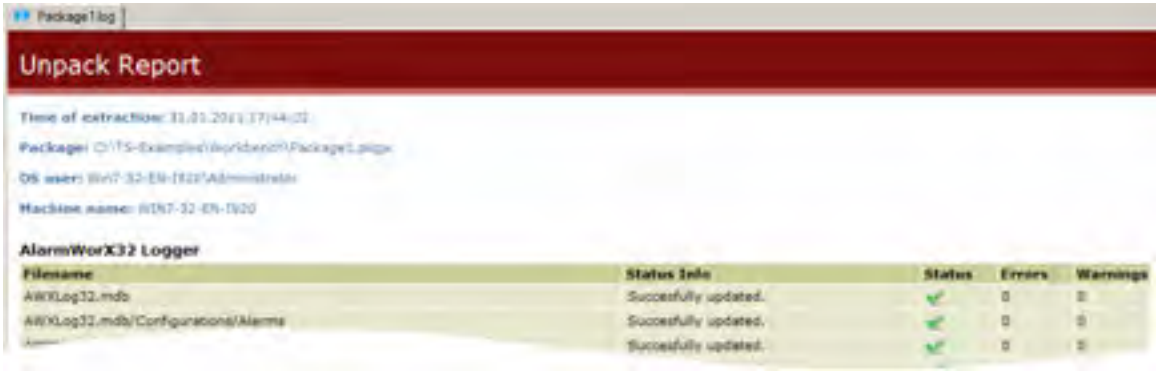


Figure 5.14 - Unpack Report

5.4. Workbench32 Runtime

Another feature of Workbench32 is its runtime mode. Not only does Workbench32 provide configuration forms for the different parts of GENESIS32, it also acts as a runtime container.

While in configuration mode you can click on the "Switch to Runtime mode" icon to put Workbench32 in runtime mode. You can manage many aspects of your application in this mode including viewing information (like alarms) and the schedule that associated with the project, as shown in Figure 5.15. Please note that this is related to the GENESIS32 Navigator. This is not used in the standard training course.

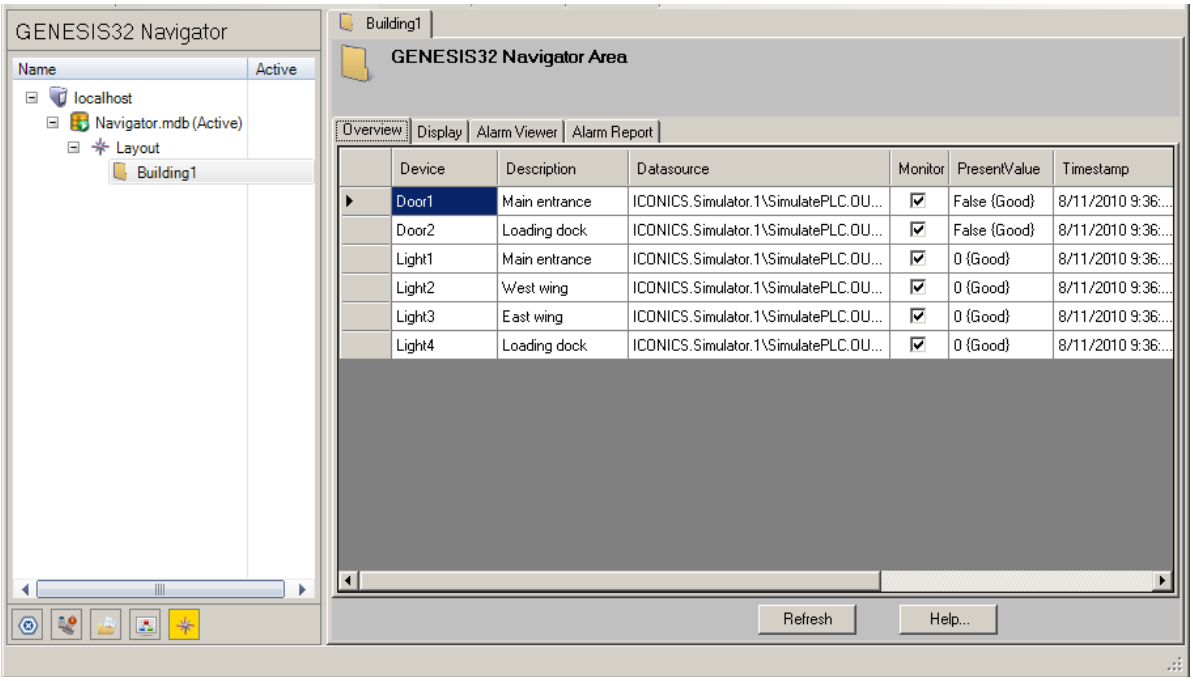


Figure 5.15 - Workbench32 in Runtime Mode

Depending on the runtime providers you add to the project, you can also manage your Multimedia Server, ReportWorX, and your BACnet Servers.

5.5. Remote Access

The final feature we will bring to your attention about Workbench32 is its ability to show configuration and runtime information over the Internet. In today's ever changing world we recognize that you may need to adapt your project wherever you may be when the occasion arises.

To configure or view your application over the Internet you will need the following on the remote client:

- Internet Explorer 6.0 or higher
- .NET Framework 3.5
- Visual C++ 2005 SP1 Redistributable
- Visual C++ 2008 SP1 Redistributable

You should also do the following in Internet Explorer:

- Add your GENESIS32 server machine into the Trusted Sites zone
- Set the security level for the Trusted Sites to Low

As one last step, you should run a command to set the Caspol security level in .NET Framework to FullTrust level.

Fortunately, you do not need to do all of this manually. You can point your Internet Explorer to http://*GENESIS32Server*/Workbench32/default.htm on the client machine, where *GENESIS32Server* in the link should be replaced by the actual server computer name or IP address. When the page loads you should see a validation page similar to Figure 5.16. You will also encounter a few popup messages telling you to install a few different ActiveX controls. You should confirm these messages and install the controls.

Figure 5.16 - First Time Loading Workbench32

Clicking on the "Validation" button will tell you the missing pieces that you need on the client computer. Simply click the links in the Solution column to resolve the issue. When you click on the links to install Visual Studio components, you will get a few different popup messages. Click on "Run" for each of the messages that come up and follow the on screen installation instructions.

Figure 5.17 - Workbench32 on a Client Showing Missing Items

Once you have passed the Validation and your client is set up you can go to http://*GENESIS32Server*/workbench32/workbench32.htm where *GENESIS32Server* in the link is replaced by the actual server computer name or IP address. It may take a few minutes the first time you try to load this page. Once it loads you should see a page similar to Figure 5.18. From here you can interact with Workbench32 as you would when running it on the server computer.

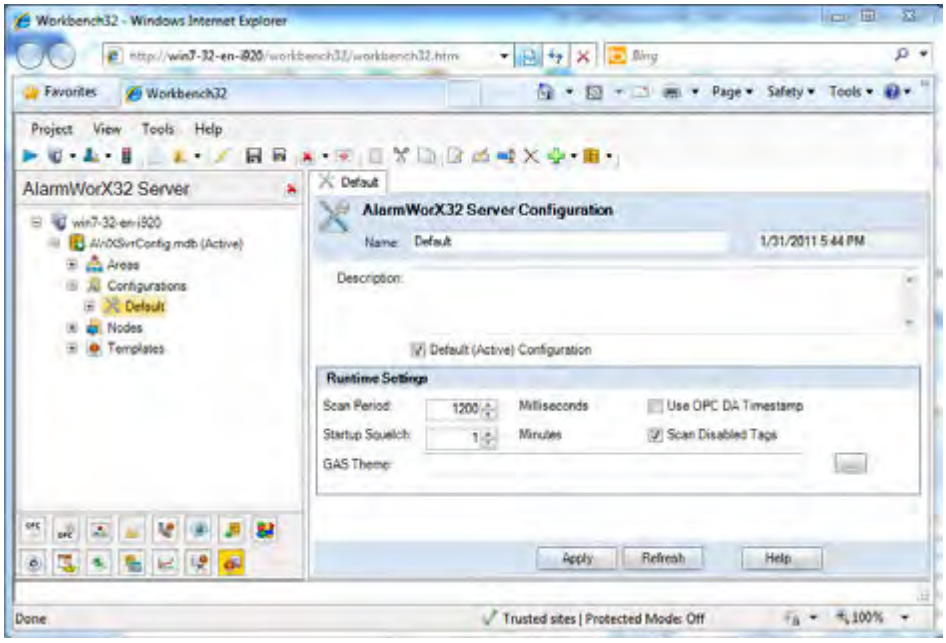


Figure 5.18 - Workbench Running in Remote IE

Lab: Acmeseutical Project *L5*

Since we will be creating alarm notifications, trending, and many other parts of the system for our project, we are going to create a project that contains everything within one file.

Estimated time to complete this lab: 20 minutes

5.1. Task Checklist

In this lab, you will complete the following task:

- Create a project file for Acmeseutical

5.2. Support Files

You do not need any support files to complete this lab.

5.3. Creating a Project with Workbench32

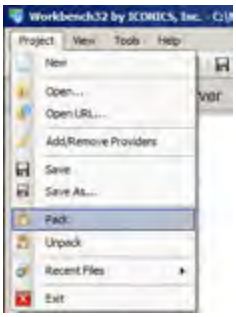
GENESIS32 provides two ways to create a project. You can use ProjectWorX32 or Workbench32. In this lab, we are going to create a project with Workbench32.

1. Launch Workbench32 by going to Start → Programs → ICONICS Workbench32 → ICONICS Workbench32.

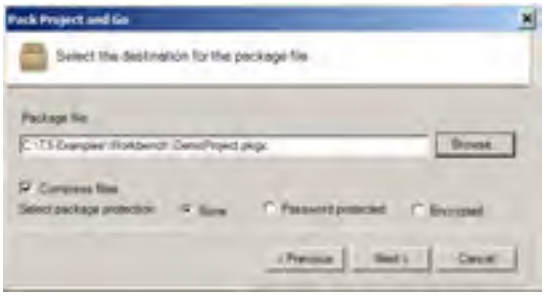
2. We are going to pack the installed project. Go to "Project → Pack."

3. Click "Next" on the welcome page.

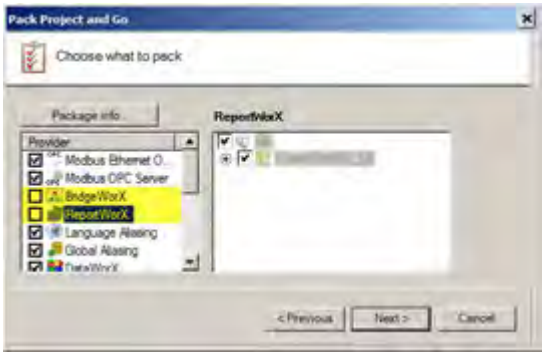
4. Select "Current Project" from the project you want to pack.



5. Browse for a suitable path. Name the file “DemoProject.pkgx”. Leave the compression enabled and disable password protection and encryption.



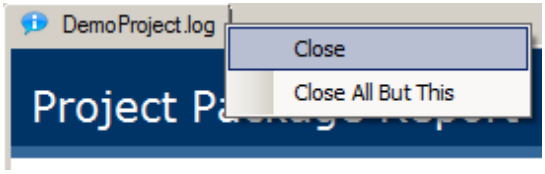
6. Select all providers BUT ReportWorX and BridgeWorX, as these may not be present in the system and are not part of the GENESIS32 training.



7. Leave find and replace unchecked and click on “Next.”

8. Enable “Show report” and click on “Finish.”

9. Have a look at the report and see if there is anything missing. Then, right click on the Log tab and select “Close.”



10. Go to “Project → New.”

11. From the list of providers, select the following:

    a. AlarmWorX32 Server

    b. AlarmWorX32 Logger

    c. TrendWorX32 Logger

    d. Unified Data Manager

    e. DataWorX

    f. Global Aliasing

    g. Language Aliasing

    h. AlarmWorx32 Multimedia (only if you are doing the MMX training).

    Leave the rest unchecked.

12. Click on "OK."

13. Go to "Project → "Save as…"

14. Change directory to C:\Training and save the file as "AcmeseuticalProject.prjx".

15. We now have a project file. From here onward, we will add different application servers to the project.

16. Feel free to take a look around the project environment and the different providers available in the project.

Review: Introduction to Workbench32

R5

5.1. Chapter Objectives

In this chapter, we discussed the Workbench32. You can create and change configurations related to your project using the Workbench as a central tool. You should know and understand the following topics:

- Workbench32
- The Layout of Workbench32
- Creating and Opening Projects in Workbench32
- Packing and Unpacking Projects

5.2. Summary

Worbench32 is a powerful application that provides configuration and runtime operations. In its configuration mode, you can create new projects, add providers, and set up configuration for the different aspects of your application. Workbench32 is laid out in an intuitive way where normal menu items are at the top of the application, a Tree Explorer to the left to select the provider to work with, and the contents area to the right where it loads individual configuration forms based on the item you select in the Tree Explorer.

Once you have configured your project on your development machine, you can easily deploy it to a different machine using the Pack and Go feature. The different configurations within your project pack up in a single "*.prjx" file that you can take with you. On the deployment machine, you will only need this file to transfer all your configuration work from the development machine.

5.3. Terms

Configuration Mode	Contents Area	Pack and Go Wizard
Provider	Runtime Mode	Tree Explorer
Workbench32		

5.4. Test Your Knowledge

In this chapter, we exposed you to Workbench32. While we did not cover all aspects of this powerful application, you should be able to answer the following questions:

1. Which of the following is NOT a provider in Workbench32?

    a. DataWorX32
    b. GraphWorX32
    c. ScheduleWorX32
    d. AlarmWorX32

2. Which are the major sections to Workbench32?

    a. Ribbons, Tree Explorer, and Contents Area
    b. Menus and toolbars, Navigation Tree, and Contents Area
    c. Menus and toolbars, Tree Explorer, and Contents Area
    d. Menus and toolbars, Navigation Tree, and Configuration Forms

3. True or False: Workbench32 allows you to view graphics in runtime mode.

4. True or False: When you unpack a project, you can choose to create new items.

5. True or False: When packing a project you do not need to pack the currently active one.

AlarmWorX32 Server 6

AlarmWorX32 is a distributed enterprise-wide alarm and events management system. Available in the standard GENESIS32 suite of applications, or as a stand-alone Open Series component, AlarmWorX32 offers the tools you need to deliver real-time alarm information throughout your system. AlarmWorX32 is a family of modular alarming products, including the Alarm Container, Alarm Server, Alarm Logger, Alarm Viewer ActiveX, Alarm Report ActiveX, and Multimedia Server.

In the next few chapters, we will discuss the major parts of AlarmWorX32. Before we can discuss any other components in AlarmWorX32, we must first understand and configure an alarm server. This will be the source from which we obtain alarm and event information. By the end of this chapter, you will be familiar with the following:

- What the AlarmWorX32 Server application does
- The Alarm Server Configurator and configuration database
- How to use and configure Alarm areas
- The different types of Alarm tags and how to configure them

6.1. Alarm Server Configurator

Alarm Configuration is easy with the **AlarmWorX32 Server Configurator**, shown in Figure 6.1. You can define your own alarms using any OPC data and/or expression (complex combinations of OPC data), choosing from analog Limit Alarms, Deviation, Rate-of-Change, and Digital Alarms just to name a few. In keeping with the open standards employed throughout the GENESIS32 architecture, your alarm configurations are saved to a Microsoft Access or Microsoft SQL Server database.

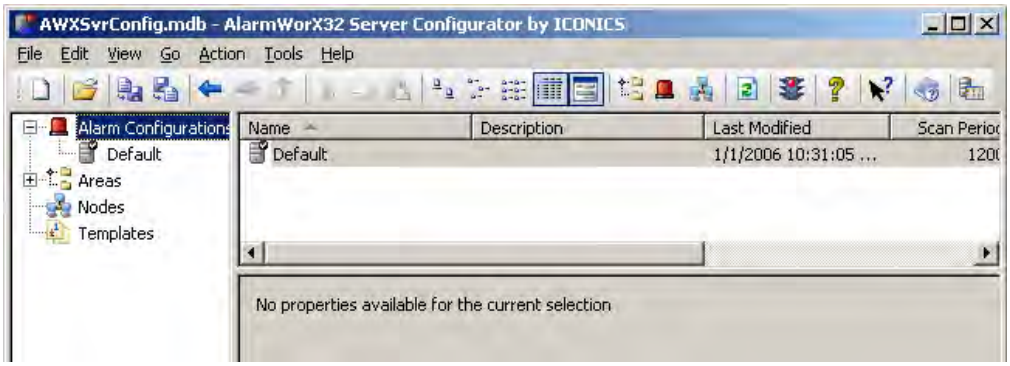


Figure 6.1 - Alarm Server Configurator

NOTE: The lab following this section outlines the steps necessary to create Alarm Server configuration databases. For more information about creating configuration databases, also refer to the "Creating Configuration Databases" section in the AlarmWorX32 Server Help documentation.

Once the alarms are configured, the **OPC Alarm Server** takes care of monitoring live data and to post new alarms throughout your network. This component runs in the background (even as a Windows NT Service if desired), and is tuned to give your system the performance required of an alarming system.

The **AlarmWorX32 Server** receives field data from any OPC-compliant Data Access (DA) server, and performs alarm detection. It then sends alarms to any OPC Alarm and Event client that is subscribed to it. The **AlarmWorX32 Viewer** and the **AlarmWorX32 Logger** are two examples of clients that can receive alarms and events from the server, and will be discussed in later chapters.

6.2. Alarm Server Configuration

The Alarm Server can have multiple configurations; however, only one configuration can be designated as the default or active configuration at any point in time. Typically, multiple configurations are only used when a single configuration database is shared by multiple Alarm Servers. Each individual Alarm Server will then use exactly one of the defined configurations.

There are a few configurable settings that provide some control for the Alarm Server behavior when it goes into runtime mode. The runtime settings are:

- **Scan Period:** Effectively the maximum speed at which an OPC DA server will send the Alarm Server the most current data values it has received from its source, typically a PLC, a controller, or other device, but it may also be another software program. It has to be noted that only updated values will be sent for those tags that have changes, as OPC is fully change-driven; therefore, it may be that at the given interval, 1000 milliseconds by default, no updates will be sent at all to the Alarm Server.
- **Use OPC DA Timestamp:** When enabled, this will ensure that the Alarm Server will use the timestamps provided by the OPC Data Access server, as opposed to having the Alarm Server generate its own timestamps.
- **Startup squelch:** This defines a period at the Alarm Server startup, one minute by default, during which no alarms will be sent to alarm clients. This is useful during system restart where the OPC Data Access server has not had the chance to startup completely and supply good quality data values to the Alarm Server.

- **Scan Disable Tags:** Any alarm tags you create can be dynamically enabled and disabled through an OPC tag or an expression. This option allows the alarm server to scan the source tag of a disabled alarm. If you have many alarm tags that often changes between the enable and disable states you should check this option. Otherwise, leaving the option unchecked would have less overhead on the OPC server.
- **GAS Theme:** This is the Global Alias Theme that is to be used by this configuration. This may be applicable when a single configuration can be used on multiple systems or in different setups, in which case, the GAS Theme can control exactly which tags are being used to generate the alarms.

NOTE: The lab following this section outlines the steps necessary to create and configure Alarm Configurations. For more information about creating Alarm Configurations, you can also refer to the "Creating a New Alarm Configuration" section in the AlarmWorX32 Server Help documentation.

6.3. Alarm Tags

An **alarm tag** represents a single object for OPC alarm information. An alarm tag can contain information for multiple types of OPC alarms, including deviation, limit, rate of change, digital, trigger level, and rate level alarms.

The **General** tab of an alarm tag definition provides the following fields:

- **Enabled:** An alarm can be enabled or disabled by setting the value of this field to 1 or 0 respectively. However, it is also possible to use an OPC Data Access Tag instead to control the Enabled state of the alarm.
- **Default Display:** An alarm can have a default (display) file associated with it. When a file is specified, a custom menu will become enabled in the Alarm Viewer ActiveX, which will launch the specified file in its default viewer. This can be especially useful to associate additional information with an alarm in the form of an external file, such as PDF or HTML.
- **Description:** In this field, a more elaborate description can be associated with the alarm.
- **Delay:** Enter a number, in seconds, into this field to have the alarm server only alarm when the value from the source remains the same for this amount of time.
- **Base Text:** The base text of the alarm, together with the actual alarm state message will create the final alarm message sent to the alarm clients.
- **Help Instruction:** Enter any help instructions into this field. The contents of this field are available to the alarm clients via the custom attribute HELP.

- **Alarm Tree Enum Type:** When you are using the AlarmWorX32 Explorer, you can configure the alarm viewer in the explorer to show the state of the alarm in textual string as opposed to the numeric value. This is comparable to using the Statefields feature in a PPT.
- **Enable Clear:** When checked, the alarm will contain an extra state. Normally, when an OPC tag value returns within the normal limit and an operator has acknowledged the alarm, the alarm is considered complete; therefore dropping out of the state server. With this option checked, the alarm will remain visible in the viewer instead of dropping out of the server; it will then enter the Clear State, at which point, if someone acknowledges the alarm again, it will drop out.
- **Delay On Alarm Only:** Normally, when you enter a Delay for the alarm, it would delay both alarm messages and return to normal messages. When this option is checked, it will only delay alarm messages.
- **Exclude Equal To:** When comparing present values to generate an alarm, the normal comparison would include the value for the alarm state. With this option checked, the system would exclude the configured value for the state.
- **Alarm-Tree Writes Enabled:** When checked, this option will allow users to change the current tag values for the Alarm Input tag in the States Tab of the AlarmWorX32 Explorer.

NOTE: Alarm Tree Enum Type and Alarm-Tree Writes Enabled are outside the scope of this class.

The ICONICS alarm server provides six different types of alarms. An alarm tag can have more than one alarm type configured. For example, it is possible to have both a limit alarm and a rate of change alarm in one alarm tag. The following alarm types are available and configurable on the **Alarms** tab of an alarm tag definition:

- **Limit alarms** generate based on an OPC input exceeding one of the four different limits (Low-Low, Low, High, and High-High).
- **Deviation alarms** generate based on the difference between two OPC inputs exceeding one of the four different limits (Low-Low, Low, High, and High-High).
- **Rate of Change alarms** triggers when an OPC input changes by more than the configured limit within a given period.
- **Digital alarms** fire when an OPC input matches exactly the configured alarm state.
- **Trigger Limit alarms (TLA)** go off when the alarm tag changes. It compares the current value with the last known value. If the values are different, the system triggers an alarm.

- **Rate Limit alarms (RLM)** are a combination of Limit and Rate of Change alarms. You can define four different rates of changes (Low-Low, Low, High, and High-High) and when the value exceeds the preset limit, the system will generate an alarm.

Limit, Deviation, and Rate Limit alarms can contain a **deadband** value in which repeating borderline alarms can be avoided. For all alarms, it is possible to use OPC tags or expressions to control the **severity** of the alarm and whether or not an alarm **requires acknowledgement.**

In this training course, the **Related Values** tabs are not used. However, these tabs make it possible to specify up to twenty additional OPC Tags, whose values will be sent together with the normal alarm information to the OPC AE Client. This OPC AE client can then use these Related Values as Custom Attributes. An example of their use would be batch or lot tracking. By using an OPC Tag with the Batch number as a related value, you could track which alarm and how many occurred during the production of a particular batch.

NOTE: For more information about configuring alarm tags, also refer to the "Creating a New Tag" section in the AlarmWorX32 Server Help documentation.

6.3.1. Limit Alarm

Now that we have a general understanding of what we need to configure an alarm, let us take a closer look at how to configure one of the most often used alarm types. In this section, we are going to digest a limit alarm in the default demonstration configuration.

One of the alarms configured in the default out-of-the-box alarm server is the Humidity alarm. It monitors the simulation Ramp tag to generate different alarms based on the value of the tag. You can see from Figure 6.2 that this alarm is always enabled. Associated with the alarm are also a GraphWorX32 display and help instructions.

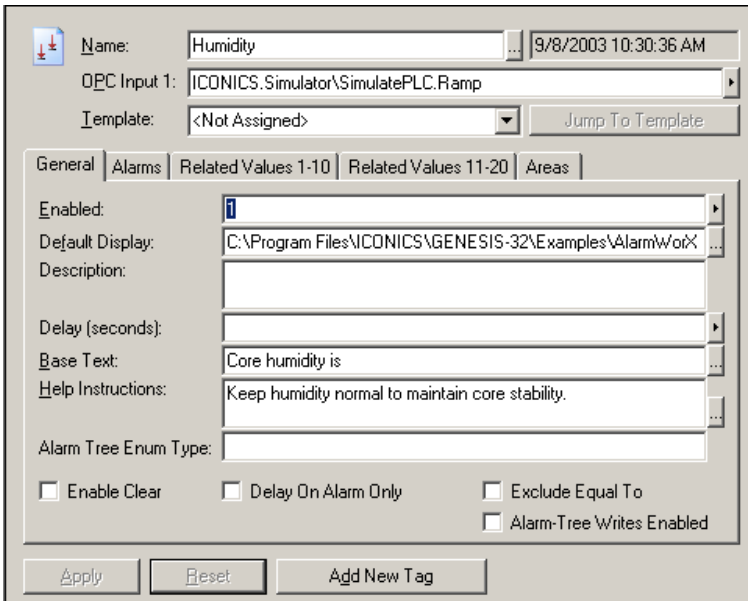


Figure 6.2 - Humidity Alarm Configuration: General Tab

NOTE: A static value of one in the Enabled field is the default value for the field when you create an alarm which means that the alarm will always be enabled. If you change the value to a zero, then the alarm will be disabled.

On the Limit under Alarms you will find the different set points that will trigger an alarm. The message text may seem incomplete at a glance, however; the resulting message for the alarm appends this text to the Base Text field on the General tab. For example, when a low alarm occurs the message display for the alarm would be “Core humidity is low.”

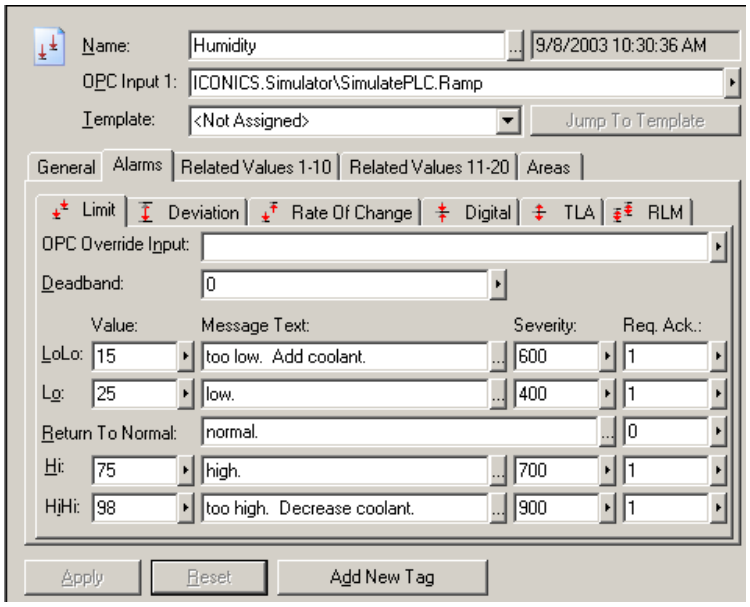


Figure 6.3 – Humidity Alarm Configuration: Alarms Limit Tab

In the limit alarm configuration, you also see a column for severity. For each alarm that triggers, you can associate a “severity,” sometimes referred to as “priority.” This allows you to indicate the order in which the operator should deal with each alarm. A higher severity alarm should take priority over an alarm with a lower severity.

> ***Tip!*** OPC defines the severity range between 0 and 1000. You can set the severity in the alarm server. You will see in later chapters that you can give different appearance to alarms with different severity values to attract operator attention.

A one in the “Req. Ack” field indicates that the alarm requires acknowledgement. After an alarm triggers, it can go into a few different states. An alarm can be in the Alarm and Unacknowledged, Alarm and Acknowledged, Normal and Unacknowledged, and a few other states that we will not mention here. When an alarm is triggered, if the “Req. Ack” is set to true (a value of one), it will go into the Alarm in an Unacknowledged state. On the other hand if the field is set to false (a value of 0) when the alarm triggers the alarm will go into the Alarm in an Acknowledged state. This is important to know so that you will understand how it affects alarm clients in later chapters.

6.4. Alarm Areas

You can use an **area** to group OPC alarm tags. Clients subscribed to the alarm server can use this information for filtering purposes. An alarm tag can exist under multiple areas, and an area can have multiple levels. An area is defined by its name and the name of its parent branches. For example, a branch called Area 1 under Plant Area A is different from Area 1 under Plant Area B. You can assign an alarm to more than one area. For example, a tank limit alarm may be part of both the "Tank 1" area as well as the "Water" area.

You can associate an alarm to an area using one of two ways. When you are configuring an alarm tag you can go to the Areas tab and click on the “Add” button toward the bottom of the configuration. This will bring up the “Select or Enter Area name” dialog. Double-click on any area to select it or type in the area name in textbox at the top of the dialog. When you are done, click on “OK” to add the area and close the dialog. In the Areas tab you should see all of the areas that you assign for the alarm. Selecting an area in the list will enable the “Remove” and “Jump To Area” buttons, which allows you to disassociate the area from the alarm and go to the area’s configuration respectively.

You can also add alarms to an area by selecting the desired area under the Areas node. In the right-click context menu of the area, select the “Insert New Link(s) To Tags…” option. The “Alarm Tag Selector” dialog will appear. You can select the configuration where the alarms reside in the dropdown list at the top of the dialog. The alarms associated with the configuration will display in the text box below. You can select multiple alarms by holding down the Shift or Ctrl key. If you want to select all of the alarms in the configuration, you can also click on the “Select All” button at the bottom of the screen. Once you click on “OK” you should see the list of alarms above the area configuration. You can click on any alarm in the list to view a summary configuration for the alarm. Clicking on the “Jump To Tag” will bring you to the alarm directly for editing.

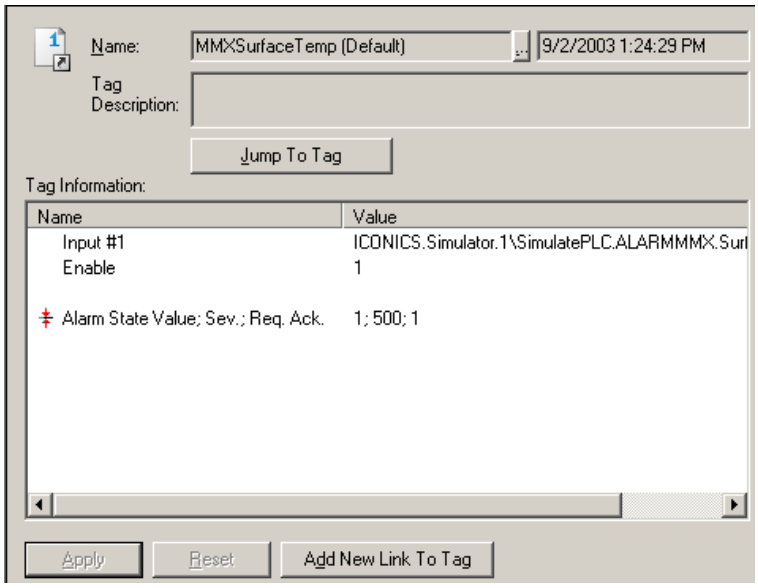


Figure 6.4 - Alarm Tag Summary in Area

6.5. Configuring Many Similar Alarms

Often we see similar configuration for different production lines in a manufacturing facility yet each of these lines require their own alarm generation. Since the lines are similar the alarms needed are similar as well. There are a few different tools available in AlarmWorX32 Server Configurator to help you duplicate similar alarms.

First, if you have already created one of the alarms that you need you can right-click on the alarm and choose the “Multiply” option. The **Multiply Item** dialog will come up as shown in Figure 6.5.

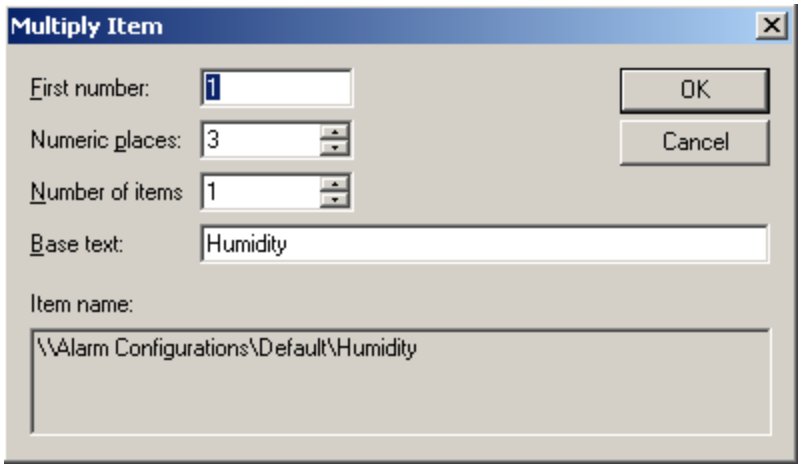


Figure 6.5 - Multiply Item

In the "Base text" field, you will see the name of the alarm. You can edit this so that the text will match the name of the alarms you are going to create. In the "Number of items" field, specify how many alarms you would like to create based on the selected one. In the "Numeric places" field, specify the number of digits you would like in the numbering system when it creates the alarms. For each alarm that you create with this dialog the name of the alarm will be the name in the "Base text" field with a number. The syntax would be something like *BaseText*XXX, where each X represents a digit configured for in the Numeric places field. In the configuration shown in Figure 6.5, when you click on OK, an alarm with the name "Humidity001" will be created. There is no limit to how many items you can create using this dialog.

You have probably noticed by now that there is a Templates node in the navigation tree and each alarm has a Template field. By using this field, you can create a "template" for alarm configurations that you use often. You can create alarms based on a template or associate of an already created alarm with the template to use it.

The template configuration form shown in Figure 6.6 is the same as the form to configure an alarm. Accept that in this form you cannot associate a template and there is an extra tab called "Assigned Sources" that will indicate which alarm uses this template.

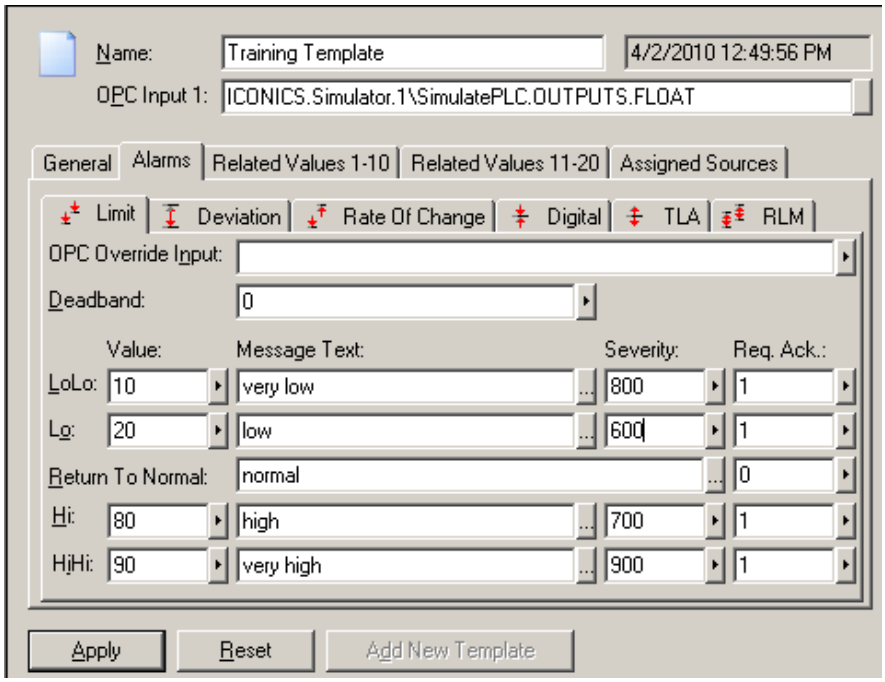


Figure 6.6 - Template Configuration

Once you have created an alarm template, you can create alarm tags based on the template. To do that, simply right-click on the configuration and select New → Tag from Template. The Alarm Tag Templates Selector will appear.

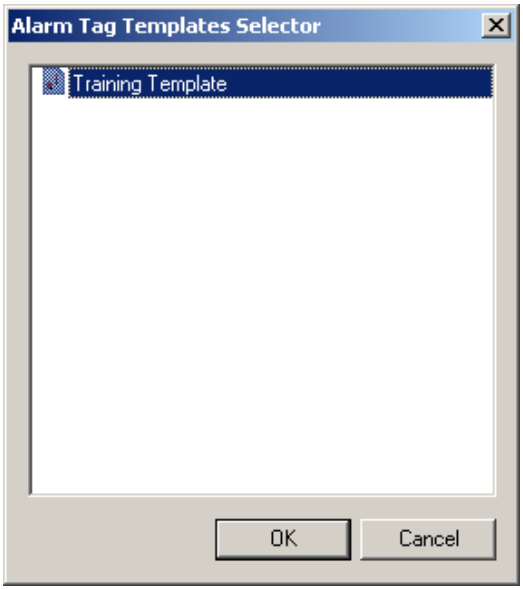


Figure 6.7 - Alarm Tag Templates Selector

All of the configured alarm templates will be listed in this dialog. Choose the one you wish to use and click on "OK" to close the dialog and create an alarm. You can give the alarm a name and the OPC tag to monitor for alarm generation. In the Template field you should see the template you selected to create the alarm.

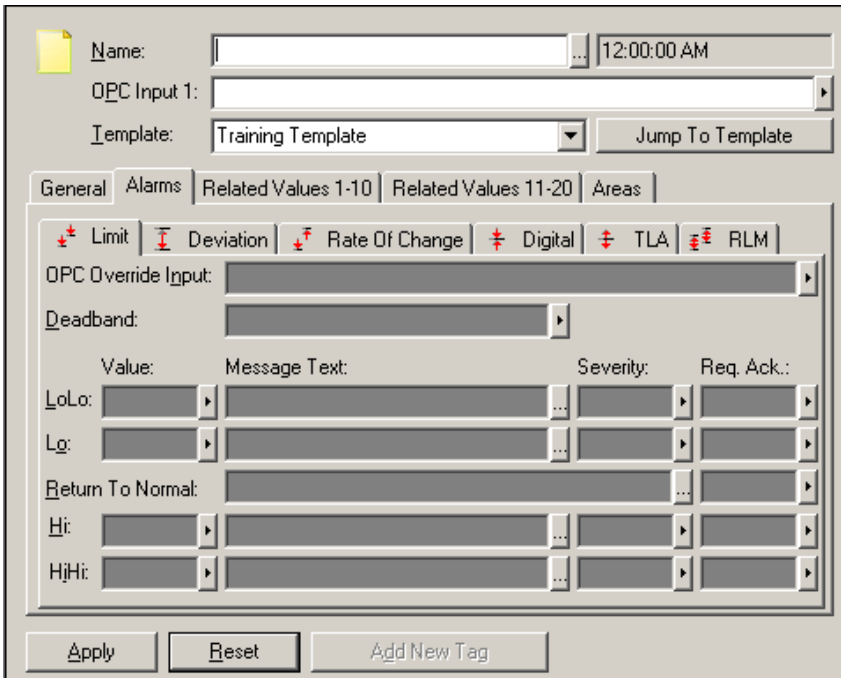


Figure 6.8 - New Alarm from a Template

As you may have noticed many of the configurable fields are grayed out. This is a visual indication that you do not need to configure these fields because it will use the information configured in the template. If you do need to overwrite information in the template you can click in any field and type in new values. The field will use the normal white background and when the alarm triggers it will use the specified information as opposed to the information configured in the template.

Although the AlarmWorX32 Server Configurator only allows you to create one alarm at a time using the template, you can use Workbench32 to create multiple alarms at the same time with a template. Using the AlarmWorX32 Server provider in Workbench32, you can create alarms as if you would in the Configurator. When you right-click on the configuration to insert an alarm tag you should notice that there is an option for “Tag Import Wizard.”

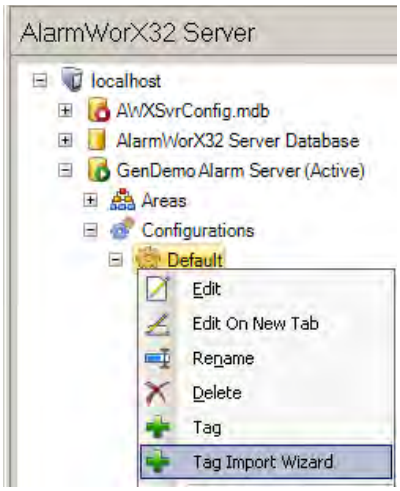


Figure 6.9 - Tag Import Wizard Option

Once you select this option you will see the **AlarmWorX32 Server Import Tags** configuration open in the contents area. In here you can select the template to use, an area for the alarms to create, and a Base Item Name. When you are done configuring these options you can click on the "Add Tags" button to bring up the Unified Data Browser. You can select multiple tags before you close the browser to add many OPC tags at once. When you have added all of the desired OPC tags, click on the "Import" button to begin the alarm tag creation. For each OPC tag you have added the system will create an alarm using that OPC tag as the OPC Input1 for the alarm. In the Item Name column you will also see the name of the alarm that the system will use. You can change the name of the alarm as desired before importing.

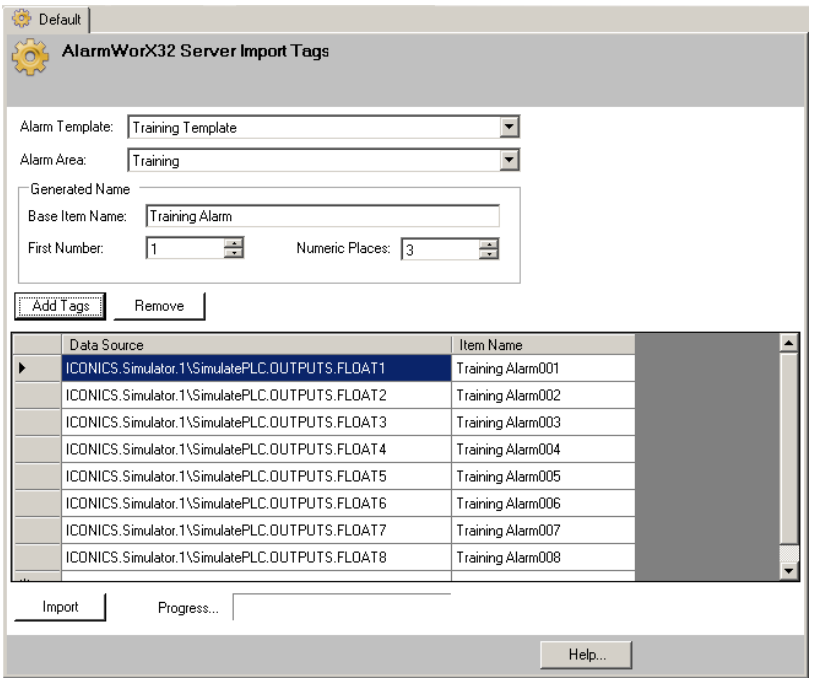


Figure 6.10 - AlarmWorX32 Server Import Tags

When the system finishes adding the alarm tags you will receive a **Results** dialog that shows the results of importing the tags. You can click on “OK” to close the dialog. If you then browse to your configuration in the Tree Explorer, you will see all of the alarms.

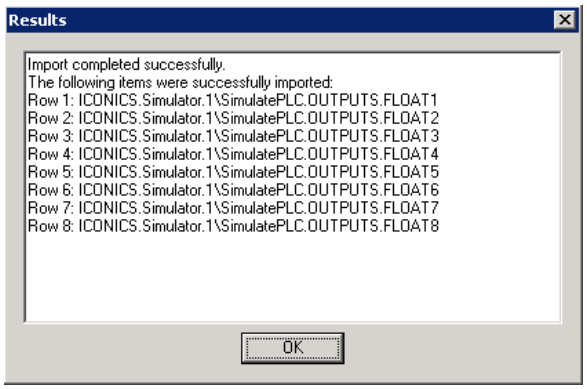


Figure 6.11 - Tag Import Wizard Results

In WorkBench32, you can right-click on a newly created alarm and click on “Edit On New Tab” to open the alarm tag configuration in the content area. Notice that there are some pink and white fields in the configuration. The pink fields represent values inherited from the template and the white fields are those with values that will overwrite those in the template.

> **NOTE:** For more information regarding AlarmWorX32 Server configuration in Workbench32, please consult the Help files.

Lab: Alarm Notifications L6

In the previous labs, we helped Acmeseutical create displays to control their pill making process. Their operators can now control the process with a click of a button. While they do like the displays, they would also like to have some visual notification on the display that alerts them when the mixers are nearly empty. To minimize downtime, the operators can fill the mixers before it is too late.

We will help Acmeseutical achieve this goal in the next few labs. First, we are going to create an alarm server that can generate the notifications that they need. Then we are going to provide the onscreen alert for their operators.

Estimated time to complete this lab: 30 minutes

6.1. Task Checklist

In this lab, you will complete the following tasks:

- Create an Alarm Server Configuration
- Configure Limit alarms
- Configure Digital alarms

6.2. Support Files

You will need the following file(s) to complete this lab:

- Acmeseutical_DWXConfig.mdb

6.3. Alarm Server Configuration Database

To give Acmeseutical the alarm notification system they need, we will first need to configure the different alarms that make sense in their situation. The alarm will reside in a configuration database that we will create now.

1. Open the AlarmWorX32 Server Configurator by going to Start → Programs → ICONICS GENESIS32 → AlarmWorX32 → Alarm Server Configurator.

2. The Alarm Server Configurator opens a default configuration database with demo data. If you would like to familiarize yourself with a complete configuration, feel free to browse this default configuration.

3. We will create our own configuration database. Click on File → New or click on the "New" button to bring up the AlarmWorX32 Server Configurator Database Wizard.

4. The Configurator Database Wizard will launch, as depicted in Figure L6.1. Click on "Next >" to proceed to the next screen.

Figure L6.1 - Configuration Database Wizard

5. Choose the "Create new configuration in MS Access database" option and click on "Next >."

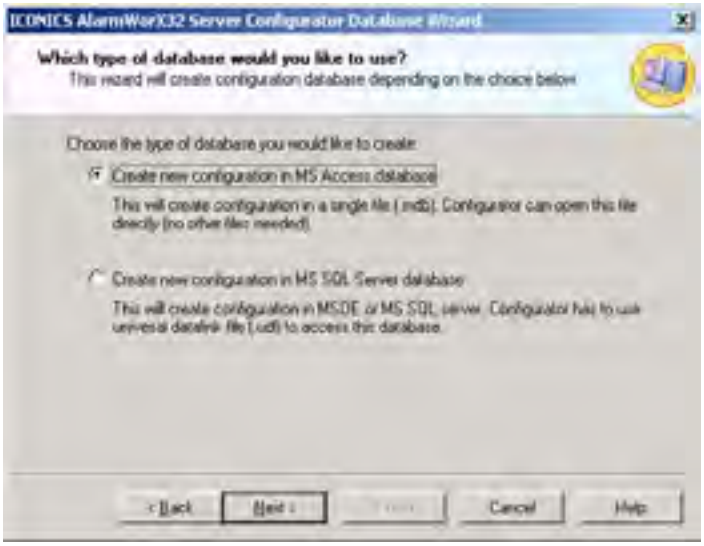


Figure L6.2 - Choosing the Configuration Database Type

6. Change the database location to C:\Training and name the configuration database "AcmeseuticalAlarmServer". If the file name already exists, you may replace it.

7. Make sure that "Make Database Active" option is checked.

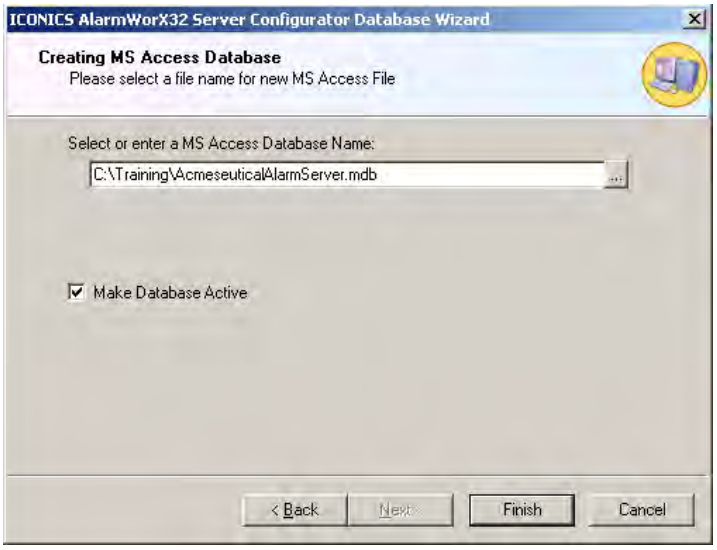


Figure L6.3 - AlarmWorX32 Server Configuration Database Wizard

8. Click on "Finish" to complete the database creation process. Confirm that the AmeseuticalAlarmServer.mdb is displayed in the header. If the File →Make Active menu selection is grayed out your new database is active.

9. The new configuration database will automatically open with a blank configuration as depicted in Figure L6.4.

Figure L6.4 - Alarm Server Configurator

6.3.1. Creating an Alarm Template

Now that we have a configuration database, we can begin setting up their alarms. Since most of Acmeseutical alarms will be very similar, we will create a template for it.

1. Right-click on Templates in the tree control on the left-hand side and select New → Template.

2. In the configuration form that opens on the right-hand side, name the template “Mixer Alarms”.

3. In the General tab, set the “Base Text” to “The level in Mixer X is”.

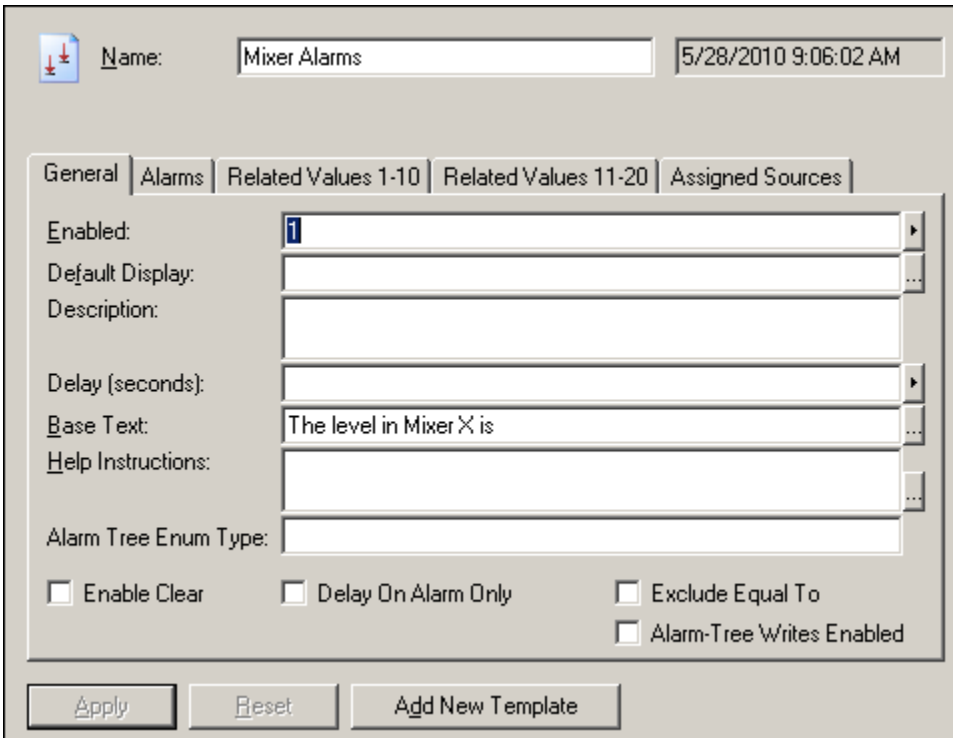


Figure L6.5 - Alarm Configuration: General Tab

4. Go to the Alarms tab and select the Limit tab.

5. Set the values, message text, and severity for the different sub conditions according to Table L6.1.

Table L6.1 - Alarm Template Sub Conditions' Configuration

Sub Condition	Value	Message Text	Severity
LoLo	10	Near empty.	800
Lo	20	Getting low.	500
Normal	N/A	Normal.	N/A
Hi	80	Getting high.	500
HiHi	90	Near overflow.	900

6. Click on “Apply” when you are done. Your Limit tab should look similar to the one show in Figure L6.6.

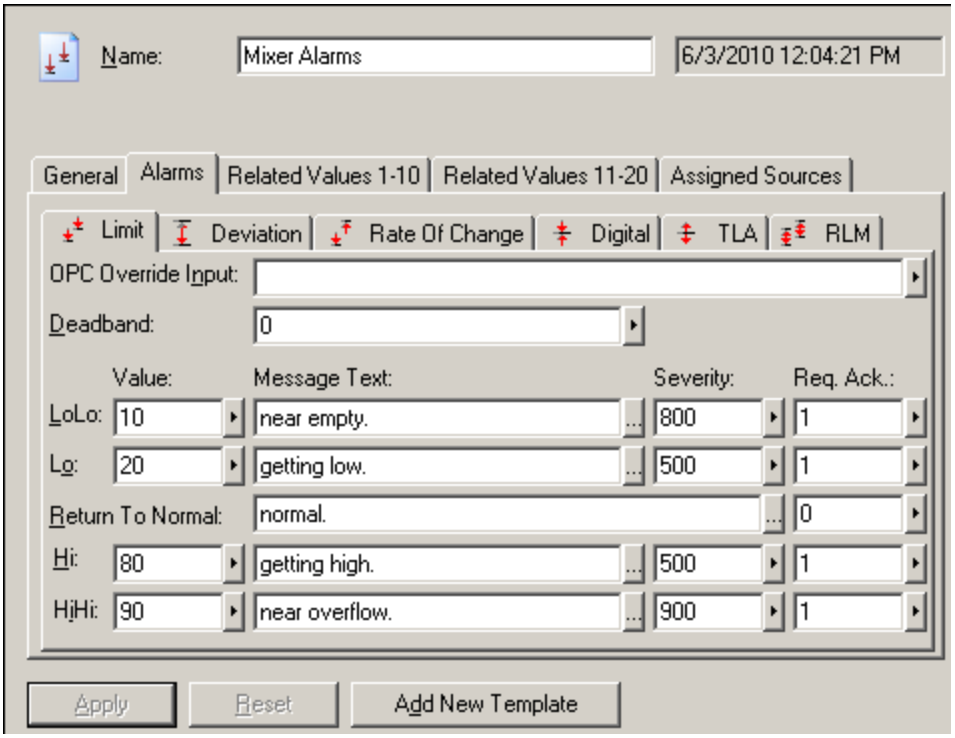


Figure L6.6 - Limit Alarm Configuration

6.3.2. Configuring Areas

For organizational purposes, we are going to assign our alarms to areas. We will create those areas now.

1. Right-click on Areas in the tree control on the left-hand side and select New → Area.

2. In the configuration form that opens on the right-hand side, name the area "Mixers" and click on "Apply."

Figure L6.7 - Area Configuration

6.3.3. Configuring Alarms

We are now ready to create the alarms.

1. Right-click on Alarm Configurations in the tree control on the left-hand side and select New → Configuration.

2. In the configuration form that opens on the right-hand side, name the configuration "Acmeseutical Alarms".

3. Check the "Default (Active) Configuration" checkbox and click on "Apply."

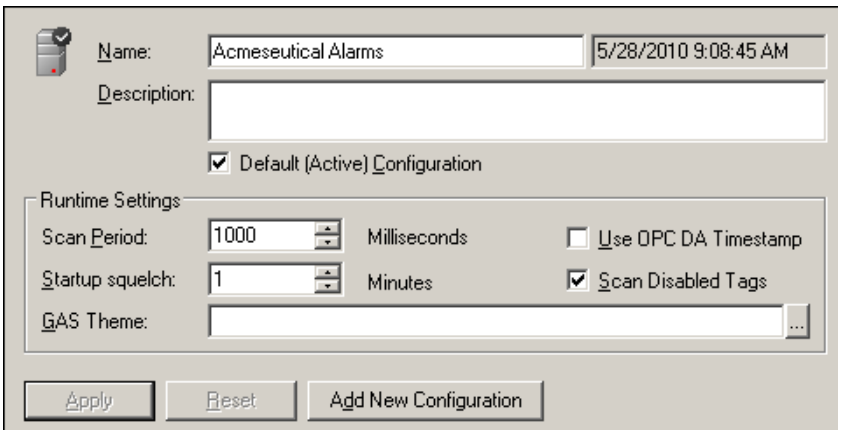


Figure L6.8 - AlarmWorX32 Server Configuration

4. Right-click on "Acmeseutical Alarms" in the tree control and select New → Tag from Template.

5. Click on "OK" in the Alarm Tag Templates Selector that comes up. It should have already highlighted the Mixer Alarms template you created earlier.

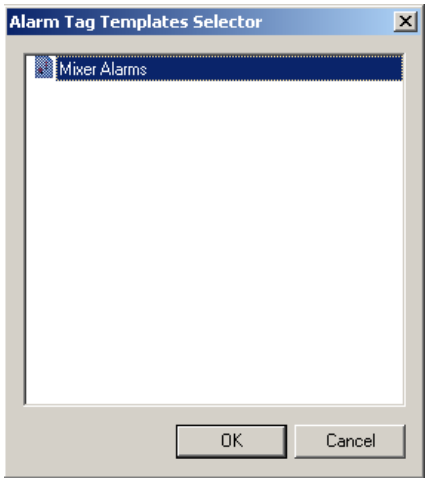


Figure L6.9 - Alarm Tag Templates Selector Dialog

6. Notice that in the configuration form that opens, the Template is already set to "Mixer Alarms" and there are grayed out fields. The grayed-out field values will be retrieved from the template.

7. Name the alarm "Foxborough Machine1 Mixer A Level".

8. Click on the arrow key to the right of the OPC Input 1 text field and select "Unified Data Browser" (UDB).

9. Select the OPC DA tab and browse to My Computer → ICONICS.DataWorX32.8 → Registers → USA → Foxborough → Machine1 → Mixers → MixerA and double-click on the Level tag. You should see the tag added in the list at the top of the UDB.

10. Click on "OK" to select the tag and close the UDB.

11. On the General tab, click in the Enabled field and type in 1. Notice that the field now appears normally with a white background. This will overwrite the value in the template.

    NOTE: Because the field is grayed out, you will not be able to see the cursor after you click in the field. However, you can type the value in.

12. Put your cursor in the Base Text field and Type in "The Level in Mixer A is".

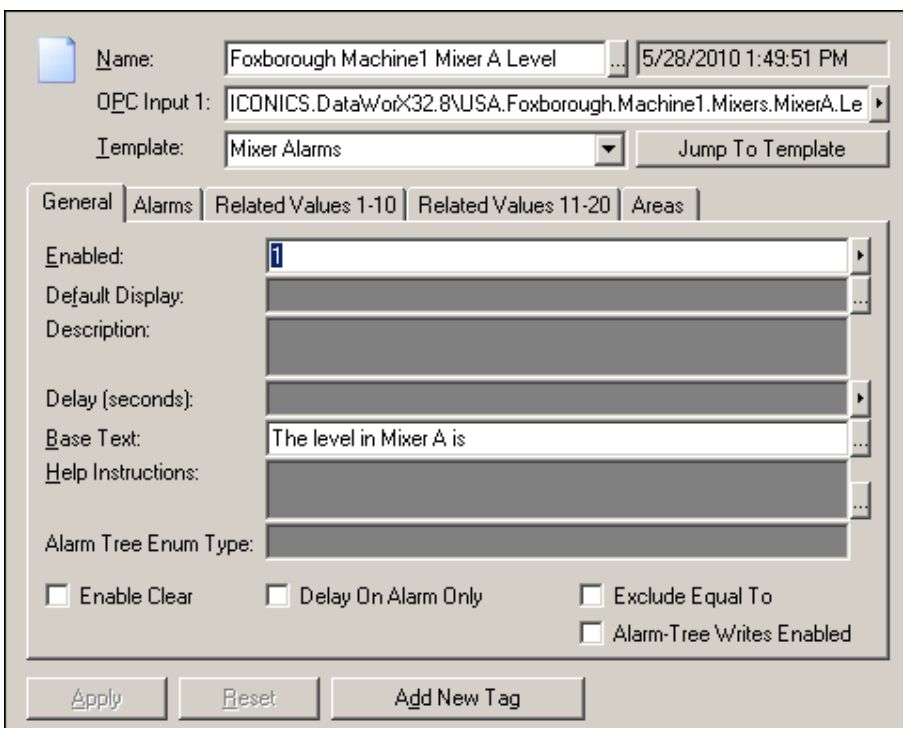


Figure L6.10 - Mixer A Level Alarm Configuration

13. Click on “Apply” to save the changes.

14. Select the Areas tab and click on the “Add” button.

15. In the “Select or Enter Area Name” dialog, double-click on Mixers and click on “OK” to select it and close the dialog.

16. Select the alarm in the dialog view above the configuration form.

17. Right-click on the alarm and select “Multiply…”

18. In the Multiply Item dialog, change the Numeric Places to 1 and Number of items to 3.

19. Leave the Base text as “Foxborough Machine1 Mixer A Level” and click on “OK.”

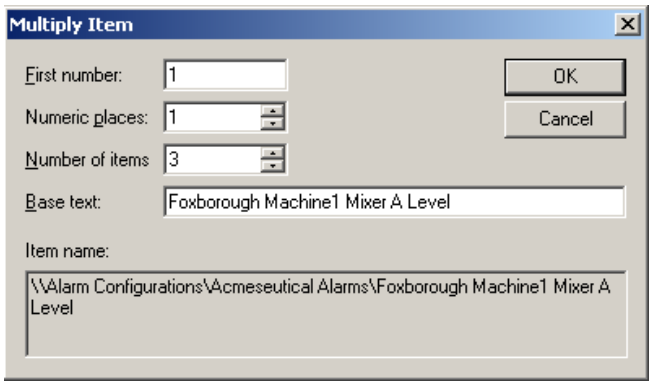


Figure L6.11 - Multiply Item

20. Go through each of the multiplied alarms, changing the alarm name, base text, and OPC Input according to Table L6.2.

Table L6.2 - New Values for Multiplied Alarms

Original Alarm	New Alarm Name	OPC Input	Base Text
Foxborough Machine1 Mixer A Level1	Foxborough Machine1 Mixer B Level	ICONICS.DataWorX32.8\USA.Foxborough.Machine1.Mixers.MixerB.Level	The level in Mixer B is
Foxborough Machine1 Mixer A Level2	Foxborough Machine1 Mixer C Level	ICONICS.DataWorX32.8\USA.Foxborough.Machine1.Mixers.MixerC.Level	The level in Mixer C is
Foxborough Machine1 Mixer A Level3	Foxborough Machine1 Mixer D Level	ICONICS.DataWorX32.8\USA.Foxborough.Machine1.Mixers.MixerD.Level	The level in Mixer D is

21. When you are done, you should see the four alarms in the dialog view.

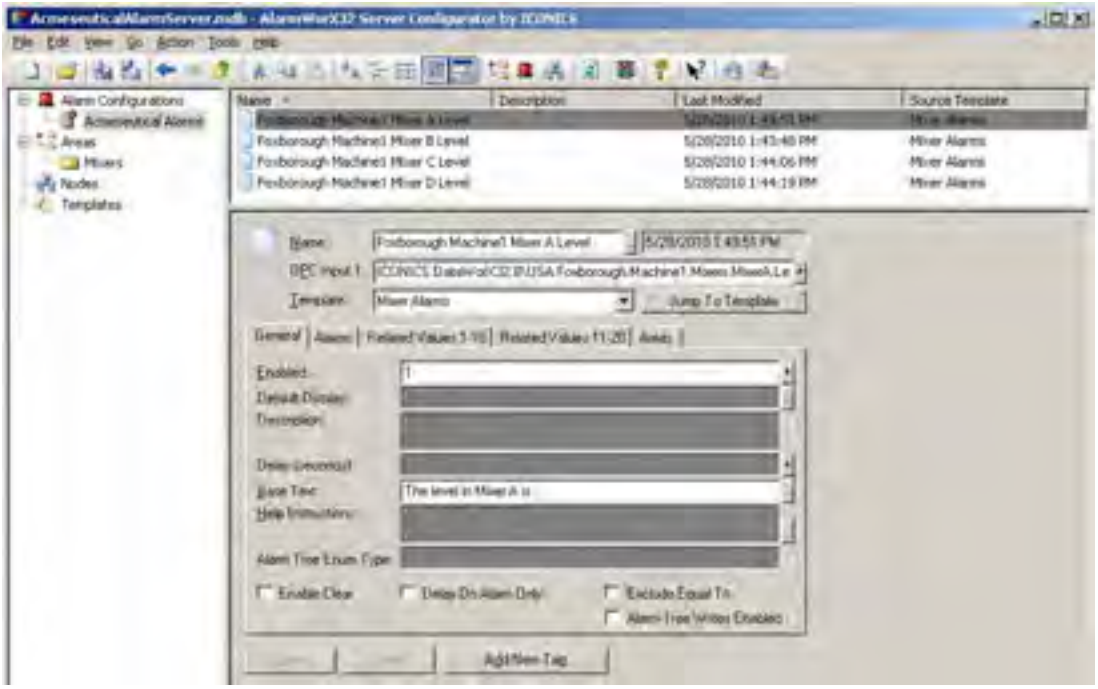


Figure L6.12 - Alarm Configuration

22. We will add one more alarm to notify us when the machine is not running.

23. Right-click on Acmeseutical Alarms in the tree control and select New → Tag.

24. In the new form that opens on the right-hand side, name the alarm "Foxborough Machine1".

25. Click on the arrow key to the right of the "OPC Input 1" text field and select "Unified Data Browser" (UDB).

26. Browse for the Foxborough.Machine1.Status tag and set the Enabled field to 1.

27. Go to the Alarms tab and select the Digital tab. Enter 500 in the Severity field.

28. Type in "0" in the "Alarm State Value" field and type in "Machine1 in Foxborough has stopped" as the message text.

29. In the "Return to Normal" field, type in "Machine1 in Foxborough is running".

30. Your configuration form should look similar to Figure L6.13. Click on "Apply" to save the changes.

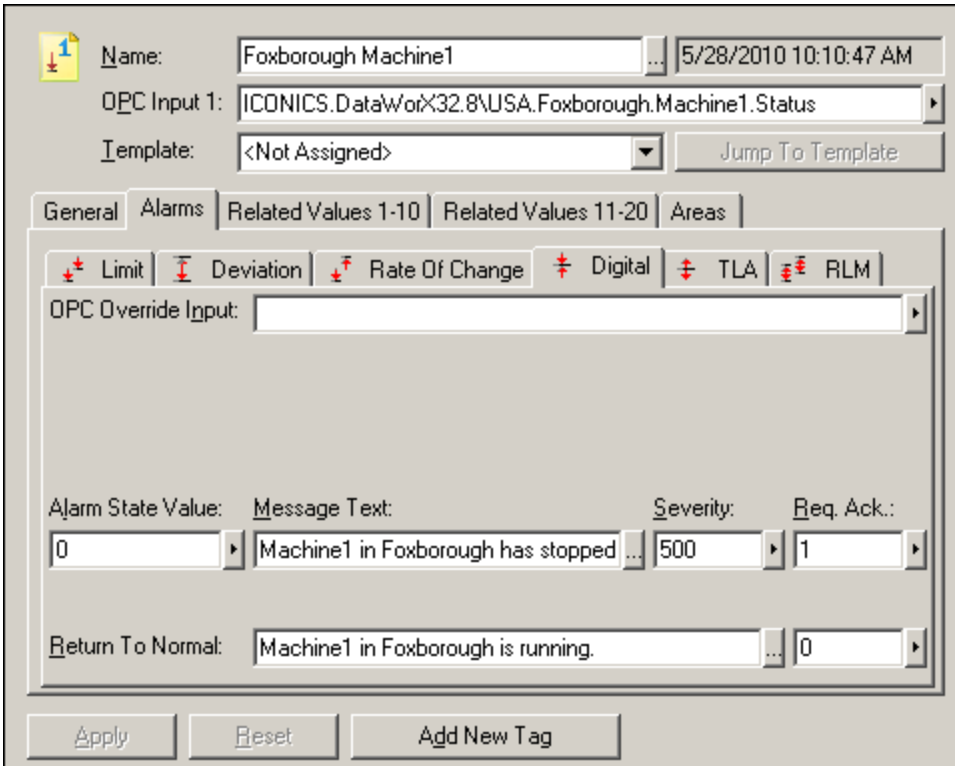


Figure L6.13 - Digital Tag Configuration

Review: AlarmWorX32 Server R6

6.1. Chapter Objectives

In this chapter, we discussed in detail how to configure the AlarmWorX32 Server. You should now understand the following topics:

- The function of the AlarmWorX32 Server
- The difference between the Alarm Server Configuration Database, the Alarm Server Configuration, and the Active Configuration
- How to use and configure alarm areas
- The different types of alarm tags available
- How to configure a limit alarm
- Different ways to generate many similar alarms at once

6.2. Summary

Using the AlarmWorX32 Server Configurator, you can define different alarm tags to generate notifications based on different values. The configuration is stored in a database: either MS Access or MS SQL. The configurator reads the information from the database and shows it graphically. You can configure many Alarm Configurations in the same database, but only one configuration can be active. The active alarm will be the configuration used to generate alarms.

You can configure six different types of alarms using the configurator: Limit, Deviation, Rate of Change, Digital, Trigger Limit, and Rate Limit alarms. Each of these alarms contains general configuration such as the name of the alarm, the OPC input, help instructions, default display, and much more.

Once you have created multiple alarms, you can organize your alarms by using areas. You can create areas in the Areas node in the configurator. Once you have created areas, you can associate an alarm to an area by going into the Area tab of the alarm or adding a link to the alarm in the area itself.

Many times, you may find yourself creating similar alarms. In these cases, you can use the Multiply option to create the alarm or use the Alarm Template.

An Alarm Template allows you to configure the common properties of an alarm once then create an alarm based off that template. If you need to overwrite values in the template, you can do that in the created alarm.

The AlarmWorX32 Server provider in Workbench32 allows for similar alarm configuration to the AlarmWorX32 Server Configurator. One notable option in the AlarmWorX32 Server provider from Workbench32 is the "Tag Import Wizard." Using this wizard, you can select multiple OPC tags to create alarms with using an Alarm Template. This makes creating many similar alarms fast and simple.

6.3. Terms

Active Configuration	Alarm Area	Alarm Tag
Alarm Template	AlarmWorX32 Explorer	AlarmWorX32 Logger
AlarmWorX32 Server	AlarmWorX32 Viewer	Configuration
Configuration Database	GAS Theme	Severity
Startup Squelch	Tag Import Wizard	

6.4. Test Your Knowledge

In this chapter, we discussed the steps required to setup the AlarmWorX32 Server. In the next few chapters, we will discuss clients that obtain data from the server. Before you continue to the next chapter, you should make sure that you can answer the following questions.

1. How many alarm configurations can you have within a single configuration database?

   a. 1
   b. 32
   c. 64
   d. Unlimited

2. How many active alarm configurations can you have within your system?

   a. 1
   b. 32
   c. 64
   d. Unlimited

3. True or False: The AlarmWorX32 Server is an OPC A&E compliant server.

4. Which types of alarms can you create through the AlarmWorX32 Configurator?

   a. Limit Alarms
   b. Rate of Change Alarms
   c. Boolean Alarms
   d. Deviation Alarms
   e. All of the Above
   f. Only A, B, and C
   g. Only A, B, and D
   h. None of the above

5. What is the severity range for an alarm defined by the OPC Foundation?

   a. 1-1000
   b. 0-1000
   c. $0\text{-}10^{32}$
   d. $1\text{-}10^{64}$

6. Which of the following alarm types can you associate with a deadband?

   a. Limit
   b. Digital
   c. Rage of Change
   d. Trigger Limit

7. How many related values can you configure for a single alarm tag?

   a. 10
   b. 20
   c. 30
   d. 40

8. What does it mean when the Req. Ack field contains a value of 1?

   a. An operator can acknowledge the alarm if they have the permission
   b. An operator can acknowledge the alarm regardless of configured security
   c. The alarm comes in already acknowledged
   d. The alarm turns to the acknowledged state as soon as someone visualizes it

9. True or False: You can delay an alarm for x-number of seconds when the value reaches the set point.

10. True or False: You can configure areas within an area.

AlarmWorX32 Viewer 7

In the previous chapter, we discussed the necessary steps to configure alarm tags in the AlarmWorX32 Server. By now, this server should be configured and ready to generate alarm notifications. In this chapter, we will trigger some alarms and visualize them. By the end of the chapter, you should be familiar with the following:

- The Alarm Viewer ActiveX and where to configure it's properties
- Configuring columns in the viewer
- Configuring local and global alarm subscriptions
- Configuring record sorting in the viewer
- Server- and client-side filtering

7.1. Alarm Viewer ActiveX

The **Alarm Viewer**, shown in Figure 7.1 is a real-time alarm client. Since this is an ActiveX component it can be placed in any ActiveX container application, such as GraphWorX32, Microsoft Visual Basic, or a web page. The Alarm Viewer displays current alarm information and handles the user interface to the alarm system (such as alarm acknowledgement). The layout of information displayed, including sort order, color, font, and displayed data, is user-configurable.

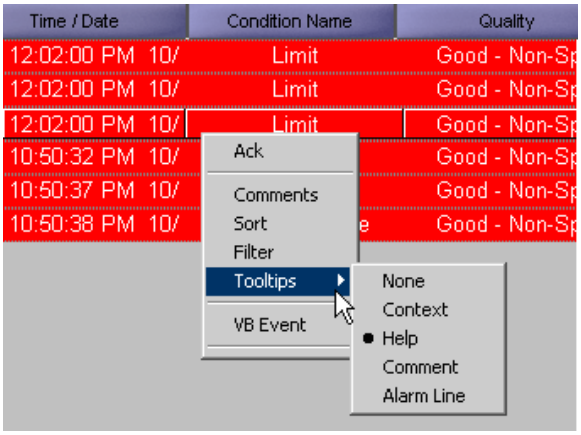


Figure 7.1 - Alarm Viewer ActiveX

You can drop this AlarmWorX32ActiveX control into any GraphWorX32 HMI display, an HTML Internet/intranet-based Web page, or any other ActiveX container. It automatically configures itself to deliver live alarms in a scrollable window.

You can easily customize the view via its properties page to control the colors, fonts, columns, rows, alarm filtering, subscriptions, hot-links, etc.

7.2. Alarm Viewer Properties

The Alarm Viewer ActiveX Properties dialog is the user configuration interface for the ActiveX control. When embedded in an ActiveX container, double-clicking anywhere in the ActiveX client opens the Properties dialog as shown in Figure 7.2**.**

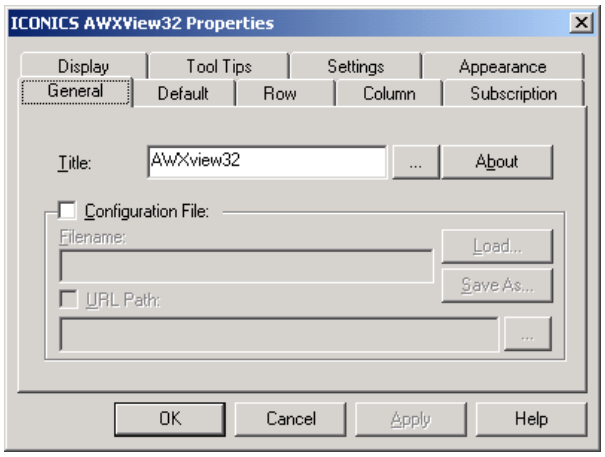


Figure 7.2 - Alarm Viewer ActiveX Properties

NOTE: Right-clicking on the Viewer ActiveX and selecting Properties also opens the Alarm Viewer ActiveX Properties dialog box.

The ActiveX Properties window contains the following configuration tabs:

- General
- Default
- Row
- Column
- Subscription
- Display
- ToolTips
- Settings
- Appearance

While we will not go through every detail in the properties, we will cover the basic information you need to connect to an OPC compliant A&E server. We will also setup the viewer to indicate different alarm information.

7.3. Alarm Subscription

The AlarmWorX32 Viewer can display data from any OPC Alarm and Event server. The way you connect to an OPC A&E server is by subscribing to it.

You can do that on the **Subscription** tab of the **Properties** dialog. You can subscribe to many OPC A&E servers in a single viewer. By default, when you add an Alarm Viewer it contains a full subscription to the ICONICS AlarmWorX32 Server and the ICONICS GenEvents Server.

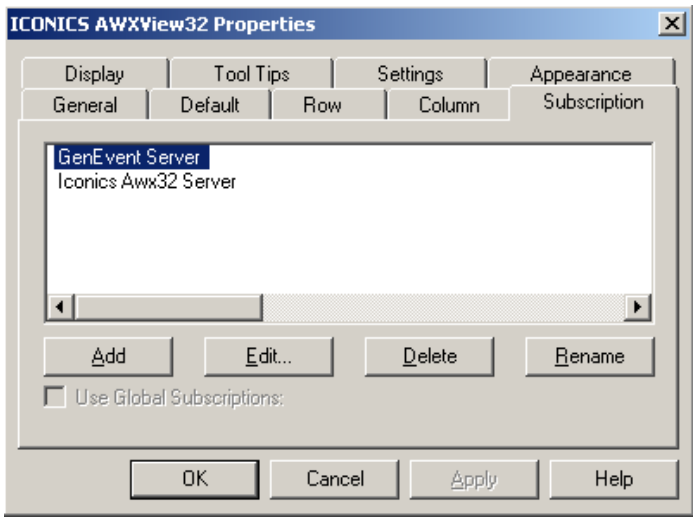


Figure 7.3 - Alarm Viewer ActiveX Control: Subscription Tab

The ICONICS AlarmWorX32 Server is where we configured our alarm notification previously, and we will work closely with the server in this chapter. The ICONICS GenEvents Server is an OPC events server that keeps track of ICONICS events such as when the AlarmWorX32 Server starts, when an operator writes a value to an OPC point, and when security is configured, it can also track information such as who is logged into the system. This server contains much information about the system that, when tracked, can provide much information about the activities on the machine.

Normally you would not subscribe to the ICONICS GenEvents server in the viewer since the information may be unnecessary for an operator. You can remove this subscription by selecting it in the Subscription tab and clicking on the “Delete” button. Similarly, you can use the “Add,” “Edit,” and “Rename” buttons in the tab to perform the respective functions.

When you add or edit a subscription you will encounter a separate dialog for the subscription properties. In this dialog, you will be able to specify the OPC A&E server to subscribe to, provide server-side filters, and configure any additional alarm attributes that you need.

NOTE: You can also add global subscriptions in which case you will use the Unified Data Browser to select the subscription instead.

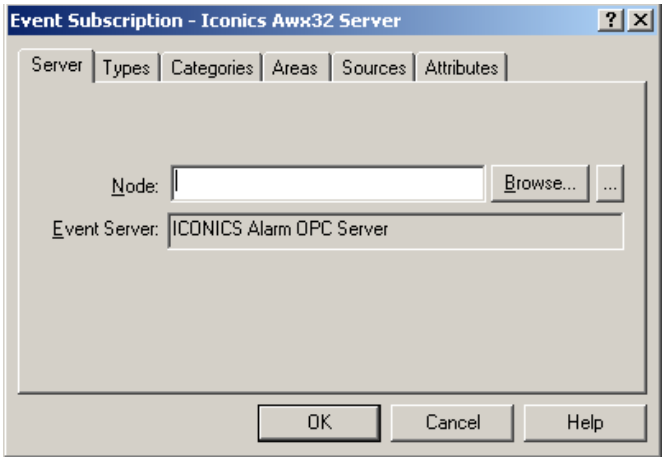


Figure 7.4 – Edit Subscription Properties

There are six tabs in this dialog:

- **Server** – Where you browse for the OPC A&E server you are subscribed to
- **Types** – Defines the event types and severity you wish to receive
- **Categories** – Defines the categories (deviation, digital, limit, rate of change, etc.) of alarms to receive
- **Areas** – Defines the areas in the alarms server to subscribe to and receive alarms
- **Source** – Defines the individual alarms you wish to receive
- **Attributes** – Defines additional or custom attributes per event category that you need to view

It is important to keep in mind that if you do not expressly specify a Category, Area, or Source in the subscription property, then you will receive all alarms and events generated by the server. Once you add anything to any of these three tabs you will only receive information specified in the tabs. Specifying individual categories, areas, and sources is also referred to creating a server-side filter since the alarm server will only send alarms that meet those criteria.

Once a subscription is setup the viewer requests a refresh of all current alarms upon animation. The OPC A&E server then sends alarms on an alarm-by-alarm basis. This decreases the number of alarms sent to each viewer and allows for server-side filtering as mentioned before.

7.4. Column Configuration

The **Column** tab shown in Figure 7.5 allows you to select which columns you want to appear in the viewer. Columns show information from the various attributes associated with an alarm.

All OPC A&E servers will have to provide a set of standard attributes such as Source (or Tag), Message, and Event Time. In addition to these mandatory attributes, an OPC Alarm and Event Server can also provide other attributes that are specific to the server. For example, the ICONICS Alarm Server provides the attributes Current Value, Area, and Help, among others. These attributes are Custom Attributes, and can be assigned in the alarm subscription to the generic attributes named Attribute 1-20 in the Available list.

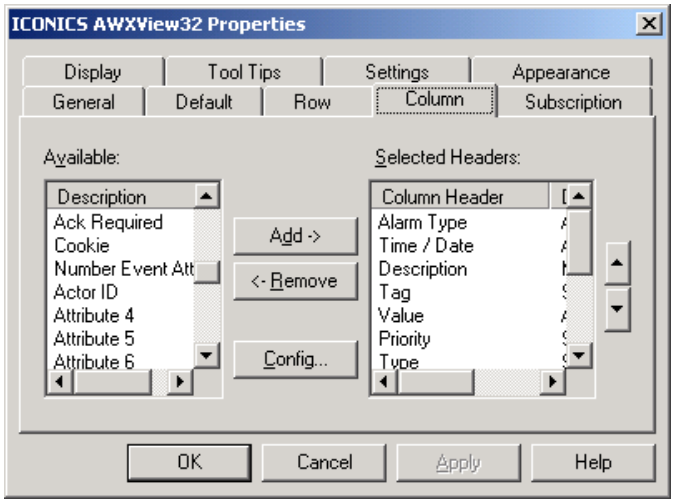


Figure 7.5 - Alarm Viewer ActiveX Control: Column Tab

NOTE: You should not confuse the additional attributes with Related Values in the alarm server. You will need to add the Related Value as one of the twenty custom attributes. The association is not automatic.

You can also change the appearance of a column by either double-clicking on an item in the "Selected Headers:" list or selecting the item and clicking the "Config…" button. The **Set Column** configuration dialog for the item would open, from which you can set the font, color, width, and cell properties by selecting and changing the various options available.

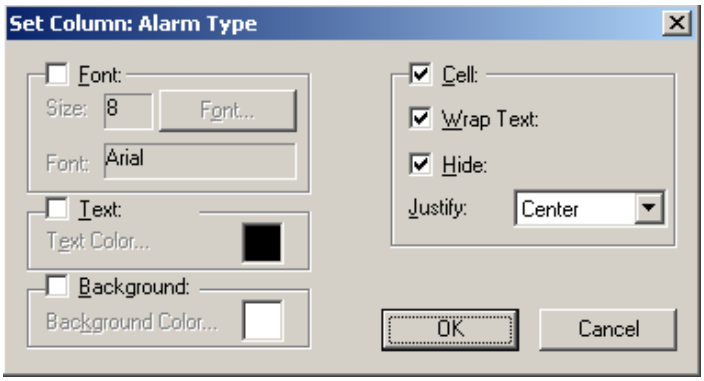


Figure 7.6 - Set Column Dialog

In addition to the runtime appearance of the columns, you can rename the Heading of each column such that it does not have to read the name of the attribute specifically. This is especially useful when using Custom Attributes.

7.5. Row Configuration

Once you define the attributes you wish to see about an alarm, you can add color indication to the alarm, which can tell you information at a quick glance. The **Row** tab, shown in Figure 7.7, allows you to configure a different appearance of an alarm based on the event type and severity.

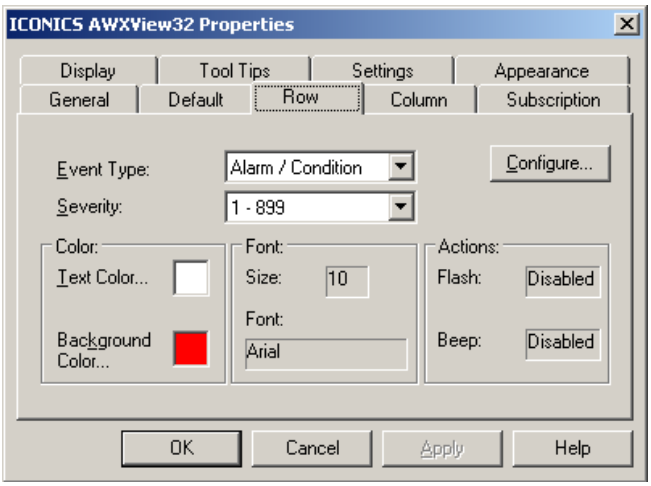


Figure 7.7 - Alarm Viewer ActiveX Control: Row Tab

For example, in Figure 7.7, for any event that comes in to the viewer in the alarm condition having a severity between 1 and 899, the event will have a red background with white text. You can add your own definition by clicking on the "Configure…" button, which will bring up the **Set Row** dialog.

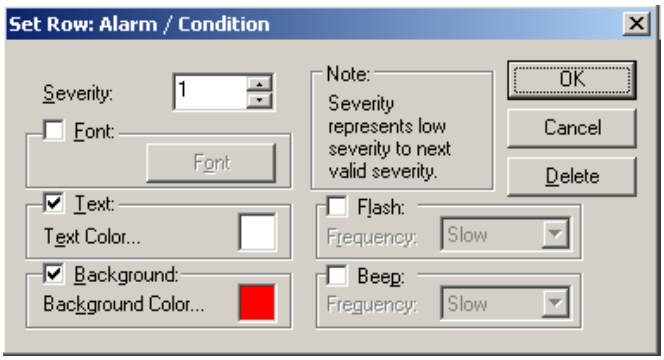


Figure 7.8 - Set Row Dialog

In the severity field of this dialog, you can enter the lower limit. For example, if you chose to configure the event type as shown in Figure 7.7, when you enter 500 in the severity, you will end up with two row configurations: one for severity between 1-499 and another for severity 500-899.

In this dialog, you can configure the text and background colors for the condition as well as adding flash animation and sound to an event. This is especially helpful in obtaining more attention on alarms with higher severity.

7.6. Record Sorting

The **Display** tab of the **Properties** dialog configures options for sorting and client-side filtering. The **Sort** dialog, shown in Figure 7.9, is accessed by clicking the "Sort" button in the **Display** tab. This is where you can configure the sorting criteria. Records can be sorted based on any column selected in the viewer. It also allows for multilevel sorting up to five levels with ascending/descending specific to each level. The default sort for a new viewer is Time/Date. In the event that the Time/Date column is removed from the viewer, you will be informed that the sort key will need to be changed.

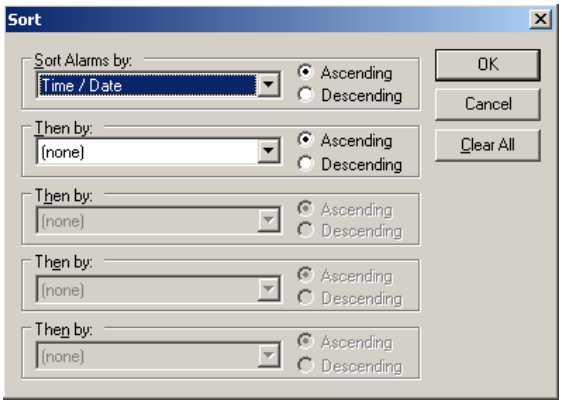


Figure 7.9 - Sorting Properties

7.7. Filtering Configurations

When the Alarm Viewer is configured for **Server-Side Filtering**, it is actually limiting what alarm information will be sent by the Alarm Server to the Alarm Viewer. This filtering mechanism can be especially useful if an Alarm Server can potentially send many thousands of alarms to an Alarm Viewer when only a small number of these alarms are actually relevant to this particular instance of the Alarm Viewer.

Server-side filtering is configured by choosing to limit your alarm subscription to specific types of alarms, or alarms from specific places. Therefore, server-side filtering is configured with the **Subscription** properties.

When the Alarm Viewer is configured for **Client-Side Filtering** it receives all alarms from the Alarm Server as normal, however, it applies a filter such that the user will only see a selection of alarms rather than the full list. This is useful when an operator needs to apply a filter temporarily to narrow down the information shown and will have to switch between different views quickly.

A client-side filter is a filter that does not change the subscription to the alarm server. This means that all alarms will still be sent to the Alarm Viewer and maintained in the background but they are simply not displayed in the Alarm Viewer. These filters can be created using the “Filter” button on the Display tab of the Properties window shown in Figure 7.10.

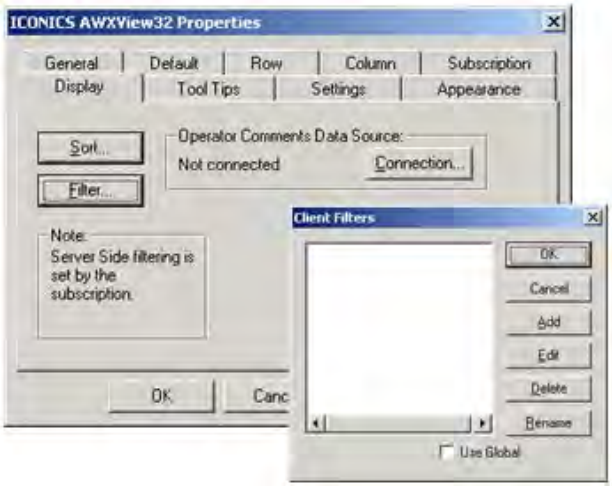


Figure 7.10 - Client-side Filter Properties

Client-side filters are created and edited using the **Expression Editor** and its **Filter Wizard**. The first time you create a filter, Filter Wizard will appear as shown in Figure 7.11. You can use Filter Wizard to quickly and easily create your filter. For more

advanced filters you can click on the "Advanced…" button to bring up the Expression Editor.

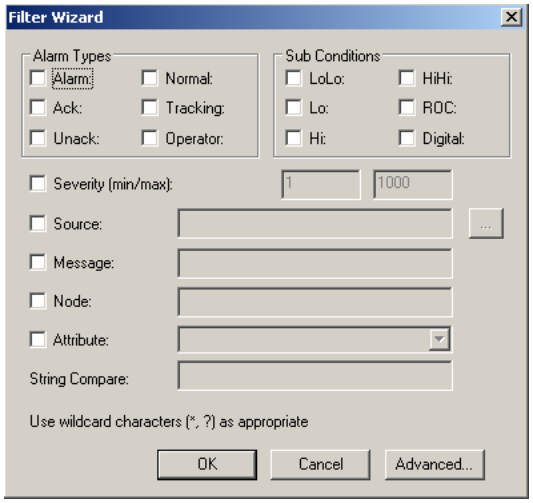


Figure 7.11 - Filter Wizard

If you need to edit a configured filter, know that all filters always appear in the Expression Editor. You can go back to the Filter Wizard from the Expression Editor, but any information you currently see in the Expression Editor will be erased once you try to configure the filter in the wizard again.

> **NOTE:** You may use global filters, but if you check this option you will be browsing for the filter in the Unified Data Browser.

7.8. ToolTips Configuration

The **ToolTips** tab in the **Properties** dialog, shown in Figure 7.12, allows you to determine which tooltip options are available for switching during runtime by selecting the corresponding check boxes. These tooltips are visible when you hover over something in runtime.

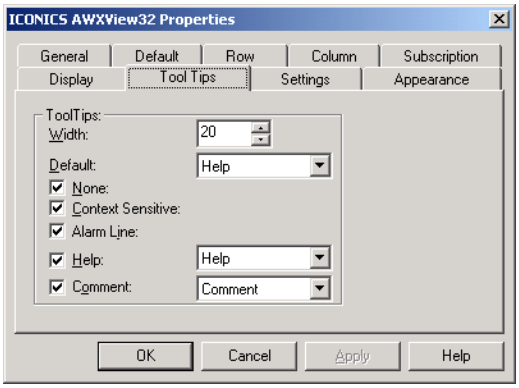


Figure 7.12 - Alarm Viewer ActiveX Control: Tooltips Tab

You can choose a default setting for tooltips here, and assign the column that will be displayed in the Help and Comment tooltip (in the event that this information is set to be shown in the tooltip). You can also choose how many characters the tooltip should display.

7.9. Runtime Options

For ease of access many of the options available in configuration mode are also available in runtime mode. First, you can sort your alarms easily in runtime by double-clicking on a column header. This will sort that column in ascending order. Double-clicking on the column again will sort it in descending order.

You can also right-click in the ActiveX control to bring up the context menu and select the sort option. The dialog you will see is the same one brought up by clicking on the "Sort" button on the **Display** tab. As you can see from Figure 7.13, there are many more options available. For example, you can configure your client-side filters during runtime as well. The filter configuration process and interface are the same as in the configuration mode.

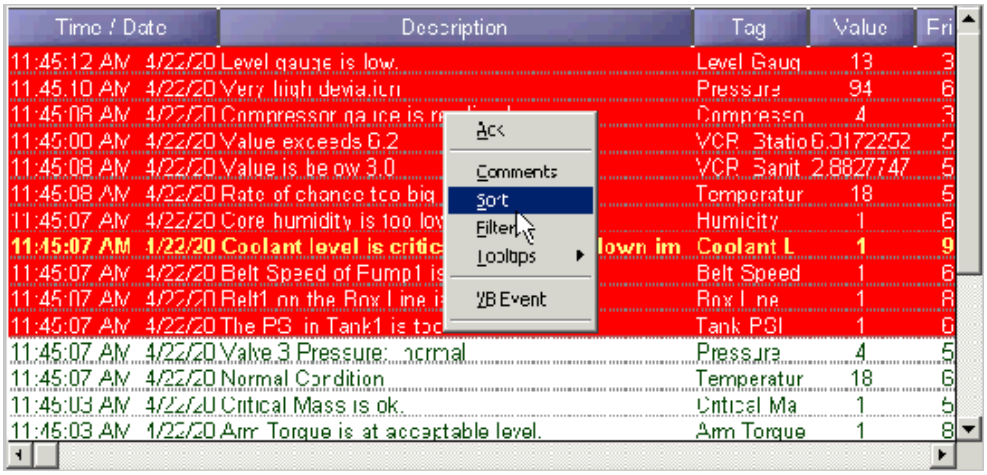


Figure 7.13 - Alarm Viewer Runtime Right-click Context Menu

There are also actions that you can perform in the viewer that are only available in runtime. For example, double-clicking on an alarm (row) will acknowledge the alarm. You can also use the right-click context menu to perform other acknowledgement type actions such as "Global Ack," "Visible Ack," "Filter Ack" and many others. You can change the behavior of the alarm viewer in runtime by configuring the **Settings** tab of the viewer's properties.

Lab: View Alarm Notifications L7

In the previous lab, we configured alarms. In this lab, we will provide the alarm notifications to the operators in a GraphWorX32 display. Using this display, the operators can view the alarms and acknowledge them.

Estimated time to complete this lab: 45 minutes

7.1. Task Checklist

In this lab, you will complete the following tasks:

- Create a display that shows alarm notifications
- Generate alarms using a display then acknowledge the alarms to test it

7.2. Support Files

You will need the following file(s) to complete this lab:

- AcmeseuticalProject.prjx
- ACMEPillMachine.gdf
- AcmeseuticalTemplate.tdf

7.3. Viewing Alarms

We will first need to add an Alarm Viewer ActiveX to a display to see the alarms.

1. Locate your AcmeseuticalTemplate.tdf in Windows Explorer and double-click on it to create a new display.
2. Once the display opens, save it in the Training folder as “AcmeseuticalAlarms.gdf”.
3. Click on the “ICONICS AWXView32” icon. This adds the ICONICS AlarmWorX32 Viewer ActiveX control to your display.
4. Size the control so that it takes up most of the screen.
5. Double-click on the control to bring up the properties dialog.

6. Select the Subscription tab. There are two default subscriptions here: one is subscribed to the ICONICS AlarmWorX32 Server and the other is subscribed to the GenEvent Server.

7. Since we are not interested in viewing the events, select the “GenEvent Server” and click on the “Delete” button to remove it.

8. Select the “Iconics Awx32 Server” subscription and click on “Edit.”

9. In the Event Subscription dialog that comes up, select the Attributes tab.

10. Select “Area” in the Available column on the left and click on the “Add ->” button to move it to the right.

11. Take note of the attribute number. It should be attribute number 4 if you have not added any other attributes yet.

12. In the Event Category dropdown list, select “Digital” and add the Area attribute as attribute number 4.

13. Do the same for Event Categories: Limit, Rate Limit, and Rate of Change.

14. Click on “OK” to save the settings and close the Event Subscription dialog when you are done.

15. Go to the Column tab in the viewer properties and remove the following columns by selecting them and clicking on the “<- Remove” button.

    - Time/Date
    - Type
    - Quality
    - Node

    Tip! You can select more than one column at a time by holding the Ctrl key and selecting the column.

16. Select “Attribute 4” in the Available list and move it to the "Selected Headers" list by clicking on the “Add ->” button.

17. Right-click on Attribute 4 in the Selected list and choose “Rename”. Rename the Attribute to “Area”.

18. Select the “Event Time” column and move it up to the second item by clicking on the up arrow button to the right of the list.

19. Arrange the Selected list by using the arrow buttons so that it looks like the list in Figure L7.1.

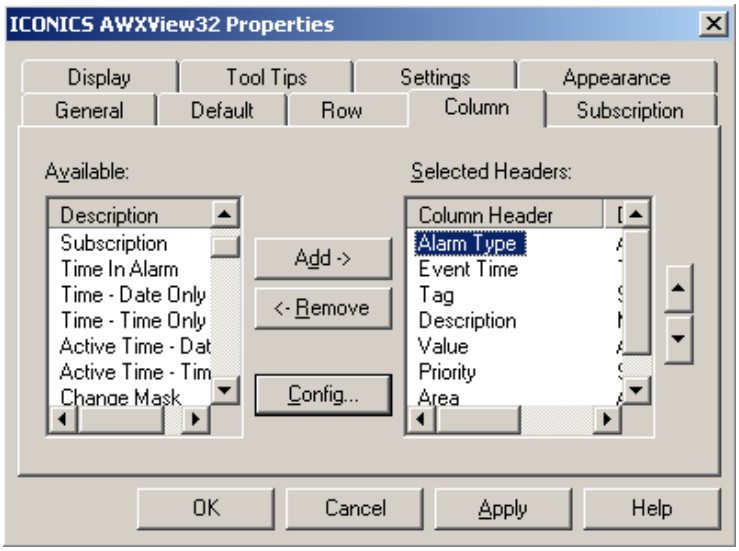


Figure L7.1 - AlarmWorX32 Viewer Properties: Column

20. Select the Row tab. Make sure the Event Type is set to “Alarm/Condition” and click on the “Config…” button.

21. In the new dialog that comes up, type in 800 for the Severity field.

22. Change the color of the Text to purple by clicking in the text color block (Red: 128, Green: 0; Blue: 255). Click on “OK” to close the dialog when you are done.

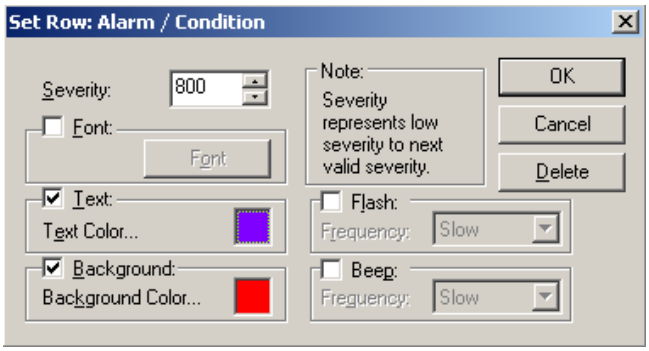


Figure L7.2 - Set Row Dialog

23. Go to the Default tab and uncheck the “Scientific Notation” option.

24. Go to the Tool Tips tab and set the Default to “Alarm” in the dropdown list.

25. Uncheck the Help and Comment checkboxes.

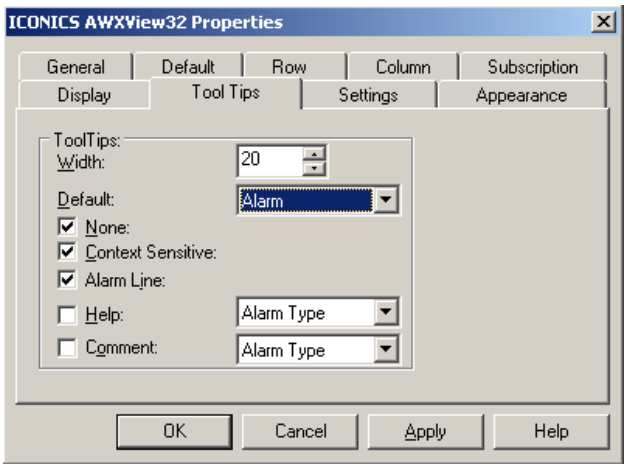


Figure L7.3 - AlarmWorX32 Viewer Properties: Tool Tips

26. Select the Appearance tab and click on the "Column Width…" button to bring up the Set Column Size dialog.

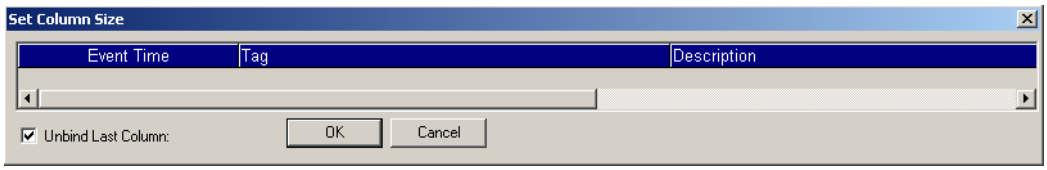


Figure L7.4 - Set Column Size

27. Widen the Tag column. Click on "OK" close the Set Column Size dialog.

28. Click on "OK" again to save the settings and close the AlarmWorX32 Viewer properties dialog. Save the display.

29. We are now ready to view the alarms. First, we will add some notification and a way to get to our alarm display.

30. Open the ACMEPillMachine.gdf.

31. Drag the Attention symbol from the Symbols Library and place it at the lower right-hand corner of the display.

32. With the symbol selected, click on the "hide/disable" icon in the dynamics toolbar.

33. In the Property Inspector that comes up, click on the "Data Tags…" button and browse through ICONICS.AlarmServer_.1 and select the Recursive_Active tag.

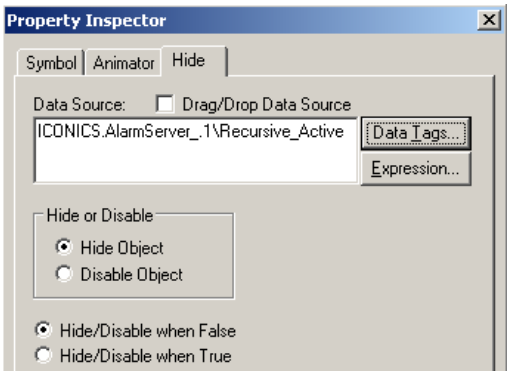


Figure L7.5 - Property Inspector: Hide

34. Click on “OK” to save the settings and close the Property Inspector dialog.

35. Still with the Attention symbol selected, click on the “Pick” icon.

36. Leave the Action as “Load Display” and type in “AcmeseuticalAlarms.gdf” in the Filename field.

37. Click on “OK” to close the Property Inspector. Save the ACMEPillMachine.gdf display.

38. Save the AcmeseuticalAlarms.gdf display.

39. Put the display into runtime. If your mixers are empty, you should have alarms and see the attention symbol at the bottom of the display.

40. Click on the Attention symbol to load the alarm display we created, previously in this lab, to view the alarms.

41. Double-click on one of the alarms to acknowledge it. Notice the acknowledged alarm has a different background and text combination to indicate that it is no longer in the alarm state.

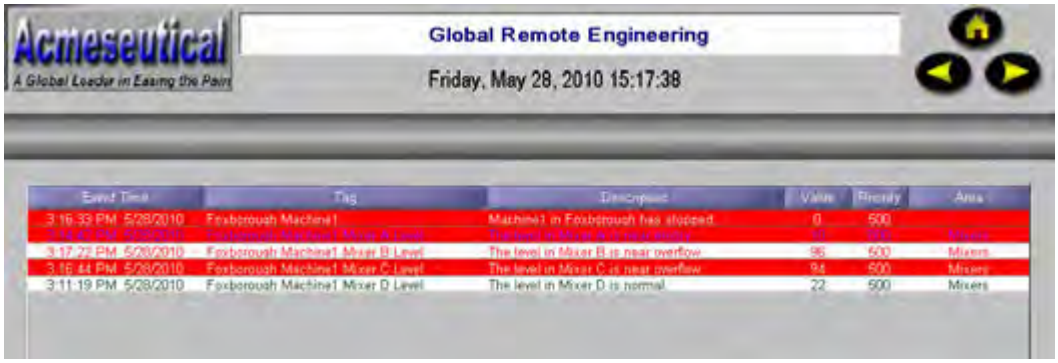


Figure L7.6 - Acmeseutical Alarms

42. Click on the Back icon at the top of the display to go back to the Acmeseutical Pill Machine display.

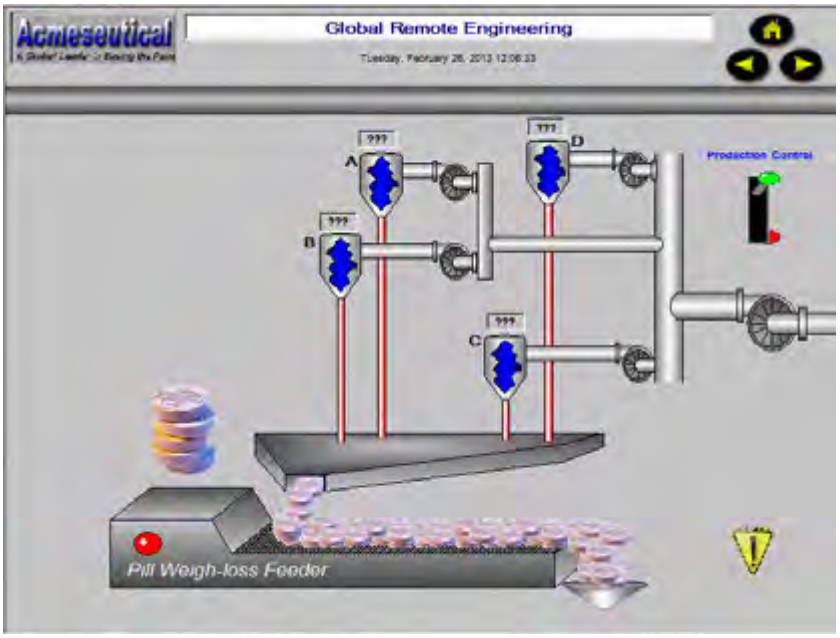


Figure L7.7 - Pill Machine Display with Active Alarms

43. As functionality is added in subsequent labs, you will start filling the Mixers to stop the alarms from generating. You'll want to make sure that the Mixers are not too full or they will generate additional alarms. Once you have a good balance of the Mixer levels, the alarm Attention symbol should disappear.

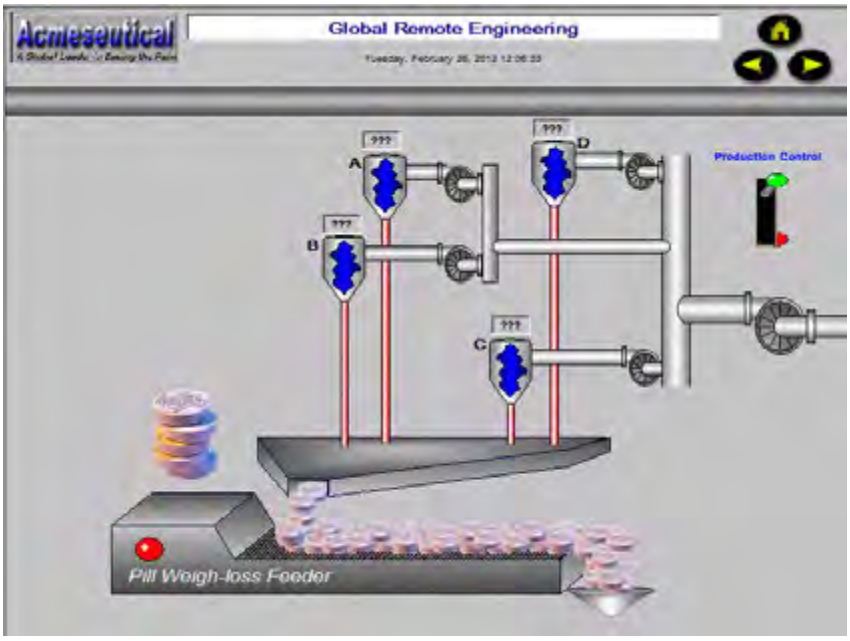


Figure L7.8 - Pill Machine Display without Active Alarms

Review: AlarmWorX32 Viewer R7

7.1. Chapter Objectives

In this chapter, we discussed how to configure the ICONICS AlarmWorX32 Viewer. You should know and understand the following topic:

- How to subscribe to an OPC A&E Server
- How to configure custom attributes
- How to customize rows and columns in the viewer
- How to create client-side filters and sort records
- The difference between server- and client-side filters

7.2. Summary

The ICONICS AlarmWorX32 Viewer ActiveX control is an OPC A&E client that can subscribe to any compliant OPC A&E servers to display alarms and event information. Since this is an ActiveX control, you can place it in any ActiveX container application such as GraphWorX32.

The alarm viewer has many configuration properties. You can setup different information about the alarm viewer in the respective tabs. On the Subscription tab, you can add connections to many different OPC A&E servers. On each subscription, you can further limit the information that the OPC A&E server will send to the client by adding Categories, Areas, or Sources. By including this additional information, you are creating a server-side filter. This can be very useful in reducing network traffic if the alarm server resides on a remote computer.

You can also add client-side filters in the viewer properties. The first time you try to create a filter, the Filter Wizard will popup, helping you create your filter. From thereon, when you want to edit the filter, you will use the Expression Editor. Client-side filters differs from server-side filters in that they only limit the visible alarms in the viewer; the alarm server still sends all available alarms to the client.

In the alarm viewer properties, you can also configure Columns, or the attributes of an alarm that you wish to see as well as Rows, which dictates how an alarm will appear based on the event category and severity. Additionally, you can also configure sort order and tooltips.

When you are in runtime, you can use the right-click context menu to perform many actions. You can configure filters, acknowledge alarms, change the sort order, and much more.

7.3. Terms

Alarm Attributes	Alarm Subscription	AlarmWorX32 Viewer
AlarmWorX32 Container	Client-side Filter	Expression Editor
Filter Wizard	Server-side Filter	

7.4. Test Your Knowledge

In this chapter, we described how you can connect to the alarm server using the alarm viewer to view real-time alarm information. You should now understand how to connect alarm clients to alarm servers as well as how to customize the alarm viewer. You should now be able to answer the following questions.

1. How do you configure the viewer to receive alarms from a particular server?

    a. Create a subscription from the viewer to the alarm server.
    b. In Workbench32, draw a line from the server to the viewer in the system diagram.
    c. Give a name to the viewer, and add that to the alarm server configuration.
    d. You do not need to do anything. The viewer will automatically detect all alarm servers in the network and subscribe to them automatically.

2. How do you view related values configured in the AlarmWorX32 Server in the alarm viewer?

    a. You can only view related values through automation methods.
    b. When you are in runtime, right-click on the alarm and select the appropriate related value in the context menu.
    c. In the subscription, add the related value as one of the custom attributes. Add the associated attribute as a column in the viewer.
    d. You do not need to do anything. The related values are automatically associated with the custom attributes in order.

3. True or False: You can subscribe to multiple alarm servers in a single viewer.

4. True or False: You can use both local and global alarm subscriptions in a single viewer.

5. Where would you define the appearance of an alarm based on its category and severity?

   a. The Appearance tab of the Alarm Viewer Properties dialog
   b. The Row tab of the Alarm Viewer Properties dialog
   c. The Column tab of the Alarm Viewer Properties dialog
   d. The Alarm Viewer VBA Properties dialog

6. Which of the following is NOT a filtering criterion that you can configure on the Filter Wizard?

   a. Alarm Types
   b. Source
   c. Attribute
   d. String comparison functions

7. True or False: You can add a beeping sound to high severity alarms to attract and operator's attention.

8. Configuring which of the following will NOT be considered as configuring a server-side filter?

   a. Categories
   b. Areas
   c. Sources
   d. Attributes

9. True or False: You must add the column in the order you wish to see them; you will not be able to recorder the columns after adding them.

10. True or False: An operator can comment on an alarm without acknowledging it.

AlarmWorX32 Logger 8

In the previous chapters, we have configured an alarm server to generate notifications as well as learned how to setup a viewer to visualize real-time alarms. It is now time for us to record the real-time alarms into a database for later analysis. We will do this with the AlarmWorX32 Logger. In this chapter we will go over the following:

- Alarm Logger Configurator
- Logging database connections
- Databases supported by the alarm logger
- Alarm subscriptions and attributes
- Column definitions
- Node configuration

8.1. The AlarmWorX32 Logger

The **AlarmWorX32 Logger** logs alarms to Microsoft Access, Microsoft SQL Server, or Oracle databases. It can also print out the information to one or even a redundant set of printers. You can create alarm reports and calculations with Microsoft's built-in Visual Basic for Applications (VBA) scripting language. Alarm calculations can be performed on individual tags or groups of tags as well.

The logger can run as an executable or as a service, and is able to process alarm information from several alarm servers locally or over a network. The database is user-configurable (i.e., it allows auto-flushing at a specified period, auto-start, circular and append logging modes, etc.). Of primary importance to the database is performance; the logger is able to handle bursts of alarms due to the design and implementation of the logger's database handling.

The **AlarmWorX32 Logger Configurator**, shown in Figure 8.1, writes configuration information into a configuration database. The AlarmWorX32 Logger reads this configuration information during runtime to determine what alarms and events to log and where to log them. A single database can contain multiple named configurations to run on one or more nodes in a network. Unlike other configurations, a single node can also run multiple configurations simultaneously. In keeping with the open standards employed throughout the GENESIS32 architecture your alarm configurations are saved to a Microsoft Access or Microsoft SQL Server database.

Figure 8.1 - Alarm Logger Configurator

8.2. Alarm Logger Configuration

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

Like many of our products, the Alarm Logger comes with a default logging configuration database. There are two configurations setup in the default: one for logging alarms and one for logging events. The simplest way to start logging/printing your alarms is to use the default configuration. Changing only the location where you want the historic alarms to reside or configuring the printers where you want to create a paper trail for the alarms.

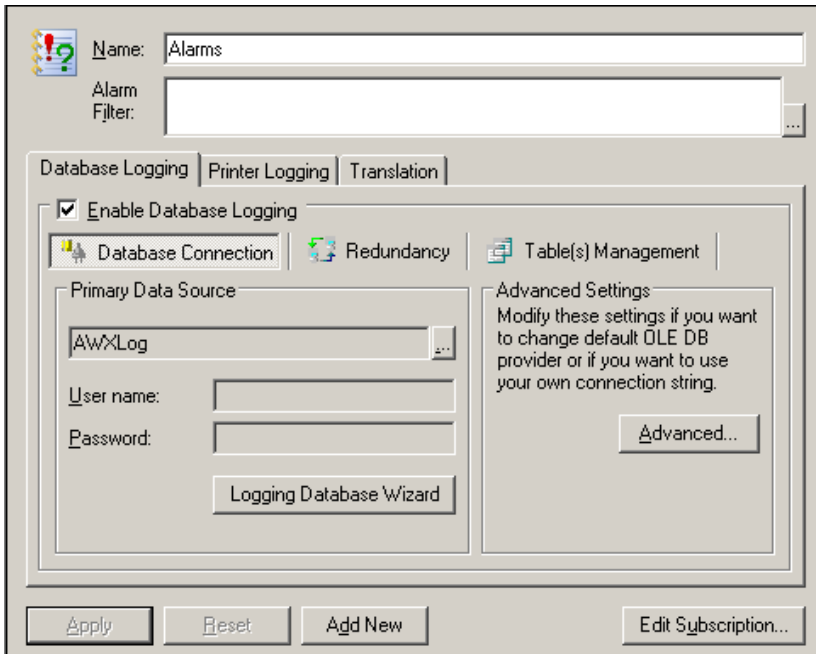


Figure 8.2 - Alarm Logging Configuration

In this course we will focus on **Database Logging**. The Configurator provides additional settings for **Printer Logging**, **Translation**, and **Alarm Snapshots** but these are out of the scope of this class. For more information on these settings refer to the "Logger Configuration Properties" section in the AlarmWorX32 Logger Help documentation.

> **NOTE:** The **AlarmWorX32 Logger** is designed to print alarms in real-time directly to your printer or backup printer. **AlarmWorX32 Report ActiveX** can be automated to display alarm reports on a periodic basis or you can use **ReportWorX** to print or save reports to files.

Let us now take a closer look at how to construct an alarm logging configuration that will record alarm information into a database.

8.2.1. Alarm Sources

The first thing you should do when creating an alarm logger configuration is to make a subscription to the alarm server that provides the alarm notifications. The alarm logger is an OPC compliant A&E client and you can subscribe to any OPC A&E server.

You can create, view, or edit subscriptions by clicking on the "Edit Subscription…" button at the bottom right portion of the alarm configuration form. This should bring up the **Subscription Properties** dialog. If you look closely, this dialog is exactly the same as the "Subscription" tab in the alarm viewer properties and the way you would add or edit a subscription is the same as doing it in the alarm viewer.

Furthermore, you can use a filter to limit the alarms that are logged. For example, if you only want a record of the high severity alarms, you can create a filter to disregard the alarms that are less severe. In the AlarmWorX32 configurator, if you click on the Alarm Filter: ellipsis button and select the type of filter (global or local), you will get the same Filter Wizard or Expression Editor, under the Advanced button, as you saw when creating a filter in the alarm viewer.

8.2.2. Database Connection

An alarm configuration consists of three tabs: Database Logging, Printer Logging, and Translations. We will focus first on the Database Logging. This is where you would go to configure the destination database where your historic alarm information would reside.

The **Database Connection** page of the configuration properties, shown in Figure 8.3, allows for the configuration of where the alarms are logged.

The configuration connects to the destination database via an ODBC Data Source, which can be created, configured, and selected here.

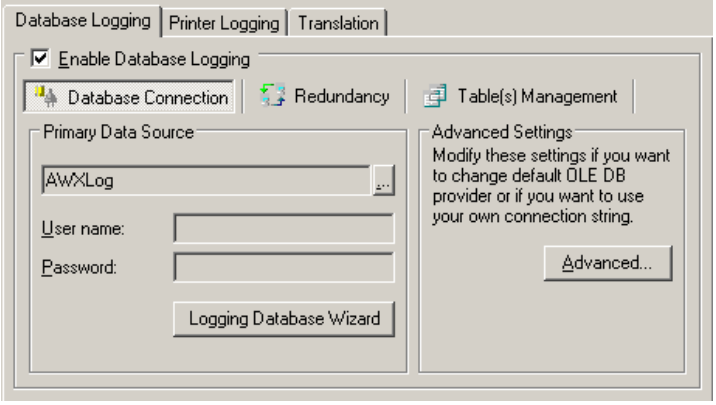


Figure 8.3 - Database Logging: Database Connection

By clicking on the ellipsis button under the Primary Data Source section you will bring up the **Select Data Source** dialog where you always want to create a "Machine Data Source." If you have already created the data source that points to the desired database, you can select it here.

> **NOTE:** If you are using a 64-bit machine and you have created your data source using the Windows Data Source (ODBC) interface, it is possible that you will not see the data source here. The general cause for this is because you have created a 64-bit data source while GENESIS32 can only use a 32-bit data source.

Otherwise, you can click on the "New…" button to create a new one. In the **Create New Data Source** dialog you should always select to create a new "System Data Source." If you create a User Data Source you will always have to login as the currently logged in user for the data source to work. Clicking on the "Next" button will display a list of drivers. Select the proper drive for the type of database you want to log the alarms to and click on "Next" to continue.

> ***Tip!*** If you are logging data to an MS Access database, select the Microsoft Access Driver (*.mdb). If you are looking to MS SQL 2000 or prior databases, select SQL Server. If you are looking to MS SQL 2005 and later, select SQL Native Client.

Depending on the type of database you are logging the alarms to the next steps will be slightly different. However, the essence of the steps would be to locate the destination database and make a connection to it. At the end of this process you will have the opportunity to test your new database connection.

This test is highly recommended. Before proceeding you should receive a message indicating a successful connection to the database.

If the test is unsuccessful you should reconfigure the data source until you can perform a successful test. Until there is a successful connection to the database you will not be able to log any information.

> **NOTE:** For more information on how to make a connection to the various types of databases that we support, please refer to the respective Help documentation and Application Notes. The lab that follows the chapter will walk you through in connecting to a MS SQL 2005 database.

8.2.2.1. Database Support and Options

The AlarmWorX32 Logger can log data to different databases, including Microsoft Access, Microsoft SQL Server, MySQL and Oracle. Different databases require a different ODBC setup and some of the dialogs may be different depending on which logging database you use.

As Microsoft Access has its limitations, a database server like Microsoft SQL Server, mySQL, or Oracle may provide better performance and functionality for larger data-logging systems. This becomes especially relevant when considering maintenance as Microsoft Access databases do not have an automatic compacting function. This means that once in a while a Microsoft Access database will have to be compacted manually. During the compacting process, the AlarmWorX32 Logger has to be stopped because it is not possible to log any data to the MS Access database while it is being compacted.

8.2.3. Table(s) Management

Once you have connected to a database you should define how much data you want to keep in the database. You can do this on the **Table(s) Management** page under the **Database Logging** configuration tab. On this page, you can define how many records or how long you would like to keep data for in the Active table. When you configure the size for the Active Table you should keep in mind that the bigger this table gets, the longer it might take to query for all the data and refresh when trying to view the historical data in GraphWorX32.

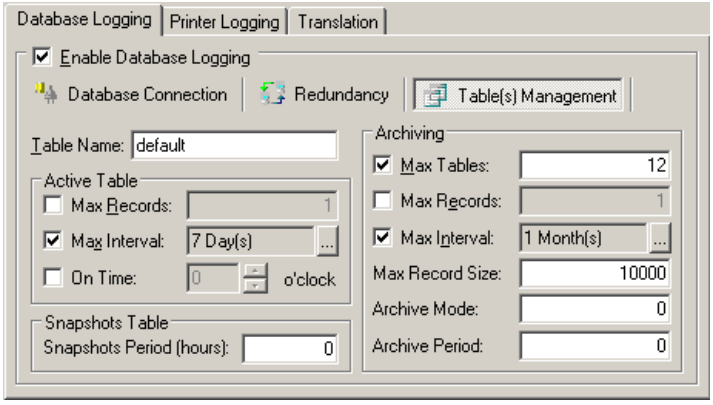


Figure 8.4 - Database Logging: Table(s) Management

The Alarm Logger also provides various archiving options for database management in the **Tables Management** section of the **Database Logging** configuration tab. The **Active Table** can be limited to a maximum number of records or a maximum time period. If no additional archiving parameters are specified, there will only be one table at any point in time observing the rules for the active table.

Additionally, it is also possible to maintain a number of **Archived Tables.** These archived tables are maintained either up to a maximum number of records, or up to a maximum time period for each table specified in the Max Table field. If any of these conditions expire the oldest table will be removed. Note: the fields under "Archiving" will become available only after fields are selected in the "Archive Table" section. Otherwise the fields are grayed out.

There are a few different values for "Archive Mode." Under normal circumstances you can leave the mode as default (value 0). We created the other archive modes for advance users who want finer control in the archiving process. Depending on the amount of records that need to be archived, using a different archive mode may also enhance the archiving performance.

> **NOTE:** For more information about the archive modes, please refer to the respective Help documentation and Application Note.

8.3. Alarm Logger Columns

Now that we know where to log the alarms and how we would like to organize them into tables, we will need to define the alarm attributes that we want to record. Notice that each alarm logging configuration contains one or more **Columns**. The columns defined for a configuration appear as child elements of the configuration in the tree control and as list items in the list view (top right pane) of the Logger Configurator.

Each column is associated with an alarm attribute either subscribed to by default or as a custom attribute. Double click on any column value to view the column details specified below.

The following fields are specified for each column:

- **Name:** Enter the name of the column. This will be the column name used in the logging database and the column heading when printing.
- **Column Type:** Select the OPC Alarm and Events data item you wish to log in this column from the dropdown list.
- **Maximum Width:** Enter the number of character spaces for this column when printing.
- **Line Number:** When printing, each event may occupy more than one printed line. Specify the line on which you want the column to print. A value of zero will prevent the column from printing.
- **Start Position:** Enter the number of character spaces from the left to begin printing. The first character space is zero.
- **Enable Line Wrapping:** When printing, field data that are wider than the configured column width will either be truncated or wrapped to the next line depending on this setting.

> ***Tip!*** In version 9.22-Rev C, we have an enhanced configurator such that when you create a new Alarm Logger Configuration we will automatically generate default columns for you. You can then add or remove columns based on the default. This should help you minimize development time.

8.4. Node Configurations

Each **Node** represents a computer on the network that is configured to run one or more of the named logger configurations in the active configuration database. When the logger starts it looks for its network computer name in the list of nodes to determine which logger configuration(s) to run. If a logger cannot find its computer name in the list of nodes it will run the configuration named “default.” If no “default” configuration exists no configuration will be run.

If you want the local machine to run one or more non-default alarm logger configurations you can use "localhost" as the node name instead of specifying the actual node name.

> ***Tip!*** In version 9.22-Rev C, we have an enhanced configurator such that when you create a new Alarm Logger Configuration named anything other than “default” it will automatically link the configuration to the localhost node. This should help you to reduce development time further.

8.5. Workbench32

You can find the AlarmWorX32 Logger configuration in Workbench32 as well. The configuration in Workbench32 is similar to the standalone application. The configuration interface may look slightly different but all of the configuration options are there. You can use the Workbench32 to configure your Alarm Logger remotely making it easy to control your system off-site.

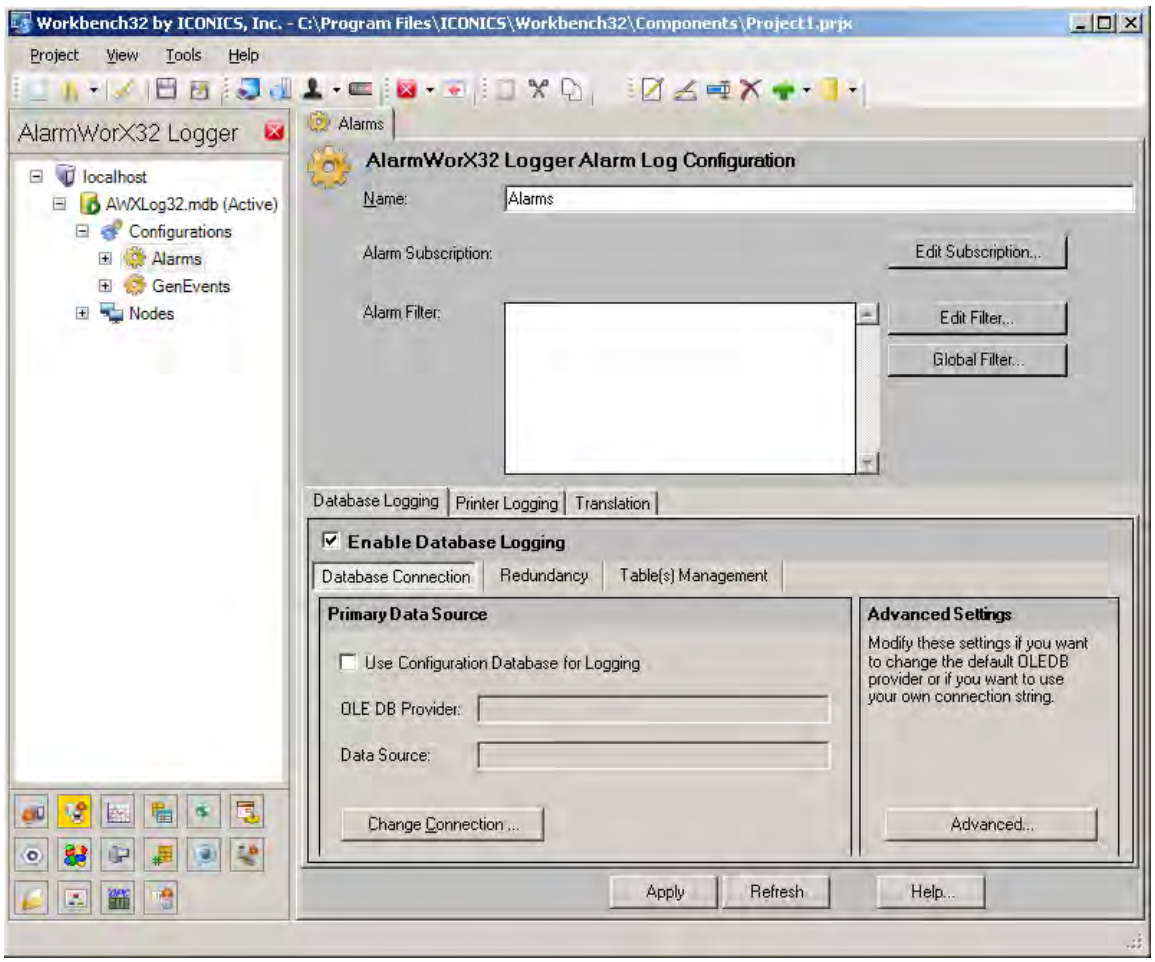


Figure 8.5 - Alarm Logger in Workbench32

A noteworthy difference between the standalone configurator and the Workbench32 configuration form is the way you create the database connection. In the standalone configuration the interface is the one provided by Data Source (ODBC) from your operating system. Since Workbench32 supports remote configuration we can no longer use the same interface.

To create a database connection in Workbench32, you would click on the "Change Connection…" button. This will launch the **Connection Strings** dialog. Listed in the dialog are the different types of ODBC data source links that we support. You can double-click on the appropriate folder to browse or create that type of data source.

Figure 8.6 - Connection Strings

In the following example we will choose to create a SQL Server Connection. Once you are in the folder you can click on the "Add" button that becomes available to add a connection. Depending on the type of connection you choose to make you will get a different interface. In the case of a SQL Server connection you will get the **Sql Connection String** dialog. The information required in this dialog is similar to that you encounter when configuring the data source in the standalone application.

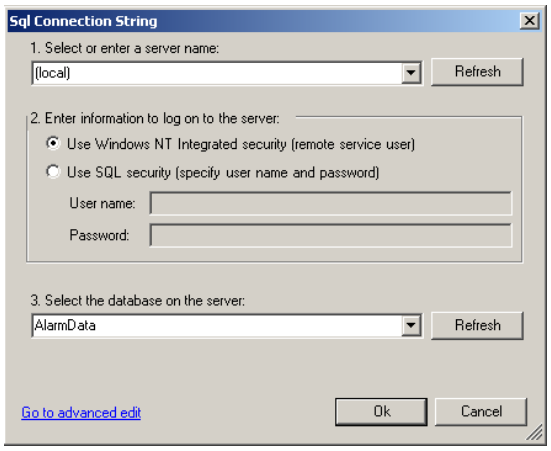


Figure 8.7 - SQL Connection String

Once you click on "OK" to complete the configuration you will see the newly configured connection in the **Connection Strings** dialog. You can change the name of the database connection as well as view the actual database connection. From here simply select the connection and click on "OK" to complete the configuration.

Lab: Maintaining History Alarms *L8*

In the previous labs, we configured alarm notification for Acmeseutical so that their operators can receive indication of alarms on the displays. Acmeseutical has requested that we keep a record of the alarms that occur so that they could analyze it later. In this lab, we are going to setup the AlarmWorX32 Logger to give Acmeseutical the historical alarm capabilities that they want.

Estimated time to complete this lab: 30 minutes

8.1. Task Checklist

In this lab, you will complete the following tasks:

- Create a new AlarmWorX32 Logger Configuration database
- Set up configurations to log alarms and events separately
- Define the different alarm attribute to record

8.2. Support Files

You will need the following file(s) to complete this lab:

- Acmeseutical_DWXConfig.mdb
- AcmeseuticalAlarmServer.mdb

8.3. Configuring the AlarmWorX32 Logger

In order to log real-time alarms to a database, we will need to configure the AlarmWorX32 Logger.

1. Start the Alarm Logger Configurator by going to Start → Programs → ICONICS GENESIS32 → AlarmWorX32 → Alarm Logger Configurator.

2. The first time you open this Configurator, the default demo configuration will load. We are going to create our own configuration.

3. Click on the "New" icon in the toolbar to launch the **ICONICS AlamrWorX32 Logger Configurator Database Wizard**.

4. Click on "Next >" in the welcome screen to proceed.

5. In the choose database type screen, select the MS Access database option and click on "Next >" to continue.

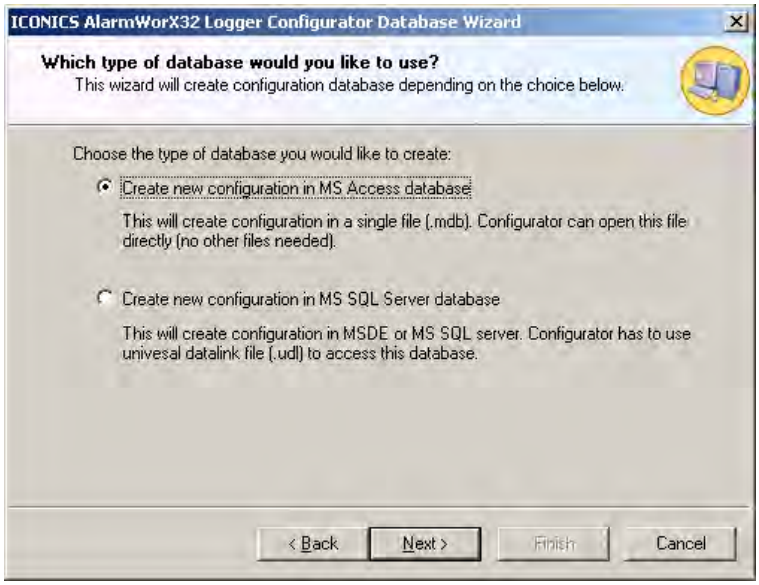


Figure L8.1 - Choose to Create a MS Access Configuration Database

6. Click on the ellipse button to bring up the File Browser. Change the directory to C:\Training and set the file name to "AcmeHistoryAlarmConfig".

7. Click on "Save" to close the File Browser and return to the Database Creation Wizard.

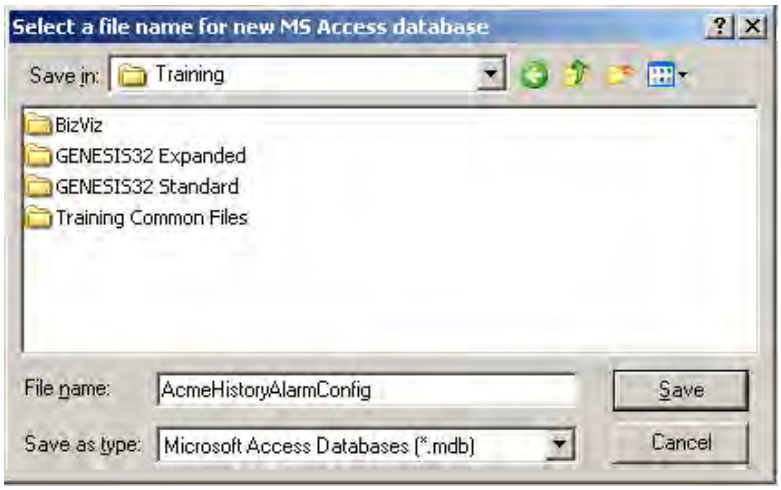


Figure L8.2 - Browsing a Location to Store the AcmeHistoryAlarmConfig Database

8. Your Database Creation Wizard should look like Figure L8.3. Click on “Finish” when you are ready. If the file name already exists, you may replace it.

Figure L8.3 - Creating the AcmeHistoryAlarmConfig Database

9. You should now see an empty AlarmWorX32 Logger configuration.

10. Right-click on “Configurations” in the tree control on the left-hand pane and select New → Configuration.

11. Name the configuration “Acmeseutical Alarms”.

12. Click on the “Edit Subscription” button to open the Subscription Properties dialog. This is the same dialog you saw when configuring the Alarm Viewer ActiveX control.

13. Click on “Add” to add a new subscription.

14. In the Event Subscription dialog that pops up, click on the “Browse” button to browse for the ICONICS AlarmWorX32 Server as shown in Figure L8.4.

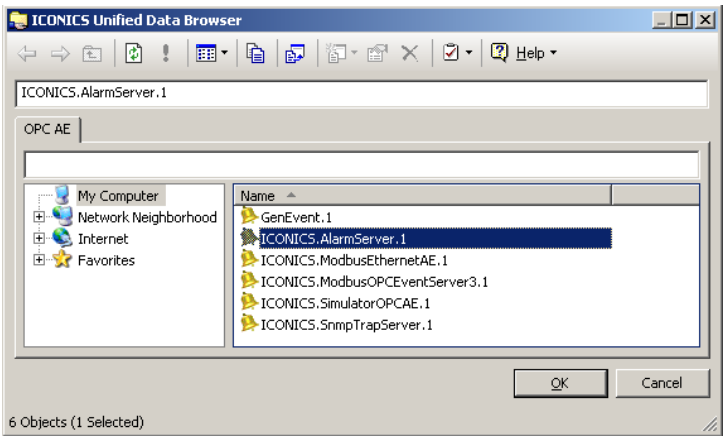


Figure L8.4 - ICONICS AlarmWorX32 Server

15. Click on “OK” to select the server and close the UDB.

16. With the “Acmeseutical Alarms” Subscription selected, click on the Edit button. Select the Attributes tab.

17. Add the following attributes to each of the Event Categories: Deviation, Digital, Limit, Rate Limit, and Rate of Change, making sure the attributes are in the same order:

    - CV
    - AREA
    - ACK COMMENT

18. Click on “OK” to close the Event Subscription dialog. In the Subscription Properties dialog, rename the subscription to “ICONICS Alarm Server”. Click on “OK” to close the dialog.

19. On the **Database Logging** Tab, select the **Database Connection** page.

20. Click on the ellipse button in the **Primary Data Source** section.

21. In the **Select Data Source** dialog, choose the **Machine Data Source** tab.

22. Click on “**New**” to create a new database link.

23. In the Create New Data Source dialog, choose the “**System Data Source**” option and click on “Next.”

24. Scroll to find the “**SQL Native Client**” driver. Select it and click on “**Next**.”

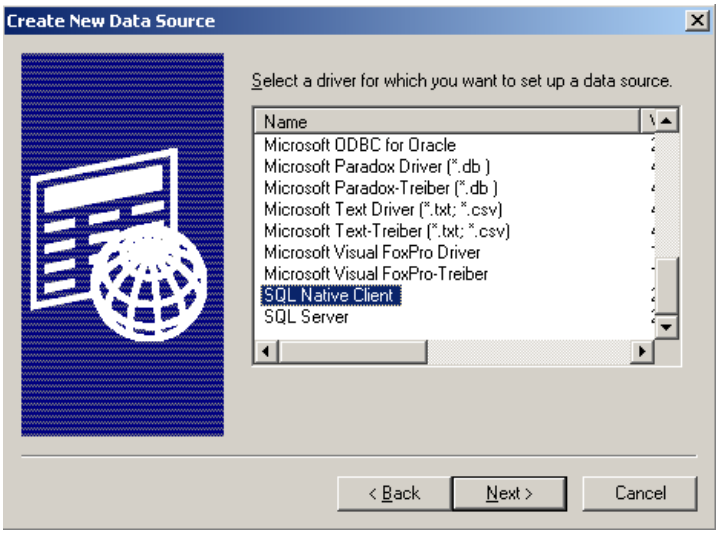


Figure L8.5 - Create New Data Source: SQL Native Client

25. Click on “Finish” and you will see the 'Create a New Data Source to SQL Server' dialog.

26. In the **Name** field, type in “**AcmeAlarmData**”. You can give it a description if you like (optional).

27. In the Server field, type in “(local)” or select a SQL Server you would like to connect to from the dropdown menu, then click on “Next >.” If you have MS-SQL2005-Express installed, then the server could be: “NodeName\SQLEXPRESS”.

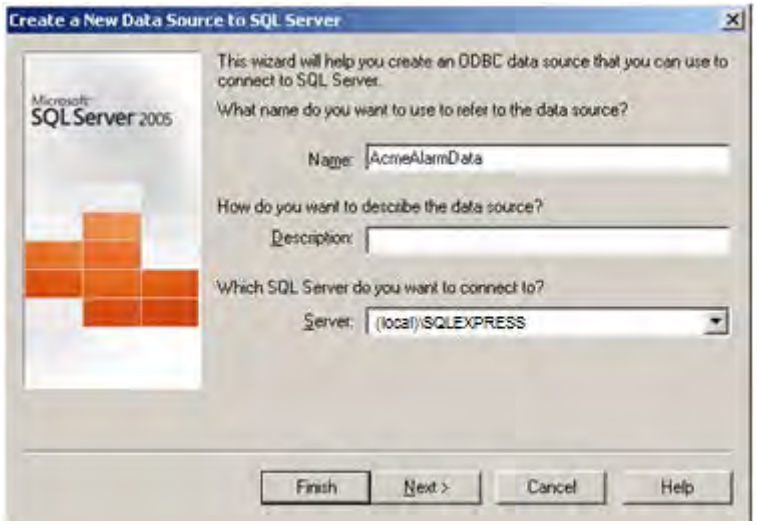


Figure L8.6 - Create a new Data Source to SQL Server

28. Leave everything default on this screen and click on “Next >.”

29. Check the “Change the default database to” option and select the “AcmeAlarmData” database in the dropdown list. Click on “Next >” to continue.

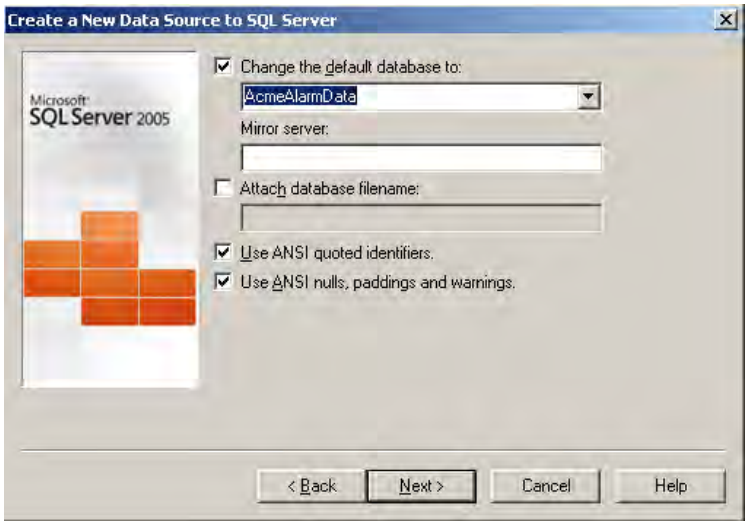


Figure L8.7 - Choose SQL Database

30. Leave this screen as default and click on “Finish.”

31. In the **ODBC Microsoft SQL Server Setup** dialog, click on the “Test Data Source” button and make sure that the test completes successfully.

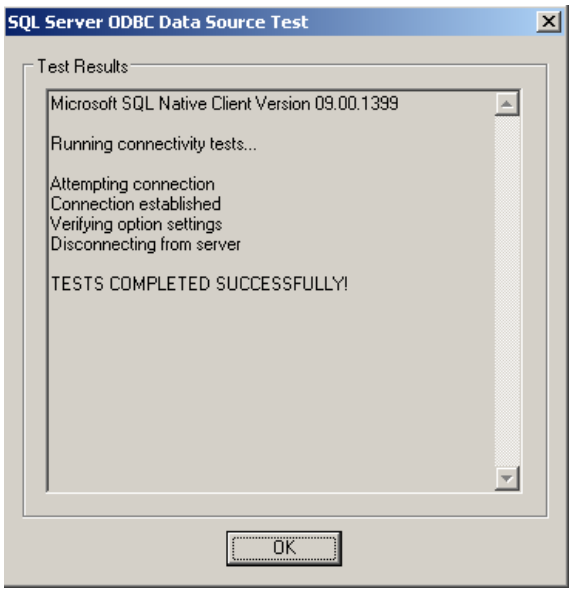


Figure L8.8 - SQL Server ODBC Data Source Test Results

32. Click on “OK” to close all dialogs. Once you are back in the AlarmWorX32 Logger Configurator, your Database Logging tab should look like Figure L8.9.

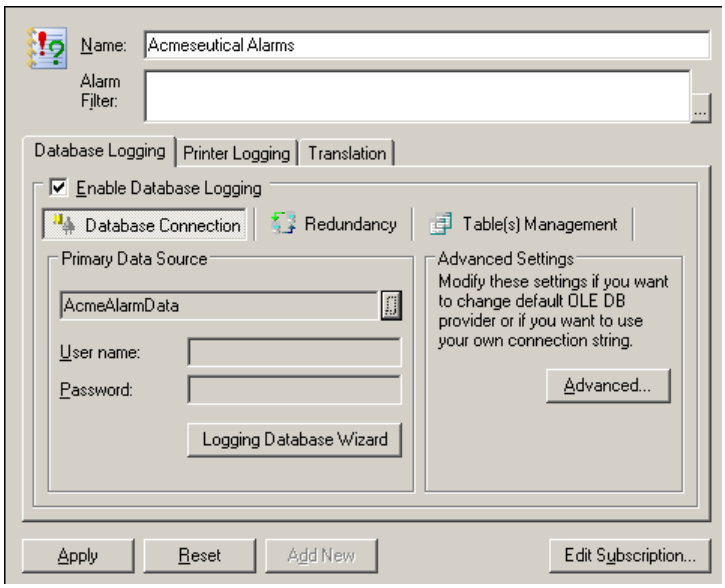


Figure L8.9 - Alarm Database Logging: Database Connection

33. Click on the “Table(s) Management” page and change the Table Name to “AlarmLog”.

34. In the **Active Table** section, check the "**Max Interval**" option and set the interval to one **(1) Month**.

35. In the **Archiving** section, check the "**Max Interval**" option and leave it as one **(1) year.**

36. Also in the **Archiving** section check the "**Max Table**" option. Set the value to **7.** The archiving system will keep seven (7) tables. Your finished configuration form should look similar to Figure L8.10.

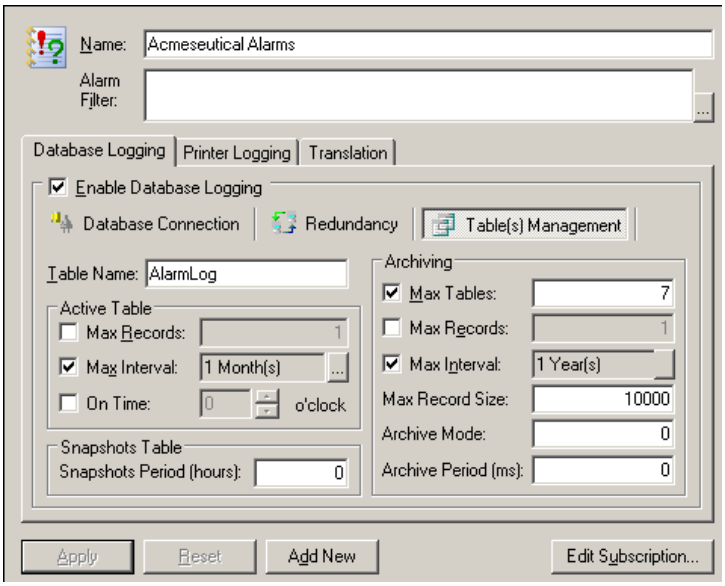


Figure L8.10 - Alarm Database Logging: Table(s) Management

37. Click on "Apply" at the bottom to save the changes.

38. The configuration adds Column Definition for you by default when you create the configuration. We just need to go through and remove the ones that we do not need.

39. Select "Acmeseutical Alarms" in the tree control on the left-hand pane. Hold down the control key to select the following columns on the upper right-hand pane:

- Acked
- ChangeMask
- Comment
- ConditionActive
- Event Category
- EventTimeUTC
- EventType
- LimitExceeded
- NameOfParameter
- NewValue
- PreviousValue
- ServerNode

40. Hit the "Delete" key when you have the columns selected. A message box will pop up, asking you to confirm the deletion. Click on "Yes" to continue.

41. Select “CV” in the tree control on the left-hand pane and change the name of this Column Attribute to “CurrentValue”.

42. Check that the Type is set to “Attribute 1”. Click on “Apply” to save the changes.

43. Select “Area” in the tree control and change the type to “Attribute 2”. Click on “Apply” to save the changes.

44. We are done configuring the Acmeseutical Alarms. Right-click on “Acmeseutical Alarms” in the Tree Explorer and select “Copy”.

45. Right-click on Configurations and select “Paste”.

46. Double click on the “Copy_of_Acmeseutical Alarms” configuration to show the configuration form in the right-hand pane.

47. Rename the configuration to “Acmeseutical Events”.

48. Click on the “Edit Subscription” button. Rename subscription to “ICONICS GenEvent Server”.

49. From the Subscription Properties dialog click on “Edit” to bring up the **Event Subscription** dialog.

50. Click on “Browse” to bring up the UDB and browse for the GenEvent server as shown in Figure L8.11.

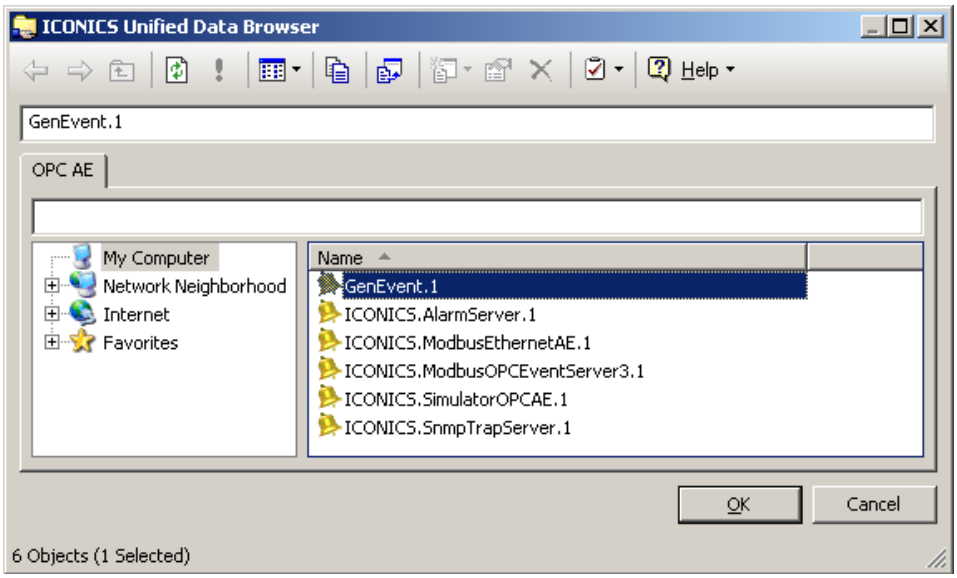


Figure L8.11 - GenEvent Server

51. Click on “OK” until you are back in the AlarmWorX32 Logger Configurator.

52. Select the Table(s) Management page and change the Table name to “EventLog”.

53. Click on “Apply” to save the settings.

54. In the tree control on the left-hand side, expand Nodes → localhost.

55. You should see that your “Acmeseutical Alarms” configuration is already linked.

56. Right-click on localhost under Nodes and select New → Link to Configuration.

57. In the configuration form that comes up on the right-hand pane, choose “Acmeseutical Events” in the dropdown list. Click on “Apply” the save the changes.

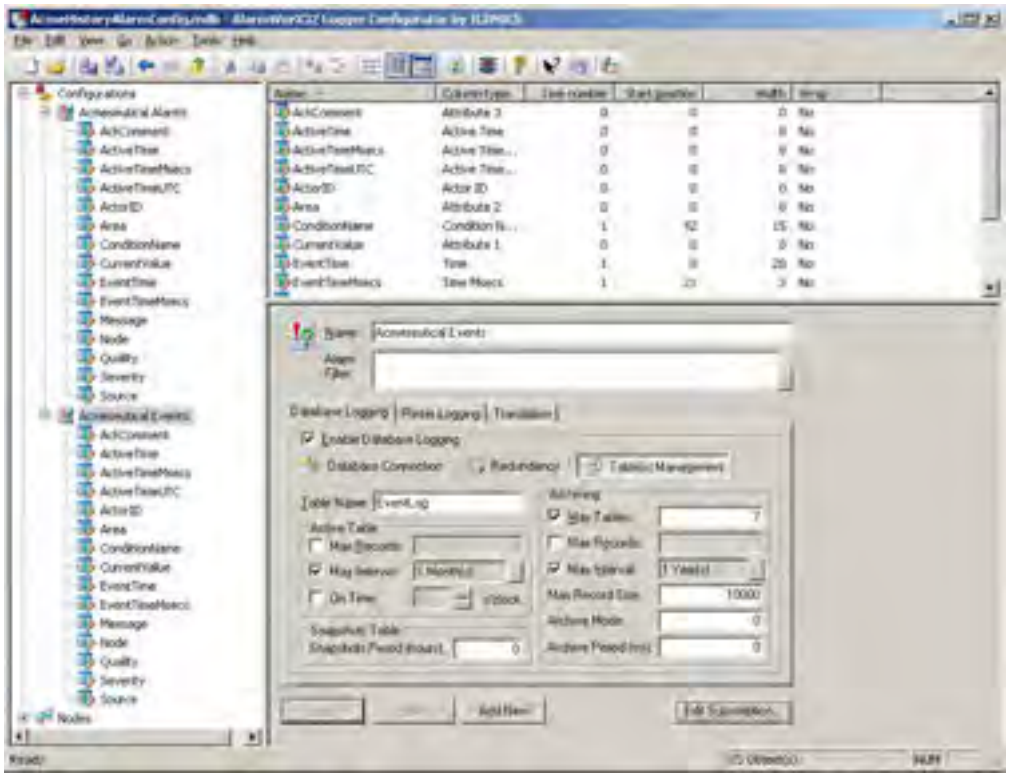


Figure L8.12 - AlarmWorX32 Logger Configuration Complete

58. Go to File → Make Active to activate the configuration database. If the “Make Active” selection is grayed out, your configuration is already active.

    NOTE: If “Make Active” is grayed-out in the File menu, it means the configuration database is already active.

59. We are done with configuring the alarm logger now. Click on the “Start/Stop Logger” icon in the toolbar to start the AlarmWorX32 Logger.

Review: AlarmWorX32 Logger *R8*

8.1. Chapter Objectives

In this chapter, we discussed using the AlarmWorX32 Logger to keep a record of the alarm notifications generated by the system. Using the logger, you can track alarm events in a database as well as create a paper trail by printing them in real-time. By now, you should know how to do the following:

- Create a new AlarmWorX32 Logger Configuration Database
- Create a database connection to the logging database
- Subscribe to an alarm server
- Add filters
- Configure alarm definitions
- Link configurations to nodes

8.2. Summary

The AlarmWorX32 Logger gives you the ability to record alarms and events from an OPC A&E compliant server. You can log the alarms to a MS Access, MS SQL, mySQL, and Oracle database for later analysis. Alternatively, you can print the alarms to create a paper trail of the alarm notifications.

You can create multiple configurations and run them at the same time. This allows you to organize your alarm information into different databases. On the alarm configuration level, you would make the connection to the destination database. The connection to this database is via an ODBC data source. Once you create a connection to the database, you can further organize how the alarms are logged by specifying table management options. From these options, you can configure the size of the active logging table as well as archive tables.

Within each configuration are column definitions. Each column definition defines an alarm attribute that you wish to log and creates a column in the database table. When you create configuration, default column definitions are automatically inserted, from which, you can add or remove definitions.

While there is a standalone AlarmWorX32 Logger configuration, you can also use the Workbench to configure it. The configuration interfaces are similar, but the Workbench32 has the added benefit of allowing remote configuration.

8.3. Terms

AlarmWorX32 Logger
Database Connection
Database Logging
ODBC Data Source
Printer Logging

8.4. Test Your Knowledge

In this chapter, we discussed how to log alarm notification into a database using AlarmWorX32 Logger. You should be able to answer the following questions now.

1. What type of database can you use to store history alarm information?

    a. MS SQL 2000 and later
    b. mySQL
    c. Oracle
    d. MS Access
    e. All of the Above
    f. Only A and D
    g. Only B and C
    h. None of the Above

2. What type of database can you use the store the alarm logging configuration?

    a. MS SQL 2000 and later
    b. mySQL
    c. Oracle
    d. MS Access
    e. All of the Above
    f. Only A and D
    g. Only B and C
    h. None of the Above

3. How do you add an alarm attribute to log in the configuration?

    a. Define a new column
    b. Define a new attribute
    c. Define a new configuration
    d. Define a new ODBC Data Source

4. True or False: You can add server- and client-side filter to an alarm logging configuration.

5. True or False: The alarm logger supports logging to two separate databases using redundancy.

6. True or False: You need to create a subscription in an alarm logging configuration to specify the server that generates the alarm notification.

7. True or False: You do not need to link the configuration name "default" to any node.

8. How many logging configurations can you run at a time?

   a. 1
   b. 32
   c. 64
   d. Unlimited

For questions 9 and 10, please refer to Figure R8.1.

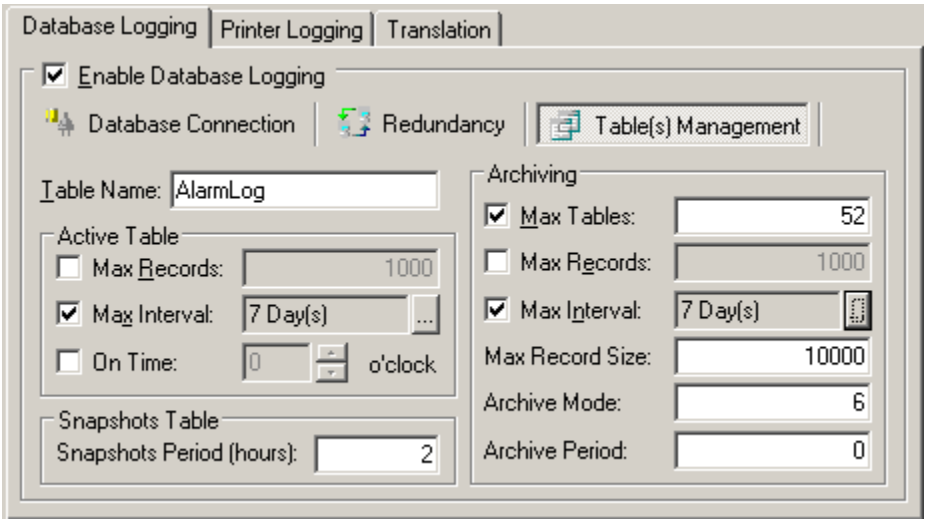


Figure R8.1 - Table(s) Management

9. What happens on the eighth day in the configuration show in Figure R8.1?

   a. A new table will be created on the 8th day to store the new alarms.
   b. The first day's data will be archived and removed from the Active table; the new data will continue to log into the Active table.
   c. All data in the Active table will be archived and removed from the active table; the new data will log into the Active table.
   d. No data will be logged on the 8th day since it's only configured to log seven days worth of data.

10. In the configuration shown in Figure R8.1, what is the maximum amount of data that you would have?

    a. One year
    b. One Month
    c. One Week (7 days)
    d. One day

AlarmWorX32 Reporter 9

In the previous chapter, we demonstrated the AlarmWorX32 Logger which allowed for logging alarm notification into a database and/or create a paper trail of what happened by printing the alarms. In this chapter, we will look at how to view the historic alarms logged into a database in the HMI system.

By the end of the chapter, you should be familiar with the following concepts:

- The Alarm Reporter ActiveX Control
- General configuration settings for the control
- How to connect to the database with alarm information
- Sorting the records in the control
- Runtime options

9.1. Alarm Report ActiveX

Alarms logged to a database can be a difficult task to sort through and analyze. The powerful Historical Alarm Analysis (Report) ActiveX solves this problem by easily creating filtered reports, pie charts, bar plots, and even exporting to custom Crystal Reports. At the click of a button you can find out which alarm is occurring most often, see if there are certain "trouble spots," review downtime, and more.

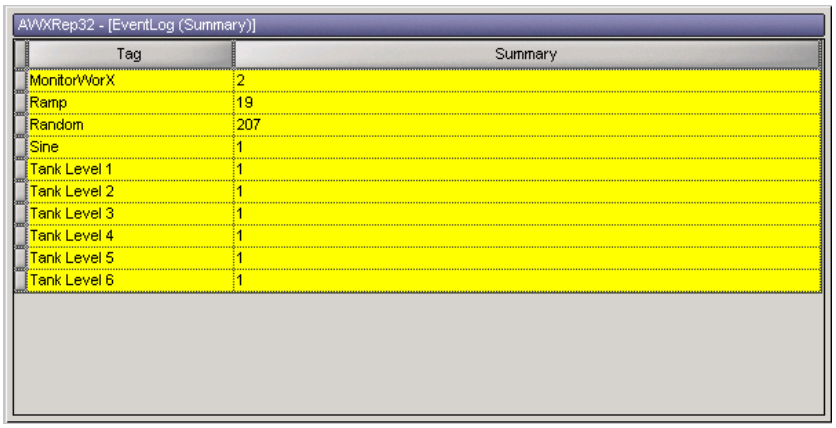


Figure 9.1 - Alarm Report ActiveX

The **Alarm Report ActiveX**, shown in Figure 9.1, allows reporting (user-configured or preconfigured) and graphing of alarms. The data source of the tracked alarms is alarms previously logged by the Alarm Logger.

Because the Alarm Report is an ActiveX control you can drop it into the AlarmWorX32 Container within any GraphWorX32 HMI display, an HTML Internet/intranet-based Web page, or any other ActiveX container.

9.2. Alarm Report Properties

Select the ICONICS AlarmWorX32 Reporter ActiveX from the ActiveX toolbar and place the control in a GraphWorX32 display. Double-clicking on the ActiveX control opens the Alarm Report ActiveX Properties dialog as shown Figure 9.2. Use these property tabs to configure how alarms will be reported (appearance, organization, etc.) and which report type will be used.

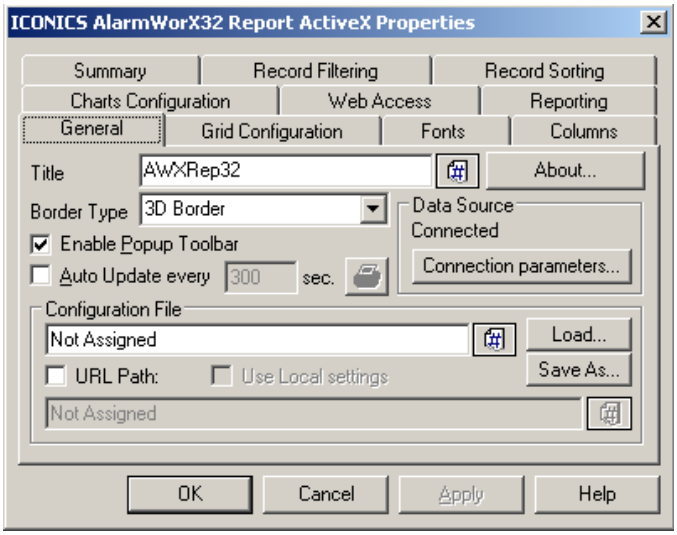


Figure 9.2 - Alarm Report ActiveX Properties

NOTE: Right-clicking on the Report ActiveX and selecting Properties also opens the Alarm Report ActiveX Properties dialog box.

The ActiveX Properties dialog contains the following configuration tabs:

- General
- Grid Configuration
- Fonts
- Columns
- Charts Configuration
- Web Access
- Reporting
- Summary
- Record Filtering
- Records Sorting

Of these configuration tabs this class will work closely with the **General**, **Columns**, and **Record Sorting** tabs.

Other configuration tabs such as the **Charts Configuration**, **Reporting**, and **Summary** tabs, have default configurations that allow you to use the features they apply to with no configuration required (although changes can be made as desired).

9.3. General Settings and Database Connection

The **General** tab of the Alarm Report ActiveX Properties dialog box shown in Figure 9.2, allows you to specify the name of the report file, select the border type and connect to a data source.

Some other important settings on this page are:

- **Enable Popup Toolbar:** When this option is checked the Report Runtime toolbar is available during runtime mode.
- **Auto Update:** Specifies the frequency (in seconds) at which the Alarm Report updates data from the Alarm Logger database.

The most important option on this configuration tab is the **Data Source Connection Parameters**. You must provide a valid data source connection in order for the ActiveX control to display data. Clicking the "Connection Parameters…" button opens the **Database Connection** dialog box shown in Figure 9.3. This defines the database connection where the alarm logger database will pull historical alarms from.

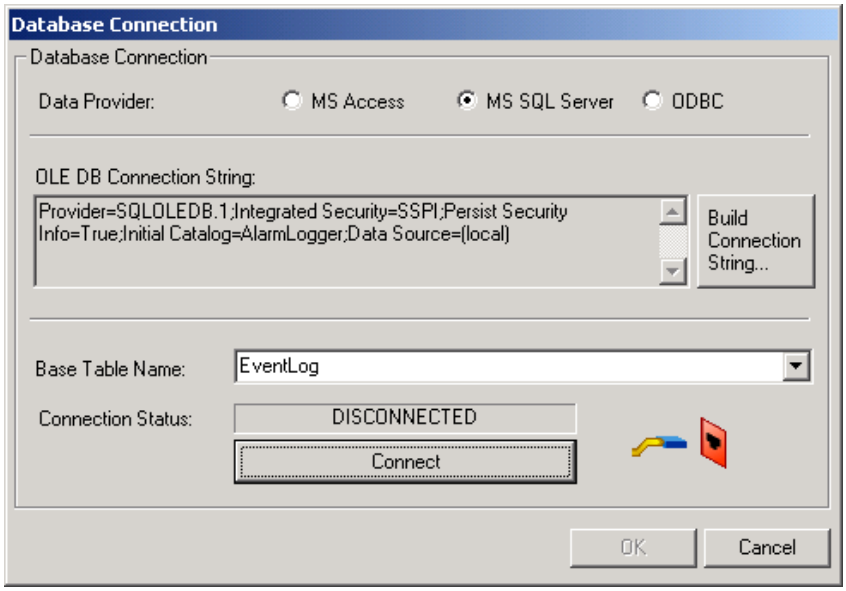


Figure 9.3 - Database Connection Properties

All data source connections are made through the **Data Link Properties** dialog. The **Connection** tab settings may vary depending on the selected data provider. The **Data Link Properties** dialog for a Microsoft SQL Server data provider is shown in Figure 9.4. Simply provide the MS SQL server name, an authentication method, and the name of the database to create a connection.

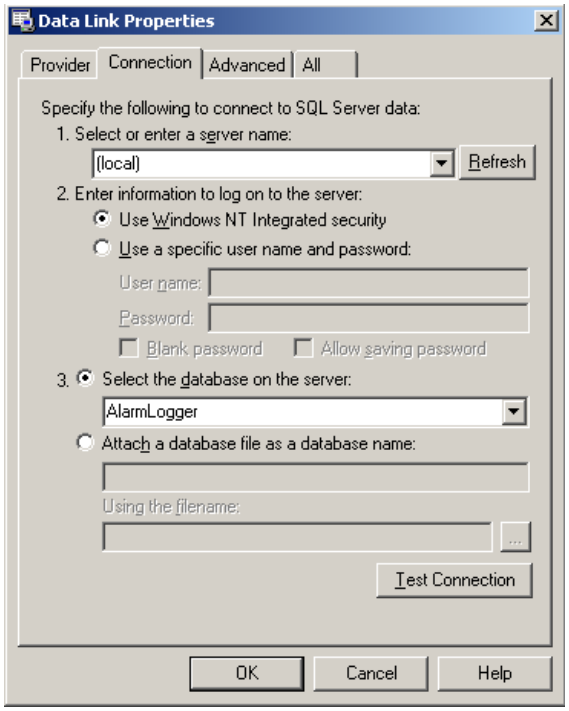


Figure 9.4 - Data Link Properties

> ***Tip!*** If the MS SQL server you are trying to connect to is the default instance, then you can use (local) as the server name. If you installed SQL 2005 Express the name of the SQL instance is by default *MyComputer*\SQLExpress where *MyComputer* is the name of the computer on which you installed the SQL 2005 Express Server.

Once you are done configuring the Data Link Properties, click on the "Test Connection" button to make sure that it can indeed connect to the specified database. Click on OK to close the dialog and save the configuration. In the Database Connection dialog choose the Base Table or the table you configured as the Active table in the Alarm Logger. Before you close the Database Connection, click on the Connect button. You should notice that the icon in the lower right-hand corner changed from a red 'disconnected' one to a green 'connected' icon. This indicates that the Alarm Reporter ActiveX control is now connected to a database and is ready to display data.

9.4. Column Configuration

In the **Columns** tab of the Alarm Report ActiveX Properties dialog, shown in Figure 9.5, you can specify which columns will appear in your alarm report. By default the following three items are included in the report: EventTime, Tag, and Message, as shown in the **Selected** list in Figure 9.5.

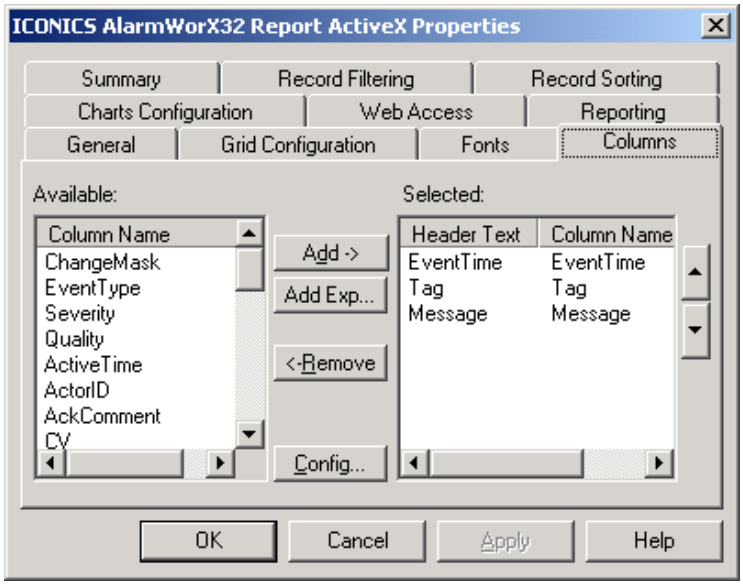


Figure 9.5 - Column Configuration

By clicking the "Add" and "Remove" buttons in the middle you can select which columns to include in the report. The up and down arrows on the right side of the dialog affect the left to right order of their appearance in the row of your report.

You can also change the appearance of a column by either double-clicking on an item in the **Selected** list or selecting the item and clicking the "Config…" button. The **Set Column** configuration dialog allows you to change the font, color, width, and cell properties by selecting them and changing the various options available.

9.5. Records Sorting

The **Records Sorting** tab of the Alarm Report ActiveX Properties dialog has just one button: the "Sort By…" button. Clicking the "Sort By…" button opens the **Sorting Fields** dialog as shown in Figure 9.6. You can choose whether to disable or enable sorting. If you choose to enable sorting you will specify the priority by which the fields in the alarm report are sorted. Records can be sorted based upon any column of information included in the report.

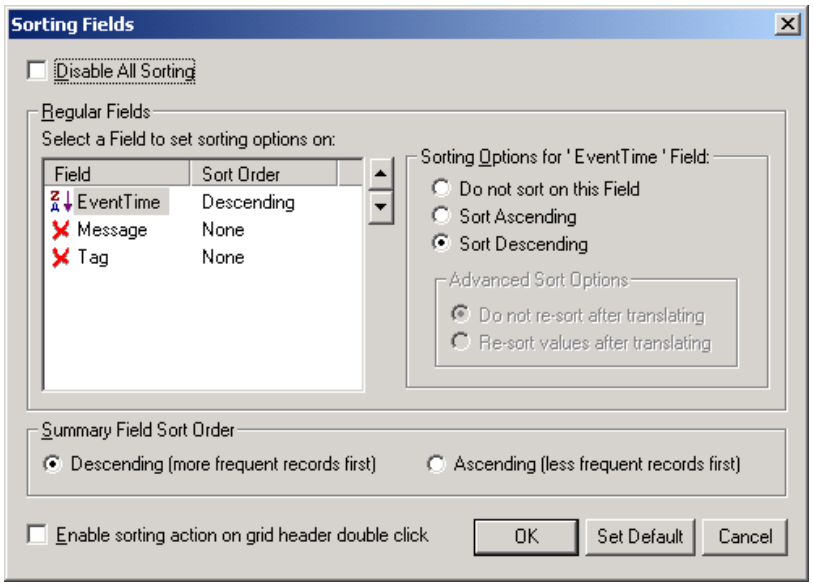


Figure 9.6 - Sorting Fields

All elements will be sorted by the first category in the list. In the event that two elements in the first category are identical, the reported alarms are sorted by the criteria configured for the next item in the list.

9.6. Alarm Report Runtime Options

Double-clicking anywhere on the Alarm Report ActiveX during runtime mode opens the Alarm Report **Runtime Toolbar** as shown in Figure 9.7. Using the toolbar you can execute certain operations and change how the data is viewed.

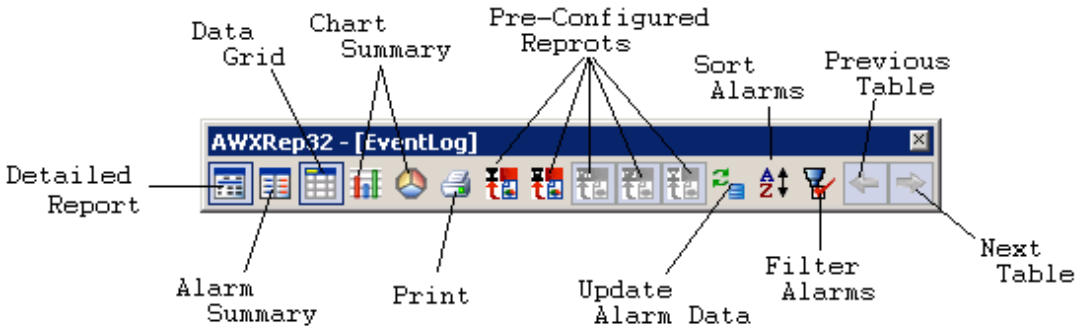


Figure 9.7 - Runtime Toolbar

There are two different views to choose from: **Data Grid** or **Chart Summary**. When viewing the Data Grid you can further choose from a detailed report or a summary report.

The **Summary** report provides a summary based on any column (configurable in the Configure mode Properties dialog), and the frequency of occurrence of items in that column.

To view a **Bar Chart** or **Pie Chart** summary of the Alarm Report data during runtime mode, choose from the **Chart Summary** buttons on the Runtime toolbar. The number of items charted depends on the options selected on the **Records Filtering** tab of the Configure mode Properties dialog. If no filters are enabled, all alarm types are charted.

Notice that sorting and filtering can be configured during Runtime just as they are configured during Configure mode.

Lab: Viewing History Alarms *L9*

In the previous lab, we set up logging to retain records of the real-time alarms for Acmeseutical. The shift managers at Acmeseutical would like to be able to review these alarms graphically. In this lab, we will build a screen for them to view the logged alarms.

Estimated time to complete this lab: 30 minutes

9.1. Task Checklist

In this lab, you will complete the following tasks:

- Create a display to view historical alarms
- Connect the Alarm Reporter ActiveX control to a database
- Configure Columns in the control
- View Crystal Reports

9.2. Support Files

You will need the following file(s) to complete this lab:

- AcmeseuticalTemplate.tdf
- ACMEPillMachine.gdf
- ACMEMixers.gdf
- AcmeseuticalAlarms.gdf

9.3. Configuring the Alarm Reporter

In order to allow shift managers to view history alarms, we will build a display that contains the information.

1. Through Windows Explorer, locate AcmeseuticalTemplate.tdf in the training folder and double-click on it to create a new display.

2. Save the display and name it “ACMEHistoryAlarms.gdf”. If the file name already exists, you may replace it.

3. Insert the Alarm Reporter ActiveX control onto the display by clicking on the "ICONICS AlarmWorX32 Reporter ActiveX" icon in the ActiveX toolbar.

4. Resize the control so that it takes up most of the screen.

5. Double-click on the control to bring up the properties dialog.

6. Click on the "Connection Parameters" button to bring up the Database Connection dialog.

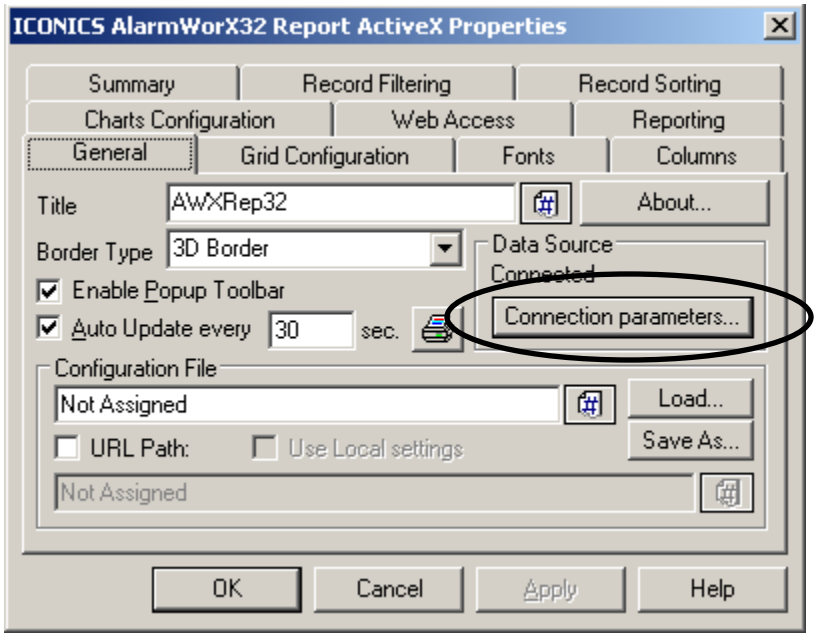


Figure L9.1 - AlarmWorX32 Reporter Properties: General

7. Make sure that the Data Provider is selected to use "MS SQL Server".

8. Click on the "**Build Connection String**" button. This brings up the Data Link Properties dialog.

9. From the Connection tab, in the Server Name field (field number 1.), type in "(local)" or select a SQL Server from the dropdown menu that you would like to connect to and click on "Next >." If you have MS-SQL2005-Express installed, then the server could be :"NodeName\SQLEXPRESS".

10. Choose the "Use Windows NT Integrated security" option.

11. In the "Select the database on the server" dropdown list, pick "**AcmeAlarmData**". The Data Link Properties dialog should look similar to Figure L9.2.

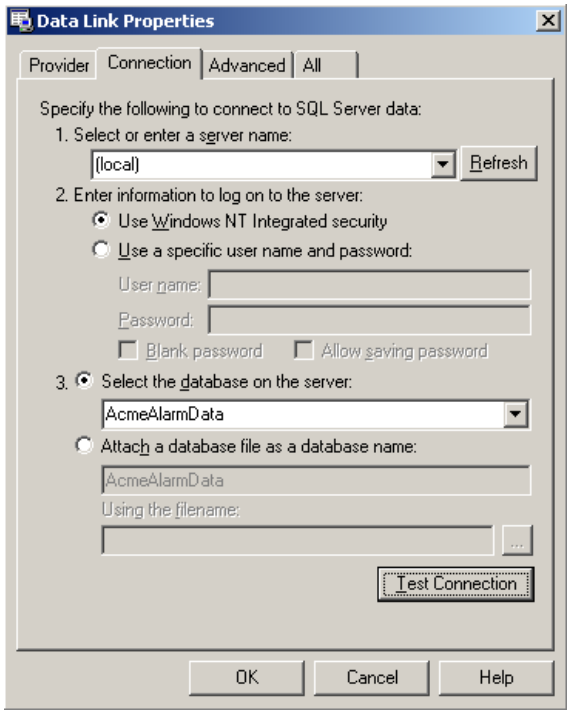


Figure L9.2 - Data Link Properties

12. Click on the "**Test Connection**" button to make sure that it can connect to the database successfully.

13. Click on "**OK**" to save the changes and close the Data Link Properties dialog.

14. In the Base Table Name dropdown list, select the "AlarmLog".

15. Click on the "Connect" button. Notice that the Connection Status changes to "Connected" and the icon to the right also changes to indicate the status.

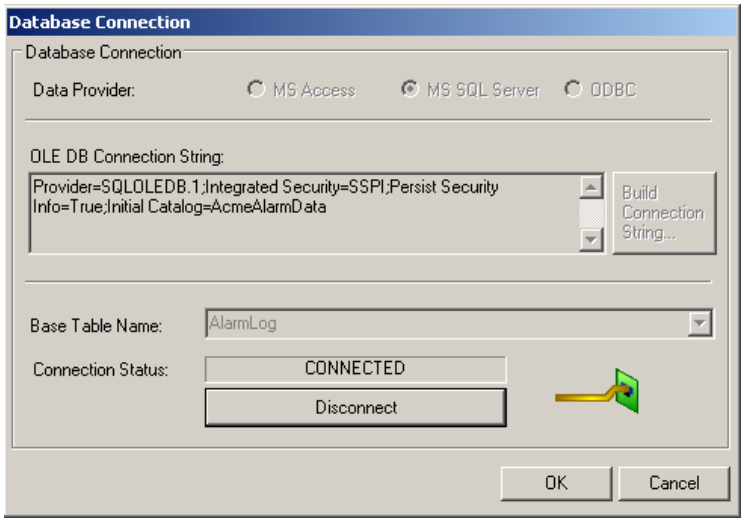


Figure L9.3 - Alarm Reporter: Database Connection

16. Click on “OK” to close the Database Connection dialog.

17. Check the “**Auto Update Every**” option and set it to update every **60** seconds.

18. Select the “**Columns**” tab.

19. Select the **Area** and **CurrentValue** in the Available list and click on the “Add” button to add these columns to the Selected list.

20. Move the Area up in the list by selecting it and clicking on the up arrow to the right of the list. The order of the columns should look similar to Figure L9.4.

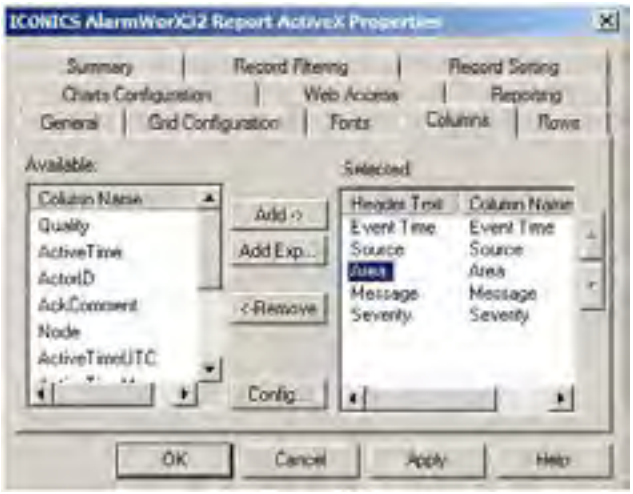


Figure L9.4 - Alarm Reporter: Columns Tab

21. Click on “OK” to close the properties dialog and return to the display.

22. Save the display.

9.3.1. Viewing the Report in Runtime

1. Click on the Runtime menu to start runtime mode and show the logged alarms, similar to what is depicted in Figure L9.5.

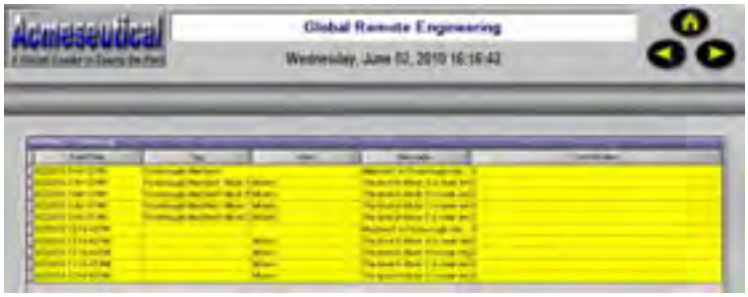


Figure L9.5 - Alarm Reporter in Runtime Mode

2. Double-click on the Alarm Report ActiveX control to open the Runtime toolbar as shown in Figure L9.6.

Figure L9.6 - Alarm Report Runtime Toolbar

3. Note that you can configure record filtering and sorting during Runtime with the "Filter Alarms" and "Sort Alarms" icon respectively. The "Update Alarm Data" icon will force an update of the latest records in the database.

4. Switch between the Detailed Report view and the Alarm Summary view by clicking on the "Detailed Report" icon and the "Alarm Summary" icon. The Alarm Summary view can be set to summarize by whatever column you prefer in the ActiveX properties window in Configure mode.

5. You can also view your data in the form of a bar chart or pie chart by clicking on the respective Chart Summary icons.

6. The five report buttons in the middle of the toolbar will execute a Crystal Report on the data you are viewing. These reports are configurable when the ActiveX is in Configure mode and up to five can be configured an enabled at a time. Two are configured for you by default. Feel free to click on either one and see the report.

7. After exploring the Runtime options of the Alarm Report ActiveX make sure that you save your changes (if desired) and close the display.

9.4. Associating the Alarm Report Display

1. Open the **ACMEPillMachine.gdf** display.

2. Add a layer to the display by clicking on the Format menu and selecting Layers → **Add Layer**.

3. In the Edit Layer Properties dialog, change the Layer name to "**Secured**" and click on "OK" to close the dialog.

4. You should be on Layer 1 now. You can check by making sure that **1** is highlighted on your Layers toolbar.

5. Add a button to this layer.

6. Leave the Action on the Pick tab to "**Load Display**" and type in "**ACMEHistoryAlarms.gdf**" in the Filename field.

7. Click on "**OK**" to save the settings and close the dialog.

8. Set the button text to "**History Alarms**" by holding down the **Shift** key and **Right Clicking** on the button.

9. Save the display and go into runtime to make sure that the button works. We will add security in a later chapter to limit the users who are allowed to click on this button.

Review: AlarmWorX32 Reporter

R9

9.1. Chapter Objectives

In this chapter, we discussed how to view your logged alarm notifications. Using the Alarm Reporter ActiveX control, you can sort through and analyze your alarms with little effort. By now, you should know how to do the following in the Alarm Report ActiveX control:

- Connect to the logging database
- Sort records in the control
- Add a column in the control
- View Bar Chart and Pie Charts

9.2. Summary

The Alarm Reporter ActiveX control allows you to view historic alarms. You can also use it to analyze alarms by generating a Bar Chart, a Pie Chart, or Crystal Reports. It is simple to configure the control. In the properties of the control, you can make a connection to the database that contains the alarm information. Once you make this connection, the Columns tab will automatically show the available columns or alarm attributes in the database. You can then choose which column to show in the Columns tab. You can further sort the alarms based on the different columns you select to show.

During runtime, there are many options available to control the appearance of the ActiveX. In the Runtime Toolbar, you can select to show the data in a Data Grid (default), a Bar Chart, or a Pie Chart. There are also preconfigured Crystal Reports available in the Runtime Toolbar, you can click on the buttons on the toolbar to generate the report and print it.

9.3. Terms

AlarmWorX32 Reporter Data Link Properties Database Connection
Runtime Toolbar

9.4. Test Your Knowledge

In this chapter, we discussed how to view logged alarm information. You should be able to answer the following questions by now.

1. Which of the following is NOT one of the configuration tabs available in the Alarm Reporter ActiveX?

    a. Columns
    b. Rows
    c. Record Sorting
    d. Record Filtering

2. True or False: You can only sort on one column in the Alarm Reporter ActiveX.

3. What do you need to configure in the Database Connection dialog?

    a. A connection to the database that house the alarm information
    b. A database table that house the alarm information
    c. The different columns that you need to retrieve from the database
    d. All of the above
    e. Only A and B
    f. Only B and C
    g. Only A and C
    h. None of the above

4. True or False: You must add the columns to the Selected side in the order that you want it to appear in; you cannot change the order after you have added the columns.

5. True or False: You cannot add your own Crystal Report configurations.

6. Which of the following is not an option in the Runtime Toolbar?

    a. Bar Chart
    b. Pie Chart
    c. Area Chart
    d. Data Grid

7. True or False: You can define how often the Alarm Reporter ActiveX queries the database for an update of the alarm information.

TrendWorX32 Logger 10

It is often necessary to log OPC tag values to create a trend and monitor the performance of the different assets. The TrendWorX32 suite of products provides just that functionality as well as reporting capability for management to analyze the logged data.

TrendWorX32 is a powerful collection of real-time trending, historical data logging, reporting, and analysis tools that seamlessly integrate with enterprise-wide information systems. Based on the OPC Historical Data Access specification for creating Plug and Play historical data servers and clients, TrendWorX32 offers an open solution to applications requiring scalable and distributed real-time performance.

The powerful Microsoft-based ADO/OLEDB data logging provider is at the core of TrendWorX32. It integrates with Microsoft Access, Microsoft SQL Server, mySQL, and Oracle using ADO and OLEDB database technologies. OPC HDA provides the standard COM and OLE interface for the ICONICS Trend Viewer ActiveX Control to display real-time and historical data, separately or simultaneously.

Several trend display types are supported, including time plots, XY plots, logarithmic plots, bar plots, the popular strip chart recorder, and even circular charts! Acquire thousands of data points and organize them into groups for very fast and efficient replay of historical and real-time information. You can use the built-in Visual Basic for Applications to create reports, calculations, and data analysis.

In this chapter, we are going to focus on the TrendWorX32 Logger. By the end of the chapter, you will be familiar with the following concepts:

- The TrendWorX32 Logger Configurator
- Basic configuration settings
- Database groups and configuring database connections
- Databases supported by the Trend Logger
- Logging groups configuration
- Logging tags configuration

10.1. TrendWorX32 Logger System Architecture

The primary purpose of TrendWorX32 data logging is to provide both a data-collection system and a data-retrieval server to supply historical data to trending and reporting clients. The TrendWorX32 SQL Logger does the following:

- Collects real-time data from OPC servers
- Logs real-time data to a database
- Provides data-retrieval capability by implementing an OPC HDA (Historical Data Access) Server

The diagram in Figure 10.1 shows the internal system architecture of the TrendWorX32 Logger. While you will not configure the TrendWorX32 HDA Server, as it is usually transparent to the end user, you should know that it is available. In case you have other OPC HDA compliant clients, you can retrieve historical data using the HDA server.

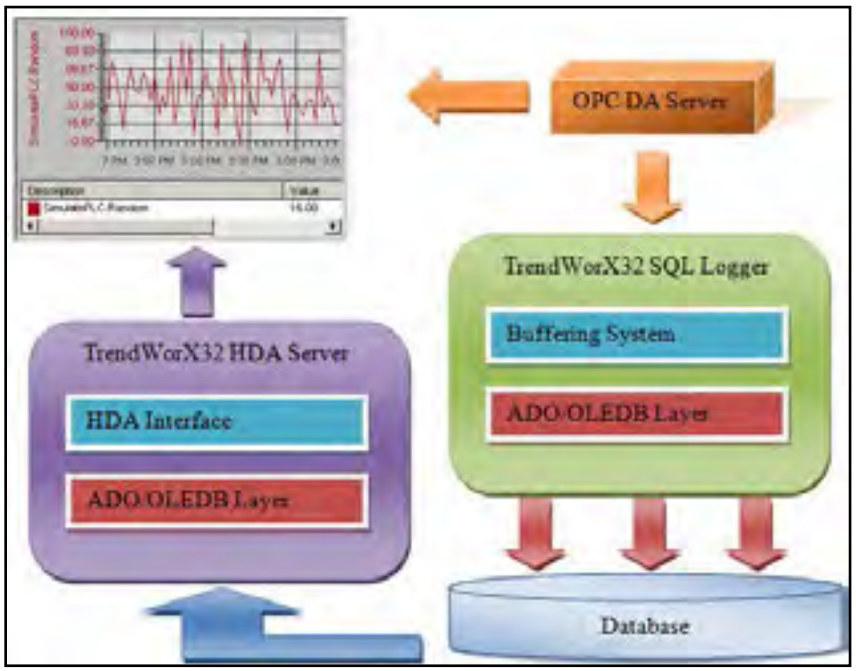


Figure 10.1 - Trending and Data Logging Architecture

The TrendWorX32 SQL Logger also adds scalable client-server architecture to the GENESIS32 trending software on top of providing data logging capabilities. This architecture ensures optimal load balancing and distribution even for large-scale applications.

The TrendWorX32 SQL Logger uses the data-access technology called ADO (ActiveX Data Objects). ADO is a database-access mechanism that provides a wrapper for OLE DB.

OLE DB is a COM approach to database programming that can be used to access any database (relational or non-relational) as long as there is a dedicated OLE DB data provider for that database. The result is that applications based on ADO/OLE DB can potentially access any database regardless of the underlying storage media, file system, and location. TrendWorX32 SQL Logger currently uses ADO, the Universal OLE DB ODBC data provider supplied by Microsoft as well as native OLE DB providers to access Microsoft Access, Microsoft SQL Server, and Oracle-compliant databases.

TrendWorX32 SQL Logger can be deployed using the same computer on which the database resides or using a different networked computer. TrendWorX32 SQL Logger deploys on multiple machines data logging to the same (or different) databases providing scalable and distributable data logging architecture.

10.2. TrendWorX32 Logger Configurator

The **TrendWorX32 Configurator** shown in Figure 10.2 is used to create and maintain the tag database used by TrendWorX32 SQL Logger. Its main function is to perform configuration for data logging. In addition to simple tag configuration, the Configurator is used to create and maintain a process-area hierarchy and to group tags into one or more named configurations to support servers running on multiple network nodes.

Figure 10.2 - TrendWorX32 Configurator

In keeping with the open standards employed throughout the GENESIS32 architecture your trend configurations are saved to a Microsoft Access or Microsoft SQL Server database.

10.3. Trend Logger Configuration

The configuration layout of the TrendWorX32 SQL Logger is quite similar to that of an OPC server configuration. The key concept here is the database group object which replaces the device of the OPC data server configuration.

A configuration holds settings for **Database groups, Logging groups, and Logging tags**.

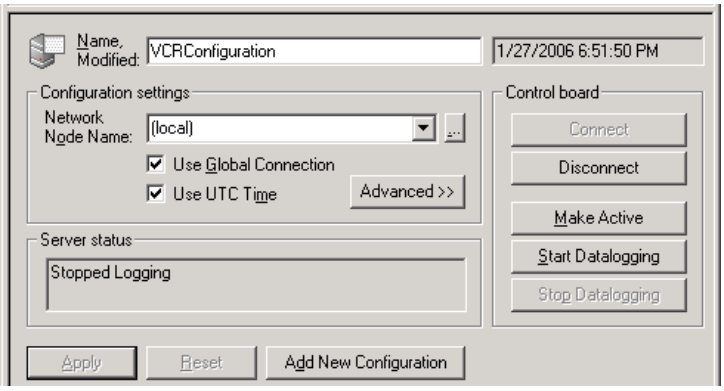


Figure 10.3 - Trend Logger Configuration Settings

To control the behavior of the Data Logger when in runtime mode the following configuration settings are provided in the Logger configuration settings:

- **Network Node Name:** The node on which the logging configuration will run; typically this value is set to **(local)** for a data logger running locally.
- **Use Global Connection:** When enabled, the connection to the database will be maintained open.
- **Use UTC Time:** When enabled, the timestamps will always be logged in UTC format. You may want to enable this option when you have systems in different time zones. You may also want to enable this option so that there is no confusing changing between daylight savings time.
- **Retrials (Advanced):** The number of times the Data Logger will try to log a given data packet if it has failed initially.
- **Retrial Delay (Advanced):** The period (in seconds) between retrials.
- **Max Samples (Advanced):** The maximum number of samples to keep in memory prior to data logging on a per-signal basis.
- **Max Logging Packets (Advanced):** The maximum number of data logging packets to keep in memory in case of data logging failure.

10.4. Database Group Configuration

To set up data logging you must first configure the underlying database and set up ODBC data sources to connect to it. An ODBC Data Source allows you to make a generic connection to any ODBC compliant database. You can consider it as the link (or the source) that provides the connection to the target database where data will be logged.

You can configure the ODBC data source within the TrendWorX32 Logger Configurator by connecting a **Database Group** to an existing database or a new database. It is imperative that the computer on which the TrendWorX32 SQL Logger resides have full access to the target database to create, add, delete, and update tables. Otherwise, data logging cannot be accomplished.

A database group object represents a:

- High-level grouping of group signals to data log
- Multithreaded object for managing data updates and logging-to-disk activities along with an extensive runtime user interface
- Physical connection to an underlying database

In addition, for troubleshooting, each database maintains a list of recent errors which can be found in the Group Status field in the TrendWorX32 Logger Configurator as shown in Figure 10.4.

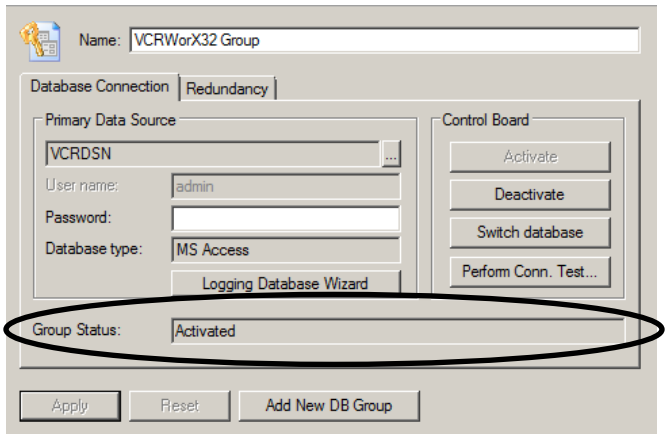


Figure 10.4 - Database Group Configuration and Status

When configuring a Database Group, you must provide a **Name** and the **Primary Data Source** in the **Database Connection** tab. The Primary Data Source allows you to setup a connection to the destination database where you would like the data to reside. It also allows you to configure a new data source. In many ways this is similar to the Data Connection you configured for the AlarmWorX32 Logger.

To the right of the Primary Data Source Section on the Database Connection tab is a **Control Board**. The options allow you to manage the database group and include:

- **Activate database group:** If the TrendWorX32 SQL Logger is in runtime mode, clicking on the "Activate" button starts data collection and data logging for all logging groups of the selected database group.

- **Deactivate database group:** If the TrendWorX32 SQL Data Logger is in runtime mode, clicking on the "Deactivate" button stops data collection and data logging for all logging groups of the selected database group.
- **Database switching:** Clicking "Switch Database" allows you to switch to a different database while in runtime mode.
- **Perform connection test:** Clicking on the "Perform Conn. Test…" button will allow the configuration to try to connect to the indicated database. The result of this test should be successful; otherwise no data will be logged.

NOTE: Database switching is supported in runtime mode; that is, when data logging is active.

10.4.1. Database Support and Options

A key feature of the TrendWorX32 SQL Logger is its ability to handle data logging and data retrieval with respect to multiple types of databases concurrently, therefore enabling you to develop a very cost-effective data logging and data-retrieval application.

TrendWorX32 SQL Data Logger will support data logging to the following types of databases:

- MSDE 2000
- Microsoft SQL Server 2000
- Microsoft SQL Server 2005 Express
- Microsoft SQL Server 2005 (all editions)
- Microsoft Access
- Oracle 8.0
- Oracle 9.0
- mySQL

As Microsoft Access has its limitations, a database server like Microsoft SQL Server or Oracle may provide better performance and functionality for larger data logging systems. This becomes especially relevant when considering maintenance, as Microsoft Access databases do not have an automatic compacting function. This means that once in a while a Microsoft Access database will have to be compacted manually. During the compacting process TrendWorX32 SQL Logger cannot log any data to the database.

NOTE: For more information about the different databases supported, please refer to the application note entitled *ICONICS – Compatible Software and Operating Systems*.

10.5. Logging Group Configuration

Each database group object is responsible for managing **Logging Groups**. A logging group includes:

- A logical collection of signals to be data-logged

- Common data-collection rate and data logging table
- Common data logging configuration
- Group style management and editing capability for a collection of signals

The logging group properties form contains three tabs: Data Collection, Logging, and Table Management.

10.5.1. Data Collection

The logging group allows you to specify two different rates on the **Data Collection** tab as shown in Figure 10.5: a **data collection rate** and a **calculation period**.

The data collection rate is the rate at which the TrendWorX32 SQL Logger receives a sample from the OPC server. It has to be noted that if there are no new values available in the OPC server, no updates will be sent to the logger.

The calculation period on the other hand, affects how the data is collected as well as over which period statistics, such as minimum and maximum, are calculated (if calculations are applied).

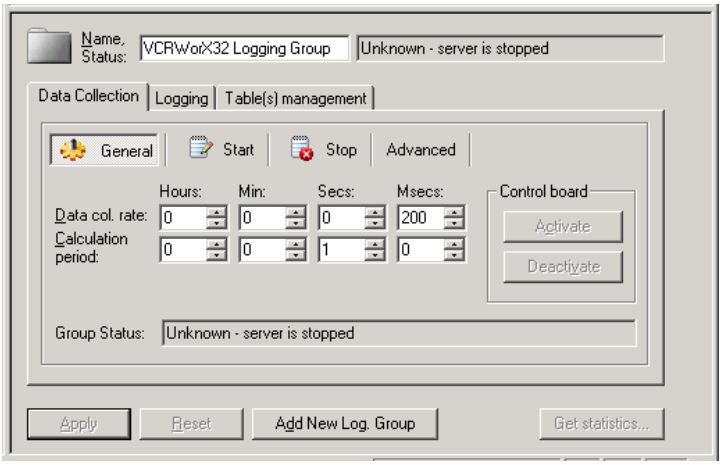


Figure 10.5 - Data Collection Configuration

You can also configure **Start Conditions** and **Stop Conditions** on the **Data Collection** tab to define the beginning and end of data collection or the period during which the logging group is connected to the data source and is logging data.

> **NOTE:** For more information about properties not covered in this class refer to the Help documentation.

10.5.2. Logging

The **Logging** tab in the **Logging Group** configuration form shown in Figure 10.6, determines the settings and conditions for the storage of historical information to the database for the selected logging group. It also provides excellent load-balancing capabilities when writing to disk.

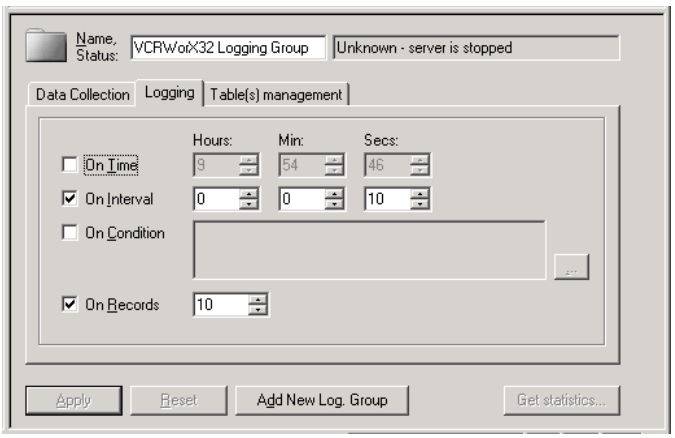


Figure 10.6 - Logging Settings

The manner in which the collected data is actually logged to the database by the Data Logger is determined by the settings in the **Logging** tab. The **On Time** checkbox refers to a data logging time. **On Interval** refers to a selected period during which data logging occurs (for example, every 30 minutes). Data written to the database can be conditionally-based or amount-based—for instance when the system has collected the defined amount of records.

NOTE: The fastest recommended time for logging data to disk is 5 seconds.

On this tab, you are defining how often to write the data to disk and you should find the best balance for how often to perform this task. Accessing disk and writing information is a resource intensive task, therefore, you want to do it as little as possible. However, until the collected samples are written to the database it also takes up your RAM. Finding the balance between how much data your system can hold in RAM without jeopardizing performance is the key.

Additionally, if your logging rate were faster than the collection rate, it would have times where it would log nothing to the database. We generally recommend that the logging rate be greater than the data collection rate.

10.5.3. Table Management

The **Table Management** tab in the **Logging Group** configuration form shown in Figure 10.7 enables you to configure the data-table creation process.

Each group of signals stores its data to a single table or a cyclic sequence of tables (user-configurable). This allows for a controlled size of database storage. For example, you can set up the data logging configuration for a group that enables data storage to multiple, sequential tables. Trend clients may be configured to retrieve data from the most recent tables while the other tables are maintained for reporting and comparison with recent data. Although historical information is retrieved by combining information from various historical tables, the ability to store data in multiple (sequential) tables can be used for database maintenance optimization as well as overall system load balancing. By controlling the size of each historical table, you can avoid creating extremely large tables which may degrade overall data logging and database performance.

To accelerate data retrieval, TrendWorX32 SQL Logger creates the necessary database schema to maintain updated information with respect to the time interval for which each table logs data.

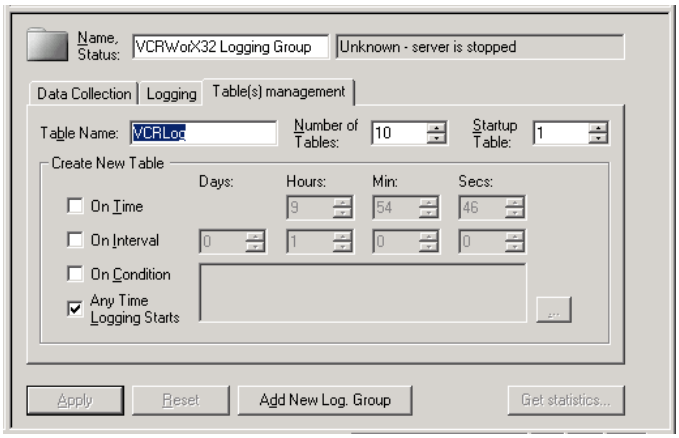


Figure 10.7 - Tables Management Settings

The **Table Management** tab determines how the data will be stored in terms of the number of current tables and when new tables will be created. The number entered in the **Number of Tables** field determines the maximum number of tables that will be created. Therefore, when that number is reached and the last table is full, the TrendWorX32 SQL Logger will wipe out all information in the first table to make room for new data. The **Create New Table** section determines the condition to create a new table. This in conjunction with the number of tables present directly affects the maximum interval of data that is stored.

10.6. Logging Tag

Each logging group maintains a collection of individual **Logging Tags**. These signals can be OPC tags or calculated values. Each signal contains various attributes such as a Logging Name and Range as shown in Figure 10.8. A logging tag can be based on an OPC server tag or an expression incorporating OPC server tags.

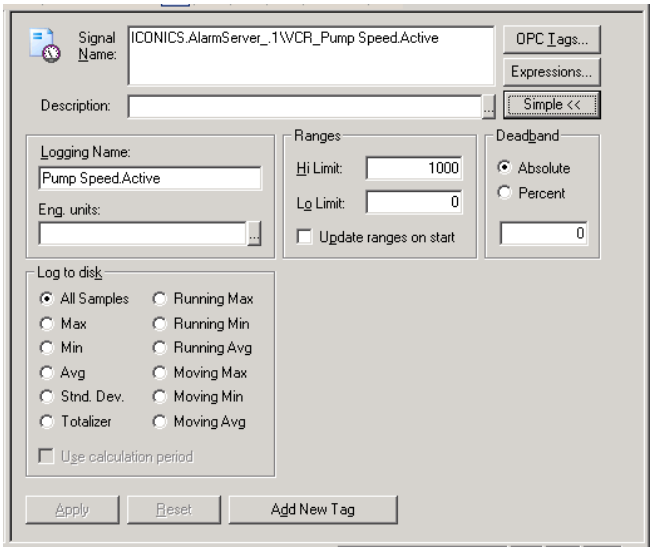


Figure 10.8 - Logging Tag Configuration

- **Signal name:** The **Signal Name** field shows the actual signal (tag or point) name or expression.
- **Description:** This field contains a description of the actual signal. For Unicode applications the description can be language-aliased so that trend clients will use the translated description during animation (runtime) mode. By default, when a signal is inserted in the TrendWorX32 Configurator, the Description entry of the signal will be its logging name.
- **Logging name:** Initially the signal's **Logging Name** is set to match the actual tag (point) name. You can override this setting and choose a more appropriate name, such as "Temperature Measurement." This, in turn will represent the "signal name" by which history clients will access the data logged for that signal. If the signal is an expression its logging name should be replaced with a description that does not include expression-specific syntax.

NOTE: Avoid using long logging names. If reporting is used, logging names should be a maximum of 60 characters. If the tag is an expression, the logging name should be modified to exclude expression symbols or mathematical symbols. Also, avoid starting the logging name with a number and avoid using special characters, such as @, #, &, *, $, !, ^, %, ~, ?.

- **Engineering units:** This field allows you to enter engineering units for the tag. The **Eng. Units** field can be language-aliased, and as a direct result, the trend clients will scale the historical data upon switching to a new language.
- **Deadband:** The **Deadband** field defines the minimum amount that the signal value must change either absolute (specifically) or by a percentage, before it is logged. The deadband is always computed using the most recent sample. A deadband value of 0 results in all samples logged.
- **Ranges:** The **Hi Limit** field sets the values for the tag's high limit. The **Lo Limit** field sets the values for the tag's low limit. In addition, if **Update Ranges on Start** is checked, the ranges for each tag will be obtained from the OPC server when logging starts.

Clicking the "Simple/Advanced" button hides/shows the **Log to Disk** options. The **Log to Disk** field allows you to select the value(s) of the signal to log to the disk. If **All Samples** is selected, the server will log all values for that signal that were collected during that specific data-collection period. If other filters are selected, the appropriate calculations will be applied to the collected values before logging the result.

10.7. TrendWorX32 Configuration in Workbench32

Like many other applications in GENESIS32, you can configure TrendWorX32 Logger using Workbench32. The hierarchy of the configuration is the same as in the standalone Configurator but the configuration forms may appear slightly different.

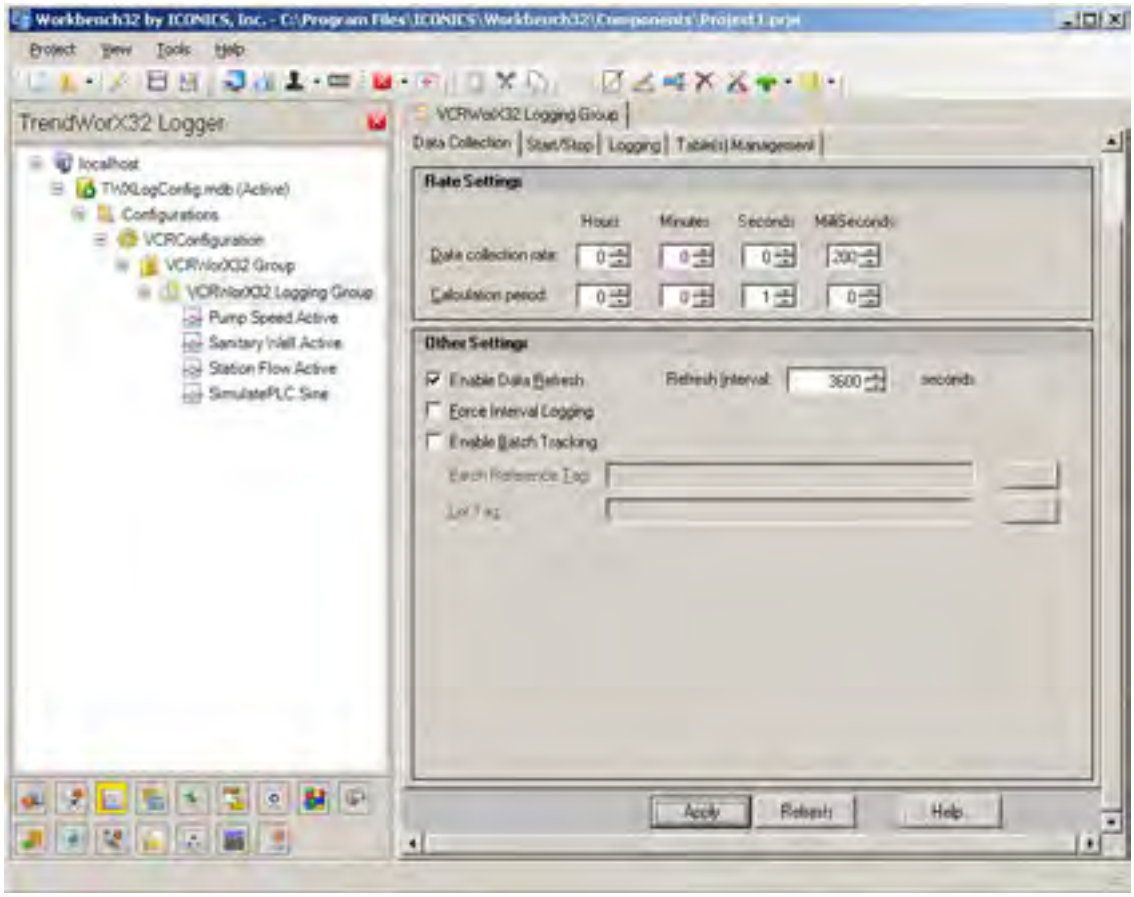


Figure 10.9 - TrendWorX32 Logger in Workbench32

One notable difference between configuring in Workbench32 and in the standalone Configurator is the database connection. Since ODBC Data Sources are specific to each machine you should configure the ODBC Data Source on the server machine first. Especially when you plan to continue configuration on a remote machine later.

Once you have added the ODBC Data Source using Windows Data Source (ODBC) manager, you can select it clicking on the "Change Connection…" button on the Database Connection form. In the **Connection Strings** dialog that comes up, double-click on the ODBC connections folder. This enables the "Add connection string" button. Clicking on this button brings up the **ODBC Connection String** dialog from which you can select the ODBC Data Source in a dropdown menu.

There are other differences when comparing the configuration interface between the standalone Configurator and Workbench32. However, these differences are minor and intuitive. As you work with the configuration forms in Workbench32 you will become familiar with the differences.

Lab: Maintaining OPC DA Records *L10*

Acmeseutical would like to maintain some sort of record to indicate how well they are doing with production. They would like to trend some information and analyze it later to find an ideal that they can compare with the current shift.

In this lab, you will create a trending configuration for Acmeseutical to log their data.

Estimated time to complete this lab: 20 minutes

10.1. Task Checklist

In this lab, you will complete the following tasks:

- Create a new TrendWorX32 configuration database
- Configure a database group
- Configure a logging group
- Add tags to a logging group

10.2. Support Files

You will need the following file(s) to complete this lab:

- AcmeseuticalProject.prjx

10.3. Trending Data

In order to trend data, we will need to first set up a configuration that tells us how often to sample data and where to log the information.

1. Open TrendWorX32 Configurator by going to Start → Programs → ICONICS GENESIS32 → TrendWorX32 → TrendWorX32 Configurator.

2. The application starts with a default configuration.

3. Expand Configurations → VCRConfiguration in the tree control on the left-hand side.

4. You may see a message indicating that the service is connecting. Once everything connects, check and see that the traffic light is green in the toolbar.

5. In the navigation tree, select VCRConfiguration. In the Control Board section, click on the "Disconnect" button.

6. We will create our own configuration database to house the tags that we want to log.

7. Click on the "New" in the toolbar to launch the **ICONICS TrendWorX32 Logger Configurator Database Wizard**.

8. Click on "Next >" in the welcome screen.

9. Choose the "Create new configuration in MS Access database" option in the database type screen and click on "Next >."

10. Click on the ellipse button and browse to C:\Training in the **File Browser**.

11. Set the database name to "**AcmeseuticalTrendConfig**".

12. The Database wizard should look similar to Figure L10.1. Click on "Finish" to complete the database creation process. If the file name already exists, you may replace it.

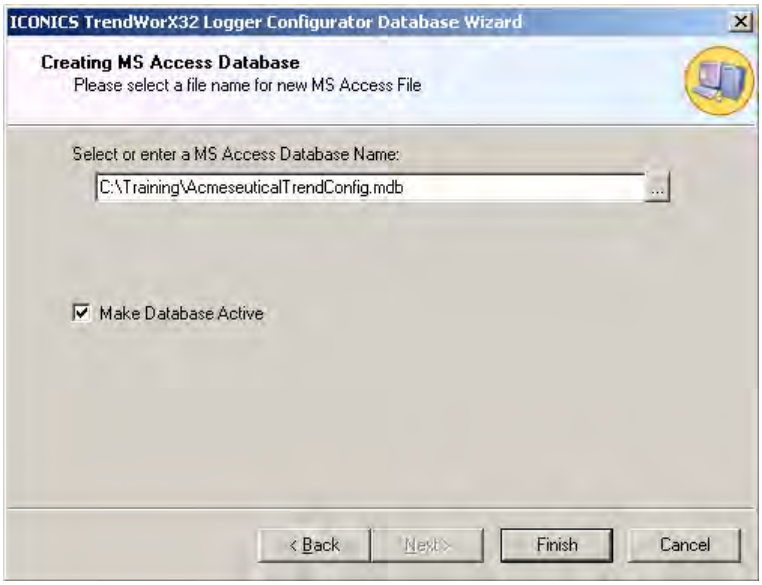


Figure L10.1 - Database Wizard: Database Name and Location

13. You should see an empty configuration in the TrendWorX32 Configurator.

14. Right-click on "Configurations" and select **New → Configuration**.

15. Name the configuration "**AcmeseuticalTrends**" and click on "**Apply**."

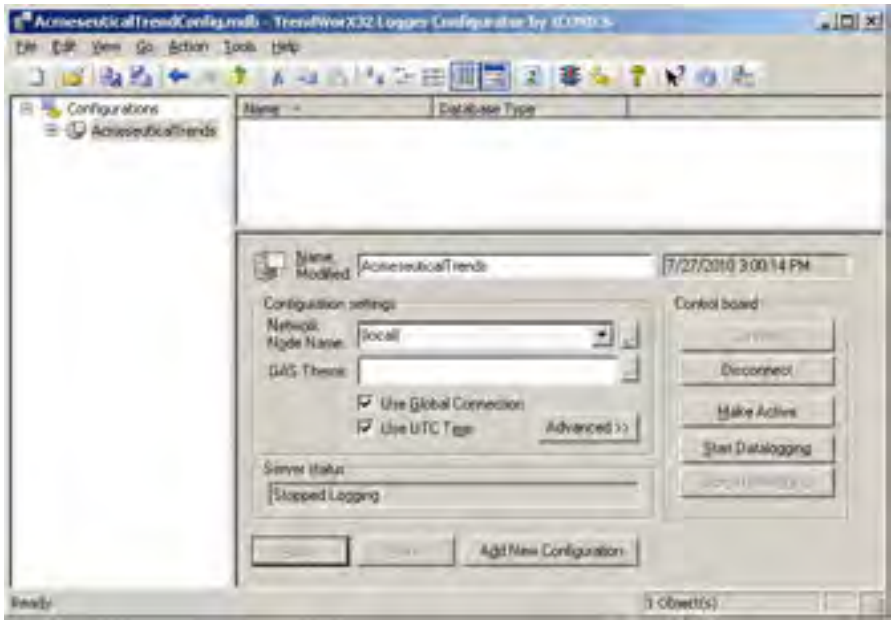


Figure L10.2 - AcmeseuticalTrends Configuration

16. Right-click on "AcmeseuticalTrends" in the tree control on the left-hand pane and select New → Database Group.

17. Name this **Database Group** "**Foxborough**".

    NOTE: In the following steps, we will create a data source to connect to the SQL Server database, AcmeTredData, that we created in Lab1 on page L1-12, section 1.5, Step 7.

18. Click on the ellipsis button in the Primary Data Source section to bring up the **Select Data Source** dialog.

19. Select the **Machine Data Source** tab and click on the "New" button.

20. In the **Create New Data Source** dialog that comes up, select the "**System Data Source**" option and click on "Next >."

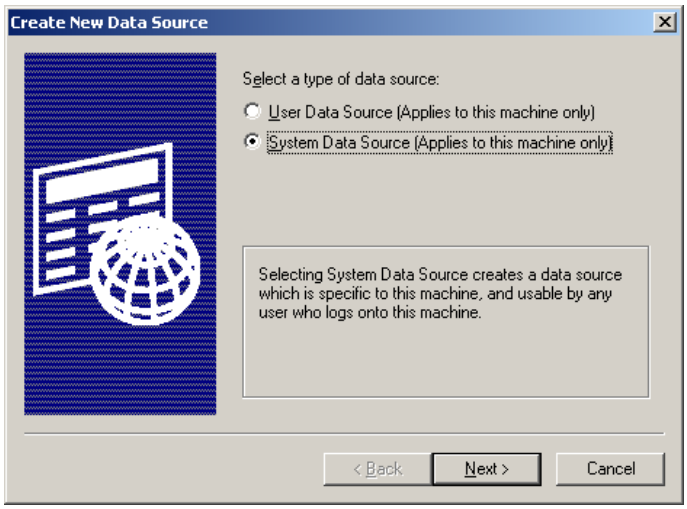


Figure L10.3 - Creating a System Data Source

21. Select the "**SQL Native Client**" as your driver and click on "Next >."

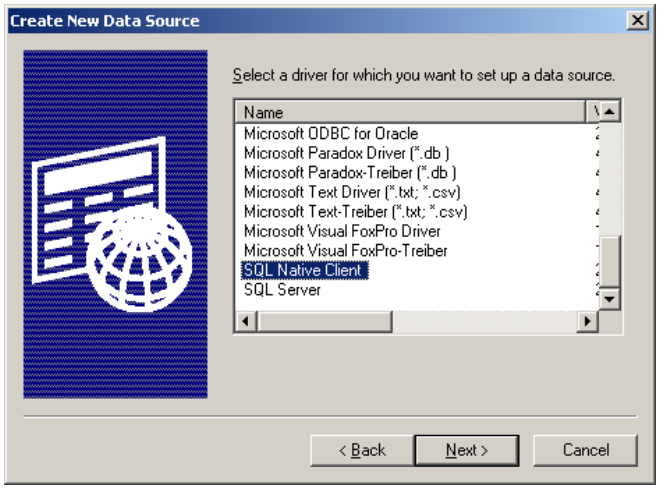


Figure L10.4 - Selecting SQL Native Client as the Driver

NOTE: If you are using SQL Server 2000, choose the "SQL Server" driver instead.

22. Click on "Finish" to complete the **Create New Data Source** process.

23. The **Create New Data Source to SQL Server** dialog opens.

NOTE: Depending on the driver you choose in the previous step, the rest of the data source creation dialog may look different.

24. Name the data source "**AcmeTrendDataSource**" and select your SQL Server in the Server dropdown list. If the return is empty, see the following **NOTE**.

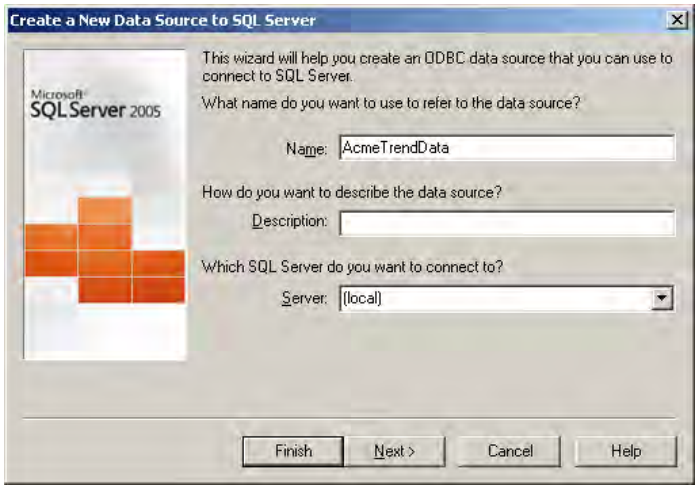


Figure L10.5 - Create a new Data Source to SQL Server

25. **NOTE:** If you are using one of the training machines provided, the SQL server is (local)\sqlexpress. Click on "Next >" to continue.

26. On the SQL Authentication screen, leave everything as default and click on "Next >."

27. Check the "**Change the default database to**:" option and choose the "AcmeTrendData" database in the dropdown menu.

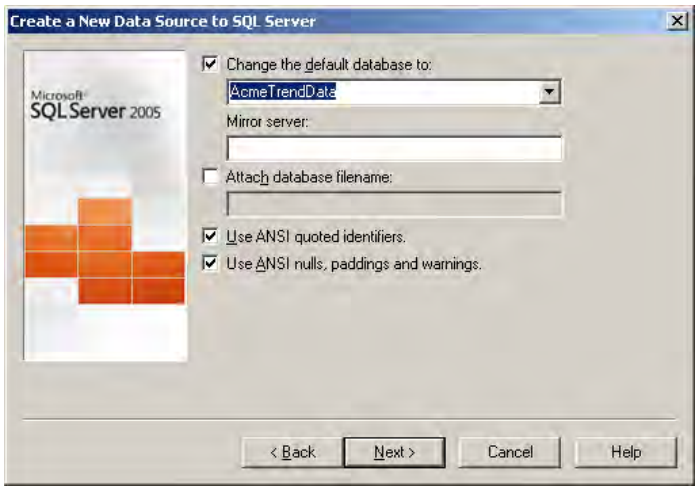


Figure L10.6 - Change Default Database to AcmeTrendData

28. Click on "Next >" to continue.

29. Leave the selections on this page as default. Click on "Finish" to complete the data source connection.

30. In ODBC Microsoft SQL Server Setup dialog, click on the "**Test Data Source**" button. The results of the test should return successfully, otherwise you should go back and double-check the settings.

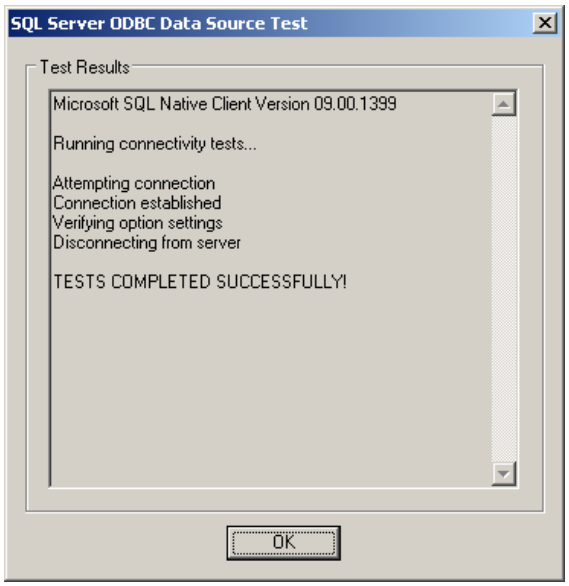


Figure L10.7 - Successfully Connected to AcmeTrendData

31. Click on "**OK**" until you are back in the TrendWorX32 Configurator.

32. Click on "Apply" to save the changes. Your Database Group configuration should look similar to Figure L10.8. If "**Group Status**" is not **Activated**, click on the traffic light on the toolbar to activate it.

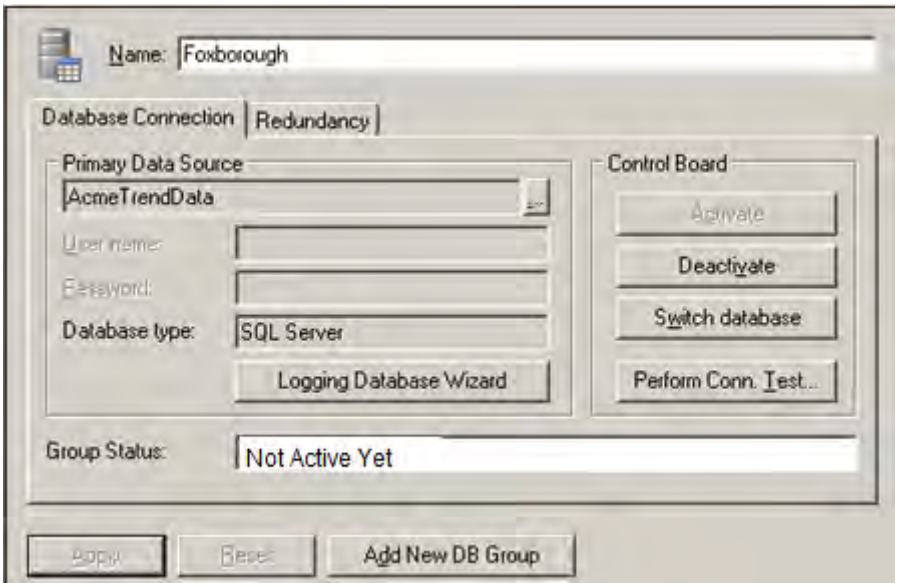


Figure L10.8 - Foxboro Logging Group

33. Right-click on "Foxborough" in the tree control on the left-hand pane and select New → **Logging Group**.

34. Name this Logging Group "**Machine1**".

35. Select the **Data Collection** tab, General page and set the **Data Collection Rate** and the **Calculation Period** to five **(5) seconds**.

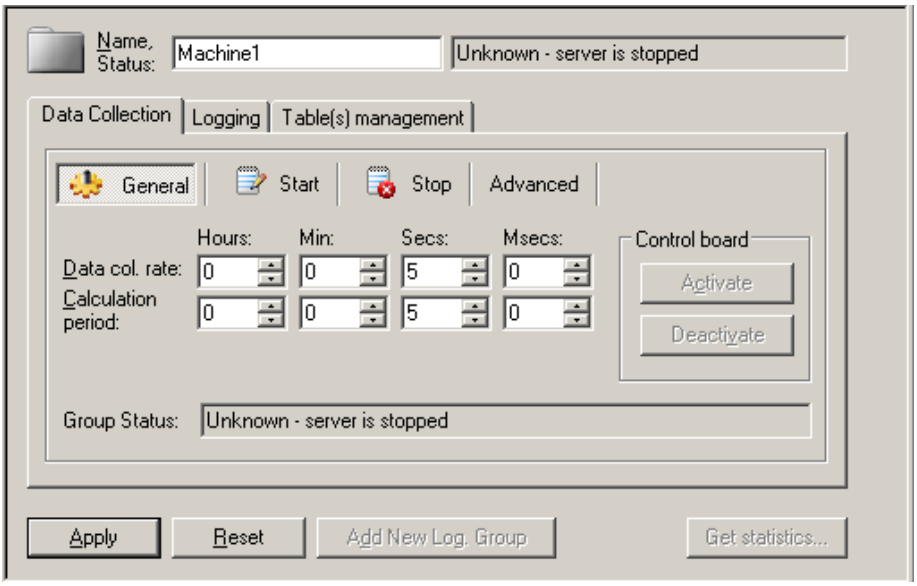


Figure L10.9 - Data Collection Rate

36. Click on the "**Advanced**" page.

37. Check the "**Force Interval Logging**" option. This option will allow us to collect a sample even if the value in the data source does not change.

38. Select the **Logging** tab and **uncheck** the "**On Records**" option.

39. Change the "**On Interval**" options to log every one **(1) minute**.

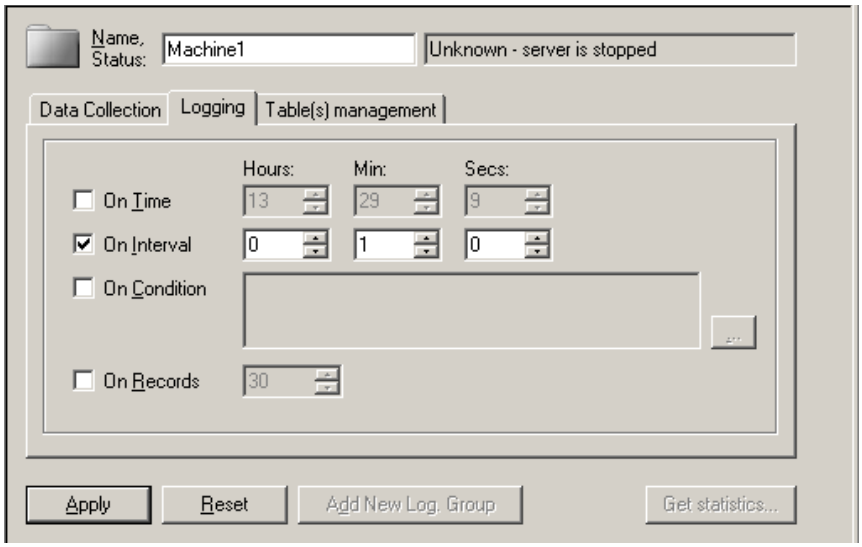


Figure L10.10 - Data Logging Rate

40. Click on the **Table(s) Management** tab and **uncheck** the "**On Time**" option.

41. Check the "**On Interval**" option and set it to one **(1) day**.

42. Change the **Number of tables** to seven **(7).** This way, we will have one day's worth of data in each table and we only keep a week's worth of data.

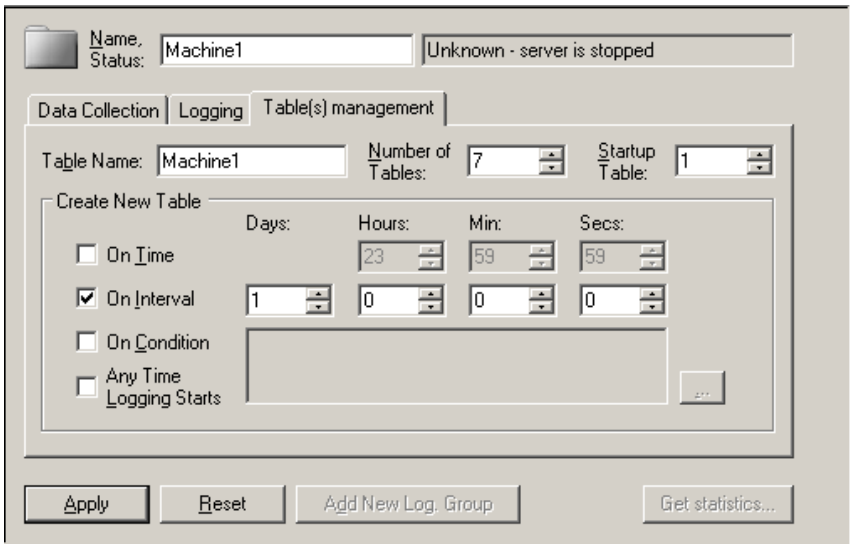


Figure L10.11 - Table(s) Management

43. Click on "**Apply**" to save the changes.

44. **Right-click** on "**Machine1**" in the tree control on the left-hand pane and select New → OPC Tags.

45. In the UDB that opens, browse to ICONICS.DataWorX32.8 → registers → USA → Foxborough → Machine1 → Mixers → MixerA.

46. Double-click on the **Level** and **Level_Setpoint** tags so that you see it added to the list in the upper half of the UDB.

47. Repeat Steps 45-46 for MixerB, MixerC, and MixerD. When you have all of the tags in the UDB, click on "OK" to add the tags.

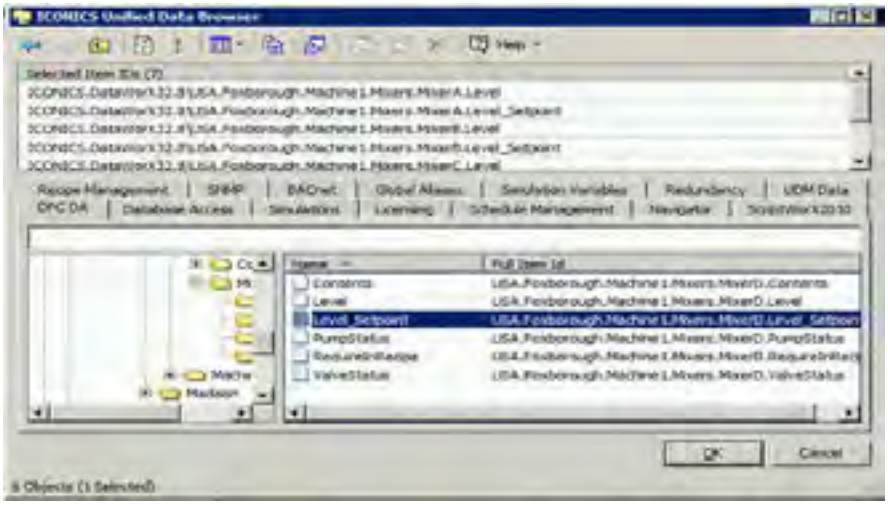


Figure L10.12 - Selecting Multiple Tags in the UDB

48. If you now expand "Machine1" in the tree control, you should see the four tags being added.

49. Right-click on "Machine1" and select New → OPC Tags again. This time, browse to ICONICS.DataWorX32.8 → registers → USA → Foxborough → Machine1 in the tree control on the left-hand pane.

50. Right-click on the "BottlingMachine" folder and select "Enter Up." All tags in that folder should now be added in the upper portion of the UDB.

51. Do the same for the "Control" folder. Click on "OK" to add the tags when you have them all.

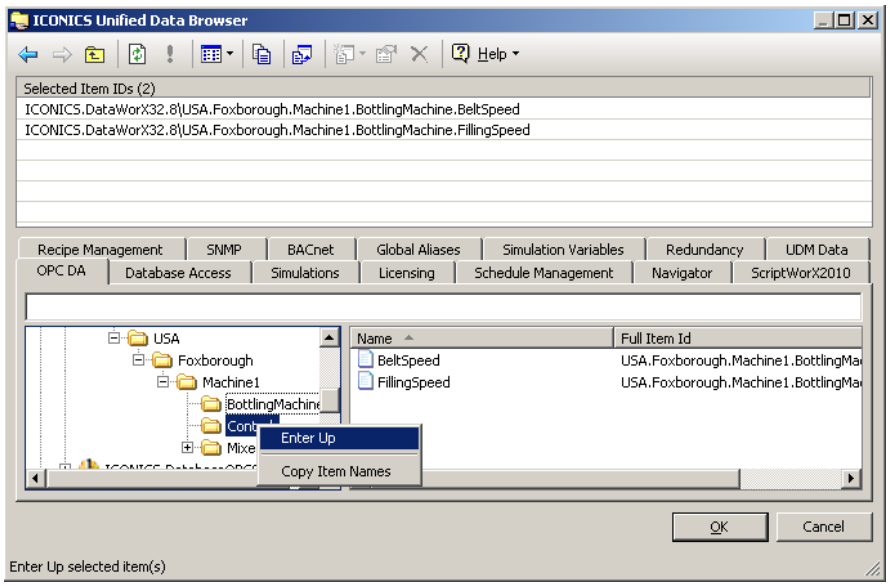


Figure L10.13 - Using "Enter Up" to Add All Tags in a Folder

52. Select "**Acmeseutical Trends**" in the tree control. In the Control Board section on the lower right-hand pane, click on "**Make Active**" and then "**Start Datalogging**."

53. We will let the data log during the next lecture. This way we will have some historical data in the next lab.

Review: TrendWorX32 Logger R10

10.1. Chapter Objectives

In this chapter, we focused on the TrendWorX32 Logger, which is a 32-bit data recorder that allows you to trace the values of your OPC tags. By now, you should know how to do the following:

- Create a new TrendWorX32 Logger Configuration Database
- Create a new TrendWorX32 Logger Configuration
- Create database groups
- Create logging groups
- Add tags to a logging group
- Choose an appropriate database to log data to

10.2. Summary

TrendWorX32 contains many different applications and is part of the GENESIS32 product family. The core of the TrendWorX32 product suite is the TrendWorX32 SQL Logger. The logger takes advantage of the ADO and OLEDB database technologies from Microsoft, allowing data to log to different databases such as MS Access, MS SQL, mySQL, and Oracle.

In the configuration of the TrendWorX32 SQL Logger, you define a Database Group, which links to one of the aforementioned database types to store the data. Beneath the Database Group, you also define Logging Groups. The Logging Group resembles a table in a database, and in the configuration of the Logging Group is where you can setup when to create new tables to store data and how much data you want to keep in the database. Also in the Logging Group, you can control how often you take a sample value as well as write the sampled values into the database. The logging tags that you add to the Logging Group will inherit the sampling rate and data logging rate configured for the group.

The TrendWorX32 HDA Server is a separate, yet equally important part of TrendWorX32. It is the application that provides data from the database to any OPC HDA compliant clients. While we do not create any configuration for this piece, it is essential in order to retrieve logged data.

10.3. Terms

Calculation Period	Data Collection Rate	Database Group
Logging Group	Logging Rate	Logging Tag
TrendWorX32	TrendWorX32 HDA Server	TrendWorX32 Logger
TrendWorX32 Viewer		

10.4. Test Your Knowledge

In this chapter, we talked about how to configure the TrendWorX32 Logger. By now, you should have a general idea of the steps required to create a configuration to trend OPC DA tags. You should be able to answer the following questions.

1. What is the fastest Data Collection Rate you can configure?

    a. 1 millisecond
    b. 10 millisecond
    c. 32 millisecond
    d. 1 second

2. Which of the following is NOT a supported database type for logging trend data?

    a. MS SQL 2000
    b. mySQL
    c. mSQL
    d. Oracle

3. Which of the following is the correct hierarchy for a TrendWorX32 Logging configuration?

    a. Configuration → Database Group → Logging Group → Logging Tag
    b. Configuration → Logging Group → Database Group → Logging Tag
    c. Configuration → Logging Tag → Database Group → Logging Group
    d. Configuration → Logging Tag → Logging Group → Database Group

4. Which of the following is NOT an option when specifying the data logging rate?

    a. On Time
    b. On Interval
    c. On Condition
    d. On Buffer Size

5. Which of the following can you define in the Logging Group?

   a. Data collection rate
   b. Data logging rate
   c. Database name
   d. Table name
   e. All of the above
   f. Only A, B, and C
   g. Only A, B, and D
   h. Only B, C, and D
   i. None of the above

6. True or False: You can change the logging database during runtime.

For questions 7-10, please refer to Figure R10.1.

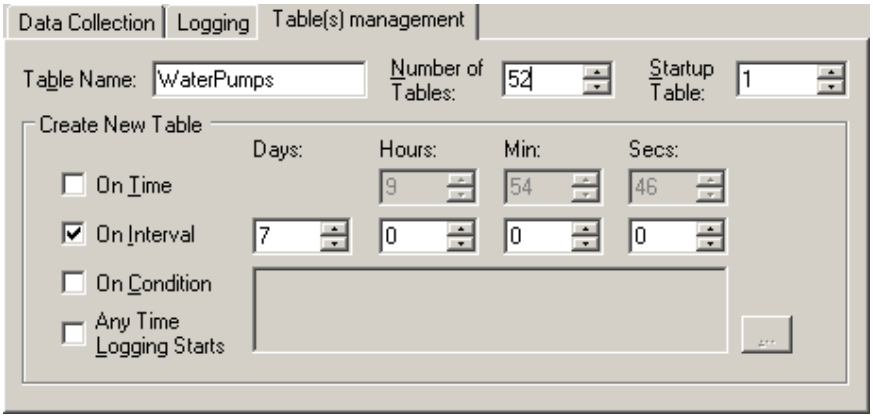


Figure R10.1 - Logging Group Configuration

7. True or False: In the configuration shown in Figure R10.1, the base table name in the database will be WaterPumps.

8. In the configuration in Figure R10.1, what will be the maximum amount of data available in the database?

   a. 1 Day
   b. 7 Days
   c. 1 Month
   d. 1 Year

9. True or False: In the configuration shown in Figure R10.1, the logger will create a new table every month.

10. True or False: In the configuration shown in Figure R10.1, the first table in the database will be named WaterPumps_1.

TrendWorX32 Viewer 11

In the previous chapter, we described how to use the TrendWorX32 Logger to record trend data into an SQL database. We also briefly discuss the TrendWorX32 HDA Server which allows you to replay the logged data. In this chapter, we will explore how to view the data. By the end of this chapter, you should be familiar with the following concepts:

- TrendWorX32 Viewer and where to configure its properties
- General settings for the control
- Appearance settings
- Configuring ranges
- Pen configuration
- Configuring rates
- Using the trend viewer in runtime

11.1. TrendWorX32 Viewer

The ICONICS TWXView32 ActiveX Control is located on the ActiveX toolbar. This control is both an OPC Data Access client (displays data near real-time) and an OPC Historical Data Access client (used for displaying historical data). The Trend Viewer is even capable of showing real-time and historical data in the same trend window.

Features of the TrendWorX32 Viewer include:

- Real-time data displays
- Historical data displays
- Data utility functions
- Sophisticated, multithreaded data buffering to guarantee data integrity with support for customer-configurable data-collection rates

The TrendWorX32 Viewer has an interface to display memory-based historical data from the GENESIS32 Persistent Trending memory-based data-logging OLE Automation server, as well as historical servers supporting the OPC HDA (Historical Data Access) Interface such as the TrendWorX32 SQL Logger.

> **NOTE:** Persistent Trending is outside the scope of this course. If you would like more information on Persistent Trending, please refer to the proper Help documentation.

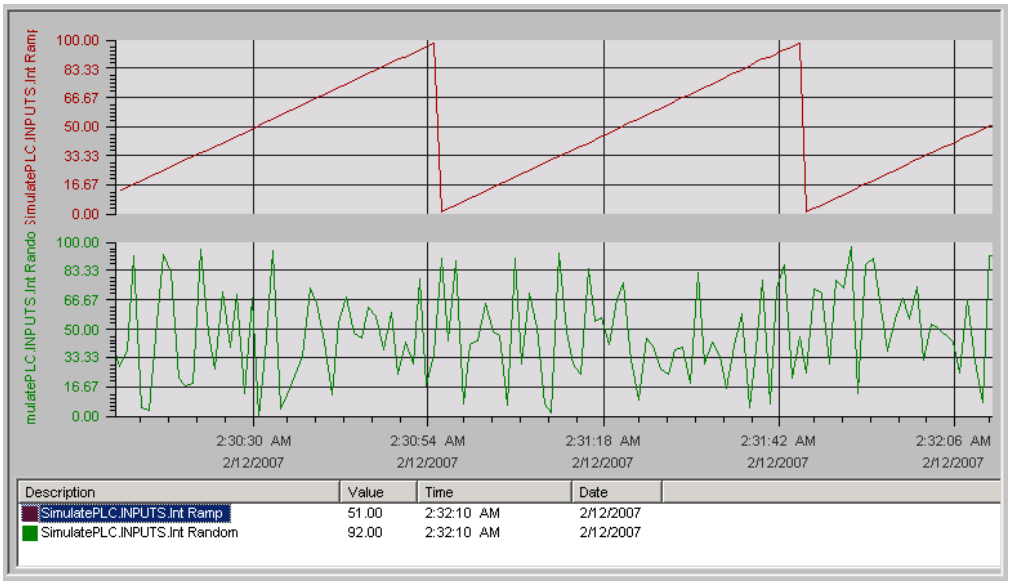


Figure 11.1 - ICONICS TWXView32 ActiveX Control in Runtime

The TrendWorX32 Viewer ActiveX Control has many other features including:

- Support for stacking multiple trend plots
- Support for background images
- Customizable background printing option
- x/y axes auto-size feature
- Tick mark interval support on x/y axes
- Customizable time stamp spacing in trend display
- Plot title support on y-axis
- Font-sensitive auto-size support on chart title
- Grid animation control for the new plot style
- User-friendliness and readability of configuration panel and trend displays
- Fully backward compatible with the previous plot style

11.2. Trend Viewer Properties

The Trend Viewer ActiveX Control Properties dialog, shown in Figure 11.2, is the configuration interface for the ActiveX control. Double-clicking on the control during configuration mode opens the Properties dialog, which contains the following tabs:

- General
- Appearance
- Ranges
- Pens

- Rates
- Settings

NOTE: Right-clicking on the Viewer ActiveX and selecting Properties also opens the ICONICS TWXView32 ActiveX Control Properties dialog.

Not all of these configuration options are within the scope of this class. For more information about Trend Viewer Properties not covered in this class, refer to the "ActiveX Properties" section of the TrendWorX32 Viewer ActiveX Help documentation.

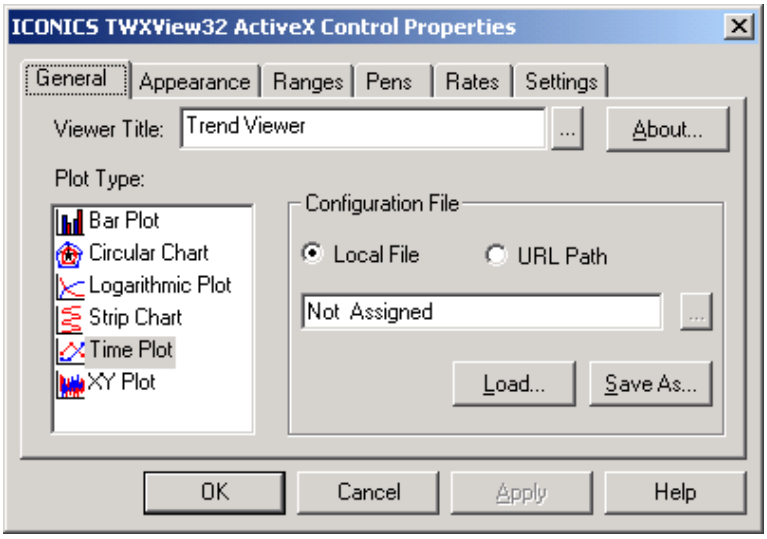


Figure 11.2 - Trend Viewer ActiveX Properties

11.3. General Configurations

The **General** tab, shown in Figure 11.2, is the default tab of the Properties dialog. The **General** tab handles the following functions:

- Viewer title
- Configuration File selection: Local or an URL
- Plot type

The Trend Viewer can show data in six different plot types: *Bar Plot*, *Circular Chart*, *Logarithmic Plot*, *Time Plot*, *Strip Chart*, and *XY Plot*. You should always choose the Plot Type first because this choice determines which options will be available on some of the other configuration tabs.

NOTE: For more information about options available on the **General** tab, refer to the "General Tab" section in the TrendWorX32 Viewer ActiveX Help documentation.

11.4. Appearance Settings

The Trend Viewer provides a wide variety of parameters for configuring its layout and appearance. It also provides the ability to adjust the appearance of up to 64 OPC tags to which it may be connected. The **Appearance** tab of the Properties dialog, shown in Figure 11.3, sets the appearance properties for the following items:

- Workspace
- Details
- Ranges
- Time
- Grids

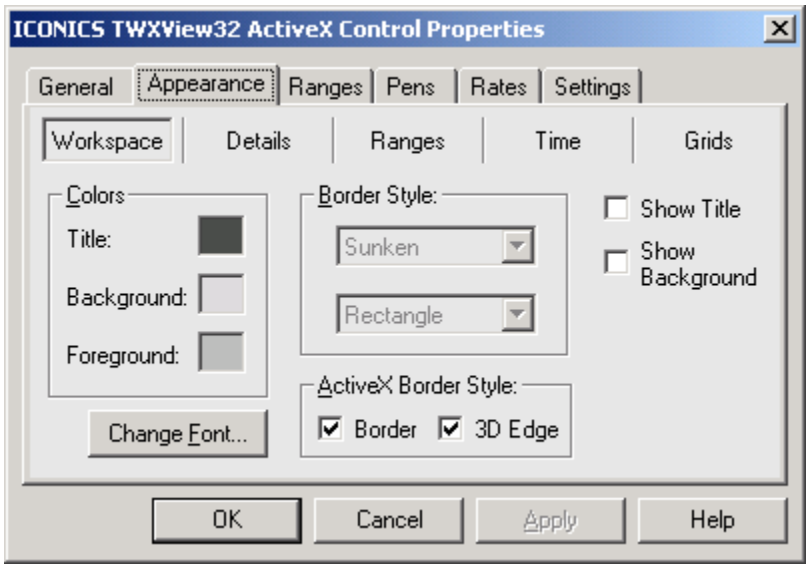


Figure 11.3 - Appearance Properties

The specific pages under the Appearance tab, which are of interest in this course, are the **Details** and **Ranges** pages.

11.4.1. Details

The **Details** section of the **Appearance** tab manages the detail information area (the tag and data summary list). The enabled choices closely match the selected plotting type. Clicking on the "**Columns**" button on the **Details** page opens the **Detail Window Columns** dialog which allows the user to specify the columns and names to be displayed in the details section (located below the chart window) of the Trend Viewer in runtime. Clicking the "Spacing" button allows you to configure the default column spacing for the Details window during runtime.

The “Number of Entries” field displays the number of pens visible in the Details window of the Viewer. If more pens exist, the details window will show a vertical scroll bar to reveal more pens. It is often helpful to be able to increase the number of entries.

11.4.2. Ranges

The **Ranges** page of the **Appearance** tab manages the Ranges Configuration for the trend display. Depending on the selected plot type, some options may not apply and therefore will not be available. You can edit the number of **Visible Ranges** here such that separate ranges for each pen in the viewer can be shown at once if desired.

> **NOTE:** For more information about the options available on the **Appearance** tab, please refer to the "Appearance Tab" section in the TrendWorX32 Viewer ActiveX Help documentation.

11.5. Configuring Ranges

The **Ranges** tab of the Properties dialog, shown in Figure 11.4 manages the global ranges configuration for the trend display. Depending on the selected plot type, some options may or may not be available.

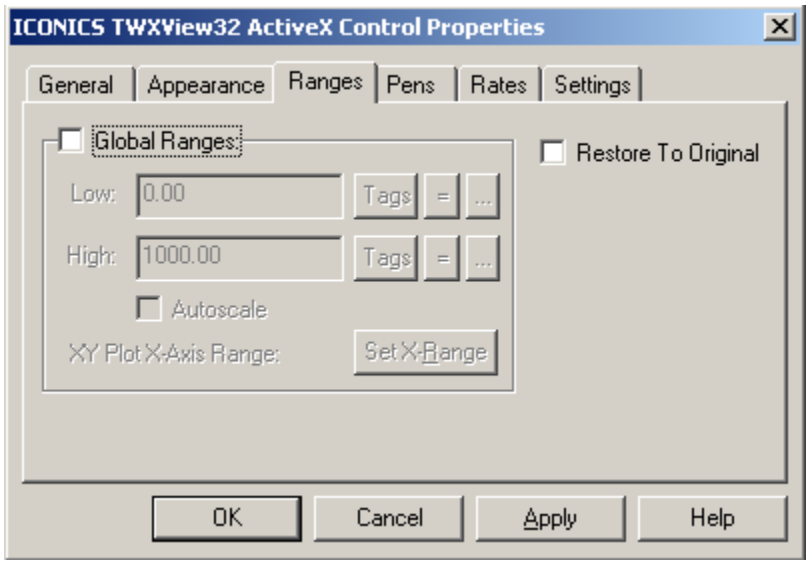


Figure 11.4 - Ranges Properties

The **Global Ranges** section of the **Ranges** tab sets the ranges for the trend display. If **Global Ranges** is checked, all pens will be drawn using the same set of ranges. These can be constants, expressions, OPC tags, or global aliases.

11.6. Pen Configuration

The **Pens** tab of the Properties dialog, shown in Figure 11.5, handles the signal connection, editing, and configuration for the data displayed in the trend. The following types of signals can be viewed in the Trend Viewer:

- Real-time (expressions and OPC tags)
- Historical
- Persistent (real-time with a small amount of immediate history)

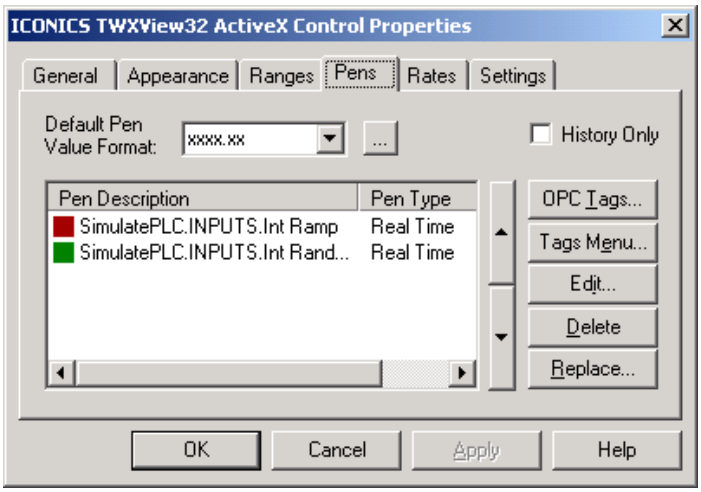


Figure 11.5 - Pens Properties

The Trend Viewer can show both real-time and historical data although you can configure it to show historical data only. In that case, the Trend Viewer will not periodically request data from the database but only request data when the operator has made a change in the window time-frame. To enable history data only mode simply enable the “History Only” checkbox.

You can use the “OPC Tags…” button to add a real-time OPC DA tag for trending. To add a history tag you will need to click on the “Tags Menu…” button and select the “History Tag…” option. Once you select this option you will notice that the only tab available in the Unified Data Browser is OPC HDA and you will only see OPC HDA compliant servers listed. If you changed your mind and want to add real-time tags instead, simply exit out of the browser and click on the “OPC Tags…” button in the TrendWorX32 Viewer properties instead.

Once you add pens into the viewer you can see the pen description and type on the Pens tab. You can configure the **Trend Pens** in greater detail using the **Edit Trend Pen** dialog. A pen's appearance can be modified using **Line** and **Width** parameters.

A pen also provides a variety of additional options: *Visible*, *Stair Plot* (to plot straight horizontal and vertical lines only), *Pen Comments* (to show if comments have been added to this pen), *Pen Marker* (a vertical marker on the side of the trend window), and *Show Samples* (to show the actual location of the samples).

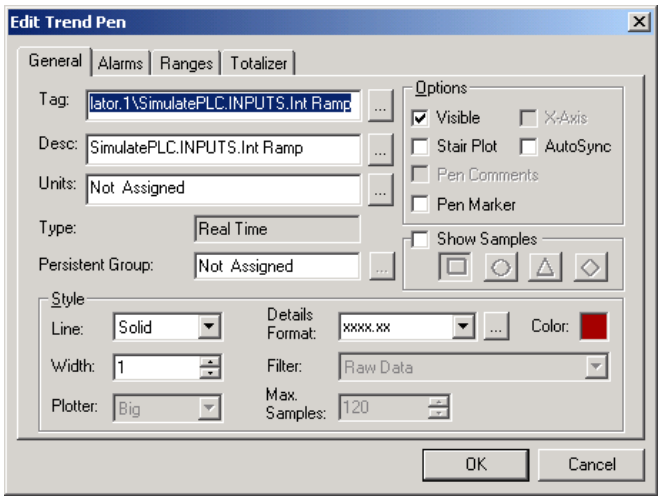


Figure 11.6 - Edit Trend Pens Properties

It is also possible to change how a historical pen requests its data from the database. The default configuration has the pen requesting its data using the “Last Sample” method which means that for every required point of the trend grid a value is plotted. If no value is available in the database, the last (previous) sample value is used. Another common option to use is “Raw Data” as this will only request those samples that are actually available in the database. The Trend Viewer will then draw lines between the actual samples.

> **NOTE:** For more information about options available on the **Pens** tab also refer to the "Pens Tab" section in the TrendWorX32 Viewer ActiveX Help documentation.

11.7. Configuring Rates

In the **Rates** tab of the Properties dialog box, shown in Figure 11.7, you can select the desired “Data-Collection Rate” and the number of “Samples” among other variables. Currently the fastest data-collection rate is 10 milliseconds (ms). The update rate is calculated by inserting the variables on the **Rates** tab into an equation. Therefore, if you change one of these variables all other variables will be updated automatically. Notice also that the data-collection rate is configured independently of the drawing action.

The fastest display update rate is 50 ms, with the exception of bar plots, in which the display rate matches the data-collection period. The maximum number of samples that can be displayed at once is 7200.

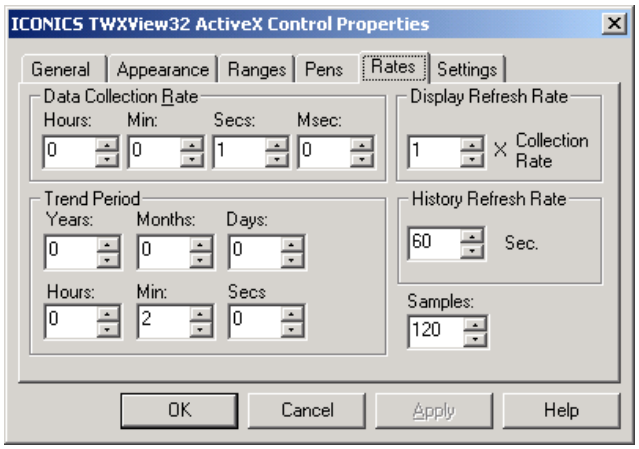


Figure 11.7 - Rates Properties

The **Display Refresh Rate** controls the speed at which the signal curves will be drawn. If, for example, the **Data-Collection Rate** is set to 250 ms and the **Display Refresh Rate** is set to 4, the samples will be collected at 250 ms but the signal curve will be updated every second without loss of data. This is a great option to minimize system-resource usage.

The **History Refresh Rate** controls the frequency at which the historical pens will be refreshed with new historical data from the TrendWorX32 historical database. This can be used as an optimization and load-balancing tool. For example, if the TrendWorX32 SQL Server is configured to log to the database every 1 minute, then the **History Refresh Rate** should be set to a minimum of 1 minute.

11.8. Trend Viewer Runtime Toolbar

The Trend Viewer **Runtime Toolbar** is the runtime user interface for the ActiveX control. The toolbar offers you access to various runtime TrendWorX32 Viewer ActiveX control functions and operations. To view or hide the toolbar, double-click the TrendWorX32 Viewer during runtime. You cannot view the toolbar in configuration mode.

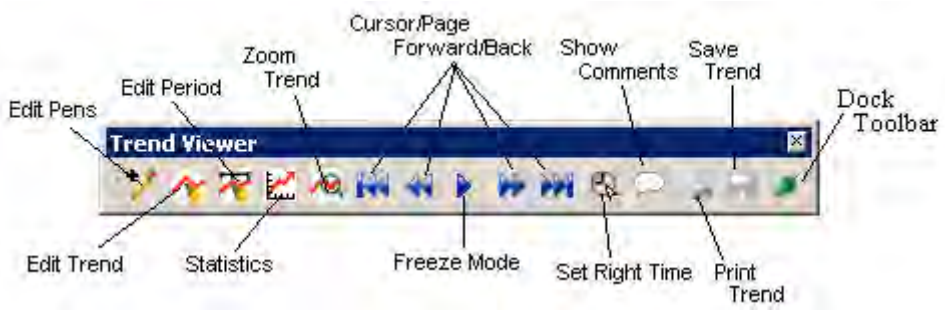


Figure 11.8 - Runtime Toolbar

Many of the buttons on the Runtime Toolbar provide access to properties configurable in both Configuration and Runtime modes, such as Edit Pens, Edit Trend, and Edit Period.

Some options such as Cursor / Page Forward/Back, Zoom Trend and Set Right Time are only available when the viewer is in **Freeze Mode.**

Clicking the "Statistics" button opens the **Trend Statistics** dialog, shown in Figure 11.9.

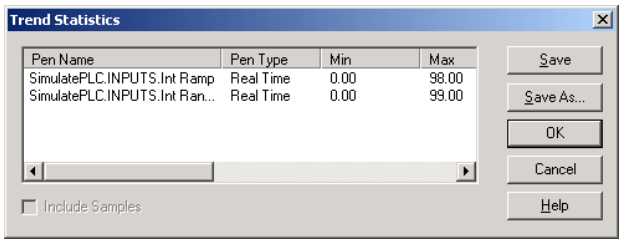


Figure 11.9 - Runtime Statistics

The **Trend Statistics** dialog shows relative statistics for all pens in a trend. In the case of bar plots the statistics window shows the statistics profile of all signals combined.

Clicking the "Set Right Time" button opens the **Set Trend Top/Right Time** dialog, shown in Figure 11.10. This dialog enables you to view a specific historical trend period. In freeze mode this causes the TrendWorX32 Viewer to retrieve and display the selected historical and persistent trend data from the TrendWorX32 Persistent Trending memory buffers.

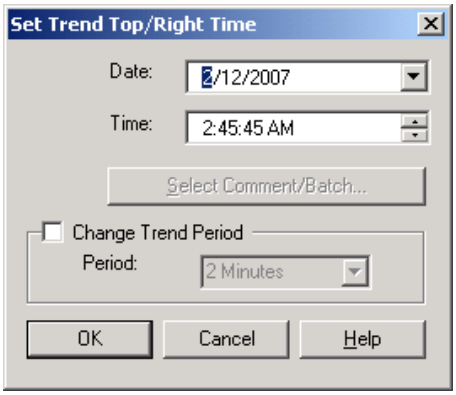


Figure 11.10 - Set Right Time

Lab: Viewing Historical Trends L11

Acmeseutical would like to allow their operators to see the historical trend information we logged in the previous chapter. We will create an HMI screen for them for this purpose.

Estimated time to complete this lab: 30 minutes

11.1. Task Checklist

In this lab, you will complete the following tasks:

- Insert a Trend Viewer ActiveX into GraphWorX32
- Configure the Trend Viewer ActiveX
- Add and configure pens
- Configure the Trend Viewer Details window
- View OPC Historical Data Access data in the Trend Viewer

11.2. Support Files

You will need the following file(s) to complete this lab:

- AcmeseuticalTemplate.tdf
- ACMEPillMachine.gdf
- ACMEMixers.gdf
- AcmeseuticalTrendConfig.mdb
- AcmeTrendData.mdf

11.3. Creating a Historical Trend Viewer

1. Open a new GraphWorX32 display by locating your AcmeseuticalTemplate.tdf and double-clicking on it.

2. Choose File → Save and save the display as ACMEHistoryTrends.gdf, checking that the file type is GraphWorX32 Displays without VBA (*.gdf). If the file name already exists, you may replace it.

3. Inside the display, insert a new **Trend Viewer ActiveX** by clicking the "ICONICS TWXView32 ActiveX Control" icon on the ActiveX toolbar.

4. Resize the Trend Viewer ActiveX control so that it fills up most of the display, as depicted in Figure L11.1.

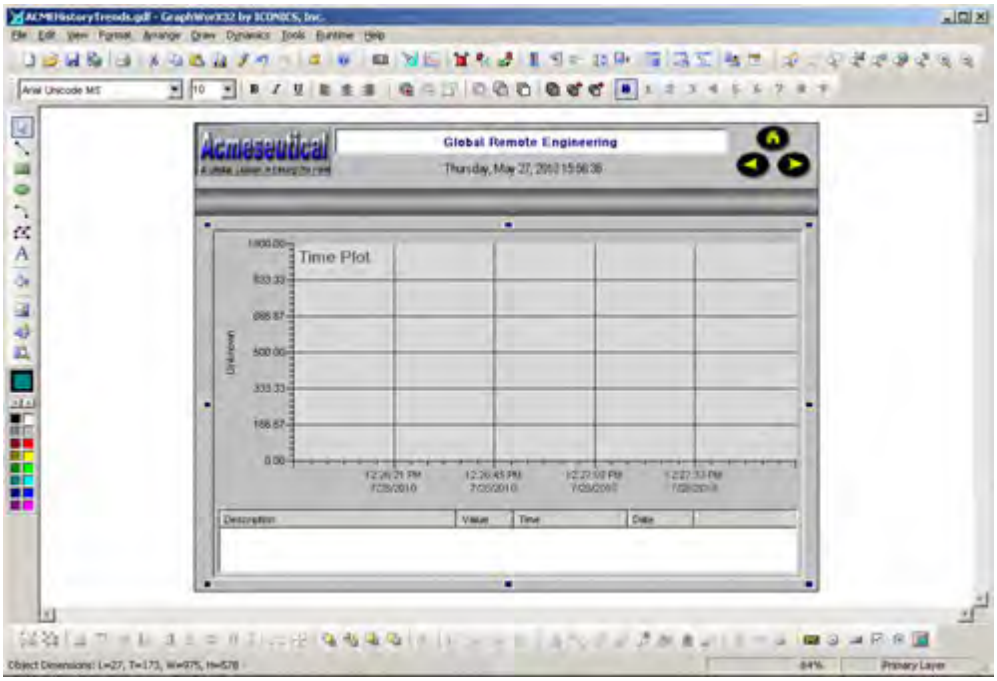


Figure L11.1 - Trend Viewer inside the Display

11.3.1. Configuring the Trend Viewer

In this application, the Trend Viewer will be configured to show data in a linear time-based plot.

1. Double-click on the Trend Viewer ActiveX Control to open its property pages.

2. Select the General tab and select the "Time Plot" as the Plot Type, as depicted in Figure L11.2.

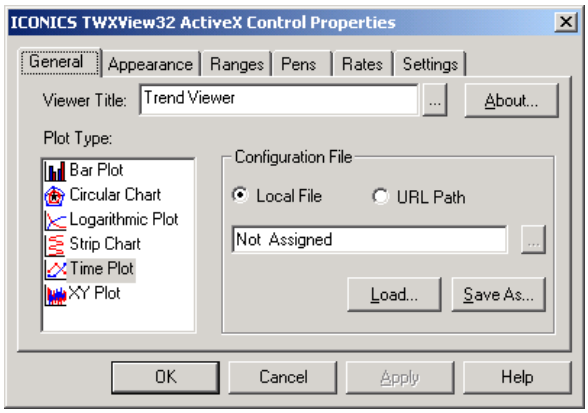


Figure L11.2 - Configuring the General Properties

3. Select the Rates tab to configure the data collection rate and trend period of the trend window.

4. Set the Data Collection Rate to five (5) seconds, as this was also the data collection rate used in the TrendWorX32 Logger. Although the data collection rates do not have to be the same between the Logger and Viewer, it is advised to make the Trend Viewer's data collection rate an integer multiple of the Data Logger's data collection rate.

5. Set the History Refresh Rate to sixty (60) seconds. The history refresh rate is the rate at which the viewer checks the database to determine if new data has been logged. Since the data logging rate set in the TrendWorX32 Logger is one minute, we do not need to pull the database more often than that.

6. Enter a Trend Period of thirty (30) minutes, as depicted in Figure L11.3.

Figure L11.3 - Configuring the Data Collection Rate

11.3.2. Configuring Trend Viewer Pens

The Trend Viewer will be connected to historical pens that we have configured in the last lab for visualization.

1. Select the Pens tab to add the logged signals to the plot.

2. Click on Tags Menu button → History Tag to open the Unified Data Browser.

3. Browse to My Computer → ICONICS.TWXSQLSvr.1 → Foxboro → Machine1, as depicted in Figure L11.4.

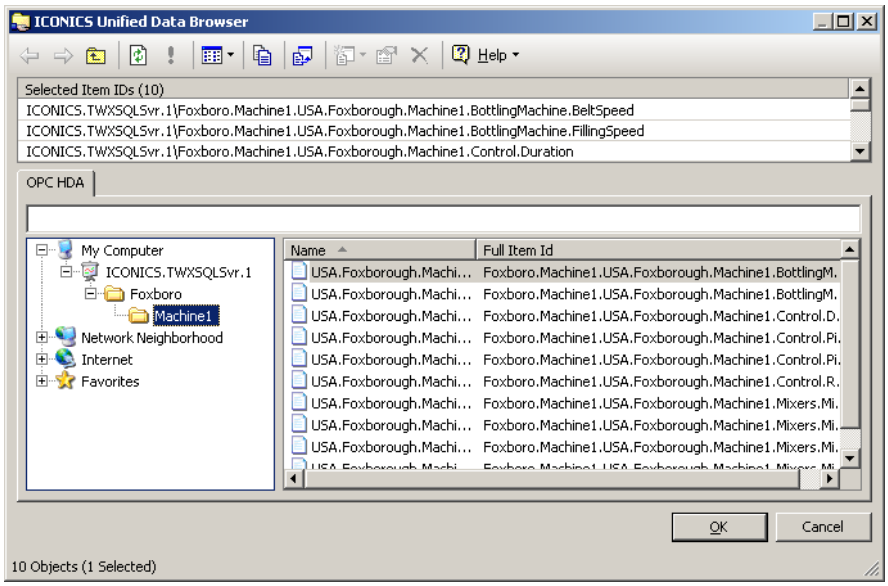


Figure L11.4 - Selecting the Logged Signals

4. Select Machine1 in the navigation tree on the left, right mouse click and select "Enter Up." All the tags under Machine1 should be added to the upper pane. Click on "OK" to return to the properties window which now has ten tags added to the list as depicted in Figure L11.5.

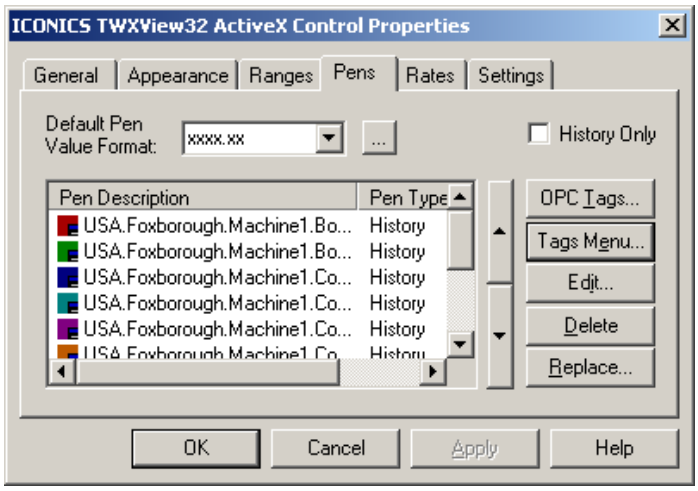


Figure L11.5 - Adding the Pens

5. Select the first pen called "USA.Foxborough.Machine1.Control.PillCount," and click on "Edit" to open the **Edit Trend Pen** dialog.

6. Enable "Stair Plot" to draw straight lines only.

7. Enable "Pen Marker" to enable the marker.

8. Change the Filter to "Raw Data" to request only actual samples from the TrendWorX32 Logger, as depicted in Figure L11.6.

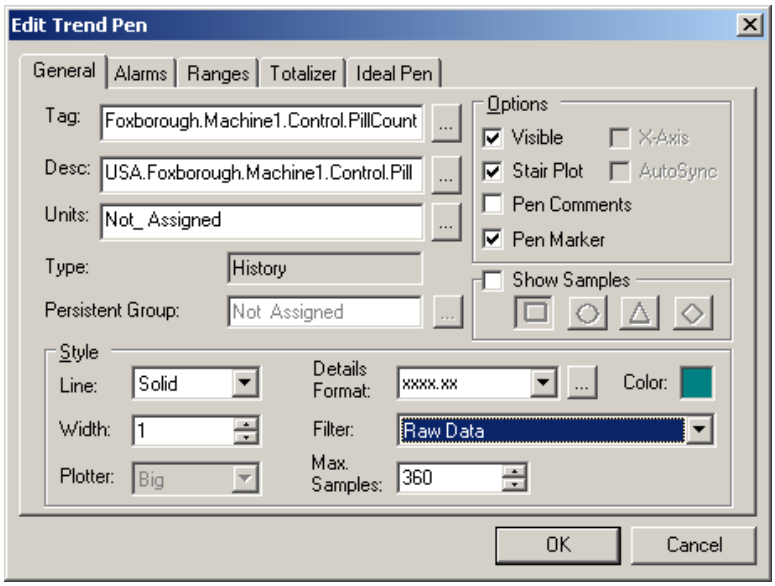


Figure L11.6 - Configuring the Pen Details

9. Click on "OK" to return to the "Pens" property page.

10. Repeat steps 5 - 9 for the Duration, PillsPerhour, and Rejected tags as highlighted in Figure L11.7. Only one pen can be edited at a time.

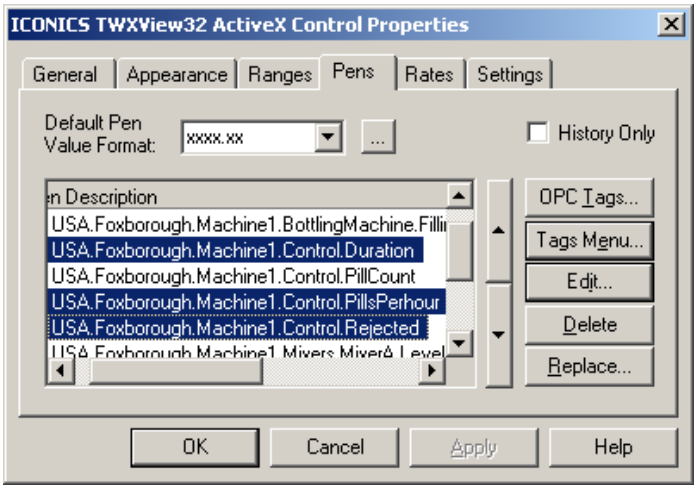


Figure L11.7 - Pens to Edit

11.3.3. Configuring the Trend Viewer Details Window

At the bottom of the Trend Viewer, there is a details window. You can configure the details to show information about the historical samples.

1. Select the Appearance tab.

2. Select the Details page, as depicted in Figure L11.8.

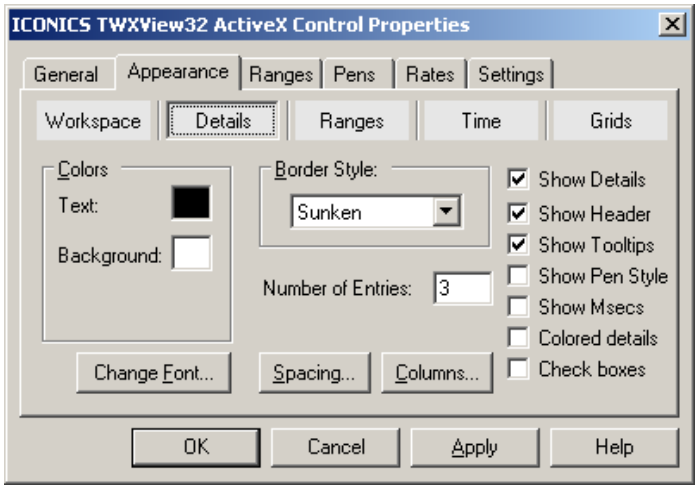


Figure L11.8 - Configuring the Details Appearance

3. Click on the "Columns" button to open the **Detail Window Columns** dialog, as depicted in Figure L11.9.

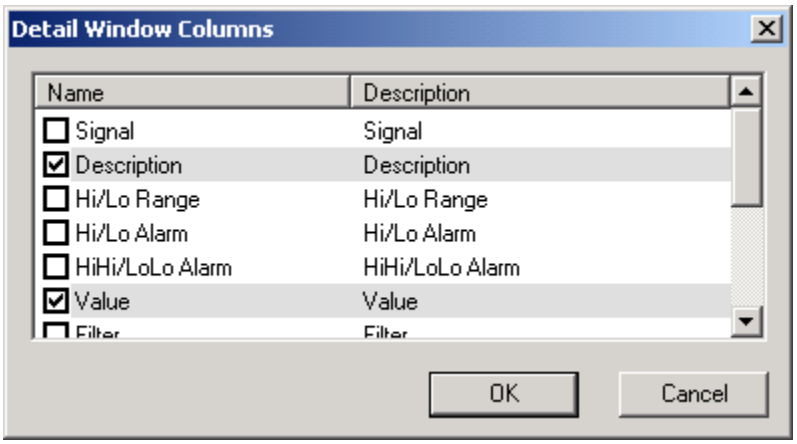


Figure L11.9 - Configuring the Details Window Columns

4. Enable the Description, Value, Quality, Time, and Date check boxes.

5. Click on "OK" to return to the Appearance tab.

6. Click on the "Spacing" button to configure the relative spacing of the selected columns. Adjust the column sizes so that all columns fit in the window properly as depicted in Figure L11.10.

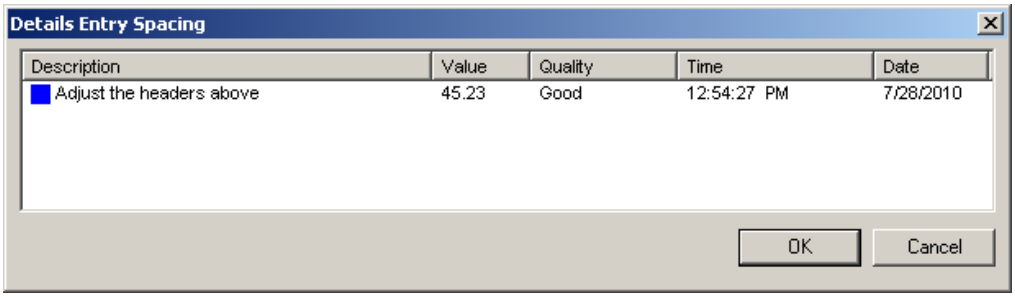


Figure L11.10 - Configuring the Column Spacing

7. Click on "OK" to accept the column spacing and to return to the Appearance tab.
8. Select the checkbox for the "Check boxes" option.
9. Select the Ranges tab and enable "Global Ranges." Set the range to be from 0-100.
10. Click on "OK" to close the property pages and return to GraphWorX32. Go to File → Save to save the display.

11.3.4. Trend Viewer in Runtime

1. Click on the "Runtime" menu to start runtime mode and look at the historical data, as depicted in Figure L11.11.

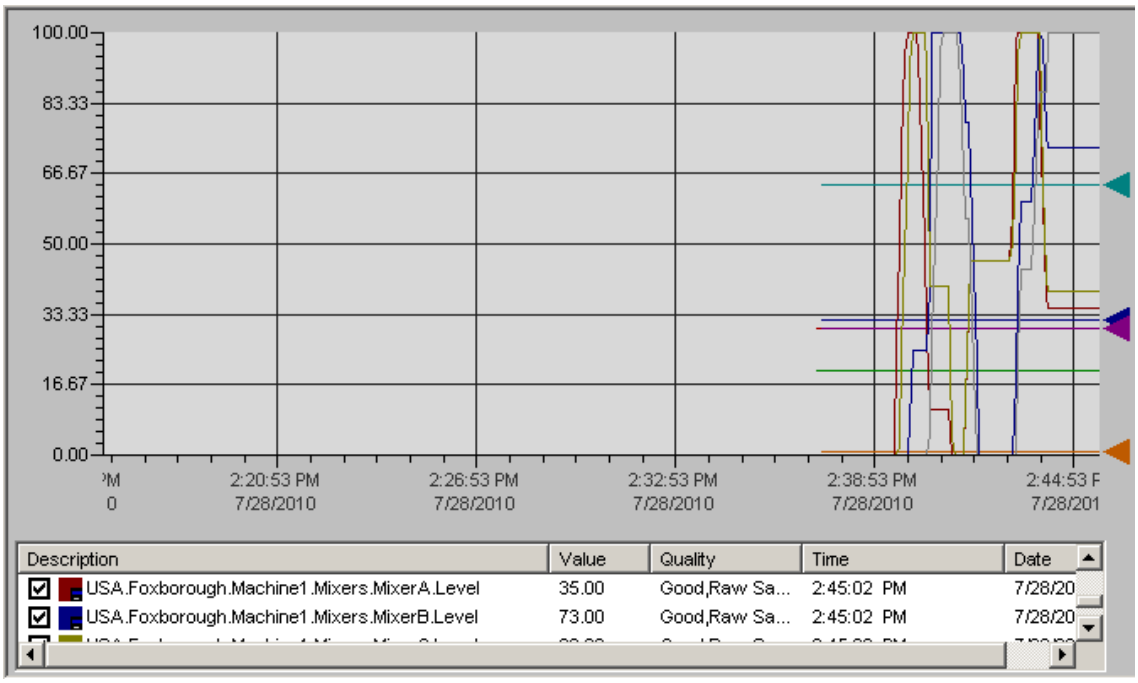


Figure L11.11 - Trend Viewer Display in Runtime

2. The checkboxes to the left of the tags control the visibility of the pens. Uncheck anyone of them to hide the pen.

3. Double-click on the Trend Viewer to open the **Trend Viewer** toolbar.

4. Click on the "Pause" button to freeze/stop the Trend Viewer.

5. Once the Trend Viewer is stopped, press and hold down the "Shift" key and drag the mouse pointer across the window to zoom into a particular area, as depicted in Figure L11.12.

6. It is also possible to drag the mouse pointer across the trend to show the historical values in the details window at the bottom of the trend.

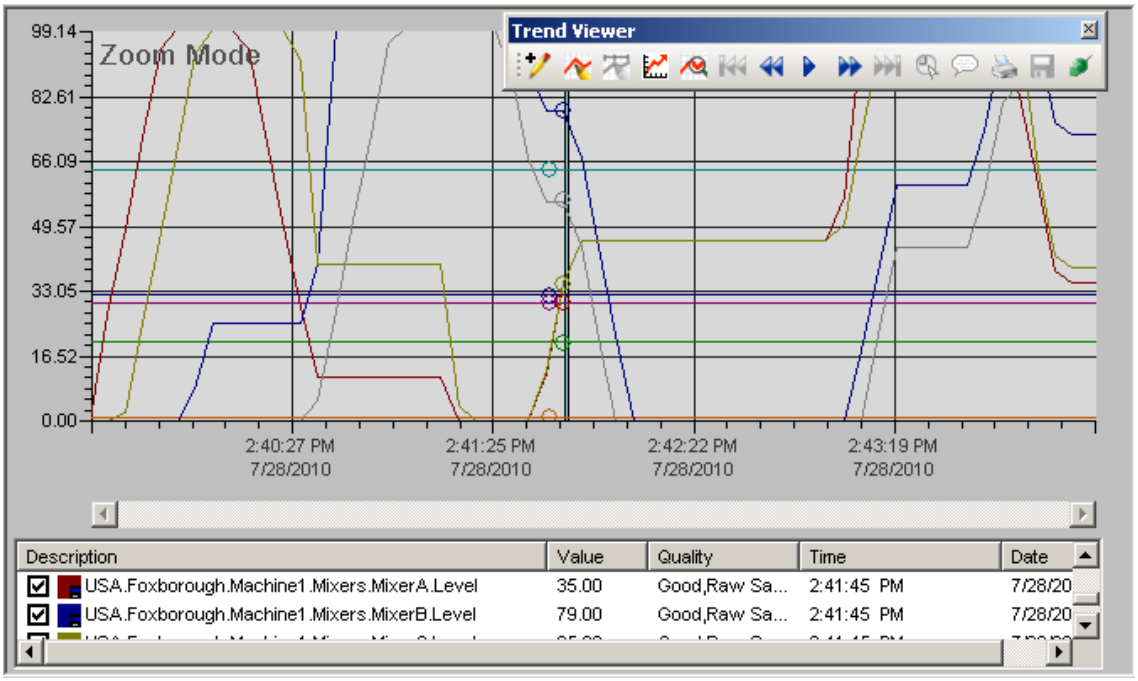


Figure L11.12 - Zoom into the Trend Window

Note: The display in Figure L11.12 is for reference only. Your display will not be populated with as much data.

11.4. Linking the Pill Machine Display with Trend History

Now that we have a display that shows us the history trend data, we want to add a load display pick action to open it while we are looking at the pill machine.

1. Open the ACMEPillMachine.gdf display.
2. Select layer one and duplicate the "History Alarms" button that is already on the layer.
3. Double-click on the duplicated button to open its Property Inspector.
4. On the Button tab change the text to read "History Trends."
5. Select the Pick tab and change the file name to "ACMEHistoryTrends.gdf".
6. Click on "OK" to save the changes and close the Property Inspector.
7. Move the two buttons to the upper left-hand corners of the display.
8. Save the display. Your finished display may look like Figure L11.13.

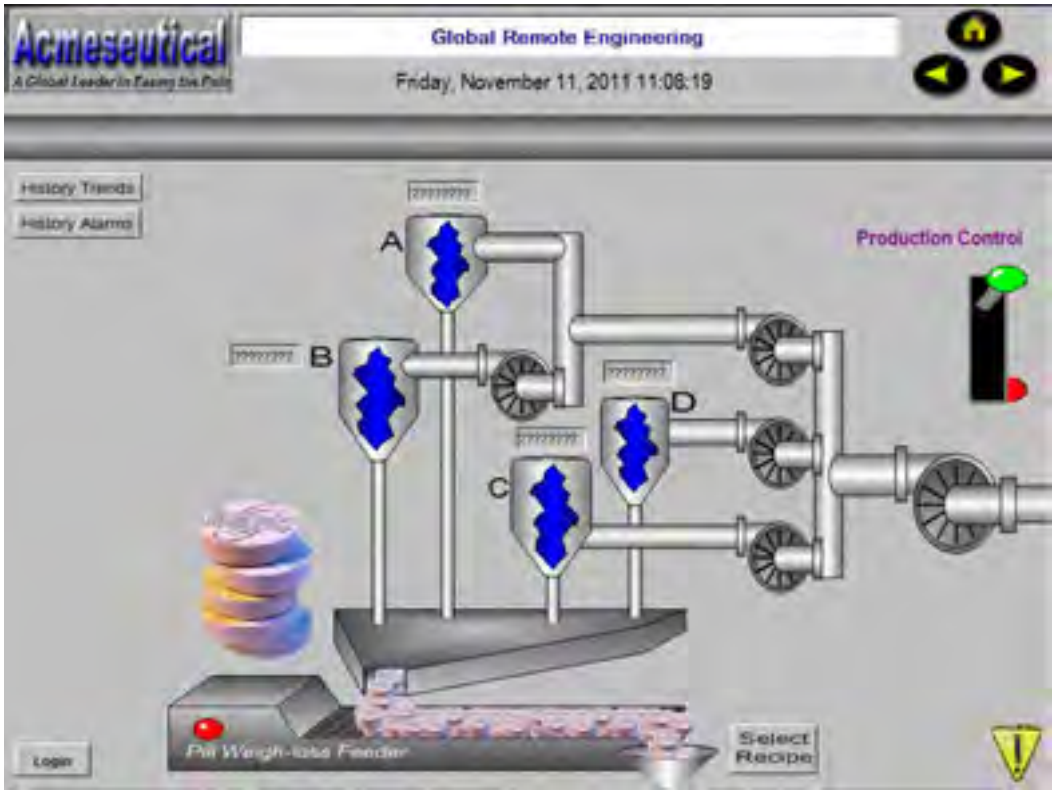


Figure L11.13 - ACMEPillMachine.gdf

Review: TrendWorX32 Viewer R11

11.1. Chapter Objectives

In this chapter, we talked about how you can configure the TrendWorX32 Viewer ActiveX Control to view real-time and history pens. You should now know how to do the following:

- Select the appropriate chart type
- Configure the details view of the viewer
- Set the y-axis range
- Add a real-time or history pen into the viewer
- Configure some properties of a pen
- Configure the rates for the control
- Bring up and use the runtime toolbar

11.2. Summary

The TrendWorX32 Viewer is an ActiveX Control that can retrieve OPC DA and HDA data. Although the viewer allows viewing for both types of data, you can configure the control to show history data only.

In the TrendWorX32 Viewer Properties Dialog, there are six tabs available: General, Appearance, Ranges, Pens, Rates, and Settings. Each tab contains settings in the named category. In the General tab, you can configure the title of the viewer, choose a plot type, and specify a configuration file if you have one.

On the Appearance tab, you can configure how the viewer looks. There are five pages on this tab: Workspace, Details, Ranges, Time, and Grid. Each of these tabs refers to a separate area of the viewer that you can configure. For example, the Details page contains settings such as Number of Entries and Columns for you to define how many pens the details window should show at once without scrolling and the different columns (or pen attributes) available for viewing.

Each pen that you add to the TrendWorX32 Viewer contains its own configurable properties. You can double-click on a pen in the Pens tab to bring up the Edit Trend Pen dialog. In this dialog, you can set the description of the pen as well as check options such as Pen Marker, Stair Plot, Show Samples, and much more.

On the Rates tab, you can configure how often the viewer will sample a real-time pen (Data Collection Rate), how often the viewer will refresh (Display Refresh Rate), and how much data should display in the trend window (Trend Period). If you are displaying history data, you can also configure the History Refresh Rate, which controls how often the viewer will refresh historical pens with new data. It is important to know that the trend viewer can only hold up to 7200 samples when configuring rates. If the combination of the Data Collection and Trend Period exceeds the maximum samples allowed, the system will adjust the variables on the page to limit the samples to 7200.

Once you have finished configuring the viewer, go into runtime and watch the pens trend in the chosen plot type. The TrendWorX32 Viewer comes with a runtime toolbar. Using the Runtime toolbar, you can add a pen, change the rates, freeze the trend, and look at trend statistics, just to name a few.

11.3. Terms

Data Collection Rate	Display Refresh Rate	Freeze Mode
History Refresh Rate	Set Right Time	Trend Period
TrendWorX32 Viewer		

11.4. Test Your Knowledge

In this chapter, we discussed how to configure the TrendWorX32 Viewer ActiveX control to play real-time and history data. You should be able to answer the following questions.

1. Which of the following is NOT a type of data that the TrendWorX32 Viewer can display?

a. Real-time OPC DA values
b. Persistent values
c. History OPC HDA values
d. OPC UA values

2. True or False: You can add new pens into the TrendWorX32 Viewer during runtime.

3. Which of the following is NOT an available plot type?

   a. Bar Plot
   b. Circular Chart
   c. Logarithmic Plot
   d. Time Plot
   e. None of the Above (All of the above plots are available)

4. Where would you go to configure the pen range for a specific pen?

   a. On the Ranges tab
   b. On the Ranges page on the Appearance tab
   c. On the Ranges tab in the Edit Trend Pen dialog
   d. You cannot configure the pen range, it is pulled in automatically from the OPC server and you cannot change it.

5. What is the maximum number of samples the TrendWorX32 Viewer can hold?

   a. 3200
   b. 7200
   c. 64000
   d. Unlimited

For the next three questions, please refer to Figure R11.1

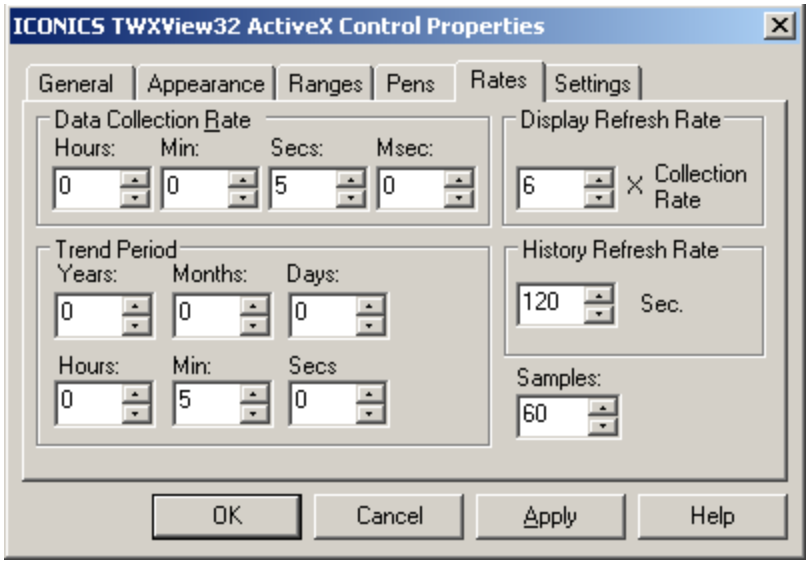


Figure R11.1 - TrendWorX32 Viewer Rates Configuration

6. True or False: In Figure R11.1, the control will display data for ten minutes at a time.

7. How often would the trend viewer update the display with real-time data with settings shown in Figure R11.1?

   a. 5 seconds
   b. 30 seconds
   c. 60 seconds
   d. 120 seconds

8. How often would the trend viewer sample a value for a real-time pen with settings shown in Figure R11.1?

   a. 5 seconds
   b. 30 seconds
   c. 60 seconds
   d. 120 seconds

9. If you want a pen to show you the collected values, which option would you turn on?

   a. Visible
   b. Pen Marker
   c. Show Samples
   d. Stair Plot

10. True or False: You can only use a Filter on a history pen; the option is NOT available for a real-time pen.

TrendWorX32 Reporting 12

In the previous chapters we learned how to use the TrendWorX32 Logger to record trend data into a database and the TrendWorX32 Viewer to view the logged data (and real-time data) in a graphical container. In this chapter we will discover how to use the logged data to generate a simple report for analysis. By the end of the chapter you will be familiar with the following concepts:

- The TrendWorX32 Reporter
- How to configure a TrendWorX32 Report
- The different available file type
- How to run a report
- Using the SQL Query Engine

12.1. Reporting with TrendWorX32

TrendWorX32 has the capability to log data to a relational database. In addition to the standard tools available for accessing such data, TrendWorX32 has built-in capability for historical data utilities and reporting. The tools available to perform historical data utilities and reporting are:

- TrendWorX32 Reporting
- TrendWorX32 SQL Tool ActiveX
- TrendWorX32 SQL Query Engine

The tools mentioned above help you retrieve history data logged by TrendWorX32 Logger. The reports generated from these tools are simple and basic. If you need to create more sophisticated reports you should consider using ReportWorX, which is in our BizViz product family.

12.2. TrendWorX32 Reporter

The TrendWorX32 Reporter allows you to create a report using your trend data logged with TrendWorX32 Logger. You can export the data specified to a text file (csv format), an Excel worksheet, or a Microsoft Access database.

In configuration mode you can add, edit, or delete reports without report processing taking place. You can access the Report Configurator by going to Start → All Programs → ICONICS GENESIS32 → TrendWorX32 → TrendWorX32 Reporting. To add a select report, **Add Report** from the **Edit** menu on the toolbar. This opens the General dialog where you can enter a Report Name: and a Target Name:. Clicking on the Next button will bring you into the **Report Configuration Wizard** which is a series of dialogs that guide you through the report configuration. Once you have configured the report you can also edit the report. The edit dialog contains six tabs, each of which appear in order when you configure a new report.

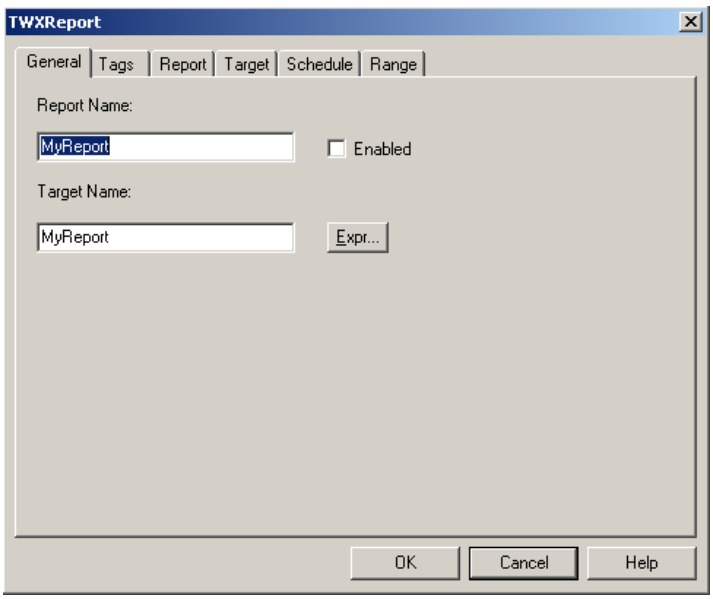


Figure 12.1 - Report Configuration Properties

The available tabs are:

- **General:** You can configure the Report Name and Target Name on this tab. The Report Name is the name of the report you are configuring. The Target Name will be the base file name to use when generating the report. You can also enable the report on this tab.
- **Tags:** On the tags tab you need to choose a connection to your logging database. This should be the same ODBC Connection you used for data logging. You will also need to choose the Database Group, Logging Group, and the Tags against which you wish to report.
- **Report:** You can choose a Data Filter on this tab. Many times you may need to create reports that perform calculation instead of spiting raw values into a report. You can use the Data Filter to generate reports with average, max, min, totalizer, and many other filters to give the report more meaning.

- **Target:** Here you can choose the file format of the report. You can use a Database table, a text file (CSV format), or an Excel worksheet. Depending on the file type you choose you will see different options on the configuration tab.
- **Schedule:** On the Schedule tab you can configure how often the report runs. Your options are:
  - **Immediately:** Runs once immediately when the TrendWorX32 Reporting starts. The report is disabled after running.
  - **One Time:** Runs once at a specific date and time. The report is disabled after running.
  - **Periodic:** Runs periodically at the specified interval.
  - **On Event:** Runs when the condition comes true. It could be when a certain OPC tag reaches a certain value.
- **Range:** The Range tab allows you to specify the date/time range during which you would like to view data for in the report.

Once you have configured your reports, you can click on the traffic light in the toolbar to turn it green to start the TrendWorX32 Reporting engine. If there are immediate reports, those reports will run and become disabled similar to "MyReport" shown in Figure 12.2. Notice that the report has a red "X" on its icon to indicate that the report is disabled.

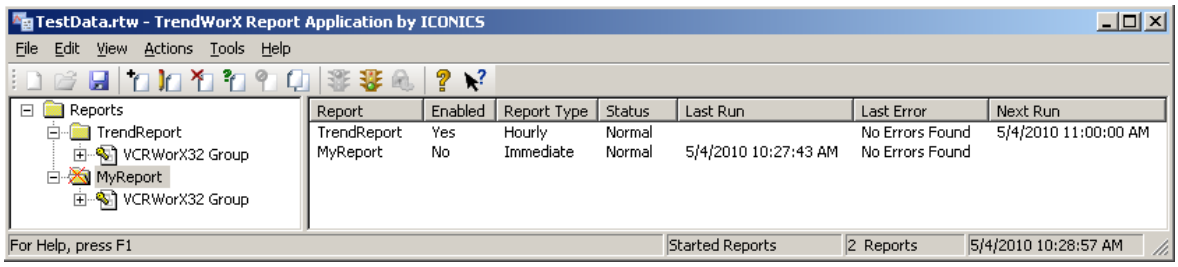


Figure 12.2 - TrendWorX32 Reporting in Runtime

Also, notice that if you have scheduled reports like the "TrendReport" in Figure 12.2, the right-hand pane will show you statistic information such as what type of report it is, when it last ran, and when it will run next. This information window can be very helpful if a report fails to run. "The Last Error" column will show you why the report failed to run. You can then rectify the problem(s) based on the error message.

12.3. TrendWorX32 SQL Tool Control

TrendWorX32 also provides an ActiveX Control for you to retrieve data in the HMI interface. You can find the TWXSQL Tool Control by clicking on the "Insert ActiveX Control/OLE Object" button in GraphWorX32.

Once you insert the control into a GraphWorX32 display you can double-click on it to configure some basic settings like the connection string for the database and the signal for which you would like to view data.

This control is invisible during runtime and you will need to add some VBA or VB code to make it report data. Since scripting is out of the scope of this class, we will not get into details of what is required. You can find an example of how to use the control in \Program Files\ICONICS\GENESIS32\Examples\TrendWorX32\TrendWorX Data Logging and Reporting\TrendSQLTool.t32. Although this is a TrendWorX32 container file the idea and code should be the same for GraphWorX32. We will discuss more about this ActiveX control in the **GENESIS32 Advanced** course.

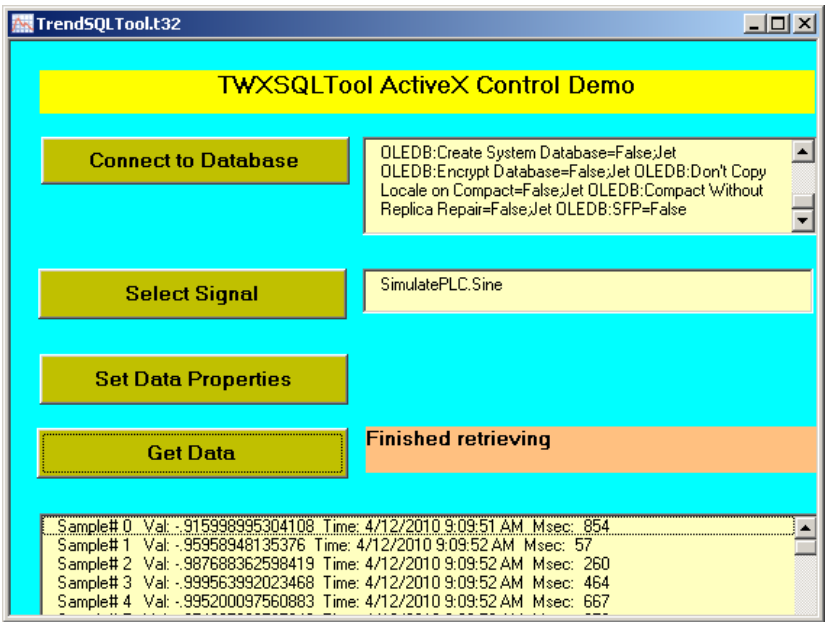


Figure 12.3 - Example of TrendWorX32 SQL Tool ActiveX Control Capabilities

12.4. TrendWorX32 SQL Query Engine

In version 9.20 we added a TrendWorX32 SQL Query Engine. Using this engine you can create your own Data Manipulations Language (DML) queries to retrieve history data. In order to use the TrendWorX32 SQL Query Engine you will need to configure it first.

To configure the SQL Query Engine you will need to bring up the TrendWorX32 Logger provider in Workbench32. Click on the "SQL Query Engine Setup" button in the toolbar. This will bring up the **TrendWorX32 Logger SQL Query Engine Setup** dialog as shown in Figure 12.4.

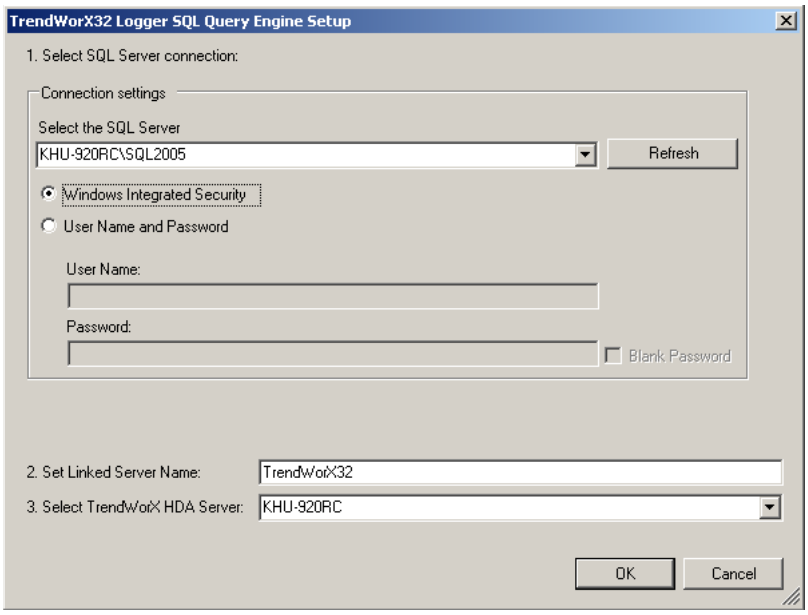


Figure 12.4 - TrendWorX32 Logger SQL Query Engine Setup Dialog

You will need to choose a SQL Server, for example (local)\sqlexpress and the authentication method. Everything else you can leave as default. Click on the "OK" button when you are done with the configuration will add a linked server to the provided SQL Server connection with a name specified in the "Set Linked Server Name" field. If you look in the SQL Server Management Studio, you should find the Linked Server as shown in Figure 12.5. From here, you can query the data using the linked server object.

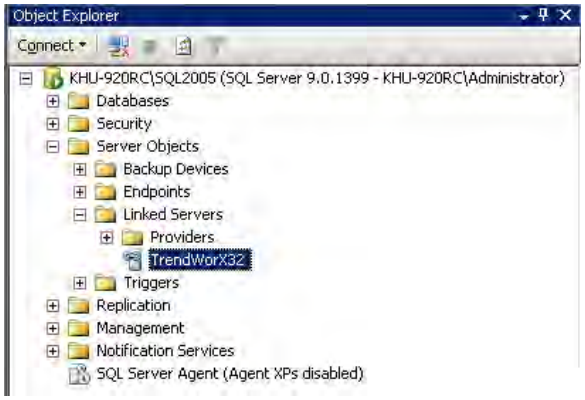


Figure 12.5 - Linked Server in SQL Server Management Studio

You can use the TrendWorX32 SQL Query Engine in many different clients including Data Mining Configurator, ReportWorX, Microsoft Excel, and any other applications that can run SQL queries.

For our discussion we will simply show the query syntax and its results directly within the SQL Server Management Studio. It is simple to retrieve data using the SQL Query Engine. Let us first try to recover the database groups and tables in our logging database.

> **NOTE:** In order to use the SQL Query Engine, you must have an active TrendWorX32 Logger configuration.

To reveal the list of database groups configured you can click on the "New Query" button in the SQL Server Management Studio. In the new query tab run the stored procedure sp_catalogs against the linked server. Your query and its result may look similar to Figure 12.6.

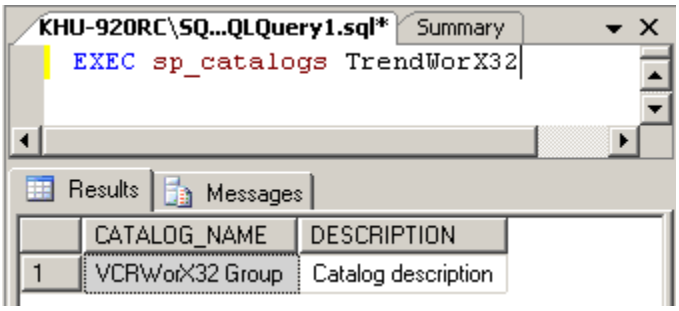


Figure 12.6 - Stored Procedure to Return a List of Configured Database Groups

Using a similar stored procedure you can return a list of table in the logger. To do this, run the stored procedure sp_tables_ex against the linked server as shown in Figure 12.7.

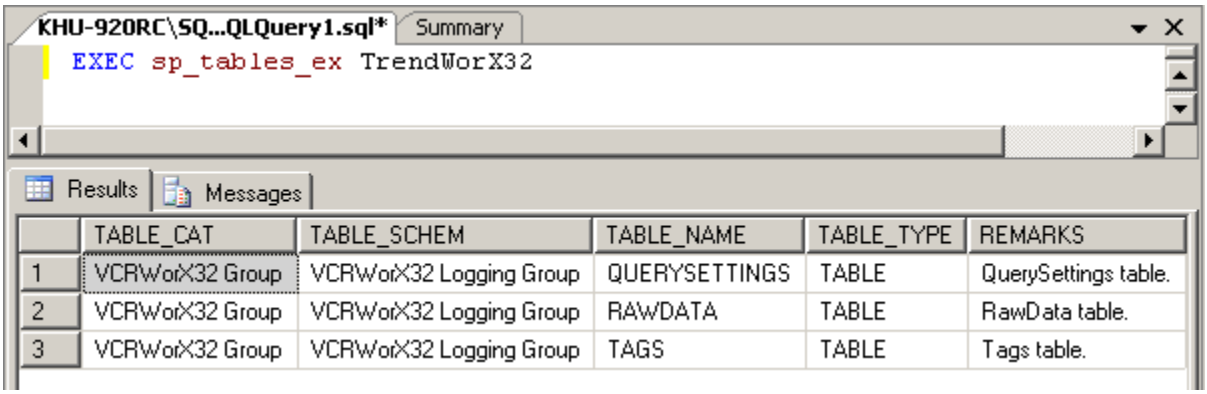
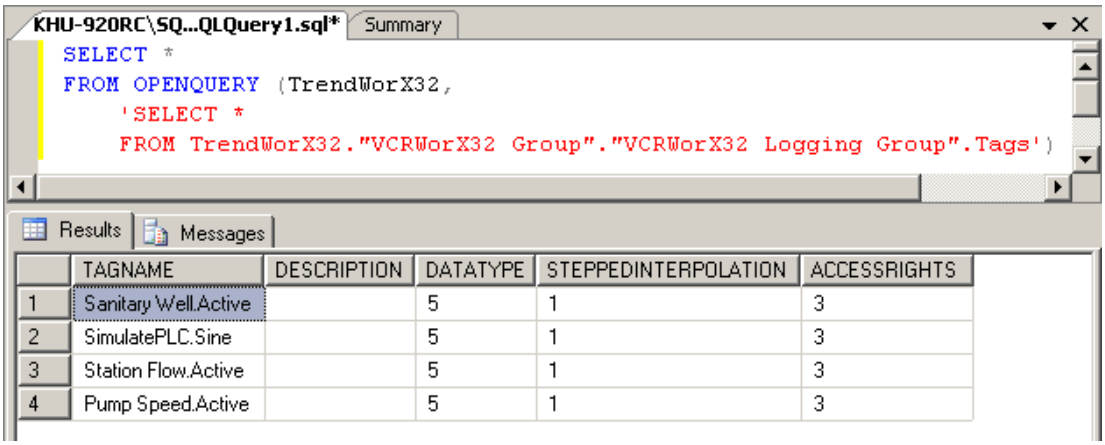


Figure 12.7 - Stored Procedure to Return a List of Available Tables

Notice in Figure 12.7 there is a RawData table. This table will be the one that we will query against to retrieve data. However, before we can retrieve data using SQL query, we will need to know the names of the available trending tags. To find out the available tags, you can run a query against the Tags table as shown in Figure 12.8.



	TAGNAME	DESCRIPTION	DATATYPE	STEPPEDINTERPOLATION	ACCESSRIGHTS
1	Sanitary Well.Active		5	1	3
2	SimulatePLC.Sine		5	1	3
3	Station Flow.Active		5	1	3
4	Pump Speed.Active		5	1	3

Figure 12.8 - Retrieving a List of Available Tags

Notice that we are using the OPENQUERY function here. This will help us improve the performance of the SQL query as well as enable us to use special keywords. If you wish to, you can still write the SQL query directly. However, know that there are many disadvantages. You may not be able to use some of the special keywords, you will need to start with the linked server name, some of the columns may be converted to use a different data type, and it may take longer for the query to return data just to name of few of the disadvantages. Because of this, we recommend that you use the OPENQUERY function.

The OPENQUERY function has two parameters: the linked server and the query string enclosed in single quotes. As mentioned before, there is a Tags table that contains all of the tags. However, it only returns the tags within a certain logging group of a database group. Therefore, the table name you see in Figure 12.8 is of the format *LinkedServer*,"*DatabaseGroup*,"and "*LoggingGroup*" Tags, where the *LinkedServer* is the one you created in Workbench32 earlier. The *DatabaseGroup* is the database group configured in the TrendWorX32 Logger configuration; the *LoggingGroup* is the logging group created within the database group. It is very useful to know your TrendWorX32 Logger configuration structure when using the SQL Query Engine.

> ***Tip!*** You may want to keep your TrendWorX32 Configurator (or the TrendWorX32 provider in Workbench32) open. This way, you can easily refer back to the configuration for your Database Group and Logging Group names.

NOTE: For more information about the OPENQUERY function, please refer to Microsoft documentations. You can find some basic information on http://msdn.microsoft.com/en-us/library/ms188427.aspx.

Now that we know the available tags in the database let us try to retrieve some data for a particular tag. All tags within the same logging group configured in TrendWorX32 Logger have a table from which you can retrieve data. The naming convention of the table is similar to the Tags table, except, instead of using Tags at the end you will use RawData. The RawData table contains the logged values of all the tags within a single logging group. To retrieve data you can use a similar SQL statement as shown in Figure 12.9. If you want to report on a single tag or only within certain time frames you can add SQL WHERE statements to refine the query results.

KHU-920RC\SQ...QLQuery1.sql* | Summary

```
SELECT *
FROM OPENQUERY (TrendWorX32,
    'SELECT *
    FROM TrendWorX32."VCRWorX32 Group"."VCRWorX32 Logging Group".RawData')
```

Results | Messages

	TAGNAME	TIMESTAMP	QUALITY	VALUE
515	SimulatePLC.Sine	2010-05-04 17:26:01.020	262336	0.86265480518341064
516	SimulatePLC.Sine	2010-05-04 17:26:01.033	262336	0.86738342046737671
517	SimulatePLC.Sine	2010-05-04 17:26:01.050	262336	0.87234228849411011
518	Sanitary Well.Active	2010-05-04 17:26:05.940	262336	0
519	Station Flow.Active	2010-05-04 17:26:05.940	262336	0
520	Pump Speed.Active	2010-05-04 17:26:05.940	262336	1
521	Sanitary Well.Active	2010-05-04 17:26:09.937	262336	1
522	Station Flow.Active	2010-05-04 17:26:09.937	262336	1
523	Pump Speed.Active	2010-05-04 17:26:09.937	262336	0
524	Sanitary Well.Active	2010-05-04 17:26:14.937	262336	0
525	Station Flow.Active	2010-05-04 17:26:14.937	262336	0
526	Pump Speed.Active	2010-05-04 17:26:14.937	262336	1
527	Sanitary Well.Active	2010-05-04 17:26:19.933	262336	1
528	Station Flow.Active	2010-05-04 17:26:19.933	262336	1
529	Pump Speed.Active	2010-05-04 17:26:19.933	262336	0
530	Sanitary Well.Active	2010-05-04 17:26:25.930	262336	0

Figure 12.9 - Data within a Logging Group

As you can see from Figure 12.9, there are four columns in the table: TagName, Timestamp, Quality, and Value. For each sample we take from an OPC tag we will log the timestamp when we take the sample, the quality at the time we took the sample, and the value of the sample. As you can see, the quality value is a big number but you may not necessarily know what that means. We can use the QualityToStr() function to translate this into a string that makes sense for us.

KHU-920RC\SQ...QLQuery1.sql* | Summary

```
SELECT *
FROM OPENQUERY (TrendWorX32,
    'SELECT TagName, Timestamp, Value, QualityToStr(Quality) AS Quality
    FROM TrendWorX32."VCRWorX32 Group"."VCRWorX32 Logging Group".RawData
    WHERE TagName = ''SimulatePLC.Sine''
        AND [Timestamp] > ''2010-04-01''
        AND [Timestamp] < ''2010-05-01'''')
ORDER BY [Timestamp] DESC
```

Results | Messages

	TagName	Timestamp	Value	Quality
1	SimulatePLC.Sine	2010-04-12 17:01:22.390	-0.81488519906997681	Good
2	SimulatePLC.Sine	2010-04-12 17:01:22.187	-0.88199543952941895	Good
3	SimulatePLC.Sine	2010-04-12 17:01:21.987	-0.9347761869430542	Good
4	SimulatePLC.Sine	2010-04-12 17:01:21.783	-0.9723699688911438	Good
5	SimulatePLC.Sine	2010-04-12 17:01:21.580	-0.99423360824584961	Good
6	SimulatePLC.Sine	2010-04-12 17:01:21.377	-0.99979788064956665	Good
7	SimulatePLC.Sine	2010-04-12 17:01:21.173	-0.98911875486373901	Good
8	SimulatePLC.Sine	2010-04-12 17:01:20.970	-0.96236985921859741	Good
9	SimulatePLC.Sine	2010-04-12 17:01:20.767	-0.91998565196990967	Good
10	SimulatePLC.Sine	2010-04-12 17:01:20.563	-0.86265480518341064	Good
11	SimulatePLC.Sine	2010-04-12 17:01:20.360	-0.79130876064300537	Good
12	SimulatePLC.Sine	2010-04-12 17:01:20.157	-0.70710659027099609	Good
13	SimulatePLC.Sine	2010-04-12 17:01:19.953	-0.61091905832290649	Good

Figure 12.10 - Data for Simulation Sine Tag within a Month

Figure 12.10 shows a query statement that will return data for the SimulatedPLC.Sine tag for the month of April using the QualityToStr() function. The returned table is sorted by the timestamp from the latest data to the oldest. Notice the Quality column now shows us “Good” instead of a number that we cannot interpret.

> **NOTE:** The TrendWorX32 SQL Query Engine supports a subset of functions compared to native SQL. You can find a list of supported functions and additional stored procedures in Appendix B.

Lab: Simple History Trend Report

L12

Acmeseutical has indicated that many of their operators like to look at their historical data with numbers rather than simply in a graphical trend. We will now create this simple report for the operators.

Estimated time to complete this lab: 20 minutes

12.1. Task Checklist

In this lab, you will complete the following tasks:

- Create a new TrendWorX32 report configuration
- Configure a database connection to the historical database
- Configure report properties and frequency
- Run the report

12.2. Support Files

You will need the following file(s) to complete this lab:

- AcmeseuticalTrendConfig.mdb
- AcmeTrendData.mdf

12.3. Creating a History Reporting System

1. Open the TrendWorX32 Report Configurator from Start → Programs → ICONICS GENESIS32 → TrendWorX32 → TrendWorX32 Reporting.

2. The TrendWorX32 Report Configurator will open, as depicted in Figure L12.1.

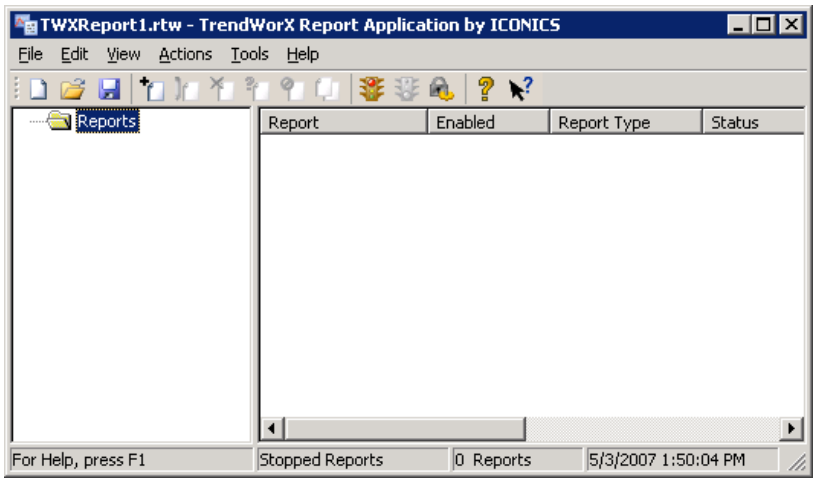


Figure L12.1 - Creating a New TrendWorX32 Report

3. Immediately choose File → Save and save this configuration file as AcmeseuticalTrendReports.rtw in your C:\Training directory.

4. In the Configurator, create a new report by selecting Edit → Add Report. A wizard will take you through all the steps necessary to have a complete report configuration.

5. The General dialog will be the first step of the wizard. Under "Report Name" enter "Machine1," and enter the same under "Target Name."

6. Also, check the "Enabled" checkbox, as depicted in Figure L12.2.

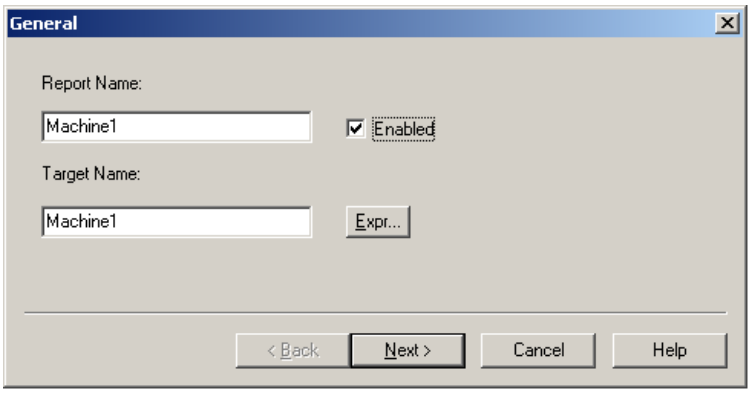


Figure L12.2 - General Report Properties

7. Click on "Next" to proceed to the **Tags** dialog.

8. Click on "**DSN**" to select the data source for this report.

9. In the **Select Data Source** dialog that comes up, click on the Machine Data Source tab.

10. Select the AcmeTrendData DSN that we created previously and are using to log our data, as depicted in Figure L12.3, and then click on "OK."

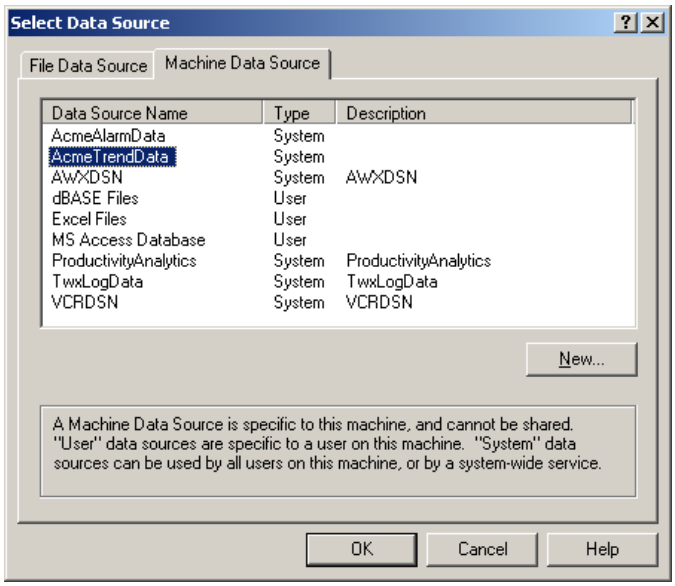


Figure L12.3 - Selecting the AcmeTrendData Data Source

11. The **SQL Server Login** dialog will appear, as depicted in Figure L12.4. But since the credentials of the currently logged-on user are used, nothing has to be configured here. Click on "OK" to return to the Trend Logger Configurator.

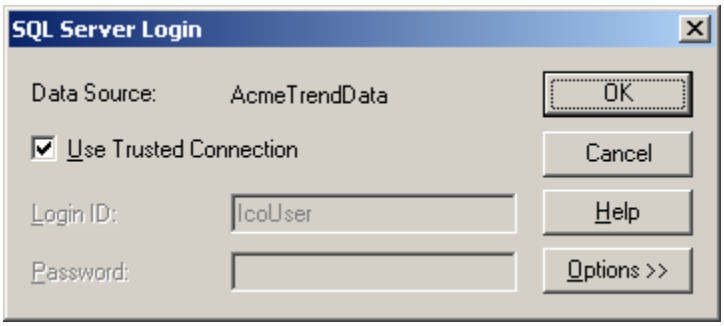


Figure L12.4 - SQL Server Login

12. Click on "DBase Group" to select the database group for this report.

13. Select the "Foxborough" database group that we have created previously, as depicted in Figure L12.5. Click on "OK" to make the selection and close the dialog.

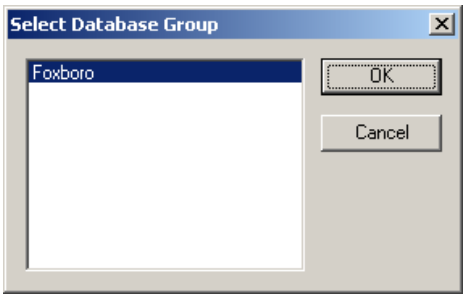


Figure L12.5 - Selecting the Database Group

14. Click on “Group” to select the logging group for this report.

15. Select the “Machine1” logging group that we have created previously, as depicted in Figure L12.6. Click on “OK” to make the selection and close the dialog.

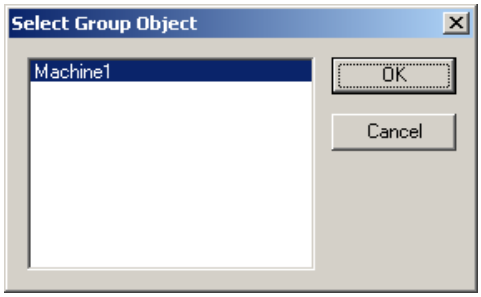


Figure L12.6 - Selecting the Logging Group

16. Click on “Tags” to select the tags from the logging group for this report.

17. Select all tags that we have created by selecting the first tag, hold down the “Shift” key and select the last tag.

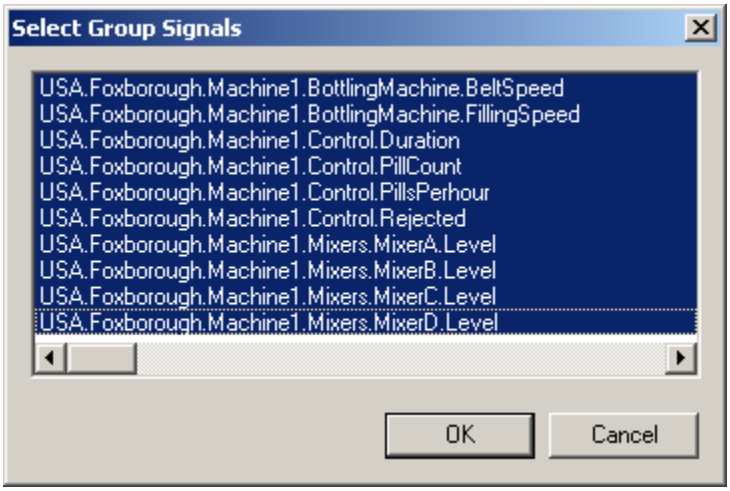


Figure L12.7 - Selecting the Tags from the Logging Group

18. Click on “OK” to return to the **Tags** dialog, as depicted in Figure L12.8.

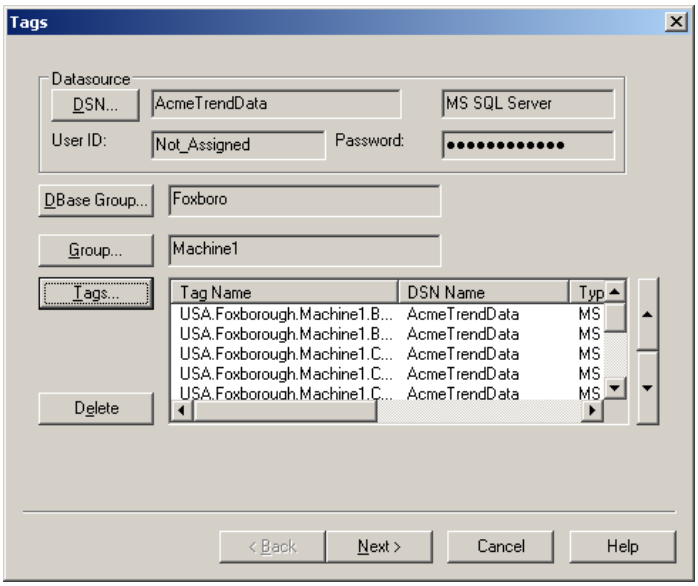


Figure L12.8 - Selected Tags Properties

19. Click on “Next” to proceed to the **Report** dialog.

20. In the **Report** dialog under “Data Filter,” select “Raw” to show only the actual logged values, as depicted in Figure L12.9.

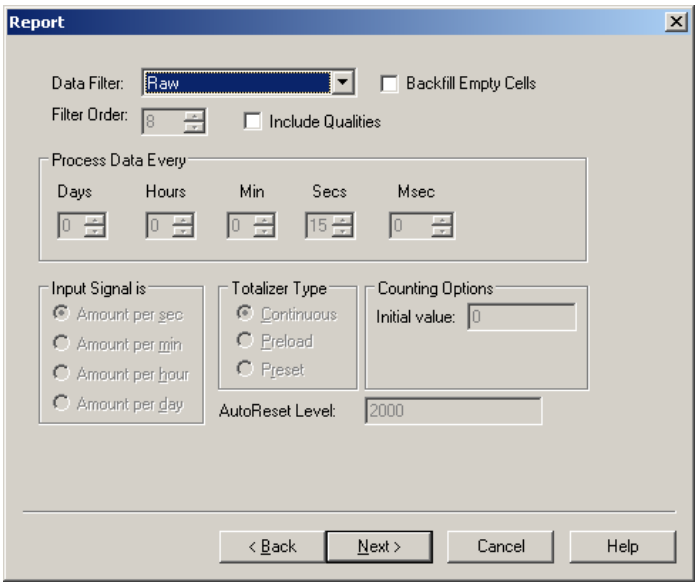


Figure L12.9 - Configuring the Report Data

21. Click on “Next” to proceed to the **Target** dialog, Select the “Text File” option.

22. Click on “Report Path” to select a directory in which the reports will be stored. We will store our reports in the “C:\Training” directory.

23. Select “Text File” and specify c:\Training as the Report Path. Select the **Create Report with Date** option, as depicted in Figure L12.10, to ensure each generated report has a different name.

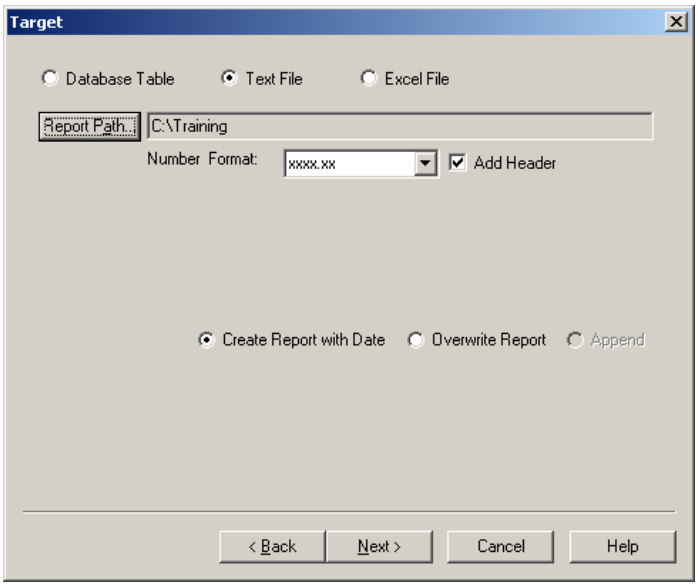


Figure L12.10 - Configuring the Report Target

24. Click on “Next” to proceed to the **Schedule** dialog. Select “**Periodic**” as the trigger.

25. Click on the “**Change**” button to change the period. In the **Edit Periodic Report** dialog, select “Hourly.”

26. Under the “Occurs every” section, specify zero (0) Hours and ten (10) Minutes, as depicted in Figure L12.11.

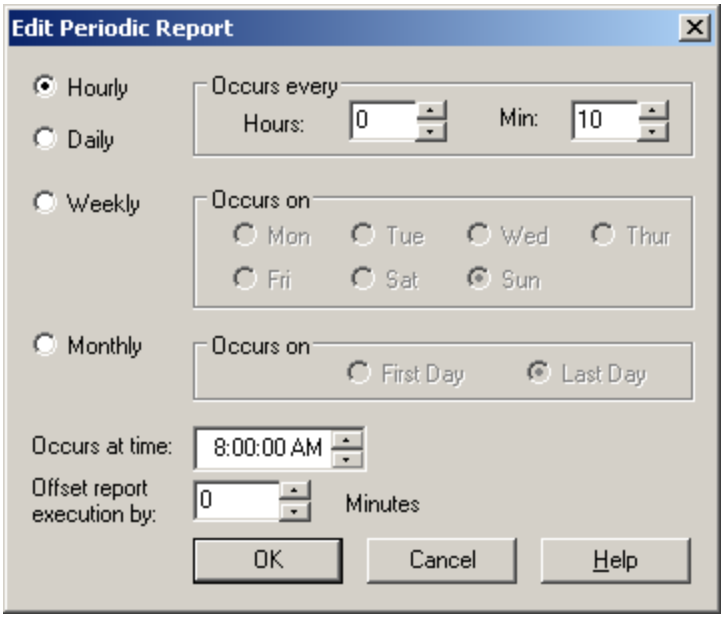


Figure L12.11 - Configuring the Periodic Report

27. Click on “OK” to close the periodic report configuration.

28. The schedule should now be configured, as depicted in Figure L12.12.

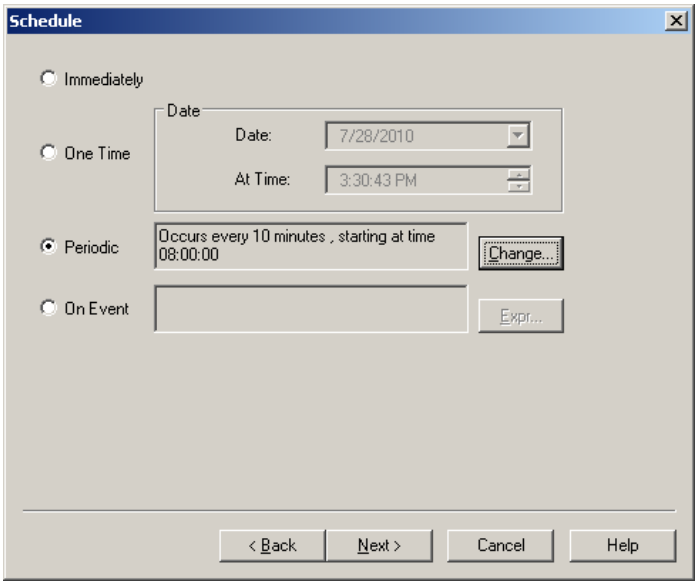


Figure L12.12 - Configuring the Report Schedule

29. Click on “Next” to proceed to the **Range** dialog.

30. Since the periodic report will use the data collected since the last time it ran, nothing has to be configured in the **Range** dialog, as depicted in Figure L12.13.

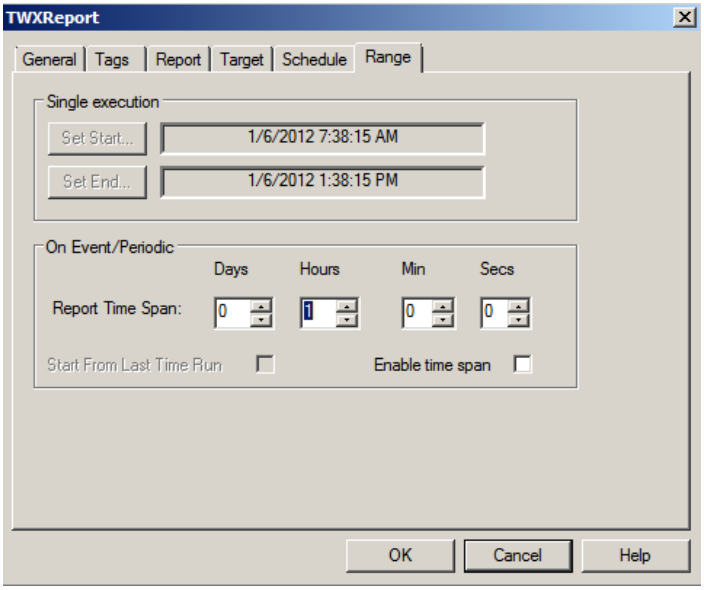


Figure L12.13 - Configuring the Report Range

31. Click on "Finish" to complete the configuration and return to the **TrendWorX32 Report Application**, as depicted in Figure L12.14.

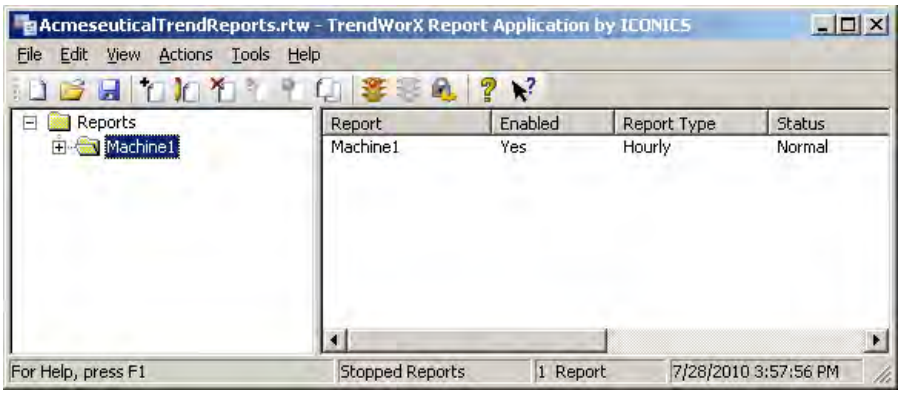


Figure L12.14 - Configured TrendWorX32 Report

32. Select File → Save to save the configuration.

33. Select Actions → Start Reports or click on the "Start Reports" button to start running the TrendWorX32 Report Application.

34. The report will run at the next 10-minute interval (e.g., 8:00, 8:10, 8:20, etc.).

35. You can monitor the status of your reports from the Configurator. You will be able to see when the last report ran and if there were any problems with it, as depicted in Figure L12.15.

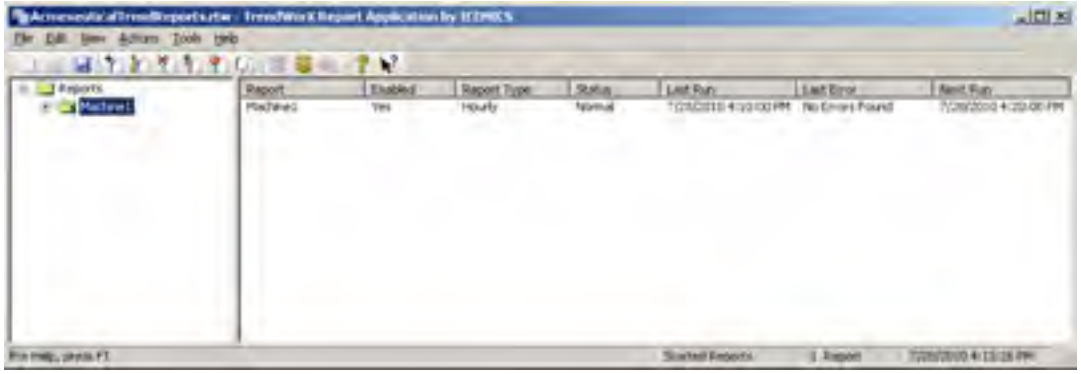
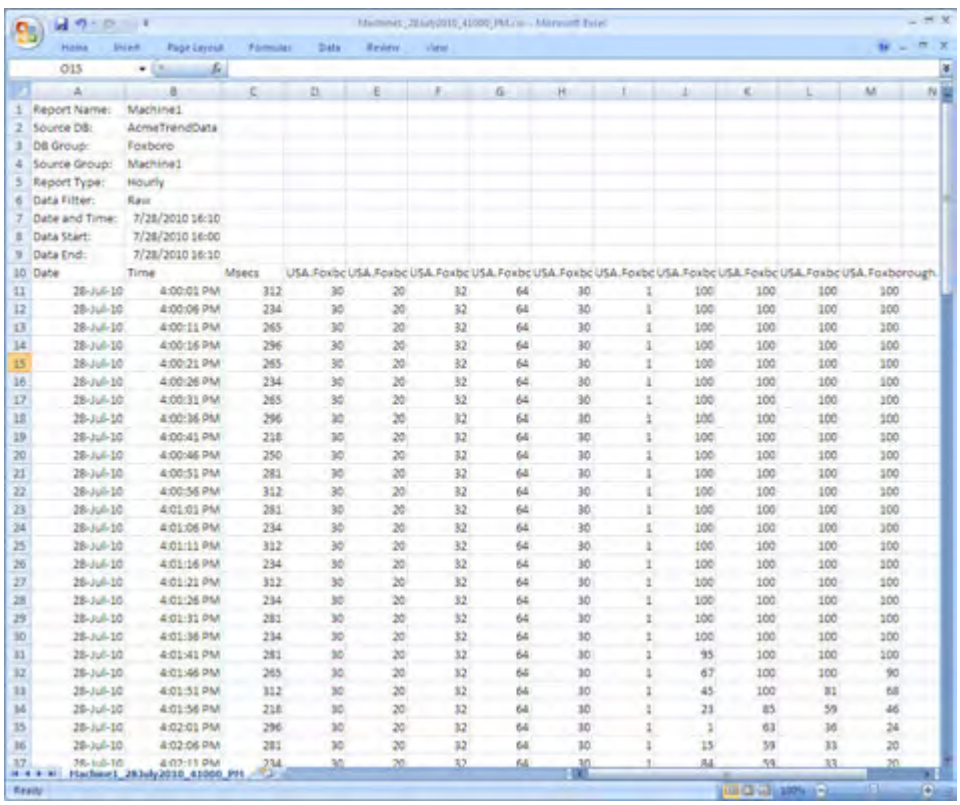


Figure L12.15 - Monitoring Report Status

36. An example of a report is shown in Figure L12.16. Notice that there are blank cells. This is because samples are only logged when the values change. Since not all values will change at the same time, there are samples for some tags while there are none for others. Locate the .csv file in the C:\Training folder to view your sample report.



1	Report Name:	Machine1												
2	Source DB:	AcmeTrendData												
3	DB Group:	Foxboro												
4	Source Group:	Machine1												
5	Report Type:	Hourly												
6	Data Filter:	Raw												
7	Date and Time:	7/28/2010 16:10												
8	Data Start:	7/28/2010 16:00												
9	Data End:	7/28/2010 16:10												
10	Date	Time	Msecs	USA.Foxbc	USA.Foxbc	USA.Foxbc	USA.Foxbc	USA.Foxbc	USA.Foxbc	USA.Foxbc	USA.Foxbc	USA.Foxbc	USA.Foxbc	USA.Foxborough.
11	28-Jul-10	4:00:01 PM	312	30	20	32	64	30	1	100	100	100	100
12	28-Jul-10	4:00:06 PM	234	30	20	32	64	30	1	100	100	100	100
13	28-Jul-10	4:00:11 PM	265	30	20	32	64	30	1	100	100	100	100
14	28-Jul-10	4:00:16 PM	296	30	20	32	64	30	1	100	100	100	100
15	28-Jul-10	4:00:21 PM	265	30	20	32	64	30	1	100	100	100	100
16	28-Jul-10	4:00:26 PM	234	30	20	32	64	30	1	100	100	100	100
17	28-Jul-10	4:00:31 PM	265	30	20	32	64	30	1	100	100	100	100
18	28-Jul-10	4:00:36 PM	296	30	20	32	64	30	1	100	100	100	100
19	28-Jul-10	4:00:41 PM	218	30	20	32	64	30	1	100	100	100	100
20	28-Jul-10	4:00:46 PM	250	30	20	32	64	30	1	100	100	100	100
21	28-Jul-10	4:00:51 PM	281	30	20	32	64	30	1	100	100	100	100
22	28-Jul-10	4:00:56 PM	312	30	20	32	64	30	1	100	100	100	100
23	28-Jul-10	4:01:01 PM	281	30	20	32	64	30	1	100	100	100	100
24	28-Jul-10	4:01:06 PM	234	30	20	32	64	30	1	100	100	100	100
25	28-Jul-10	4:01:11 PM	312	30	20	32	64	30	1	100	100	100	100
26	28-Jul-10	4:01:16 PM	234	30	20	32	64	30	1	100	100	100	100
27	28-Jul-10	4:01:21 PM	312	30	20	32	64	30	1	100	100	100	100
28	28-Jul-10	4:01:26 PM	234	30	20	32	64	30	1	100	100	100	100
29	28-Jul-10	4:01:31 PM	281	30	20	32	64	30	1	100	100	100	100
30	28-Jul-10	4:01:36 PM	234	30	20	32	64	30	1	100	100	100	100
31	28-Jul-10	4:01:41 PM	281	30	20	32	64	30	1	95	100	100	100
32	28-Jul-10	4:01:46 PM	265	30	20	32	64	30	1	67	100	100	90
33	28-Jul-10	4:01:51 PM	312	30	20	32	64	30	1	45	100	81	68
34	28-Jul-10	4:01:56 PM	218	30	20	32	64	30	1	23	85	59	46
35	28-Jul-10	4:02:01 PM	296	30	20	32	64	30	1	1	63	36	24
36	28-Jul-10	4:02:06 PM	281	30	20	32	64	30	1	15	59	33	20
37	28-Jul-10	4:02:11 PM	234	30	20	32	64	30	1	84	59	33	20

Figure L12.16 - Report Example

NOTE: If you would like blank cells to be filled in with the last logged sample (which would be the current known value since samples are only logged when the tags change), simply enable the "Backfill Empty Cells" option on the **Report** tab of the report configurations dialog. With this option, there would not be empty cells in the report depicted in Figure L12.16.

Review: TrendWorX32 Reporting

R12

12.1. Chapter Objectives

In this chapter, we discussed the different ways available to retrieve history data to create a report. You have the option to use applications such as TrendWorX32 Reporter with complete user interface to generate a report or you can write your own SQL Queries to retrieve the data. You should now know and understand the following topics:

- How to configure a TrendWorX32 Reporter report
- The different reporting schedule options
- The different report file types
- How to run a report
- How to use the SQL Query Engine to retrieve data

12.2. Summary

From time to time, it is necessary to generate a report with the logged trend data for analysis. TrendWorX32 provides a few different ways for you to generate such a report. First, you can use the TrendWorX32 Reporter. Using this application, you can graphically create a report, choose the OPC tags to include in the report, set a filter if you wish, and set when the report should run. The TrendWorX32 Reporter can export data to three different file formats; a text file, a Microsoft Excel Worksheet, and a database.

You can also use the TrendWorX32 SQL Tool ActiveX control in an ActiveX container such as GraphWorX32 to retrieve data. This control allows you to retrieve data with configurable parameters and displays the data directly on your HMI system.

New in version 9.20 is the TrendWorX32 SQL Query Engine. You can setup a linked server in Microsoft SQL Server Management Studio to interact with the logging database. You can use any application that can run SQL queries to retrieve data logged by the TrendWorX32 Logger. This makes reporting very flexible and allows you to receive data into almost any report format you need.

12.3. Terms

BizViz	Data Mining Configurator	Linked Server
ReportWorX	TrendWorX32 Reporting	TrendWorX32 SQL Query Engine
		TrendWorX32 SQL Tool ActiveX

12.4. Test Your Knowledge

In this chapter, we discussed the different ways that you can use the retrieved data logged by TrendWorX32 Logger and export it into a report. You should now be able to answer the following questions.

1. What type of scheduling does TrendWorX32 Reporting offer?

    a. Immediate
    b. One Time
    c. Periodic
    d. On Event
    e. All of the above
    f. None of the above

2. What is the difference between the Target Name and the Report name in the report configuration?

    a. There is no difference; the report name is the target name.
    b. The report name is the name of the report in the configuration; the target name is the base file name of the generated report.
    c. The report name is the base file name of the generated report; the target name is the name of the report in the configurator
    d. The report name is the name of the report in the configuration; the target name is the name of the configuration file.

3. True or False: You can automatically send out an email with a generated TrendWorX32 report.

4. True or False: You can tell if a report ran successfully by looking at the TrendWorX32 Reporting Configurator.

5. True or False: Using the SQL Query Engine, you can retrieve data from all tags within the same logging group by querying one table.

6. What is the structure of a table name that contains the history data in the SQL Query Engine?

    a. LinkedServer.DatabaseGroup.LoggingGroup.RawData
    b. LinkedServer.LoggingGroup.DatabaseGroup.Tags
    c. LinkedServer.DatabaseGroup.LoggingGroup.Tags
    d. DatabaseGroup.LoggingGroup.Tags.LinkedServer

7. Which of the following is NOT a column in a RawData table?

    a. TagName
    b. HighRange
    c. Quality
    d. Value

8. True or False: You have to manually add the linked server through the Microsoft SQL Server Management Studio; ICONICS does not provide any application interface to help you with this.

Aliasing 13

In many projects, you may find that you need the same graphics to support many pieces of the same type of machinery. It could be a duplicate of a production line or multiple rooms where you want to control the lights and temperature. In these situations, instead of keeping multiple copies of the same graphics tied to different OPC data sources, you can keep just one file and alias the parts of the OPC tags that are different. This way, when you need to make a change to such a graphic you only need to update one file making it easy for maintenance and while reducing your development time.

In this chapter, we will discuss how using aliases in a project will make development faster and easier. By the end of the chapter, you should know and understand the following topics:

- How aliases ease development
- How to configure Global Aliases
- How to use Global Aliases in different applications
- How to use Local Aliases
- How Global Aliases differ from Local Aliases

13.1. Aliasing in GENESIS32

Aliasing is the ability to define a string (usually a short name) to represent another string (usually part or all of a tag name). Aliasing in GENESIS32 is designed to be a placeholder such that during runtime it will resolve to different values.

There are a few different types of aliases in GENESIS32: Local Alias, Global Alias, and Language Alias. Each type of alias is defined to perform a specific function. However, they are the same in that during configuration mode the alias takes place of the real value shown in runtime.

In this chapter we will focus on Local Alias and Global Alias as they are very similar to each other in use. These two types of aliases are generally used to represent an OPC tag, part of a tag, or a simple text string. At this point you may wonder why we have two different types of aliases that are used for the same function. While the two different aliases are used in similar ways they have a different scope of usage.

Local Alias, as the name suggests, is local to GraphWorX32 and therefore can only be used inside GraphWorX32. While many of the ActiveX Controls such as the Alarm Viewer and Trend Viewer ActiveX Controls, are used inside GraphWorX32 these cannot resolve local aliases because they are separate programs and GraphWorX32 just happens to be the application that hosts them.

Global Alias, on the other hand, is available to most of your GENESIS32 applications including ActiveX Controls. Because many different applications can access Global Aliases it requires a configuration to store the alias and the predefined alias values. When you are choosing which type of alias to use in your project you should consider the differences between these two types of alias methods.

13.2. Local Aliasing

Local Aliases resolve to the Alias Definition which can be a Data Tag, Expression or a User Defined value. Local Aliasing, as mentioned before, only works within GraphWorX32, hence the name Local Aliasing. *It is easy to create a local alias; simply type in the local aliasing syntax wherever you can add a data source to create it*. The syntax for a local alias is double triangular brackets around the name of the alias: <<local_alias_name>>. The special delimiters "<<" and ">>" identify the beginning and end of a local alias name.

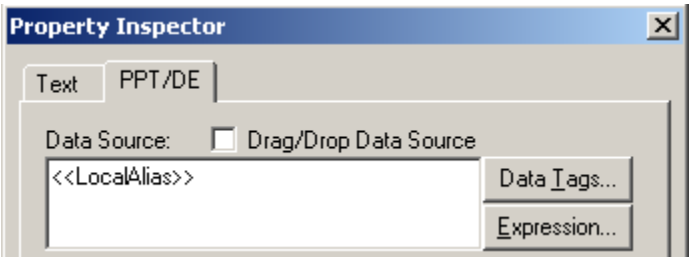


Figure 13.1 - Local Alias Configuration in Data Source

You can use a local alias by itself, as is, in the prior example or you can use it within an expression with the following syntax: x= <<local_alias_name>>. You can also use a local alias as part of a tag representing a specific part of an OPC data source connection. The syntax would be similar to the following: [opcserver].<<local_alias_name>>.out.

Once you have created the alias in a display you can right-click on the object with the alias and select “Edit Local Aliases…” from the popup menu as shown in Figure 13.2. This will open the **Edit Local Aliases** dialog. In this dialog you can set the alias’ value. You can also access this dialog by accessing the **Edit Aliases** command on the **Dynamics** menu.

> ***Tip!*** If you want to show the Local Alias as a string in a Process Point you can enclose the alias in the string literal delimiter as shown here: $“<<LocalAlias>>”$. Remember to set the Client Display Type to STRING, otherwise, it will show you a numeric value instead of text.

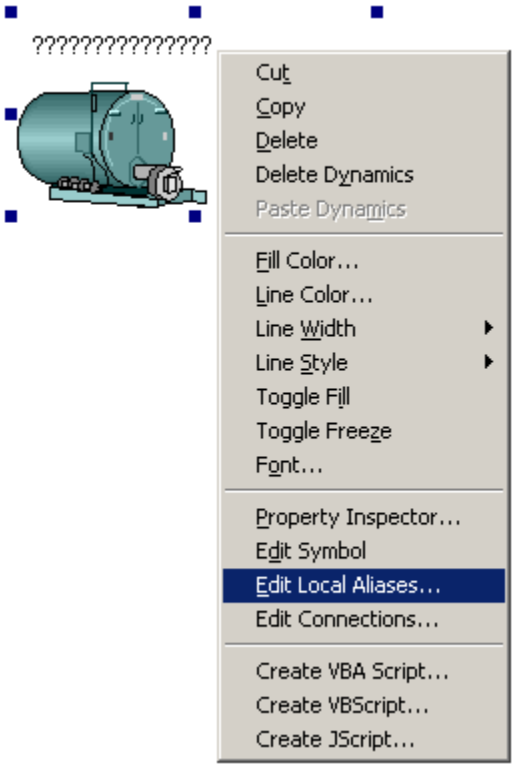


Figure 13.2 - Edit Local Aliases

13.2.1. Object-Level and Second Level Local Aliasing

Although we can simply define and use an alias throughout a display that can be set on load or in runtime for the entire display, there are other ways that you can set values for an alias. You can even set multiple values for the same alias used on different objects. Figure 13.3 shows three different broiler objects with three different alias values for each.

Figure 13.3 - Same Alias with Different Alias Values

We have the ability in GraphWorX32 to use the same local alias in several objects and have a different resolved value for that alias depending on the object where it is used. This is what we call **Object-Level Aliasing**. The alias value is actually set as part of the object properties. When you right-click on an object you will notice that the menu has the Set **Local Aliases** option available. These settings will be specific to this object. If you named your objects you will notice that in the **Edit Local Aliases** dialog, it will show you the name of the object with its associated alias and the alias definition as shown in Figure 13.4.

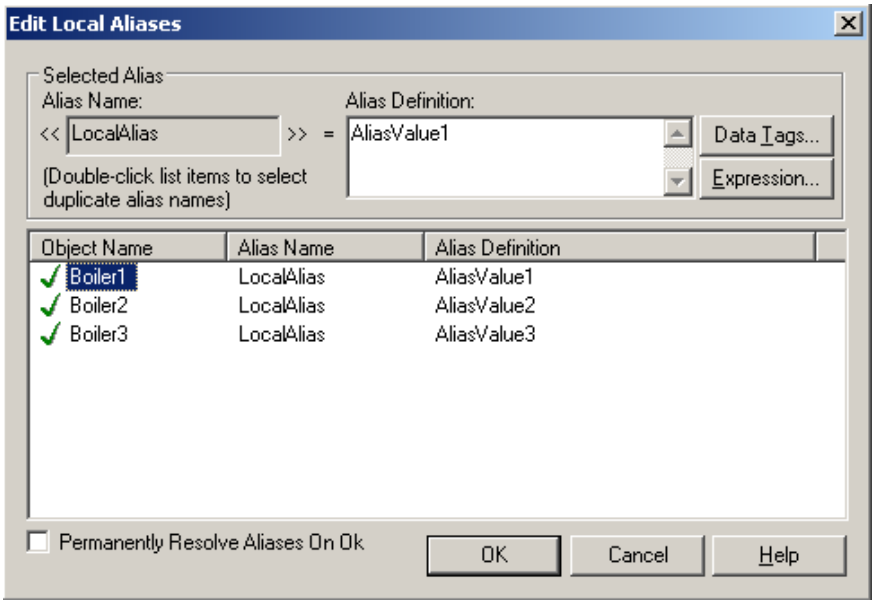


Figure 13.4 - Edit Local Aliases: Set Alias Value

Second-Level aliasing allows an alias name to be entered as the alias definition. This second alias will be resolved prior to the set-aliases command. This means that you can grab the current resolution of an alias in the current display and set an alias in a display being loaded to that value. The originating and destination aliases can have the same alias name, but it is not required. This is common to pass the current context of a display into a popup window to show the appropriate data.

13.2.2. Local Alias Pick Actions

There are two pick actions specifically created for Local Aliasing: *Set Local Aliases* and *Local Alias Dialog*. Both pick actions allow you to set a local alias definition during runtime. With the former, you have to pick a specific value during configuration. The latter option lets you choose the value to set in a drop down during runtime.To set the value of your local alias in Runtime mode directly, use the “Set Local Aliases” pick action. Clicking on the “Set Aliases” button in the Property Inspector of the pick action brings up the **Local Aliases Configuration** dialog as shown in Figure 13.5.

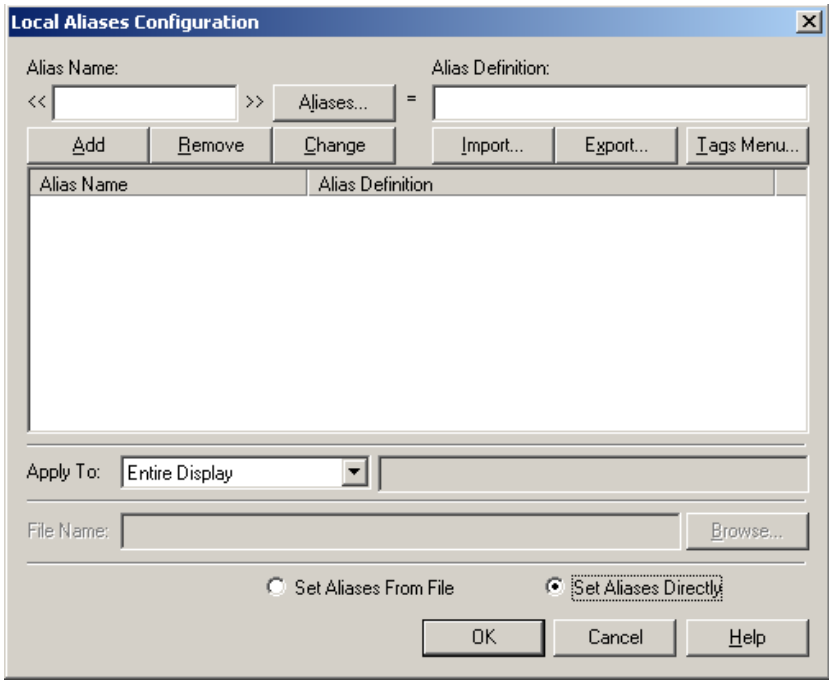


Figure 13.5 - Local Alias Configuration

Using this dialog you can define the different local alias you have on the display along with their values. The alias set by the pick action can apply to the entire display or to a particular layer or object. You can choose the scope of the local alias by making your selection from the dropdown list of the “Apply To” field.

If you have a separate text file where you defined the alias and its values, you can choose the “Set Aliases From File” option at the bottom. The “File Name” field will become available and you can click on the “Browse…” button to select the text file. If you are not sure of the format in which a local alias file should take, you can define a few aliases along with their values and use the “Export” button near the top to create an alias file. Similarly, if you want to import aliases and their definitions from a file, you can use the “Import” button.

If you want to give the operator a dropdown list of values to choose from during runtime, you can use the “Local Alias Dialog” pick action. Configure the alias definition the same way in the **Local Alias Configuration** dialog. However, during runtime, when you click on the pick action button, it will display the **Select New Alias Values** dialog as shown in Figure 13.6. For each alias you define on the pick action, you will see a tab.

When you click on the dropdown list in a tab, it will show you the alias value that you have defined. Selecting value and clicking on the “OK” button will allow you to set the local alias value for the specified alias.

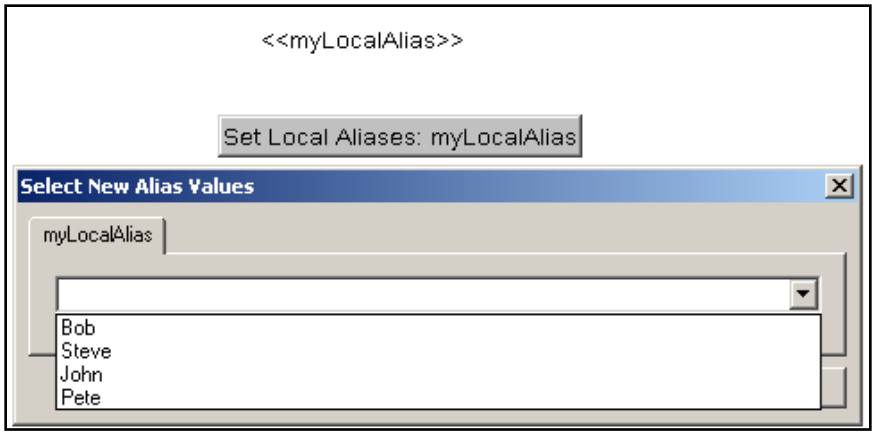


Figure 13.6 - Local Alias Dialog

13.3. Global Aliasing

The Global Aliasing System is a powerful feature implemented in GENESIS32 clients and servers, including GraphWorX32, TrendWorX32, and AlarmWorX32. Global aliases are stored in the central global database (Microsoft SQL Server or Microsoft Access) and can be accessed over the network from miscellaneous clients. To provide a reasonable performance, the local clients do not connect directly to the database (Global Aliasing Engine Server) but they communicate with a Global Aliasing Client, which will obtain the requested information and cache it locally. Figure 13.7 shows the Global Aliasing System architecture.

A **global alias** is a data string that enables you to reference multiple data sources (e.g., a trend pen in TrendWorX32 or a process point in GraphWorX32) with one unique name. Because you can reference multiple data sources from a single location, global aliasing can reduce the overall number of individual display files you need to create.

Global Aliasing

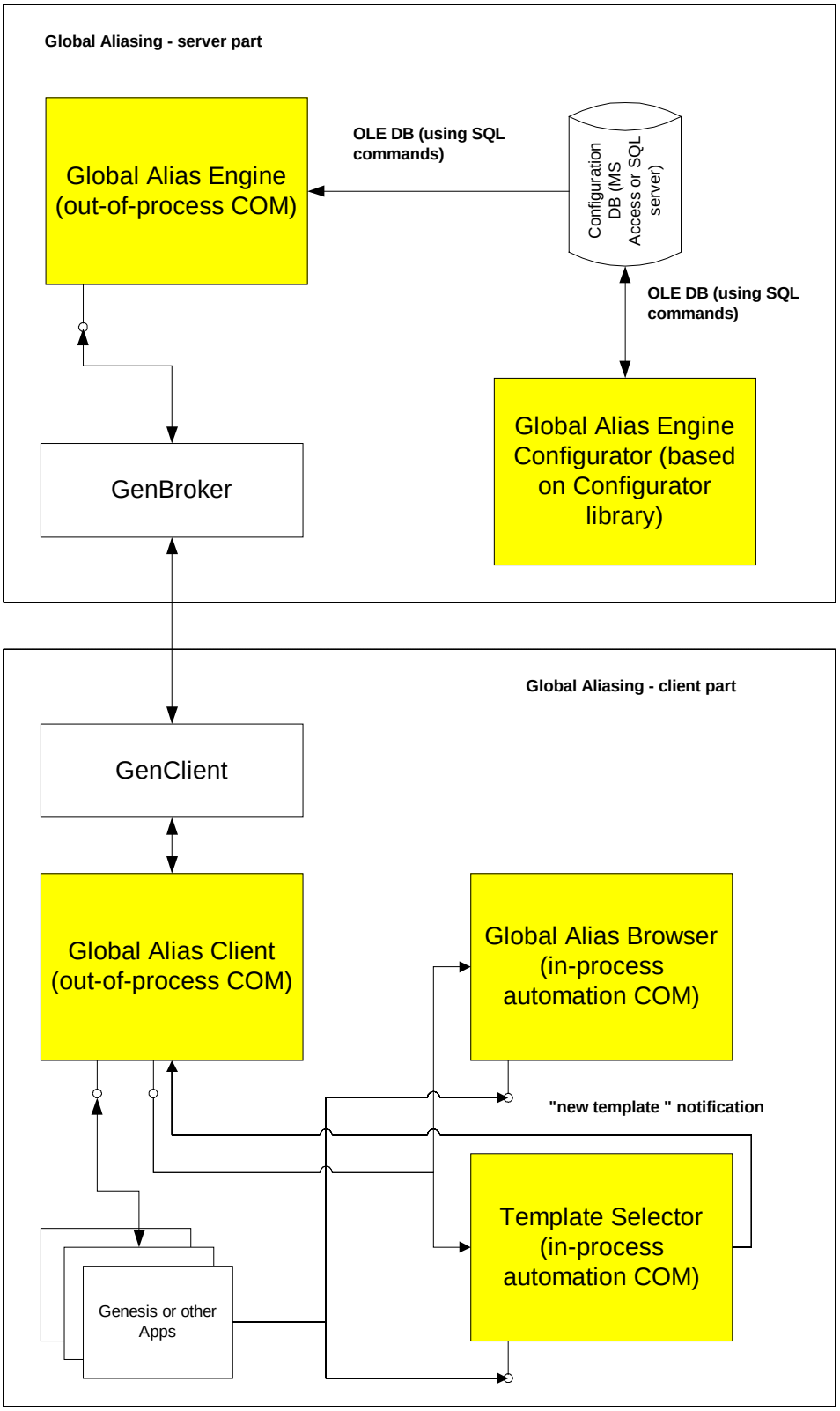


Figure 13.7 - Global Aliasing System Architecture

Global aliases use the following syntax: **<#global_alias_name#>**. The Global Aliasing System provides global storage for a string of aliases and resolves the aliases in runtime. The aliases can be used in OPC tags, menu items, file names, etc. Figure 13.8 shows how a global alias links to multiple OPC tags in a GraphWorX32 display for example. The data values with which the global aliases are associated are called alias values. Global aliases and alias values are usually created and configured in the Global Aliasing Configurator.

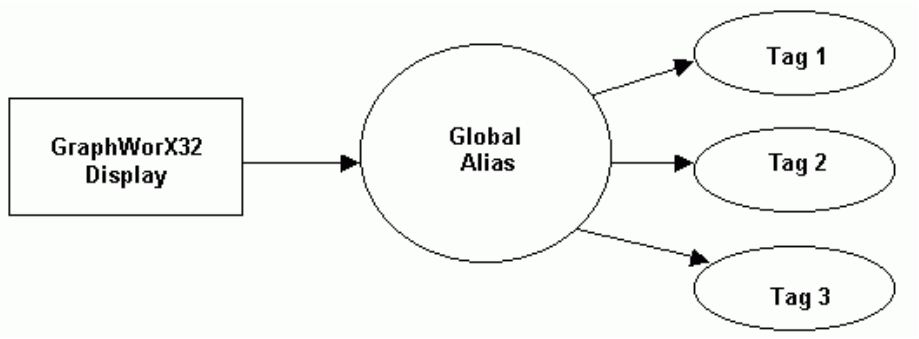


Figure 13.8 - Single Alias for Multiple Data Sources

13.3.1. Alias Resolution and Themes

Alias resolution is the process of assigning a specific tag value to an alias. You can assign one of multiple tags to the alias (e.g., Tag 1), as shown in Figure 13.9.

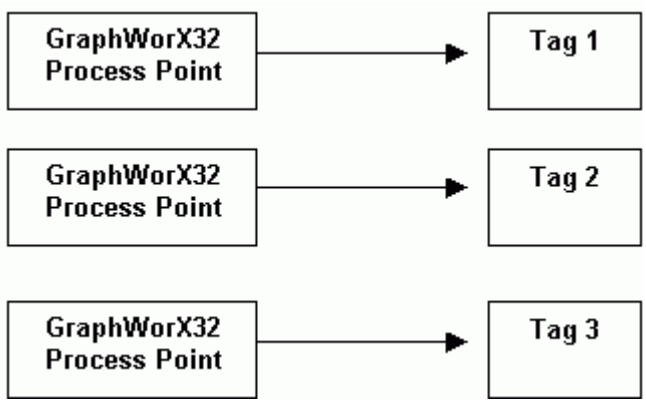


Figure 13.9 - Aliasing Multiple Tags

Each global alias can be associated with a theme. A **theme** links a global alias to one or more data values (e.g., tags). The Global Aliasing database allows you to assign different values to aliases based on the currently selected theme. The themes can change dynamically either programmatically through automation API or by a GraphWorX32 pick action. Figure 13.10 shows how a generic single alias can be used to link multiple tags to a single process point in GraphWorX32.

The Global Aliasing Engine Server resolves the alias values and theme items that are associated with the data values (e.g., tags), as shown in Figure 13.10.

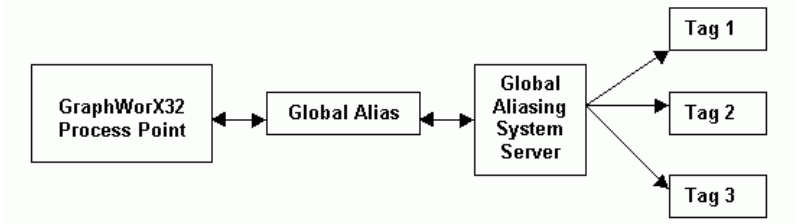


Figure 13.10 - Alias Resolution

Each theme can have one or more theme items associated with it and each alias value can be associated with a theme item. A **theme item** allows you to switch between the data values that are linked to the theme, as shown in Figure 13.11. For example, if you have one theme with three theme items, and each theme item is associated with a tag, you can toggle between the theme items (and therefore the tags). Thus, you have a single process point associated with three different values, as shown in Figure 13.11. All themes and theme items are created and configured in the Global Aliasing Configurator.

In addition, a single alias can be read by many different applications (e.g., GraphWorX32, TrendWorX32, and AlarmWorX32), as shown in Figure 13.11. You can also associate multiple aliases with a single theme.

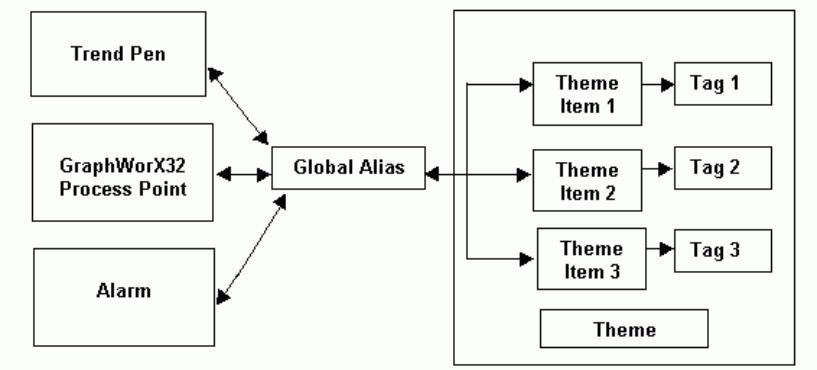


Figure 13.11 - Using Theme Items to Switch Between Data Values

Themes are useful when you have objects that need different aliases. You can simply choose a theme for the system to resolve the different aliases into the values in the theme. For example, consider a building control, like in the diagram shown in Figure 13.12, where you have different rooms on a particular floor and you want to know the temperature in a room, if the door is open, and if the lights are turned on.

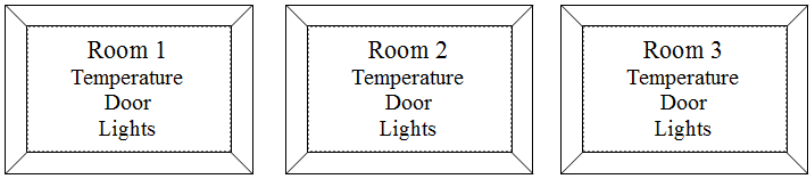


Figure 13.12 - Global Alias Scenario

In this case, your theme would be the rooms and the aliases would be their temperatures, door statuses, and light statuses so that when you select the theme to change to a different room, the different aliases will show the value for that particular room. Sometimes it is easier to think of a theme as an object, with the alias being the different parts that makes up the object.

13.3.2. Theme Scope

The resolution of aliases is controlled by the theme scope (i.e. on what level a theme is applied to your application). The Global Aliasing system supports three different levels of aliases, as shown in Figure 13.13:

- **Machine-Level:** Applies to the whole computer.
- **Process-Level:** Applies to the current process (e.g., GraphWorX32 with pop-up windows, a TrendWorX32 pen, or an AlarmWorX32 Viewer).
- **Document-Level:** Applies only to the current document/display and its embedded controls (e.g., a GraphWorX32 pop-up window).

NOTE: Document level aliasing is available only in GraphWorX32. It does not apply to the TrendWorX32 and AlarmWorX32 Viewer ActiveX controls.

In GraphWorX32, you can have multiple levels of documents, or pop-up windows, as shown in Figure 13.13.

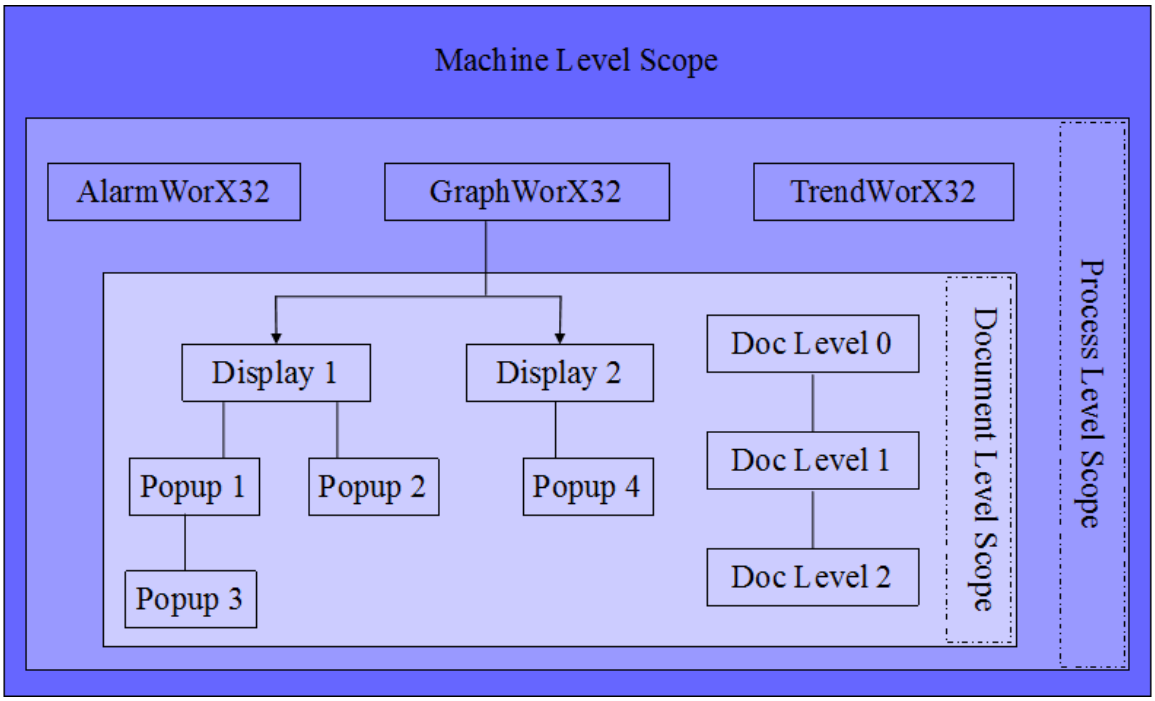


Figure 13.13 - Overview of Theme Scopes

13.3.3. Global Aliasing Configurator

The **Global Aliasing Configurator**, shown in Figure 13.14, is used to create and maintain a global alias database for use by the Global Aliasing Engine. Its main function is to perform configuration for aliases and themes. In keeping with the open standards employed throughout the GENESIS32 architecture, your global alias configurations are saved to a Microsoft Access or Microsoft SQL Server database.

In the Global Aliasing Configurator, the parameters for global aliases are set up in the **Aliases** tree control, as shown in Figure 13.14. When configuring global aliases, you are able to:

- Create folders to organize aliases
- Create new aliases
- Define new alias values
- Enable or disable aliases and folders
- Add folders and aliases to the Global Alias Browser
- Determine how folders and aliases are displayed and described in the **Global Aliases** tab of the Unified Browser
- Associate aliases with themes
- Associate alias values with theme items
- Specify default alias values

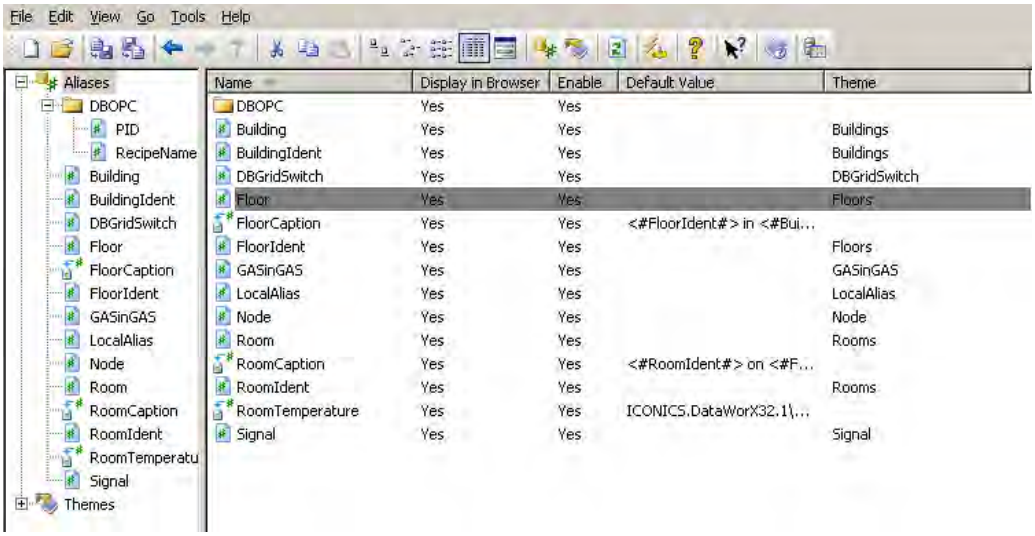


Figure 13.14 - Global Aliasing Configurator

When creating an alias, you can assign a theme (or group of theme values) to that alias. An alias can have an alias value for every theme value available. Once a configuration is in place, you can browse the aliases using the **Unified Data Browser** in any application that supports global aliasing.

The **Global Aliases** tab of the Unified Data Browser mimics the alias structure of the tree control in the Global Aliasing Configurator, as shown in Figure 13.15.

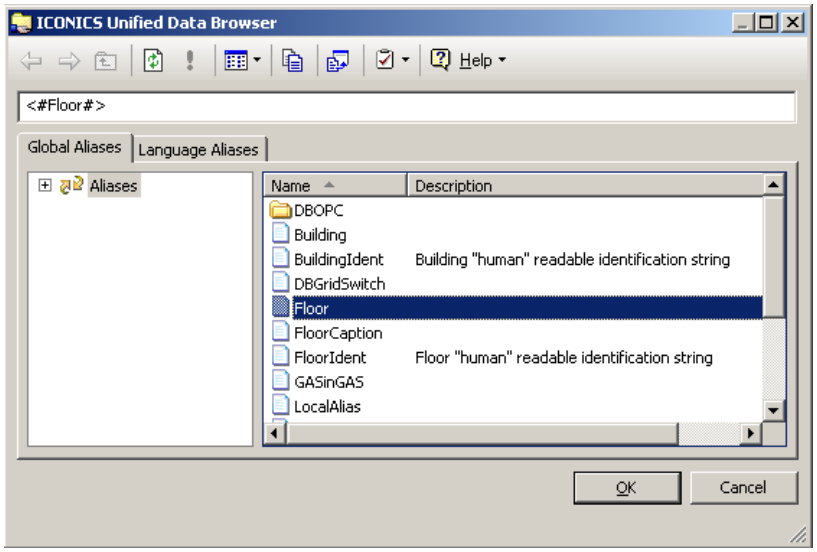


Figure 13.15 - Selecting a Global Alias From the Unified Data Browser

Once you browse for the alias, you will need to select the theme and theme values using the **Themes** browser. This way, you can switch themes easily. We will discuss how you can do that in the next section.

13.3.4. Using Global Aliases

Now that we know how to configure global aliases, let us look at how we can use them. You can take advantage of global aliases in many applications including GraphWorX32, AlarmWorX32 Server Configurator, ICONICS ActiveX Controls, and many others.

The syntax is the same regardless of where you want to use a global alias. Like local aliases, global aliases use delimiters to indicate its use. The delimiters <#and #>surrounds the alias to denote the beginning and end of a global alias.

In this section, we will look at a few of the applications in which you would most likely use global aliases.

13.3.4.1. Using Global Aliases in GraphWorX32

GraphWorX32 supports Global Aliasing by interfacing with the ICONICS Global Aliasing Server. Global aliasing is integrated into the following GraphWorX32 features:

- Support for visible objects (rectangles, ellipses, lines, etc.)
  - Description
  - Custom command
  - Global Alias Browser for general page
- Support for dynamics (size, location, rotation, etc.)
  - OPC Tag, High Range and Low Range
  - Description
  - Custom command
  - Global Alias Browser for expressions and connections
- Support for text objects (text, button, etc.)
  - Label
  - Global Alias Browser for label and descriptions
- Support for display buttons
  - Display file (supported automatically by inheritance from PICK)
  - Label (supported automatically by inheritance from Text)
  - Global Alias Browser for labels and descriptions
- Support for state fields
  - StateValue
  - StateString
  - Default Value
- Access the Global Alias Browser from the description and language edit boxes for Analog Selector, Animator, Color, ColorAnalog, and Digital Selector
- Flash, Hide, Location, Pick, Process Points, Rotation, Size, Time/Date

- State Fields and connected the code behind the pages
  - Added browse buttons to Local Variables - InitVal, LoRange, HiRange, DisplayTabs, Windows Properties
- Display Properties
  - Caption
  - Download Value 2x
  - Toggle Value 3x
- New pick action Select GAS (Global Aliasing System) Theme
  Command Line Support: GraphWorX32 allows you to set the initial global alias themes using a command line argument: Themes="*GlobalAlias*=*GlobalAliasValue*/*ThemeScope*"

  The syntax conforms to the pick action Select GAS (Global Aliasing System) Theme syntax.

Global Aliases are often used in Process Points, other dynamics, or labels (the configuration of which is shown in the left of Figure 13.16). These Global Aliases resolve when a Global Alias Theme is chosen in or before Runtime. When the aliases in this same configuration resolve in runtime, they look as shown below on the right side of Figure 13.16.

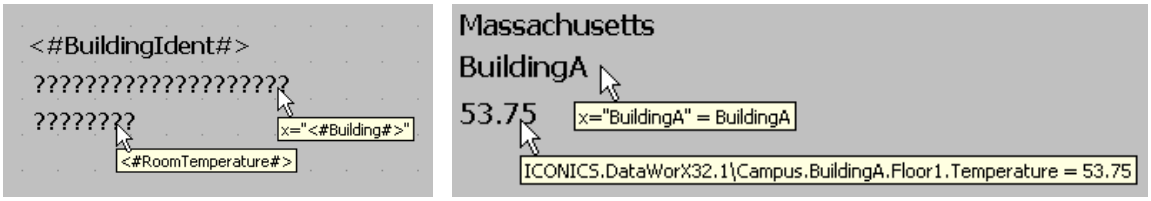


Figure 13.16 - Global Aliases Configuration (left) and Resolved Alias Values (right)

Themes are often chosen in one of three ways: by the default Start Theme, from a pick action, or during the loading process of a display.

You can define a display's start theme by choosing it in the GraphWorX32 Display Properties on the **Runtime Adv.** tab, as shown in Figure 13.17. The **Global Alias Start Themes** textbox houses this definition. You can also click on the ellipsis (…) button to bring up the Themes browser.

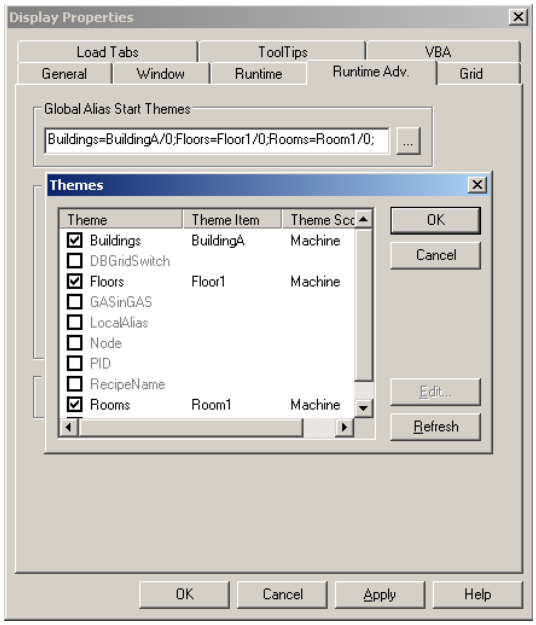


Figure 13.17 - Choosing a Global Alias Start Theme for a GraphWorX32 Display

In the Themes Browser you can check the themes you wish to set. Immediately after you check a theme it will become available instead of grayed-out with red text. The red text indicates that you need to define the theme item and theme-scope for the selected item. Clicking on the “Edit” button with the theme selected brings up the **Theme Editor** dialog.

The Theme Editor, shown in Figure 13.18, will show you the available theme items for the selected theme. In this dialog you can also choose the Theme Scope. As long as the theme item in this dialog remains “<none>,” the selected theme in the Themes Browser will remain red to indicate the configuration is incomplete. Once you choose the Theme Item and Scope you can click “OK” to close this dialog. When you do, you will notice that the selected theme no longer shows up in red and the respective columns in the browser display the configured values.

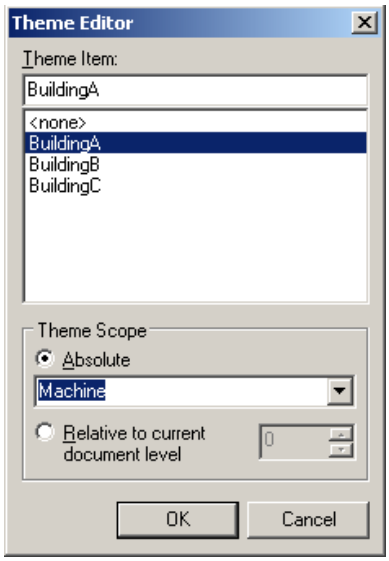


Figure 13.18 - Theme Editor

Themes are often chosen from a **Select GAS Theme** pick action to change the value of global aliases during Runtime, as shown in Figure 13.19. Notice the Select GAS Theme textbox contains the string "Buildings=BuildingA/0." This string indicates the global alias theme is Buildings and the theme item is BuildingA. The "/0" at the end signifies the theme scope. In this case, "/0" equates to the machines level scope.

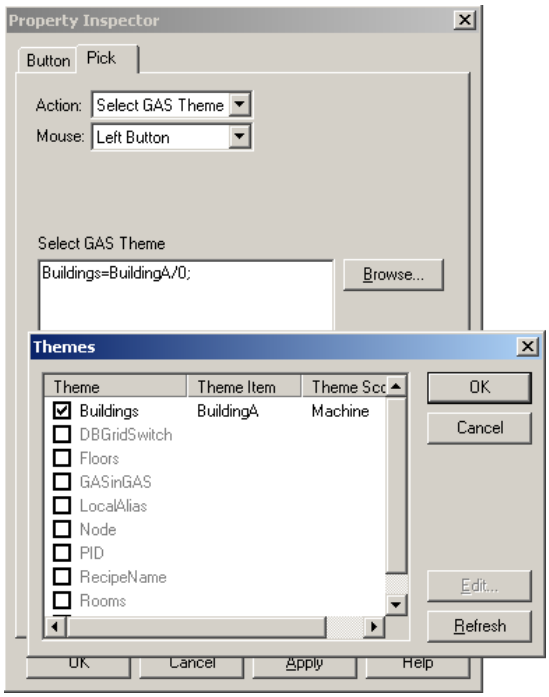


Figure 13.19 - Choosing a Theme with a Pick Action

Another common way of choosing themes is when loading a display. A display can load with a specific set of Global Alias values. For instance, if a **Load Display** or **Popup Window** pick action is on your display, you can configure a set of Global Alias themes to be applied to that display or popup window. The configuration of this is depicted in Figure 13.20.

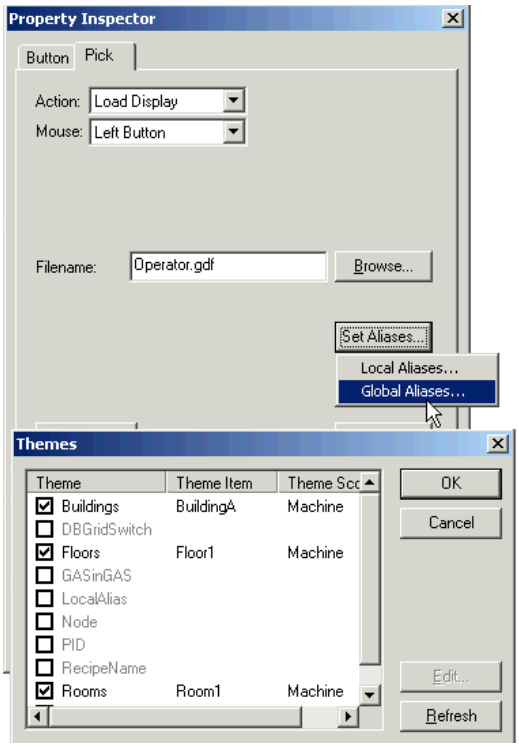


Figure 13.20 - Choosing a Theme with a Load Display Pick Action

13.3.4.2. Using Global Aliases in the Alarm Viewer

The AlarmWorX32 Viewer supports Global Aliasing by interfacing to the Global Aliasing Server. The AlarmWorX32 Viewer only supports machine and process-level aliasing. All fields that support aliasing utilize the Unified Data Browser which is opened by clicking on the ellipsis (…) button in the configuration.

When specifying a global alias in the **Alarm Viewer ActiveX Properties** dialog, you can also select a global alias from the **Global Aliases** tab of the **Unified Data Browser**, which includes all global aliases in the Global Aliasing System database. This eliminates the need to manually type in the alias name. You can browse Global Alias for the following:

- **Title, Configuration File,** and **URL Path** sections of the Alarm Viewer ActiveX and Alarm Report ActiveX Properties dialog **General** tab
- Alarm filtering via the Expression Editor

- Event Subscriptions dialog: Server, Areas, and Sources
- Alarm Report ActiveX records filtering

13.3.4.3. Using Global Aliases in the Trend Viewer

The TrendWorX32 Viewer supports Global Aliasing by interfacing to the ICONICS Global Aliasing Server. The following strings can be aliased:

- Title
- Ranges
  - Hi Tag, Lo Tag, Format, Format XY
- Details
  - Format
- Pen
  - Point Name, Hi Tag, Low Tag, Description, Units, Details Format, Ranges Format, Persistent Group Name
- URL Path
- Expressions

The TrendWorX32 Viewer only supports machine and process-level aliasing. All fields that support aliasing utilize the Unified Data Browser, which is evoked by clicking on the ellipsis **(…)** button in the configuration.

The user first connects a real-time, persistent, or historical tag and then has the option to alias the entire pen string or part of it. Global aliasing is also supported for drag-and-drop operations as well as related OLE automation methods.

13.3.5. Dynamic Global Aliases

In recent versions of GENESIS32, we have enhanced the Global Aliasing System such that it no longer requires preliminary configuration. You can use a global alias directly within GraphWorX32. It is important to keep in mind that since there is no configuration database to store your configured alias and value, once the Global Aliasing Engine stops, the dynamic global alias definition is lost until you recreate it.

It is simple to define a global alias dynamically. You can use the same global alias syntax where you want to use the alias. To set the dynamic global alias value, you will need to know the set alias syntax. Recall that in the previous section, we noted that the Select GAS Theme pick action contains a string in the format: *AliasTheme=ThemeItem/ThemeScope*.

We will use the same syntax to set the value of a dynamic global alias, except you will need to append “A:” in front of this syntax to indicate that this is a dynamically created global alias. The entire string should look similar to Figure 13.21.

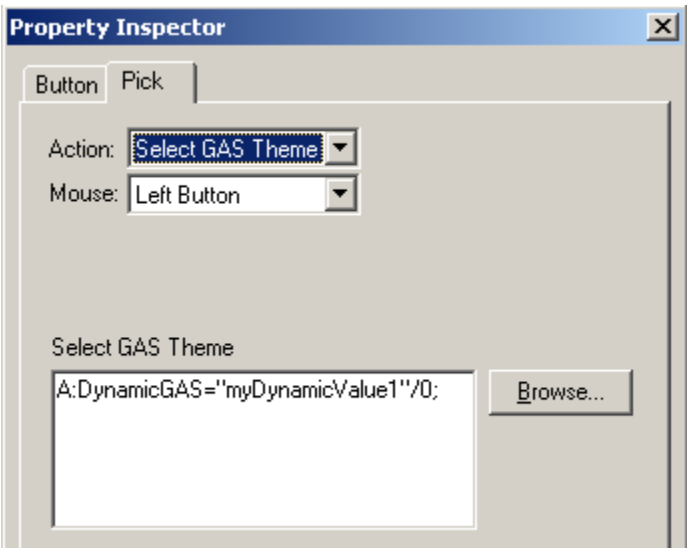


Figure 13.21 - Set Dynamic Global Aliasing Theme Syntax

NOTE: Take care that you do NOT add in extra spaces in the syntax. Having extra spaces may cause setting the alias theme to fail.

Once you configure this you can go into runtime, click on the button to set the theme and see that your alias resolves to the correct value.

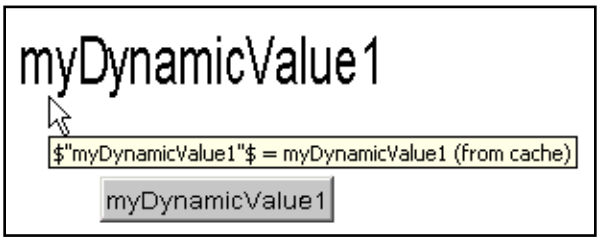


Figure 13.22 - Resolved Dynamic Alias Value

Lab: Simplified Development with Global Aliasing *L13*

Acmeseutical wants us to enhance the ACMEMixers display so that we can open and close the valves of a mixer on the display as well as view the trends and alarms for that mixer. To satisfy this requirement, we will need to use Global Aliasing. This lab will take you through the necessary steps to make such a display.

Estimated time to complete this lab: 60 minutes

13.1. Task Checklist

In this lab, you will complete the following tasks:

- Add and configure global themes and aliases
- Activate the global aliasing configuration
- Configure size dynamics based on data sources driven by global aliases
- Add pens to the Trend Viewer with data sources driven by global aliases
- Configure an Alarm Viewer subscription
- Configure Alarm Viewer columns
- Configure sorting in the Alarm Viewer
- Configure filtering that uses global aliases in the Alarm Viewer
- Configure a pop-up window pick action with global aliases incorporated

13.2. Support Files

You will need the following file(s) to complete this lab:

- ACMEMixers.gdf
- ACMEPillMachine.gdf
- AcmeseuticalTrendConfig.mdb
- AcmeTrendData.mdf
- AcmeseuticalAlarmServer.mdb
- AcmeGASConfig.csv
- AcmeseuticalTemplate.tdf
- PainReleifWorldMap.gdf

13.3. Global Alias Configuration

In order to take advantage of global aliasing, we will need to first configure it. Unlike local aliasing where alias values can only be resolved within GraphWorX32, global aliases can resolve values in the entire GENESIS32 system including applications like AlarmWorX32 Server and the different ActiveX controls within GraphWorX32. In this section, we will create the global aliasing configuration necessary.

1. Open the Global Aliasing Configurator from Start → Programs → ICONICS Tools → Global Aliasing Configurator.

1. The first time you open this Configurator the default demo configuration will be in place. You will need to create a new configuration database.

2. Click on the "New" button, or choose File → New and follow the **Configuration Database Wizard** to create an MS Access configuration database of the name "**AcmeseuticalGAConfig.mdb**". Save the database in your C:\Training directory.

3. The new blank configuration will open in the Configurator, as depicted in Figure L13.1.

Figure L13.1 - New Global Aliasing Configuration Database

13.3.1. Adding the Mixers Theme

1. Right-click on "Themes" in the tree control and choose New → Theme, or select Edit → New → Theme, to create a new theme.

2. In the "**Name**" field on the right-hand pane, enter "**Mixers**", as depicted in Figure L13.2.

3. Click on "**Apply**" to commit the changes to the database.

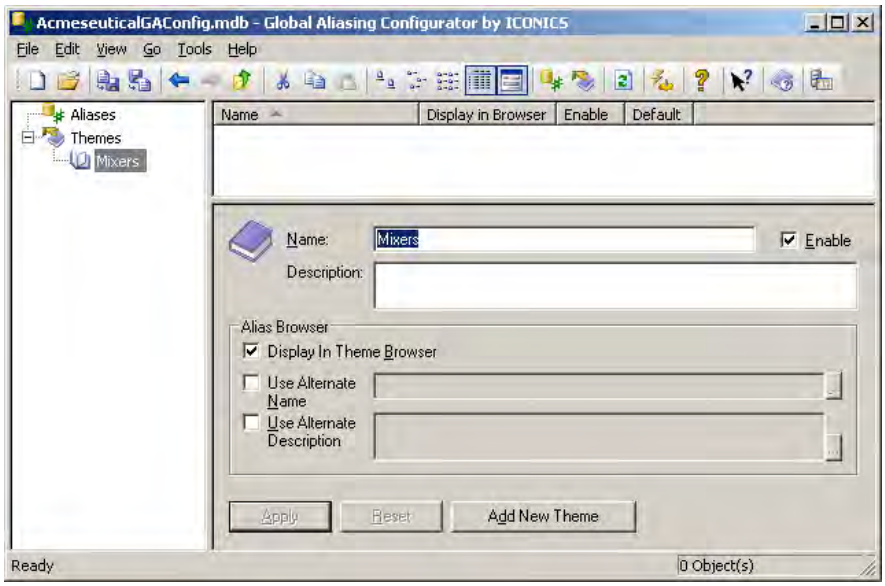


Figure L13.2 - New Tanks Theme

4. Right-click on "Mixers" in the tree control and choose New → Theme Item, or click on Mixers and select Edit → New → Theme Item, to create a new theme item.

5. In the right-hand pane, change the Name field to read "**MixerA**" as depicted in Figure L13.3.

6. Click on "**Apply**" to commit the changes to the database.

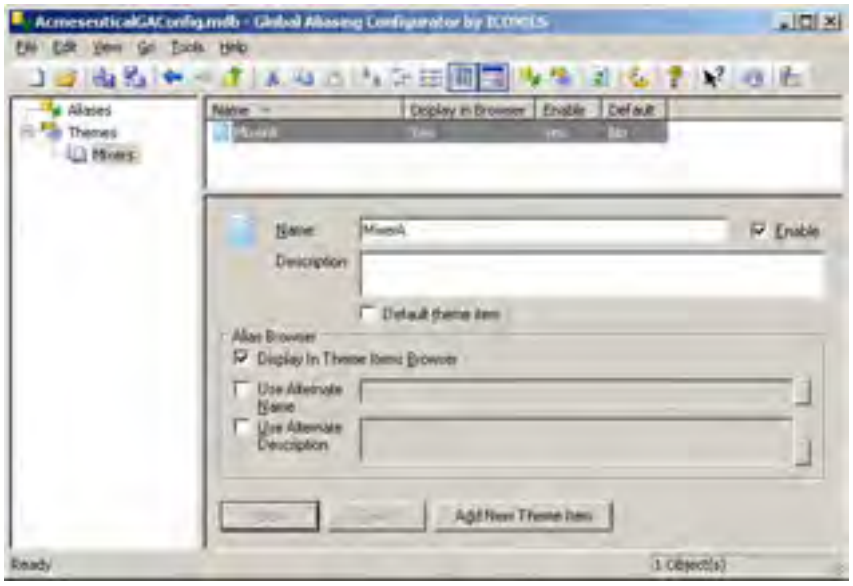


Figure L13.3 - New MixerA Theme Item

7. Repeat Steps 5-6 to create three (3) more theme items with the following names: MixerB, MixerC, and MixerD.

13.3.2. Adding the Mixers Alias

1. Right-click on "Aliases" in the tree control and choose **New → Alias**, or select Edit → New → Alias, to create a new alias.

2. On the right-hand pane, set the Name field to read "**Mixers**".

3. In the Theme dropdown list, choose "**Mixers**" as depicted in Figure L13.4.

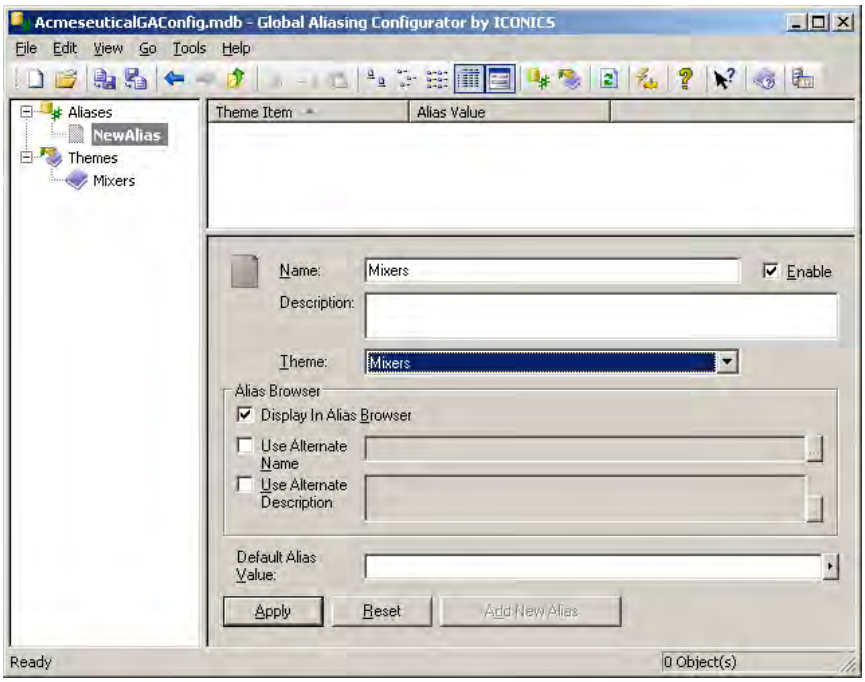


Figure L13.4 - New Mixers Alias

4. Click on "**Apply**" to commit the changes to the database.

5. Under Aliases, right-click on "Mixers" in the tree control and choose New → All Alias Values, or select Edit → New → All Alias Values, to add all alias values for the items defined in the theme.

13.3.3. Adding the AlarmFilter Alias

The alarm tags have a naming convention different from that of the mixer levels. Therefore, when creating the alarm filter, it is easier to derive the values from a different global alias. This AlarmFilter alias is driven by the same Mixers theme, but provides different outputs applicable to the Alarm Viewer filter.

1. Right-click on "Aliases" in the tree control and choose New → Alias, or select Edit → New → Alias, to create a new alias.

2. On the right-hand pane, set the Name field to read "**AlarmFilter**".

3. In the Theme **dropdown** list, choose "**Mixers**" as depicted in Figure L13.5.

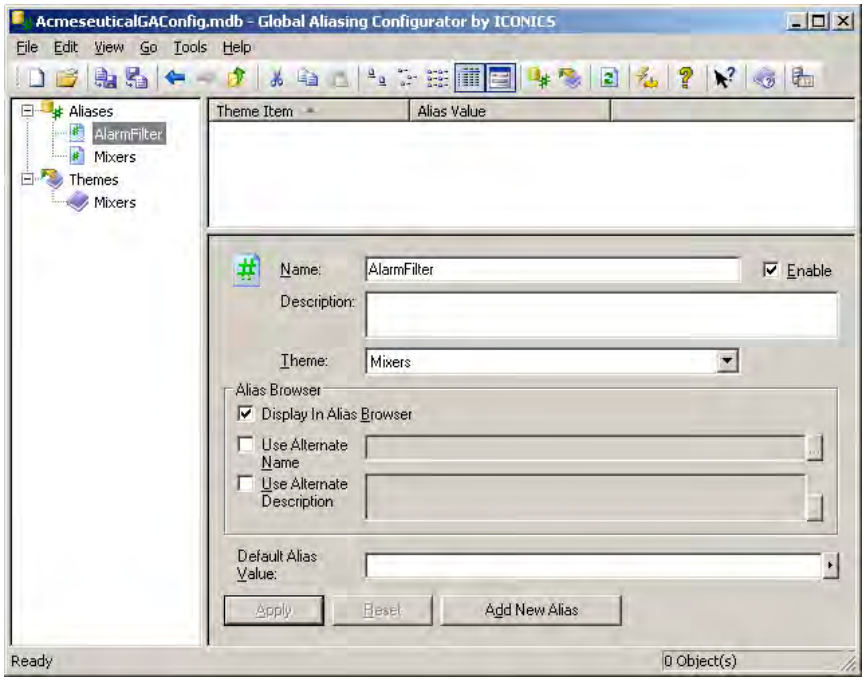


Figure L13.5 - New Alarm Filter Alias

4. Click on "**Apply**" to commit the changes to the database.

5. Right-click on "AlarmFilter" in the tree control and choose New → All Alias Values, or select Edit → New → All Alias Values, to add all alias values for the items defined in the theme.

6. Select "**Mixers=MixerA**" in the upper right-hand pane.

7. Change the Alias Value to "**MixerA Level**," as depicted in Figure L13.6. Click "Apply" to commit the changes to the database.

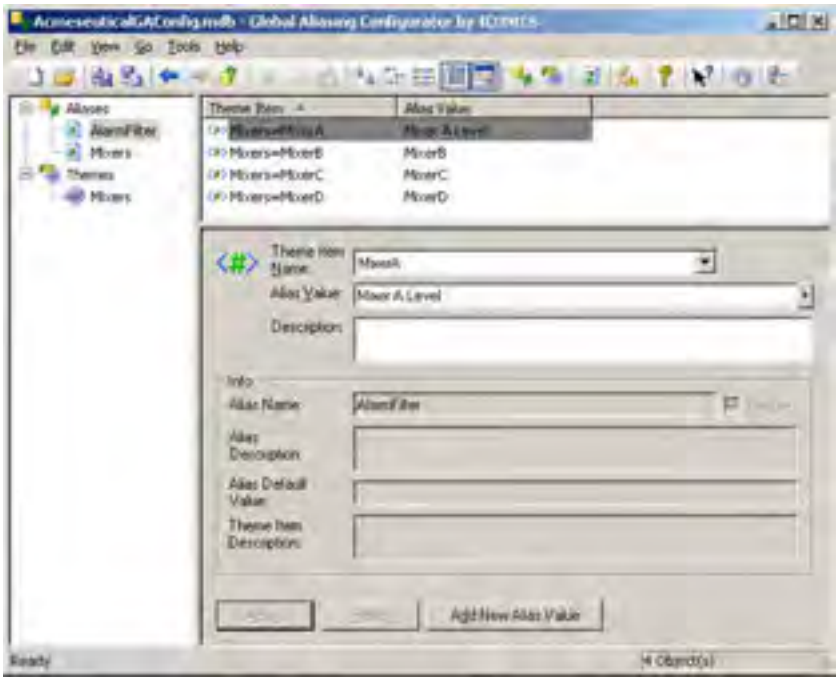


Figure L13.6 - New Alarm Filter Alias Value

8. Repeating Steps 6-7, changing the AlarmFilter Alias Values to the ones shown in Table L13.1.

Table L13.1 - AlarmFilter Alias Values

Theme Item Name	Alias Value
Mixers=MixerA	Mixer A Level
Mixers=MixerB	Mixer B Level
Mixers=MixerC	Mixer C Level
Mixers=MixerD	Mixer D Level

13.3.4. Importing a Configuration

Since the project involves many different locations worldwide, we will need aliases and themes for those different locations. Fortunately, your colleagues have done some of the configuration and provided you with their work. In this section, we will import the rest of the configuration.

1. Click on the **File** menu of the Global Aliasing Configurator and select **"CSV Import."**

2. In the **Import Configuration Data from File** dialog browse to C:\Training folder and select the "**AcmeGASConfig.csv**" file.

3. Leave the Import Settings as default and click on "Open" to begin the configuration import.

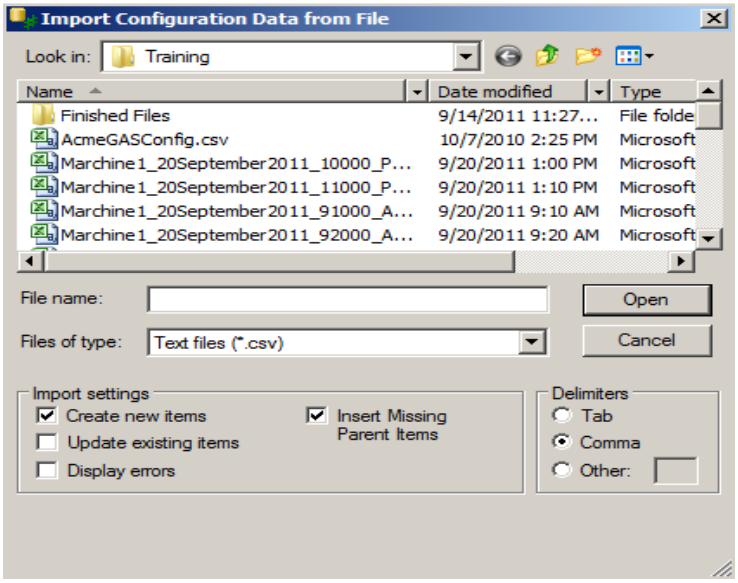


Figure L13.7 - Importing the AcmeGASConfig.csv File

4. Once the import completes, you should see the results dialog as shown in Figure L13.8. Click on "OK" to close the dialog.

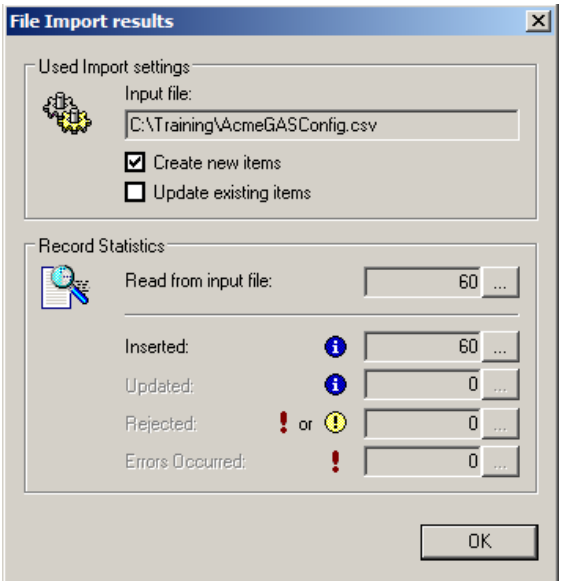


Figure L13.8 - File Import Results

5. Your finished configuration should look similar to Figure L13.9.

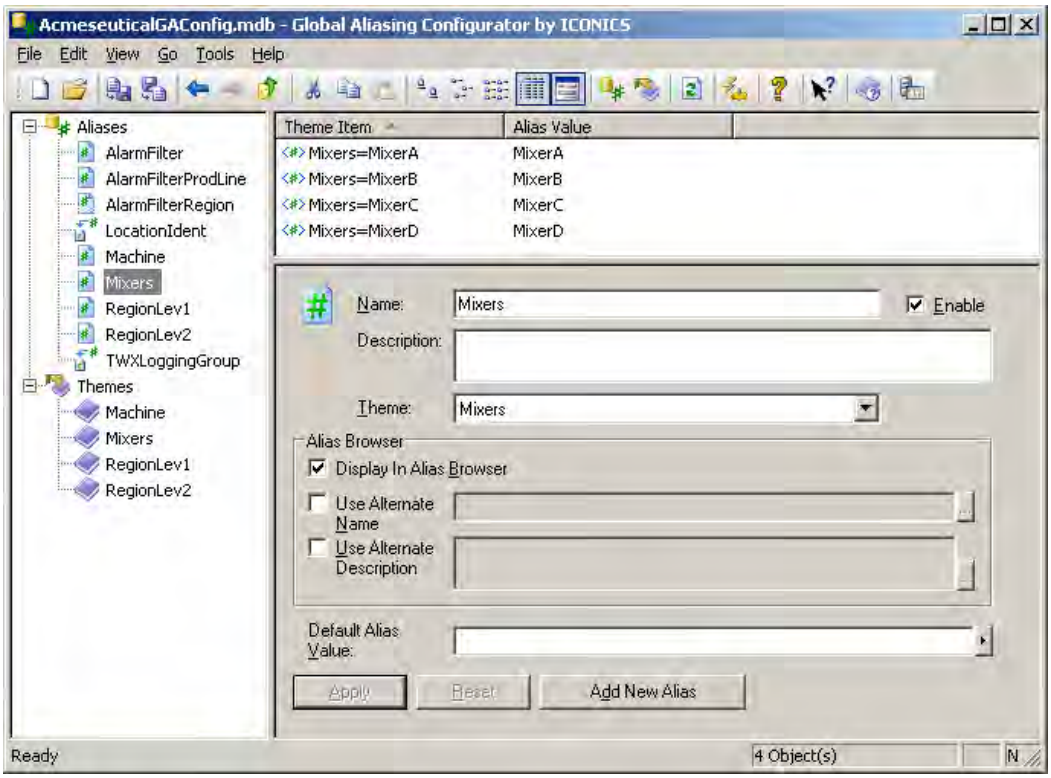


Figure L13.9 - Complete Global Aliasing Configuration

13.3.5. Activating the Database

1. Select **File → Make Active** to check that the database is the active one.

   NOTE: If the "Make Active" option is grayed out, then the current database is already the active database and nothing needs to be done.

2. If this is not your active configuration database, you will receive a dialog to confirm that you want to make it the active one. Click on "**OK**" to activate the configuration.

3. Once you have activated your configuration database, click on the "**Update Runtime**" button in the toolbar.

4. You will receive a confirmation dialog; click on "**Yes**" to continue the update.

Figure L13.10 - Confirmation to Reload Global Aliasing Engine

13.4. Global Aliases in GraphWorX32

Now that we have a Global Aliasing configuration, we will need to update our displays to take advantage of it.

1. First, open the **AcmeseuticalTemplate.tdf** file. From a GraphWorX32 session, select File → Open, browse to the C:\Training folder and select "GraphWorX32 Templates (*.tdf) in the Files of type: drop down". Make sure you are editing the template file and not a display with the template applied.

2. Double-click on the gray divider to bring up the **Property Inspector** for the text object.

   Note: You may have to select Format → Template → Edit Applied Template.

3. In the text field, enter "**<#LocationIdent#> <#Machine#>**" (without quotes). This will show the location and machine we are currently working on within the divider.

4. Click on "OK" to save the changes and close the dialog. Click No to update display files. We will update in step number 10 below.

5. Open the **PainReliefWorldMap.gdf** file in another instance of GraphWorX32.

6. Select Format → Template → Edit Applied template from the menu bar.

7. Copy the "**Navigate To**" button on the right-hand side of the divider and paste it into your **AcmeseuticalTemplate.tdf.**

8. Move the button to the right of the display on the divider similar to where it was on the **PainReliefWorldMap.gdf**. Your template should look similar to the following figure.

Figure L13.11 - Updated Template

9. Save the template. You should see a message asking if you would like to update the display file that uses this template, click on "Yes."

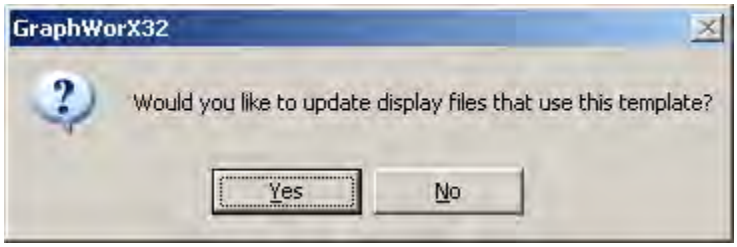


Figure L13.12 - Update Displays that Uses this Template

10. In the **Update Displays with Current Template** dialog, click the "Add" button and select the ACMEHistoryAlarms.gdf, the ACMEHistoryTrends.gdf, the ACMEPillMachine.gdf, and the AcmeseuticalAlarms.gdf. These files should be in your C:\Training directory.

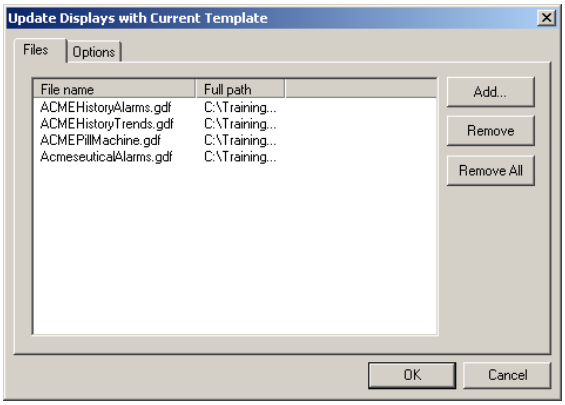


Figure L13.13 - Update Displays with Current Template

11. Be sure all the displays selected in step 10 are closed or they will not update. Click on "OK" to save and begin updating the template.

12. Close the Template file and open your ACMEMixers.gdf display. We will enhance this display to show us more details.

13. Since the **ACMEMixers** display was previously created to only allow for the opening and closing of the valves for each mixer, the display was just big enough for this purpose. We will need to change the size of the display in order to allow for more detail about the mixer. To do this, select **Format → Display Properties** from the top menu bar.

14. On the **General** tab, change the **Display Dimensions**. Change the Width to **800**, and the Height to **600**.

15. On the Window tab, in the "**Runtime Titlebar Options**" section, we will confirm that the "**Override Caption**" is set to "**Mixer Valve Status**."

16. Check that the **Outer Frame Edge** is set to "**Raised** (Resizable)," and the Inner Frame Edge to "Sunken."

17. Also, **uncheck** all other options except for "**Always on Top**," as depicted in Figure L13.14.

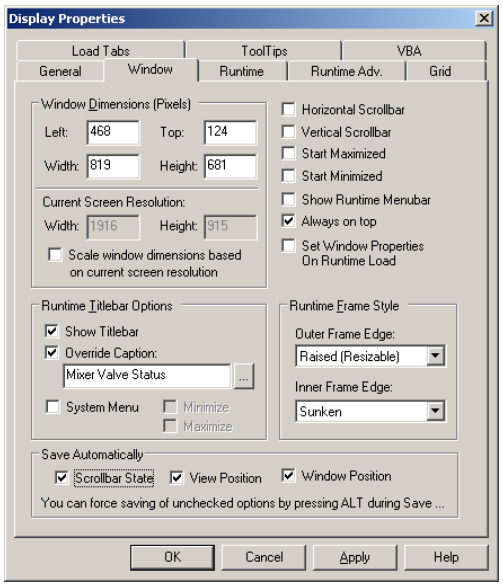


Figure L13.14 - Display Properties: Window Tab

Note: The Windows Dimensions in Figure L13.14 may not match what you have currently configured. Leave your dimensions as they are.

18. Click on "OK" to commit the display properties changes.

19. The purpose of this part of the training course is to create the display depicted in Figure L13.15.

Figure L13.15 - Popup Window to be Created in GraphWorX32

13.4.1. Inserting the Symbols

1. If the Symbol Library is not visible, click the "Symbols" button in the Draw toolbar to open it. Or you can select View → Toggle Symbol Toolbar.
2. Move the "Dismiss Display" button to the lower right-hand corner.
3. Align the process point, valve and the two buttons that open and close the valve into a line. You may want to use the "align middle" button on the Arrange toolbar.
4. Move this line of objects near the top of the display.
5. Confirm the data source says: $"Mixer<<MixerID>>"$, as shown below.

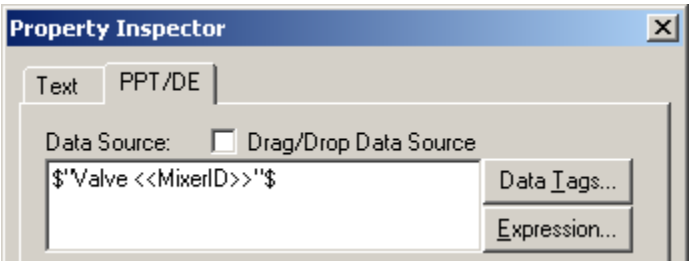


Figure L13.16 - Property Inspector for Local Alias Tag

6. In the Symbol Library, select the TrainingCourseV9.20 category in 2_Training folder, and drag the "Tank" and "Jagged Cut-Away" symbols from this category onto the display. Place the cut-away objects on top of the tank, and then place them below the process point.

7. Insert a new Alarm Viewer ActiveX by clicking on the Alarm Viewer button on the ActiveX toolbar. Adjust the size and position appropriately.

8. Insert a new Trend Viewer ActiveX by clicking the Trend Viewer button on the ActiveX toolbar. Adjust the size and position appropriately.

9. Click on the "Text" button on the Draw toolbar to insert a text label, and click in the area above the valve.

10. Double-click on the text object to bring up the **Property Inspector**.

11. Click on the ellipsis button next to Font … and choose Global Aliases to bring up the Unified Data Browser. Click on the refresh button.

12. In the **Unified Data Browser**, select "RegionLev2" in the right-hand pane, the global alias will be shown in the top field where it becomes editable.

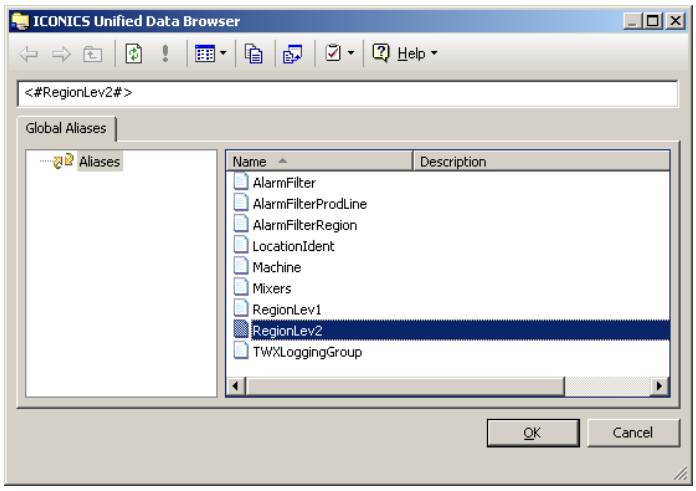


Figure L13.17 - Insert the Global Alias

13. Click on "OK" to choose the UDB.

14. The text area of the Properties inspector, after <#RegionLev2#> type in <#Machine#> <#Mixers#>.

15. Change the alignments "Text" and "Bound" to center.

16. Uncheck the "Stretch Text on Resize" option and click on the "Font" button.

17. Set the font size to eighteen (18) and click on “OK” to close the **Font** dialog.

18. Check the “Fill Color” and “Gradient Fill” property in the Property Inspector and click on the ellipse button for the Gradient Fill property.

19. Click on the color square for Color 1 and change it to blue (RGB: 198, 198, 255).

20. Set the Detail (steps) to fifty (50) and the Offset Y (%) to 0.

21. In the quick set section choose the lower left hand option.

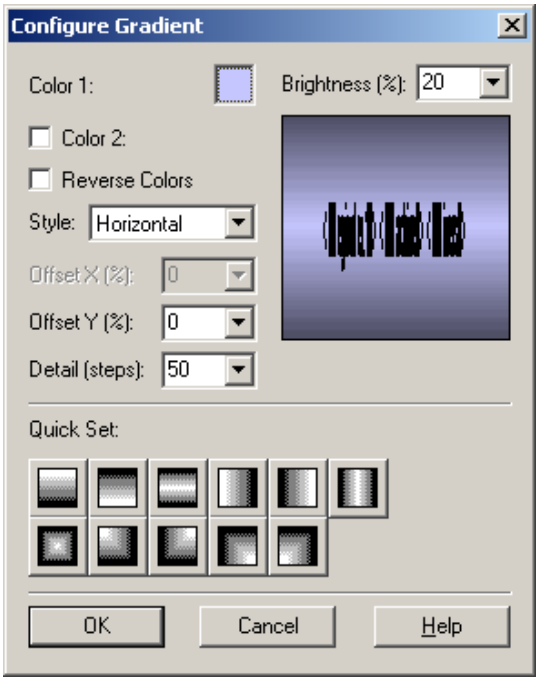


Figure L13.18 - Configure Gradient

22. Click on “OK” to close the **Configure Gradient** dialog.

23. Your Property Inspector should look similar to Figure L13.19.

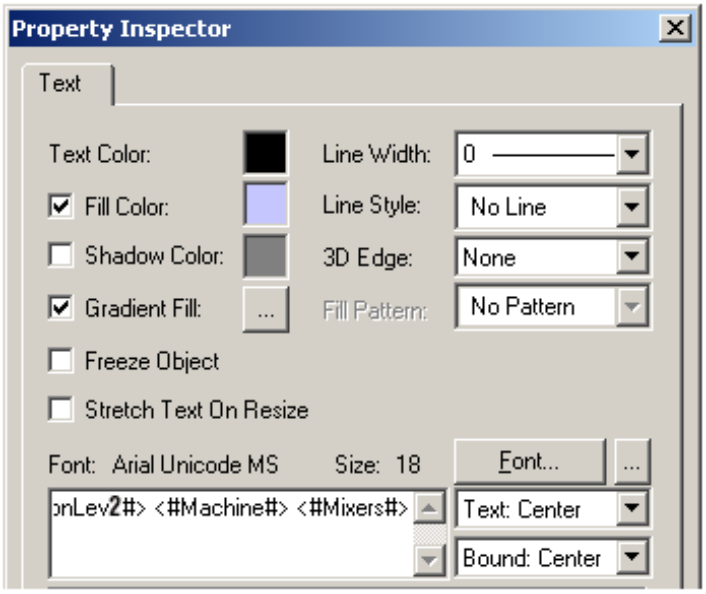


Figure L13.19 - Text Label Property Inspector

24. Click on “OK” to save the settings and close the Property Inspector.

25. Resize the text object so that it fits the entire width of the display and change the height of the object accordingly. The text object should look like a header for the display.

26. Click on the “Process Point” button on the Dynamics toolbar to insert a Processpoint/DataEntry dynamic. Click on the area under the tank.

27. The **Property Inspector** will appear with the PPT/DE tab active.

28. Browse into ICONICS.DataWorX32.8 → Registers → USA → Foxborough → Machine1 → Mixers → MixerA and select the Level tag.

29. After clicking on a tag, it is moved to the entry box at the top of the **Unified Data Browser**, where it becomes editable.

30. Highlight **MixerA**, as depicted in Figure L13.14.

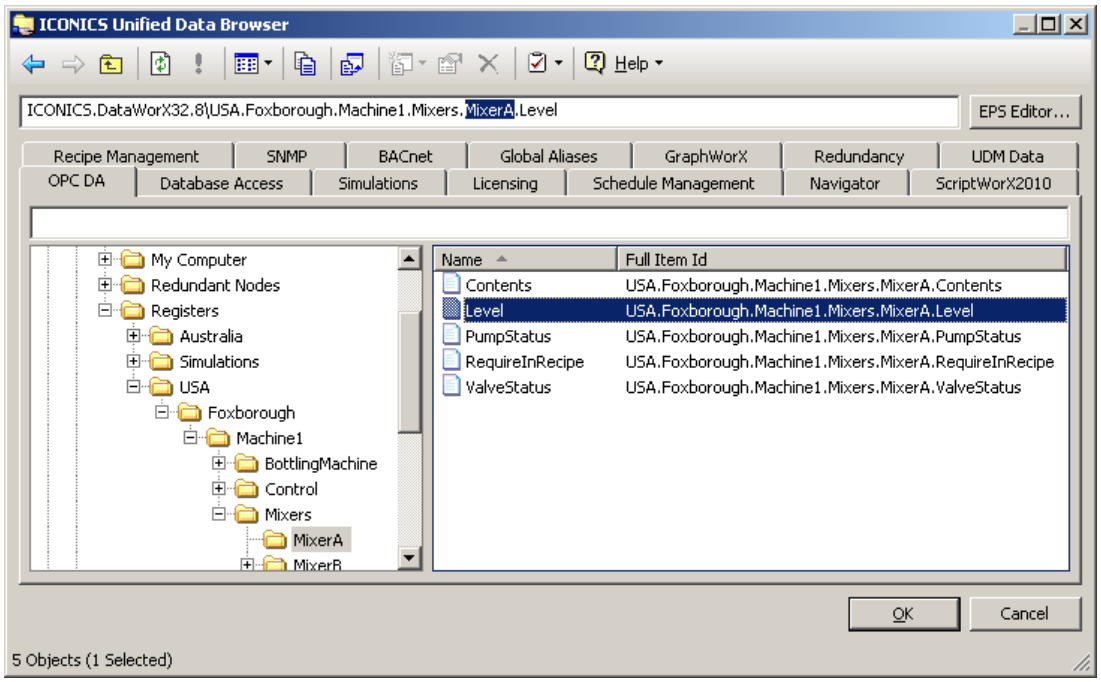


Figure L13.20 - Highlighting the Mixer in the Tag Name

31. Click on the Global Aliases tab and then click on Aliases on the left-hand side.

32. Select Mixers on the right-hand pane. The selected MixerA will be replaced by the global alias, as depicted in Figure L13.21.

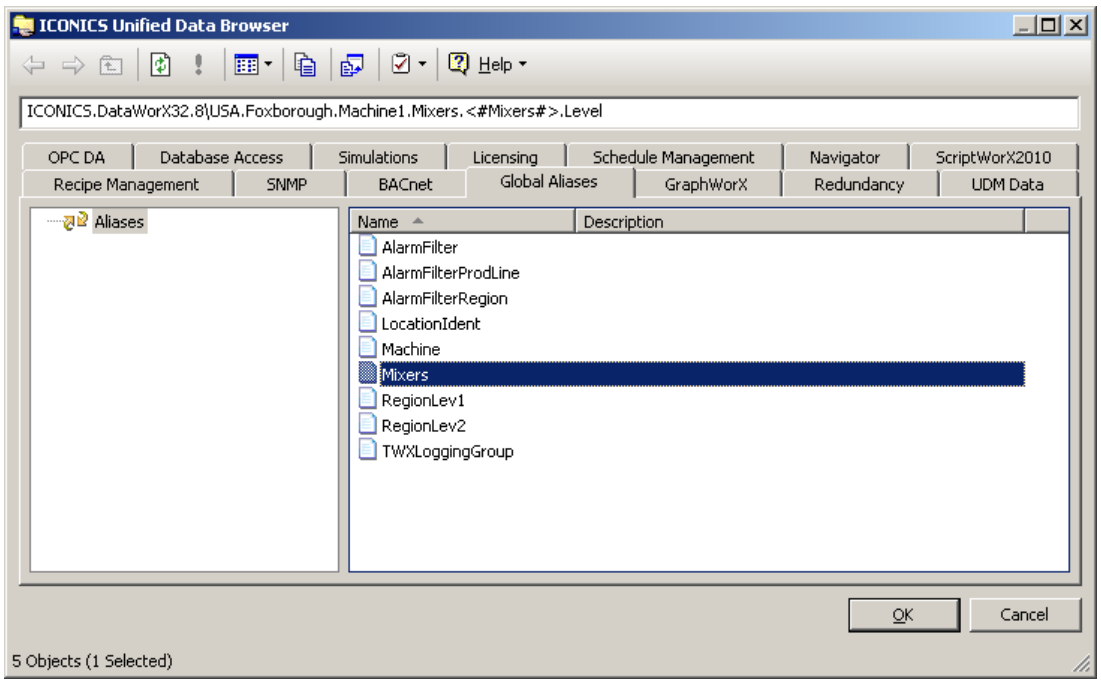


Figure L13.21 - Inserting the Global Alias Part in the Tag

33. Click on “OK” to choose this path as the data source of the process point and return to the Property Inspector. Click on “OK” again to close the Property Inspector.

34. Resize the tag so that it is bigger and easier to read.

35. We will need to replace other parts of the tag, but for this, we will use the search and replace utility. Click File → Save to save this display. Close the AcmePillMachine display if you have it open.

36. Click in the Edit menu and select “Replace.”

37. In the **Replace** dialog, Find: field type in Machine1.

38. Type in <#Machine#> in the Replace with field.

39. Choose the “Multiple Files” option in the Scope section and click on the ellipse button next to it.

40. In the **Select File** dialog click on the “Add” button. Choose the ACMEPillMachine.gdf and ACMEMixers.gdf displays. Click on “Open” to add the file.

41. Click on “OK” in the **Select File** dialog to close it.

42. Click on “Apply” in the **Replace** dialog to perform the search and replace.

43. You will see a dialog asking you to confirm the replace, click on “Yes to All.”

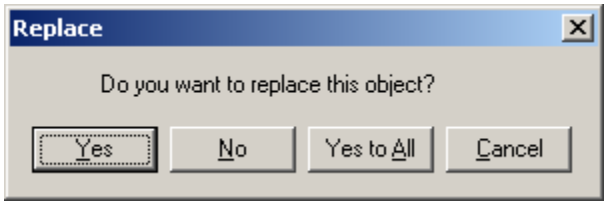


Figure L13.22 - Replace Confirmation Dialog

44. When the replace finishes, you will receive a dialog giving you the summary of the search and replace process.

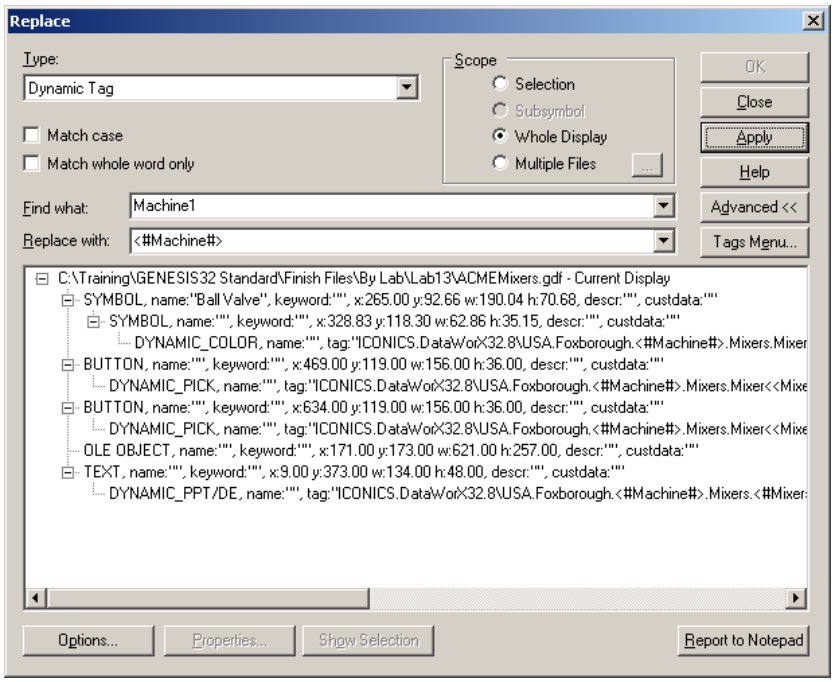


Figure L13.23 - Search and Replace

45. Repeat steps 37-44 giving it search and replace text as listed in Table L13.2.

Table L13.2 - Search and Replace

Find what	Replace with
USA	<#RegionLev1#>
Foxborough	<#RegionLev2#>

46. When you are done with the search and replace, click on "Close" to close the **Replace** dialog.

13.4.2. Adding the Size Dynamic

1. Select the cut-away object, right-click on the symbol and choose "Edit Symbol" from the context-sensitive menu, or press and hold the SHIFT key and right-click to edit the symbol.

2. Select the blue layer of the cut-away symbol and click the size dynamic button on the Dynamics toolbar. A Property Inspector will appear with the **Size** dynamic properties page visible.

3. Click on Data Tags to open the Unified Data Browser.

4. Browse to ICONICS.DataWorX32.8 → Registers → USA → Foxborough → Machine1 → Mixers → MixerA and select the Level tag on the OPC DA tab.

5. After clicking on the tag, it is moved to the entry box at the top of the **Unified Data Browser** where it becomes editable.

6. Highlight "MixerA" in the tag path, as depicted in Figure L13.24.

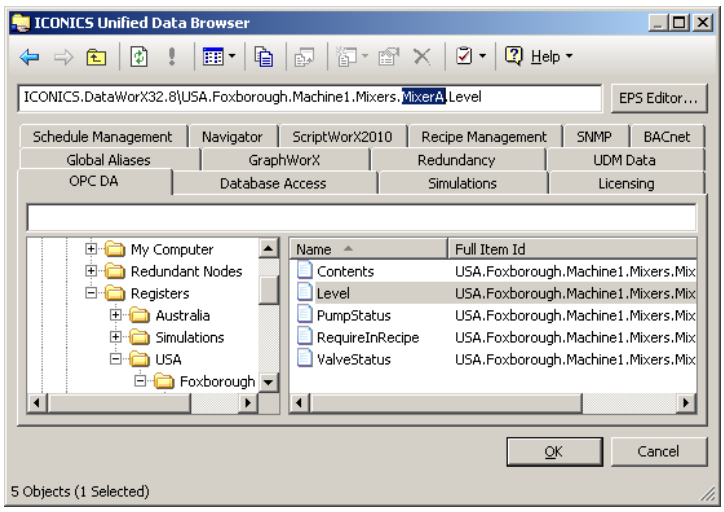


Figure L13.24 - Highlighting the Mixer in the Tag Name

7. Click on the Global Aliases tab. Select "Mixers" in the right-hand pane. The selected MixerA will be replaced by the global alias, as depicted in Figure L13.25.

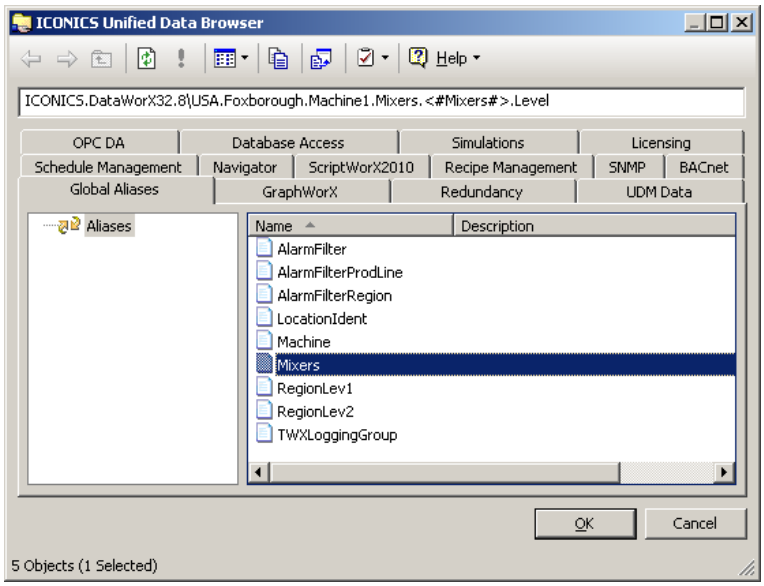


Figure L13.25 - Inserting the Global Alias Part in the Tag

8. Highlight Machine1 in the tag path and select the "Machine" alias.

9. Highlight Foxborough in the tag path and select the "RegionLev2" alias.

10. Highlight USA in the tag path and select the "RegionLev1" alias.

11. Your tag path should look similar to Figure L13.26. Click on “OK” to return to the Property Inspector.

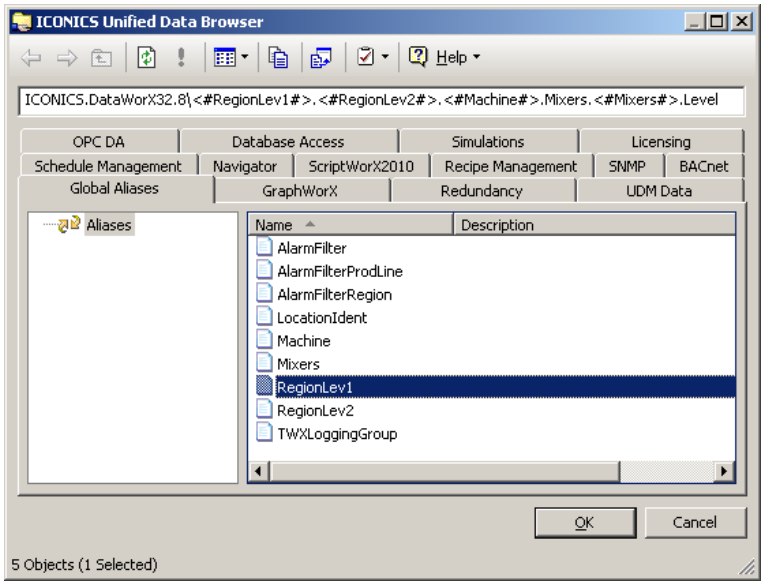


Figure L13.26 - Tag Path in UDB

12. Select the button on the top/left corner in the Size tab of the Property Inspector.

13. Enable “Range Override” to allow the entry of ranges. Enter zero (0) as the Low Range and 100 as the High Range.

14. Change the Sizing mode from Scale to Clip.

15. Click on “OK” to apply the changes and close the Property Inspector. Save the display.

13.5. Global Aliases in the Trend Viewer

1. Double-click on the **Trend Viewer ActiveX** in the ACMEMixers display to open its property pages.

2. Select the Rates tab to configure the data collection rate and trend period of the trend window.

3. Set the Data Collection Rate to five (5) seconds as this was also the data collection rate used in the Data Logger. Although the data collection rates do not have to be the same in the Logger and Viewer, it is advised to make the Trend Viewer's data collection rate an integer multiple of the Data Logger's data collection rate.

4. Set the Display Refresh Rate to one (1) times the collection rate so that the display will refresh every 5 seconds.

5. Set the History Refresh Rate to 60 seconds. The history refresh rate is the rate at which the viewer checks in the database to see if new data have been logged. Recall that the Data Logger is configured to commit all collected data to the database every minute so this history refresh rate makes sense.

6. Enter a Trend Period of five (5) minutes, as depicted in Figure L13.27.

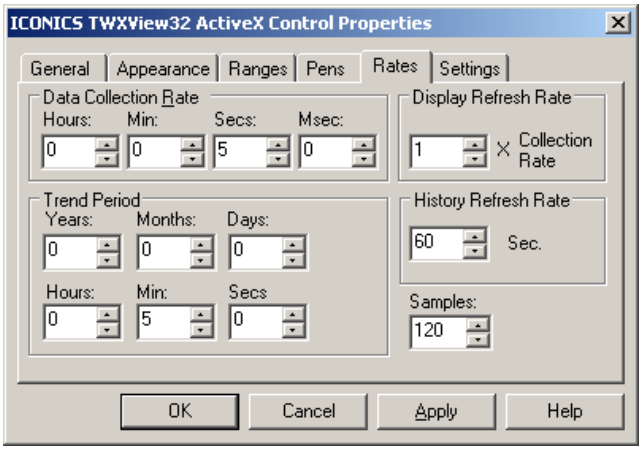


Figure L13.27 - Configuring the Data Collection Rate

7. Select the Appearance tab and select the Workspace page to configure the appearance of the main trend window.

8. Uncheck Border and 3D Edge.

9. Select the Details page to configure the appearance of the detail window.

10. Change Number of Entries to one (1), and uncheck Show Header.

11. Click on the "Spacing" button to open the **Details Entry Spacing** dialog, as depicted in Figure L13.28. This tool allows you to configure the spacing of the pen details to your liking. Fix the spacing so that all columns can be seen and there is no scroll bar at the bottom.

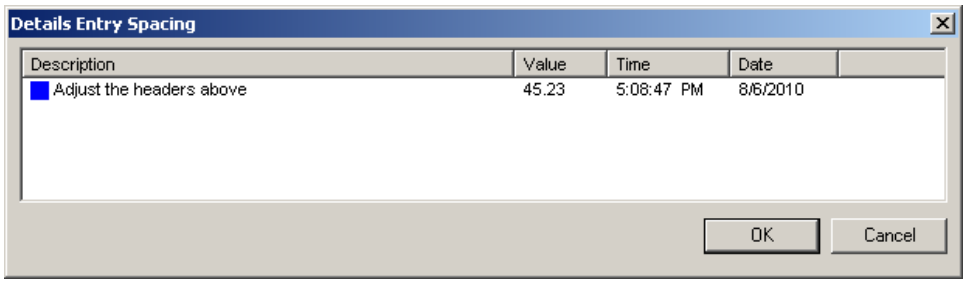


Figure L13.28 - Detail Section Column Spacing

12. Click on “OK” to return to the Appearance tab, as depicted in Figure L13.29.

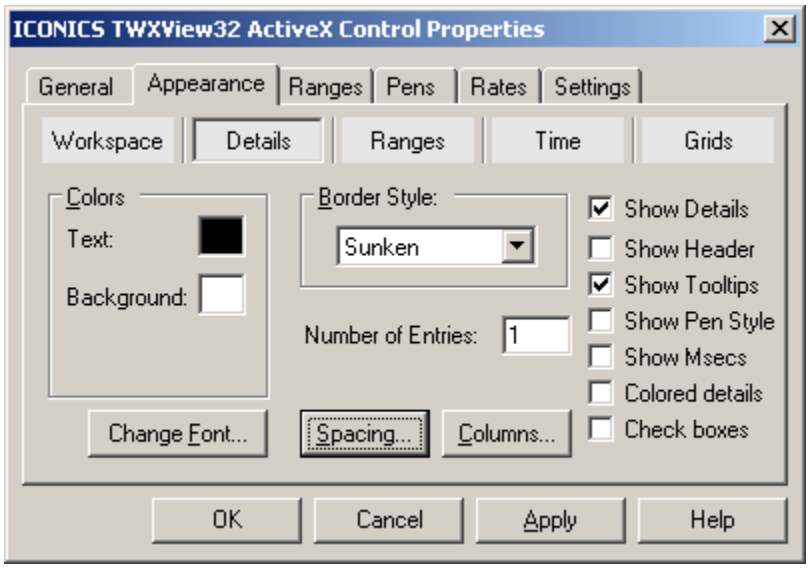


Figure L13.29 - Trend Viewer Appearance: Details

13. Select the Pens tab to add the logged signals to the plot.

14. Click on Tags Menu → History Tags to open the Unified Data Browser.

15. Browse to ICONICS.TWXSQLSvr.1 → Foxboro → Machine1.

16. Double-click on MixerA.level in the right-hand panel to add it to the Selected Item IDs section, as depicted in Figure L13.30. Click on “OK” to add the OPC HDA tag and return to the Trend Viewer Properties window.

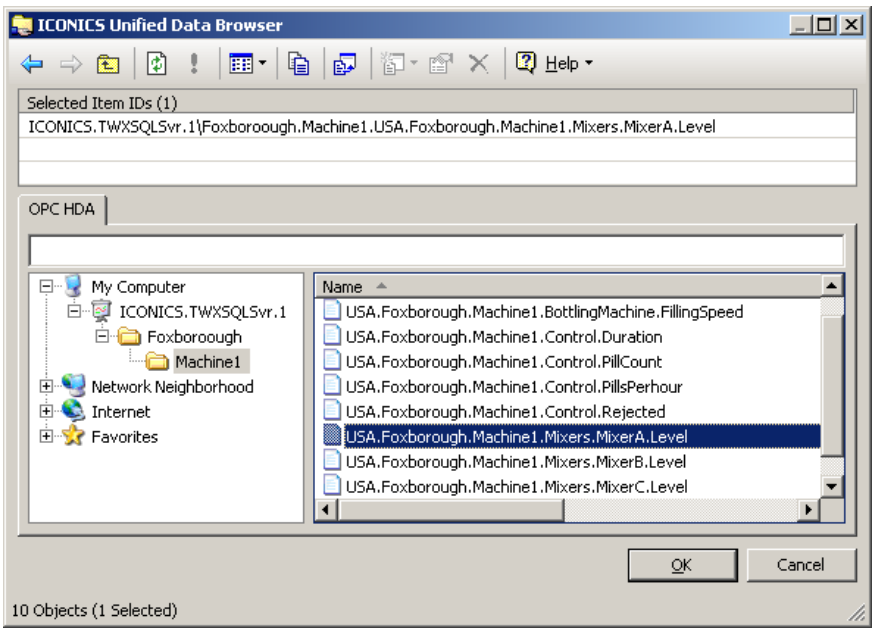


Figure L13.30 - Selecting the Logged Signals

17. Select the MixerA.Level pen in the Pens tab and click on the "Edit" button, as depicted in Figure L13.31.

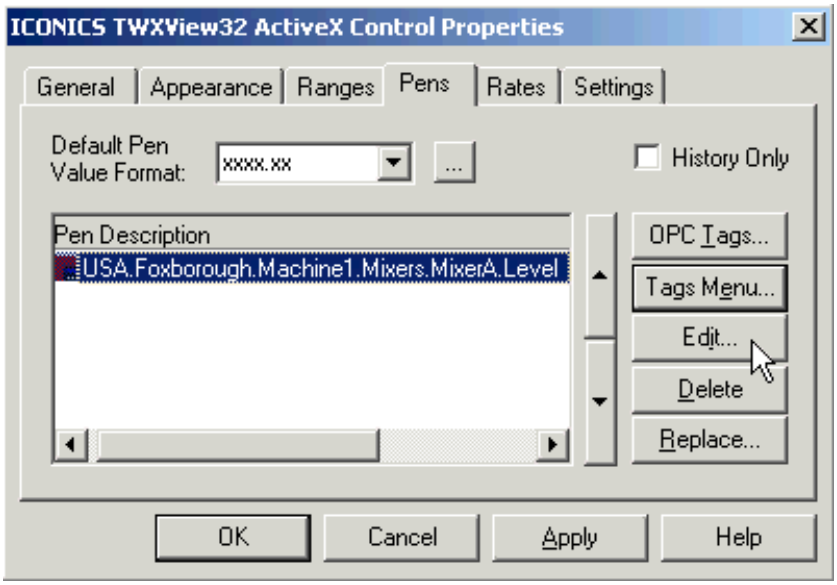


Figure L13.31 - Editing the History Pen

18. In the "Tag" field change **MixerA** to **<#Mixers#>**, **Machine1** to **<#Machine#>, Foxborough** to **<#RegionLev2#>, USA** to **<#RegionLev1#>**. The resulting may read something like:

ICONICS.TWXSQLSvr.1\<#RegionLev2#>.<#Machine#>.<#RegionLev1#>.<#RegionLev2#>.<#Machine#>.Mixers.<#Mixers#>.Level

> ***Tip!*** Be sure to change all instances of Foxborough and Machine1 to ensure that the tag will work properly for all machines in all regions later.

19. Change the **Description** field to **<#LocationIdent#> <#Machine#> <#Mixers#> Level**, as depicted in Figure L13.32.

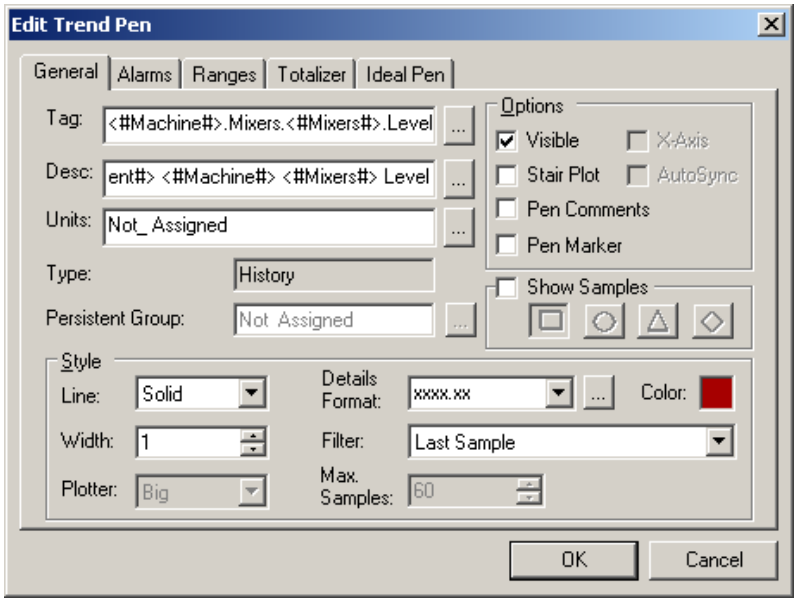


Figure L13.32 - Configuring the Pen Details

20. Enable “Stair Plot” to draw straight lines only and the “Pen Marker” to enable the marker.

21. Change the Width to two (2), and the Filter to “Raw Data.”

22. Click on “OK” to close the **Edit Trend Pen** dialog and return to the Trend Viewer Properties window.

23. Click on “OK” to return to the display. Save the display.

13.6. Global Aliases in the Alarm Viewer

1. Double-click on the **Alarm Viewer ActiveX** in the ACMEMixers display to open its property pages.

2. Select the Subscription tab where you will find two preconfigured subscriptions, as depicted in Figure L13.33.

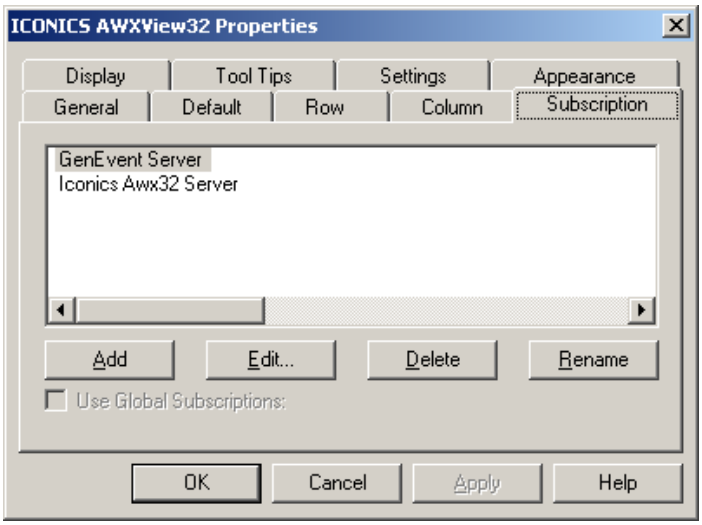


Figure L13.33 - Alarm Viewer Subscription

3. In this display, only the Alarm Server will be used. Select the "GenEvent Server" and click on "Delete." Click on "Yes" in the confirmation dialog, as shown in Figure L13.34.

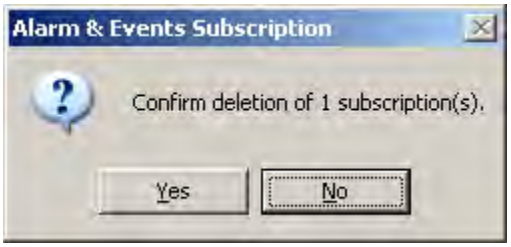


Figure L13.34 - Deleting a Subscription

13.6.1. Column Configuration

1. Select the Column tab in the Alarm Viewer Properties window where two lists of columns will appear. The left-hand side shows all Available columns and the right-hand side shows the Selected Headers columns that will be used by the Alarm Viewer.

2. In the **Selected Headers** list, select **Time / Date**, **Priority**, **Type**, **Quality**, and **Node** by clicking on the relevant entries while pressing and holding down the **CTRL** key, as depicted in Figure L13.35.

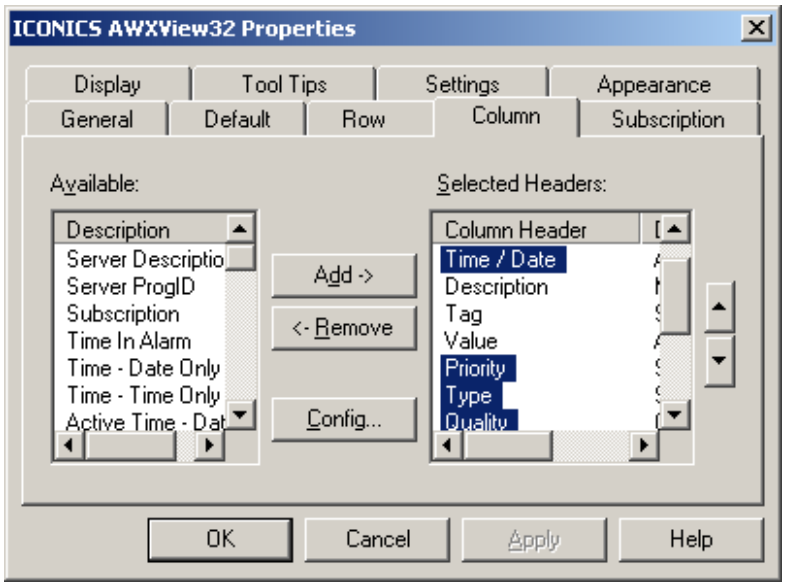


Figure L13.35 - Alarm Viewer Selected Columns

3. Once all aforementioned selections have been made, click on “Remove” to remove these columns.

4. Select “Event Time” from the bottom of the list and move it upwards until it is the second item in the list using the arrow buttons on the right of the list.

5. Right-click on “Comment” and select “Configure.” Check **Cell** and **Hide**, as depicted in Figure L13.36. This hides the comment column in runtime.

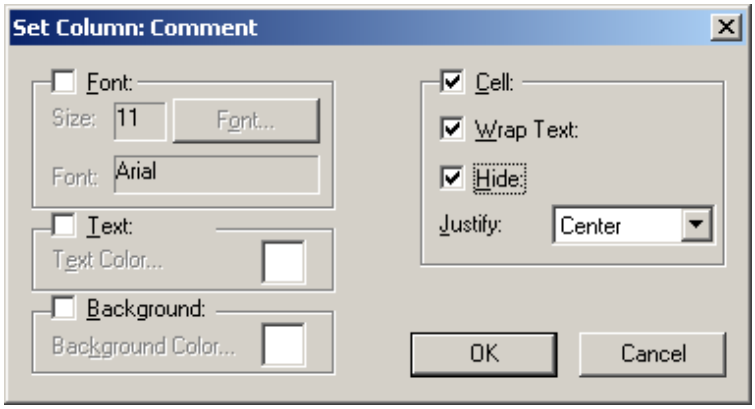


Figure L13.36 - Hide the Comment Column

6. Click on “OK” to return.

7. Click on “Apply” to commit the changes made thus far.

13.6.2. Sorting and ToolTips

1. Select the Display tab to change the sort order.
2. Click on "Sort" and set the "Sort Alarms by" dropdown list to "Event Time." Set the sort order to "Descending."

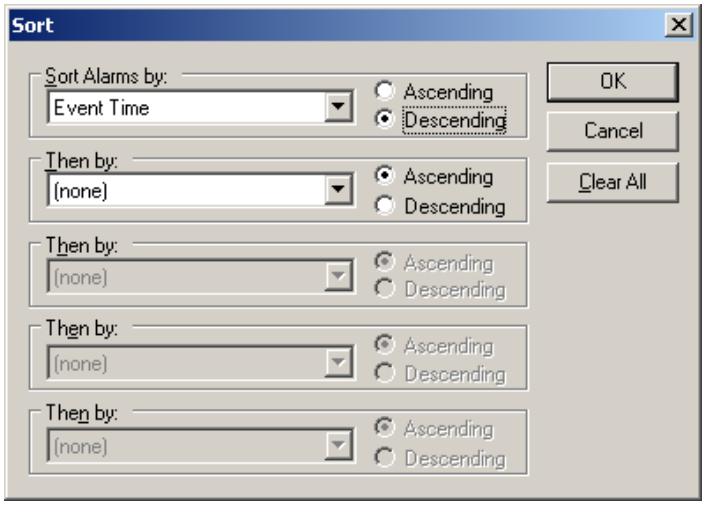


Figure L13.37 - Changing the Sort Order

3. Click on "OK" to return to the properties dialog.
4. Select the ToolTips tab to change the ToolTip properties.
5. Set the Width to fifty (50), allowing the ToolTips to be half of the Alarm Viewer window width, as depicted in Figure L13.38.

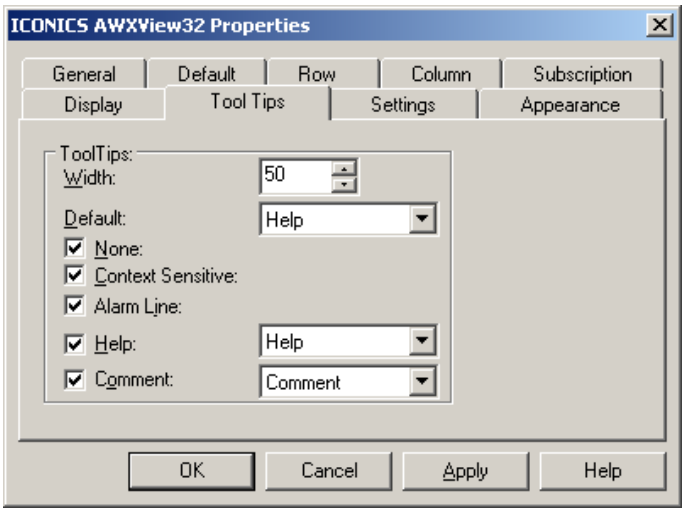


Figure L13.38 - Changing the Tooltip Properties

6. Select the Default tab and uncheck the "Scientific Notation" option.

13.6.3. Alarm Filtering Using Global Aliases

The Alarm Viewer can apply a client-side filter. Recall that this does not actually change the subscription the Alarm Viewer makes to the Alarm Server, but it does limit which alarms are visible to the operator. In this application, a client-side filter will be created with aliasing that will resolve in the Alarm Viewer when the pop-up window is shown.

1. Select the Display tab to add the filter.
2. Click on "Filter" button to open the **Client Filters** dialog.
3. Click on "Add" to create a new filter. The **Filter Wizard** dialog opens, as depicted in Figure L13.39.

Figure L13.39 - Filter Wizard Dialog

4. Because the Filter Wizard does not have any global aliasing options, we will need to use the **Expression Editor**. Click on the "Advanced" button.
5. In the Expression Editor, choose Functions → String → like (StringSource, StringCompare, BooleanMatchCase) Wildcard String Compare, as depicted in Figure L13.40.

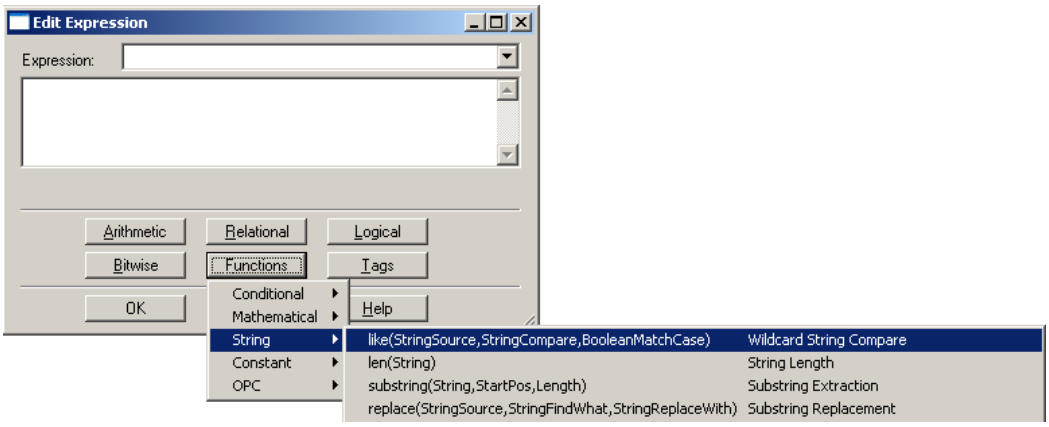


Figure L13.40 - Choosing the Like() Function

6. For the first argument, choose Tags →Advanced, as depicted in Figure L13.41, to open the **Attribute Filter** dialog.

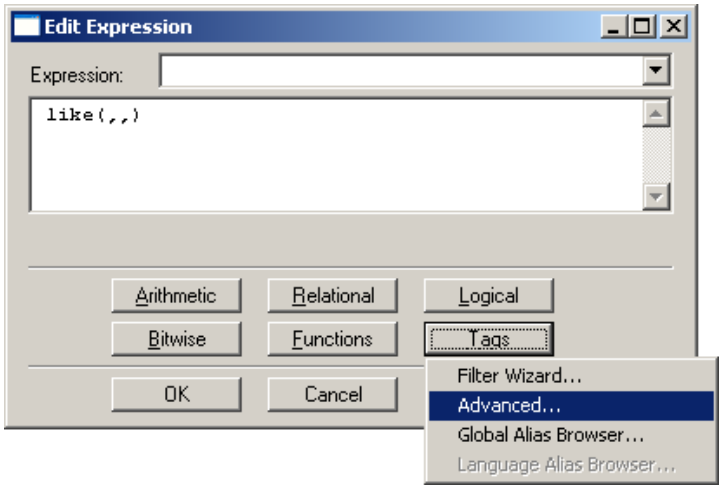


Figure L13.41 - Choosing Advanced Data Sources

7. In the **Attribute Filter** dialog choose **Source** as the string source, as depicted in Figure L13.42. Click on "OK" to return to the Expression Editor.

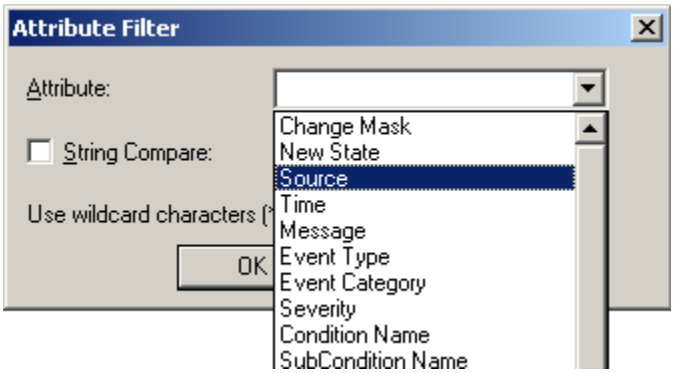


Figure L13.42 - Choosing the Source Alarm Tag Attribute

8. Place your cursor after the first comma and type in a beginning quotation mark.

9. Click Tags → Global Alias Browser, as depicted in Figure L13.43.

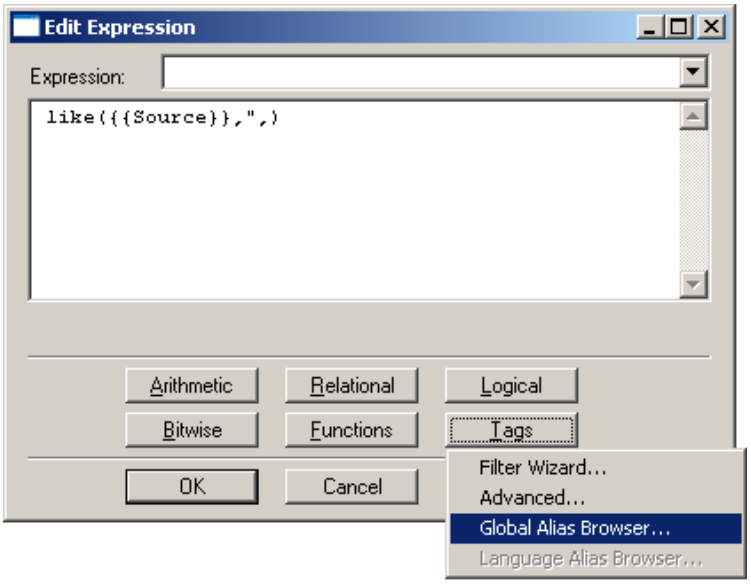


Figure L13.43 - Choosing the Global Alias Browser

10. Select "RegionLev2." This global alias will be shown in the top field, as depicted in Figure L13.44. Click on "OK" to close the UDB.

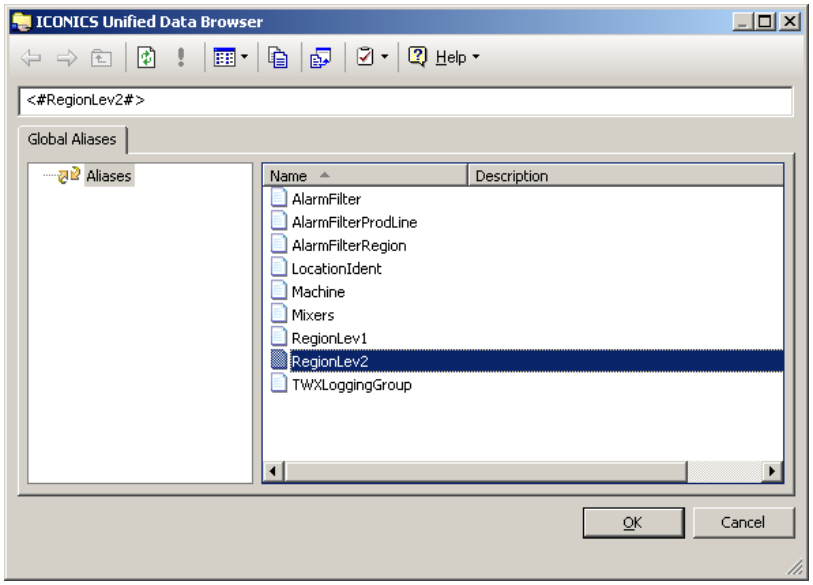


Figure L13.44 - Inserting the Global Alias as the Comparison String

11. Type an asterisk (*) in the Expression Editor.

12. Repeat steps 9-11, choosing the Machine and AlarmFilter as the alias each time. When you are done, type in a quotation mark.

13. Finally, place your cursor after the second comma, and type a **0** to make the comparison not case sensitive. The final expression should look like Figure L13.45. Click on "OK" to save the expression and return to the **Client Filter** dialog.

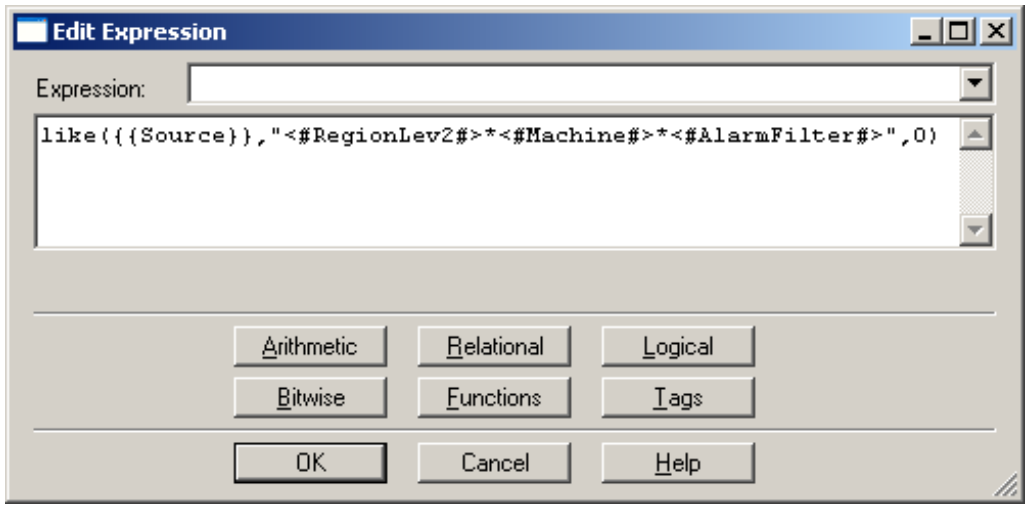


Figure L13.45 - Final String Comparison Expression

14. Check the "New Filter" to enable it, as shown in Figure L13.16, and click on "OK" twice to return to the display.

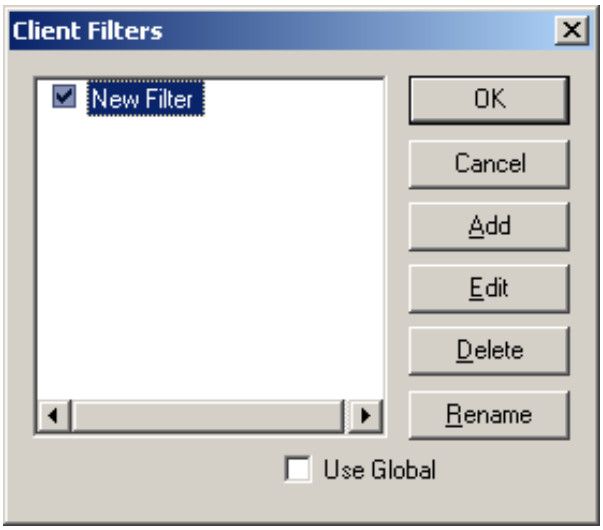


Figure L13.46 - Enabling the Client-Side Filter

15. Save the display using File → Save.

16. Close the display using File → Exit.

13.7. Changing Pick Actions in the Pill Machine Display

1. Open your **ACMEPIllMachine.gdf** display and select MixerA.
2. Eight handles appear around the symbol.
3. Double-click on the object to bring up the **Property Inspector**.
4. In the **Pick** tab of the **Property Inspector** dialog, click on Set Aliases → Global Aliases to open the Themes dialog.
5. Select and check "Mixers." Click on "Edit" to open the **Theme Editor** dialog.
6. Set the Theme Scope to Absolute Machine.
7. Under Theme Item, select "MixerA" as depicted in Figure L13.47.

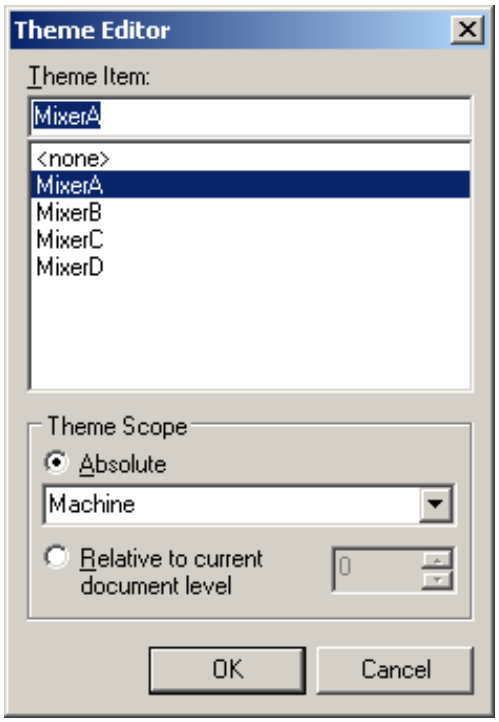


Figure L13.47 - Theme Editor

8. Click on "OK" to close the **Theme Editor** and return to the **Themes** dialog, as depicted in Figure L13.48.

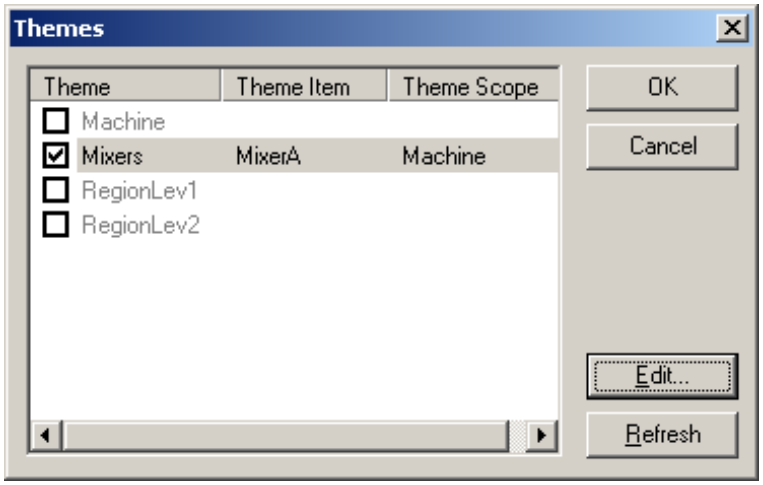


Figure L13.48 - Theme Selection

9. Click on "OK" to close the **Themes** and return to the **Property Inspector** dialog.

10. Click on "OK" to close the **Property Inspector** dialog.

11. Repeat the above steps for all mixers, setting the global aliases as follows:

Table L13.3 - Alias Settings for the Pick Fields

Mixer	Global Alias Setting
Mixer B	MixerB
Mixer C	MixerC
Mixer D	MixerD

12. Save the display using File → Save.

13. When the file is saved choose **Runtime** from the menu.

14. Click on one of the levels to show the display and a pop-up window in runtime mode, as shown in Figure L13.49.

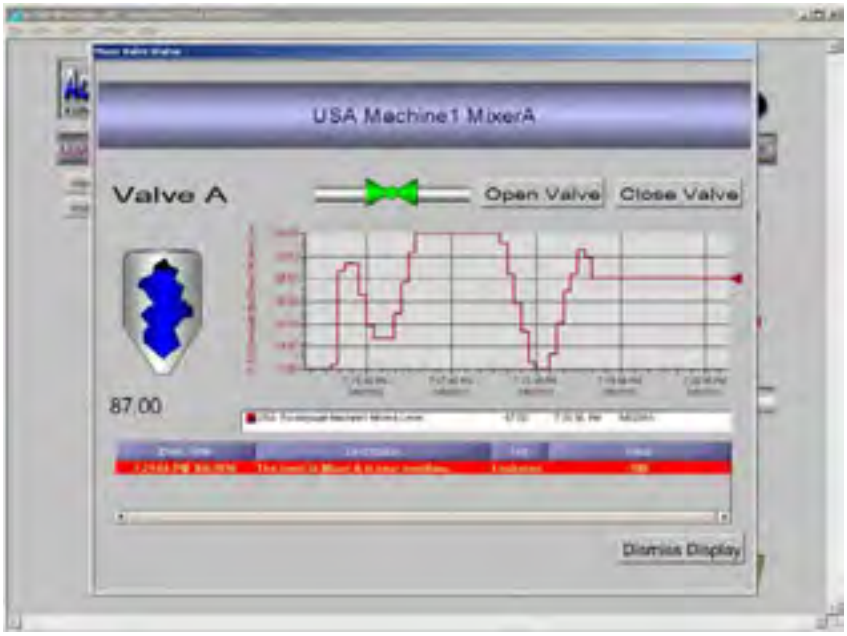


Figure L13.49 - Using Global Alias in Runtime

15. If you would like, with little instruction you can add a Button to Control the Pump and change the Level value PPT to connect to the Level_Setpoint value as a PPT/DE (Data Entry).

16. Your final display should resemble the figure below.

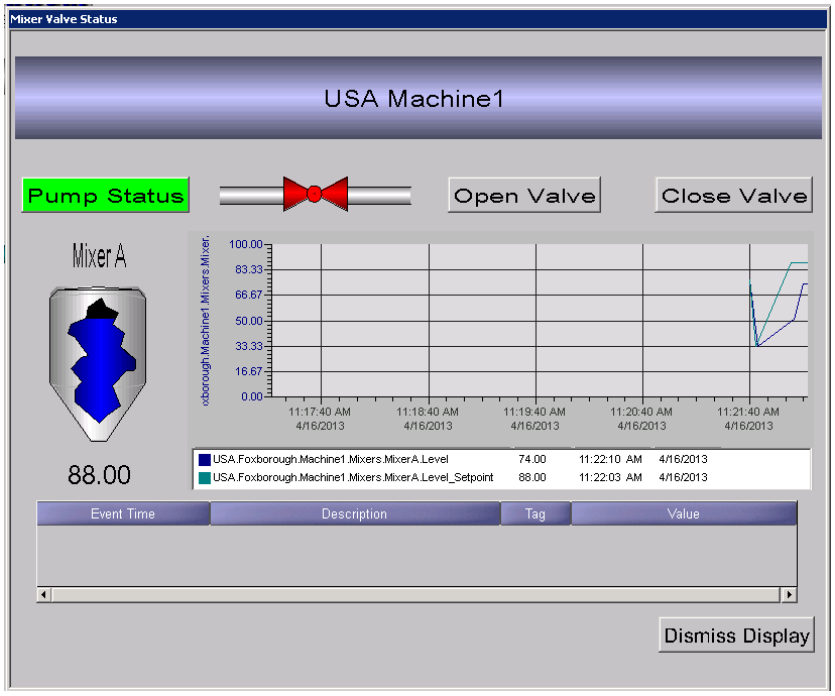


Figure L13.50 – Pump Status and Level Setpoint Added to the AcmeMixers Display

17. As an added element of assistance, we've included the figures below.

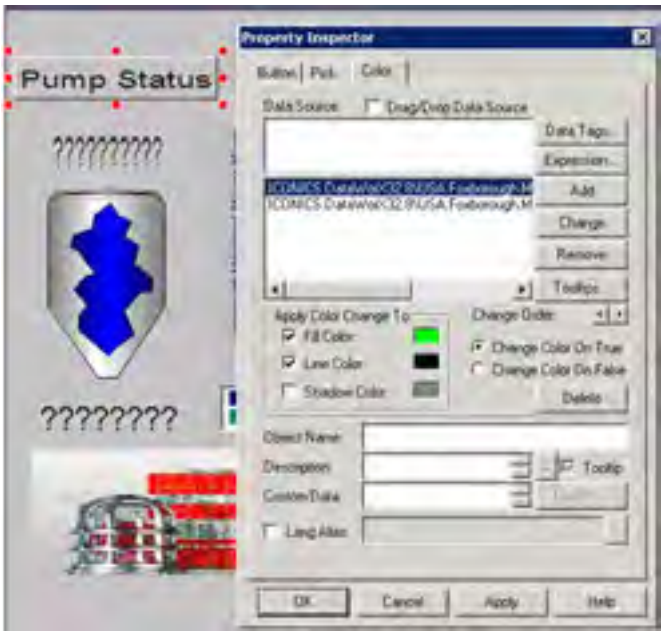


Figure L13.51 – Pump Status Button Pick and Color Dynamics

Figure L13.52 – Level_Setpoint Data Entry (PPT/DE)

18. Make sure the file is saved.

19. When you're done, click on File → Exit to close GraphWorX32.

Review: Aliasing R13

13.1. Chapter Objectives

In this chapter, we discussed how Aliasing can help you reduce development time. There are three different kinds of aliasing in GENESIS32 and we touch two commonly used alias types: Local and Global Aliasing in this chapter. By now, you should know how to do the following:

- Configured Local Aliases
- Set Local Alias during runtime
- Configure Global Aliases
- Configure Themes and Theme Items
- Associate Global Aliases with Themes and set values for theme items
- Using a Global Alias in a display
- Select theme items during runtime

13.2. Summary

Aliasing is sometimes thought of as a variable whereby a string in configuration mode will become another string during runtime depending on the selected value. Aliases are generally used as a complete data source, parts of a data source, and as simple strings. Through the use of aliasing, you can simply keep one display instead of multiple. For example, you may have ten different broilers that you need to monitor in a campus. These broilers are of the same model and you need to show the same information from each, but you need to have ten different displays, one for each broiler. Through using aliases, you can keep just one display, aliasing the parts where the broilers differ from each other. There are two types of aliases that can help you achieve this type of functionality: Local Alias and Global Alias.

Local alias, as the name suggests, is local to GraphWorX32. You can only use this type of aliasing within GraphWorX32. Although you can add ActiveX Control into GraphWorX32 displays, you may not use local aliases within ActiveX Controls. Local Alias has the ability to alias on an Object level. This means that you can use the same alias on different objects in a display and define the alias value individually according to the object. You can also pass local alias values from one display to another.

Global Aliasing the other hand can be used throughout GENESIS32. Because it can work in many different applications, you will need to define a list of alias and themes. During runtime, the client will resolve the alias value base on the chosen theme item. In recent years, we have also added the ability to use Global Alias dynamically. That means you can create a global alias and its value as you develop your displays without using the Global Aliasing Configurator.

13.3. Terms

Alias Resolution	GAS Theme	Global Alias
Global Aliasing Configurator	Local Alias	Object Level Aliasing
Second Level Aliasing	Theme Item	

13.4. Test Your Knowledge

In this chapter, we discussed local alias and global alias. You should now be able to answer the questions below.

1. What are some benefits to using aliasing?

    a. Reduces development time
    b. Maintain less files
    c. Easier to update displays
    d. All of the above
    e. None of the above

2. What does Second Level aliasing mean?

    a. Resolving a local alias that becomes another local alias to get to the final value
    b. Forwarding a local alias value to a child display
    c. Deducting a local alias value based on the prior selected alias value
    d. Creating a local alias based on input values

3. True or False: By using themes, Global Alias allows you to change the value of many aliases at the same time with a click of a button.

4. True or False: You do NOT always need to configure global alias in the Global Aliasing Configurator before using it.

5. Which of the following does NOT support local Alias?

    a. Process Points
    b. Color Dynamics
    c. Symbols
    d. ActiveX Controls

6. Which of the following pick actions allows you to assign a value to a local alias?

    a. Set Local Aliases
    b. Select Local Aliases
    c. Local Alias Dialog
    d. Download Local Aliases Values
    e. All of the above
    f. Only A and B
    g. Only A and C
    h. Only B and D
    i. Only C and D
    j. None of the above

7. Which of the following is NOT a valid theme scope?

    a. Machine Level
    b. Document Level
    c. Process Level
    d. All of the Above
    e. None of the Above

8. True or False: You can use automation method to set a global and local alias value.

9. True or False: You can import and export local alias and it's values from a text file.

Language Aliasing 14

Many companies have facilities around the world and operators in other countries may not be comfortable operating in English. By using Language Aliasing, we can easily change the language of an application rather than recreating completely new applications whenever a facility opens in a new country. By the end of this chapter, you should be familiar with the following topics:

- Create a language configuration database in the Language Configurator
- Configure aliases for several different languages
- Use expressions to change values between different measure systems
- Switch language inside a GraphWorX32 display

14.1. Introduction to Language Aliasing

The Language Aliasing System (LAS) provides a way to translate text strings and to convert measurements for any languages required for use in ICONICS applications:

- **Translation**: A specially delimited or tagged string is replaced by an appropriate value for the selected language.
- **Conversion**: Different cultures use different units of measurement; the conversion system also converts one type of measurement unit to another.

The LAS translates and converts using aliases. As mentioned in the previous chapter, an *alias* is a defined string that represents another string during runtime. In English, a rose is “a rose,” but in Italian a rose is “una rosa,” in German it is “eine Rose,” in Dutch “een roos,” and so forth. Using a language alias you can define the text string for “a rose” in each language such that the displays you create can switch to the local client machine’s language; your Italian users see “una rosa” and your German users see “eine Rose.”

There are two major components of the LAS: the **centralized database** and any **client application** that uses the language switching mechanism. All language variations and conversion-expressions are stored in and retrieved from a centralized database configuration (Microsoft Access or Microsoft SQL Server). The centralized database is maintained using the Language Aliasing configurator.

14.2. Background

The language/locale setting in Windows is based on an ISO standard. A locale is similar to a culture, differentiating English (United States) from English (United Kingdom). There are 264 defined settings available in recent versions of Windows operating systems, including language/locale. When a language is set in Windows on a client machine, Windows' user interfaces retrieve the language alias translations for that language and cache the information locally on the client displaying the translations in the local display.

This language/locale setting establishes the language for the user account. GENESIS32 uses that same setting to determine the language/locale properties to be used in GENESIS32 applications at runtime. GENESIS32's LAS returns language settings for that language for any alias objects needing it for language translation. Similarly, GENESIS32 reads the unit binding for the current language to obtain unit conversions for values in the display.

Although GENESIS32 uses the language setting to determine the program action, it does not use any other Windows feature or dictionary to supply conversions. In order for your displays to have the correct language alias support you must configure translations and conversions for the languages in which you need them by using the LAS configurator.

Language aliases are applied to the client based on the active configuration. Only one configuration is active at any one time and that database can use only one language setting for displaying data on the client. Languages cannot be mixed in a display. This means that when you use a pick action to change the language/locale in a display, all of the language aliases defined in that display change to the currently selected language. In fact, the entire system changes to use the selected language.

14.3. Configuring Language Aliasing

There are two ways to configure language aliasing, either from Workbench32 or from the standalone configurator which can be launched by going to **Start → Programs → ICONICS Tools → Languages → Language Configurator**. The standalone configuration interface is shown in Figure 14.1.

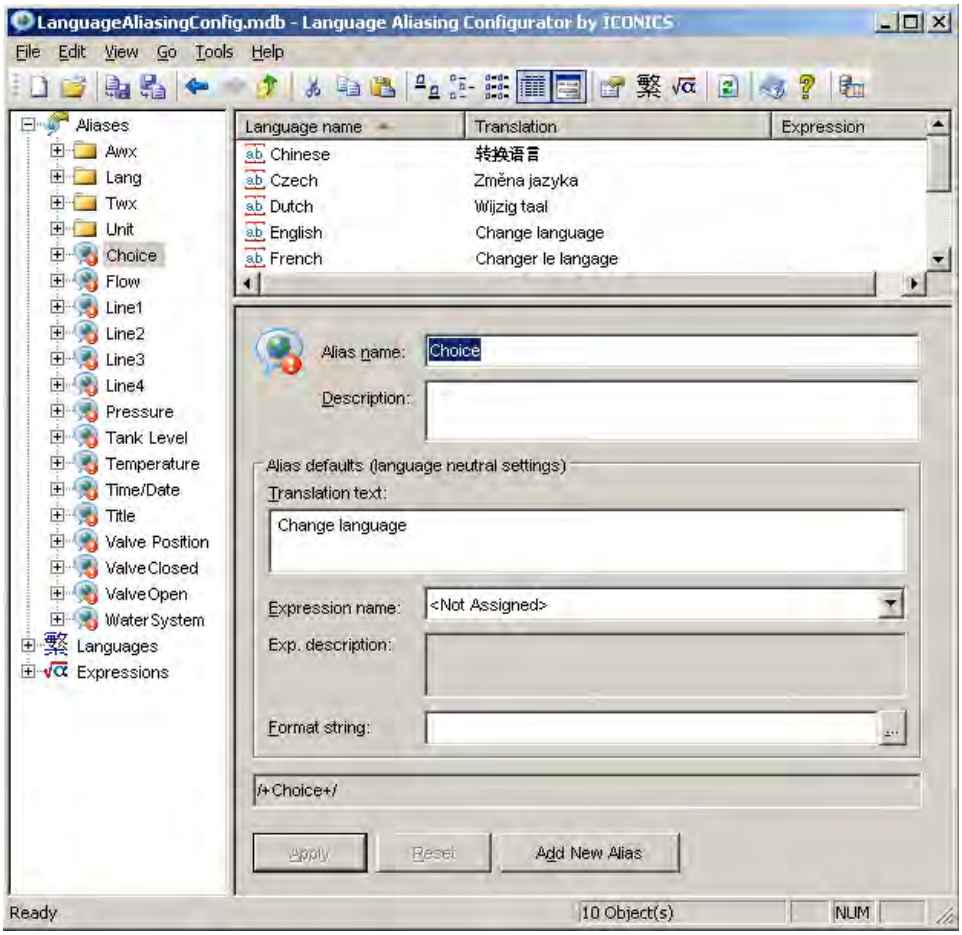


Figure 14.1 - Language Aliasing Configuration Interface

The Language Configurator, like many of the ICONICS configurators, contains a tree control on the left-hand side where you can select items to edit or add. Selecting an item in the tree changes the contents area, where it will display forms for you to configure the item.

14.3.1. Configuring Languages

Start configuring the LAS by deciding which languages you are going to use. Then define them by right-clicking on Languages in the tree control and add each language. These languages will be used for translating text as well as converting units of measurement for the locale. Therefore, both the language and the locale are important for each language definition.

Adding a language is a simple process. You can right-click on Languages in the tree control and select New → Language. Then in the form that opens, click on the "Browse…" button to select the language.

The other required fields populate automatically after you select the language as shown in Figure 14.2. You can define the following information in the configuration form:

- **Language name:** This is the name of the language. It should match with the settings configured in the Language setting. If you browse for the language in the Regional Specification section, the system will automatically fill-in the information.
- **Description:** This is an optional field. You can add in descriptions for the language as desired.
- **Enable translations for this language:** This option is checked by default. If you uncheck the option we will not consider this language for translation but we will consider it for conversions.
- **Default Language:** When you check this option, the language will become the default one in the GENESIS32 system.
- **Language:** You will have to browse for a language. This is to make sure that the computer you are using to generate the configuration recognizes the language. After you have chosen your language, the Language name and Local ID fields will be populated by the system.
- **Local ID:** The computer's identification for a selected language. This field is populated automatically after choosing a language.

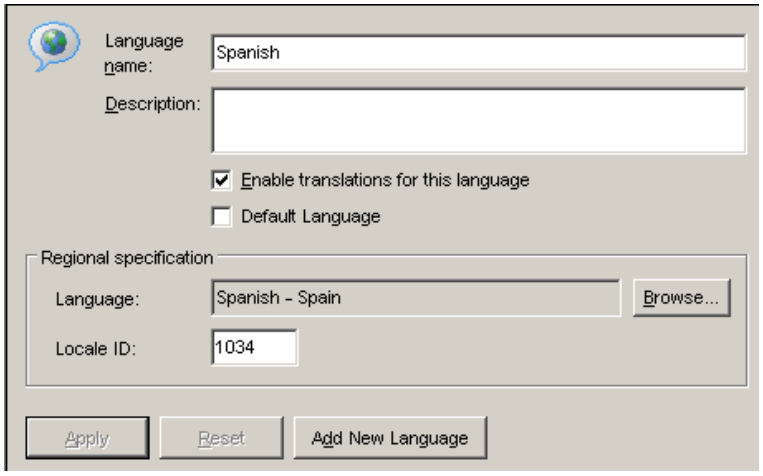


Figure 14.2 - Language Definition for Spanish

Once you have defined all of your languages you are ready to create language aliases.

14.3.2. Configuring Expressions

Because languages and cultures use varying methods of measurements such as distance, temperature, and currency, the language aliasing system (LAS) provides a way to convert units of measure and their expressions.

You define **Expressions** in the Language Configurator. Let us understand this with an example that converts values from centimeters to inches.

To define a conversion from centimeters to inches you start by defining an expression, as shown in Figure 14.3.

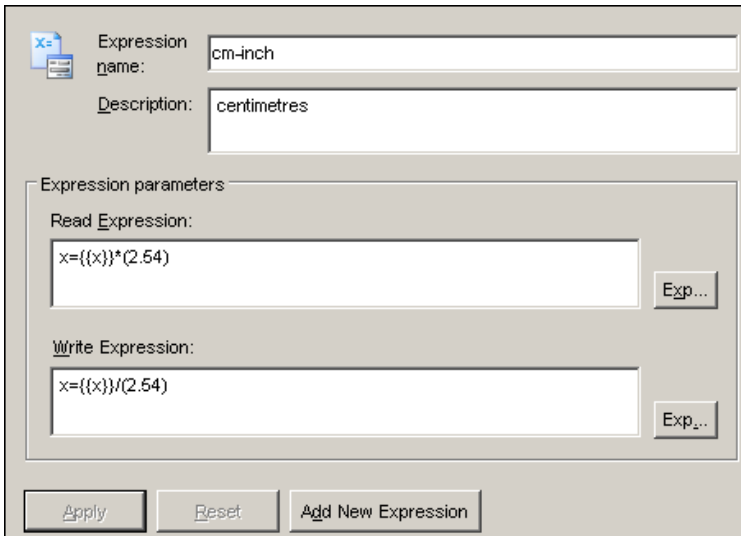


Figure 14.3 - Unit Conversion Definition

An Expression definition consists of the following:

- **Expression name:** Assign a name that describes the conversion direction (such as Centimeters to Inches).
- **Description:** A description if you think it is necessary.
- **Read Expression:** Specifies the calculation that will be applied to the original read value – **{{x}}** – in order to display it in its converted form. In our example it would be the calculation that converts a read value in centimeters and converts it to inches for display. You can use the Expression Editor to create your expression.

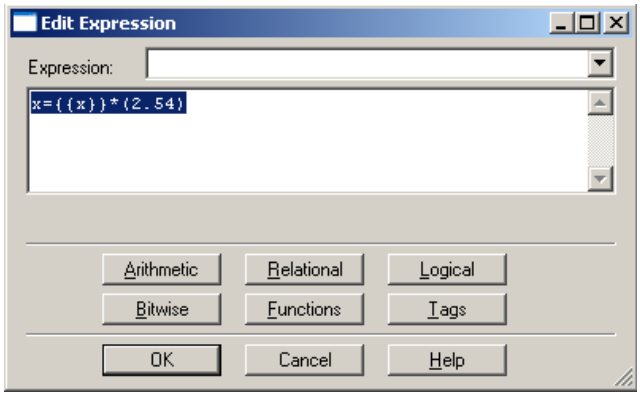


Figure 14.4 - Expression Tab in the Data Browser

- **Write Expression:** Specifies the calculation that will be performed on a value – **{{x}}** – that a user manually inputs into a converted data point. In our example it would be the calculation for taking a user-provided value in inches and converting the value to centimeters and write it to the process point.

14.3.3. Configuring Language Aliases

You can create language aliases in the Aliases root folder or you can create subfolders for them. To add an alias, you need to specify the following information, as is shown in Figure 14.5:

- **Alias name:** Give the alias a name that identifies the text that it translates.
- **Description:** Describe the alias if you wish. This field is optional.
- **Translation text:** Specify the Default Translation text that will be used if no language-specific translation is available for the language that is active during runtime.
- **Expression name:** Choose an expression if you want to perform conversions using this alias; otherwise, you can leave this field blank.
- **Format string:** Define the format of the converted value. For example, some countries use a period (.) as the decimal point and others use a comma (,). You can define the default format here.

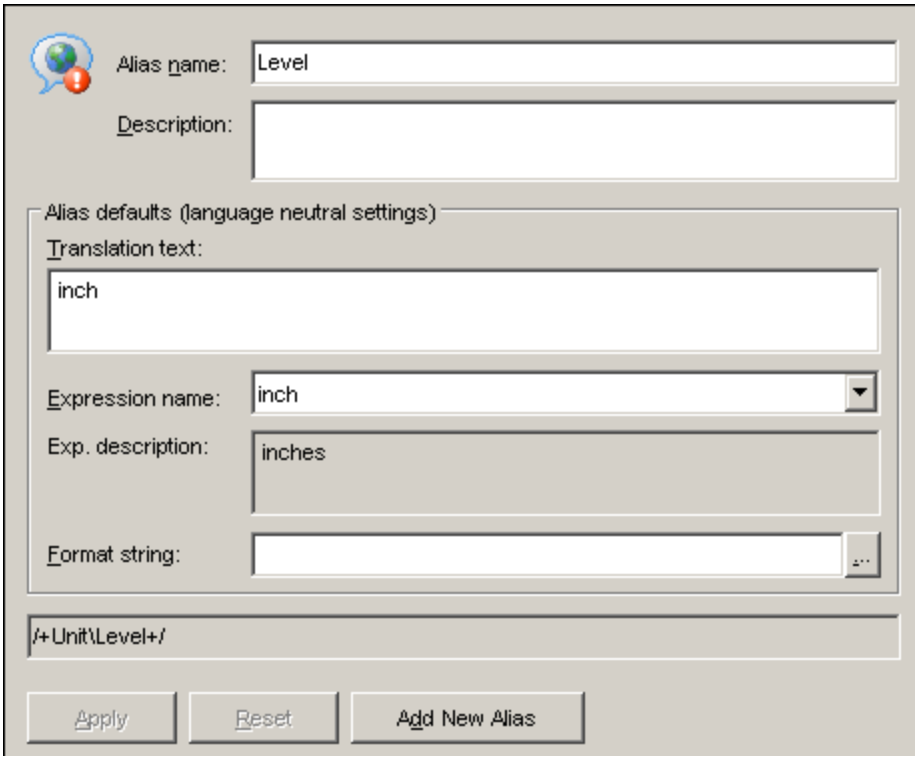


Figure 14.5 - Language Alias Definition

Notice that the bottom of the configuration form shows the syntax of the alias. You can use this text in client applications to insert the language alias.

Notice also that the alias contains child items which are the defined languages. Until you provide a translation for each language the icon for the alias in the tree control contains an exclamation mark to indicate its incompleteness. You can find a similar behavior with the child item under the alias. For languages that you have provided a translation string it shows an icon with text ; for ones that you have provided an expression it shows an icon with an equation ; and for ones that you have provided both text translation and conversion expression it shows an icon with both . For languages in which you have not defined any translations or expressions it will show a question mark ? to indicate its lacking configuration.

For each language under an alias there are two tabs: Text and Expression. On the Text tab you can provide the text translation to show for the alias when the user changes to the selected language. On the Expression tab you can choose the appropriate expression from the dropdown list and provide the proper formatting string that makes sense for the language.

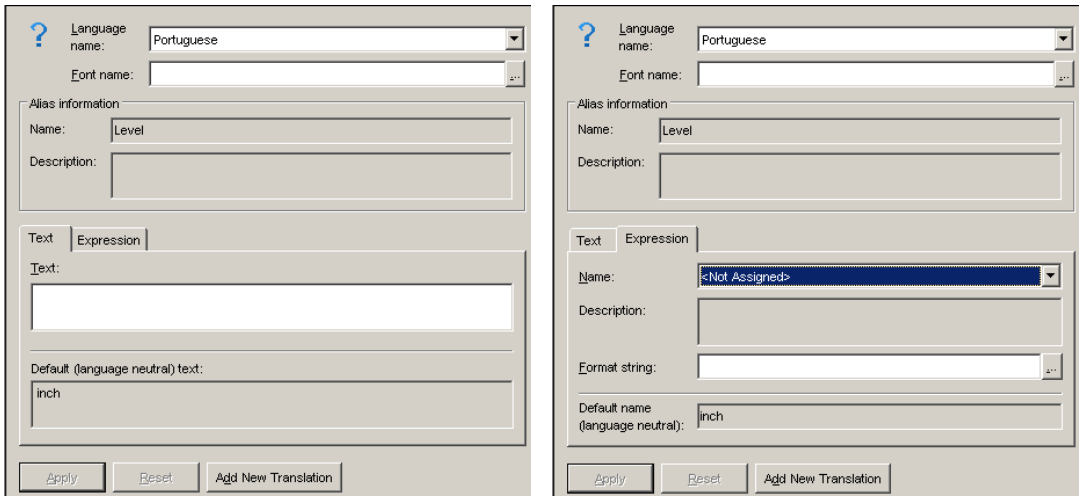


Figure 14.6 - Language Translation Configuration: Text (left) and Expression (right)

14.4. Using Language Aliases

Once a language alias is created you can use it in any GENESIS32 application. To use a translation alias, browse for it using the Unified Data Browser's **Language Aliases** tab, shown in Figure 14.7.

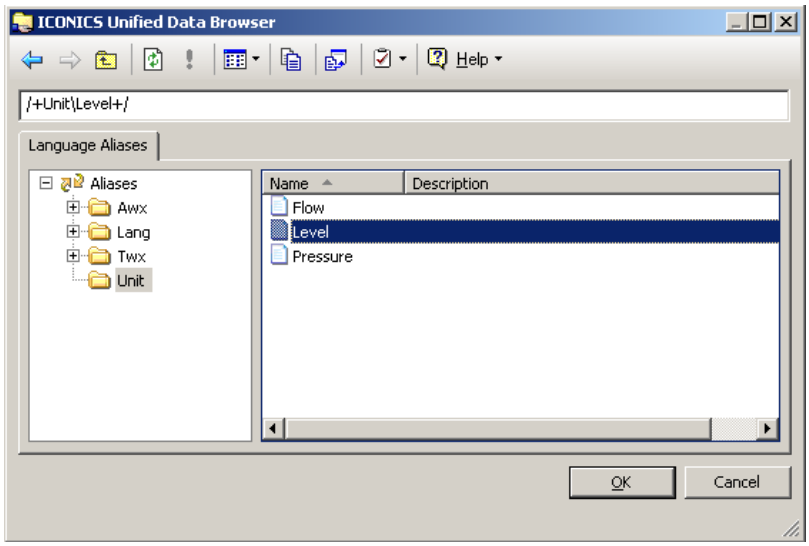


Figure 14.7 - Language Alias Selection in Unified Data Browser

In the field at the top of the tab you can easily recognize the language alias by its special syntax. The language alias name is delimited by the characters /+ at the beginning and +/ at the end. For example, the alias name **Title** when used in your application will look like this: **/+Title+/**. The alias in Figure 14.7 is **Level** and it is in a folder call **Units**. The backslash (\) in the alias indicates this structure.

NOTE: If you are planning a new project consider the advantages of integrating translation capabilities as part of the initial project development. Integrating translation capabilities after a project is done will require a considerable amount of rework.

14.4.1. Using Expressions

While you can browse for language alias in most text fields in any GENESIS32 applications you can also browse for it in objects where you provide a data source such as a process point.

Most objects in GraphWorX32 contain a checkbox for language alias near the bottom of the Property Inspector. You can enable this option and click on the ellipsis (…) button to select a language alias that applies to the data source. This allows the configured conversion in the language alias to apply to the associated data source.

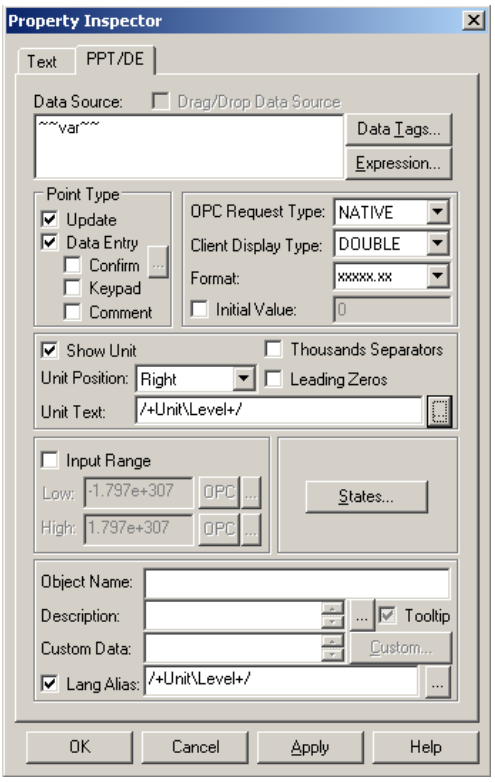


Figure 14.8 - Process Point with Language Alias Configured

14.5. Language Alias Switching

The easiest way to see a language alias at work in runtime is to use a button with the **Select Language** pick action in a GraphWorX32 display. Figure 14.9 shows the Property Inspector of a Select Language pick action.

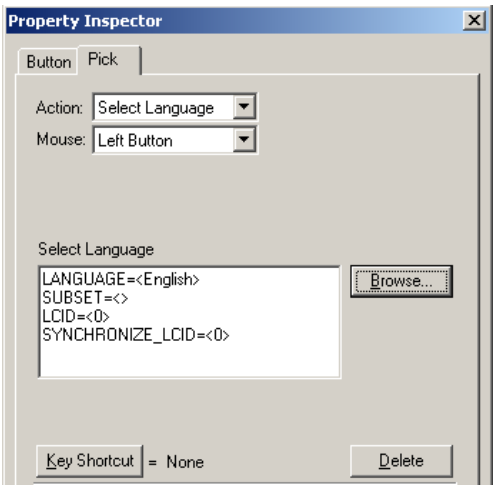


Figure 14.9 - Object with the Pick Action Set Language

Clicking on the "Browse" button brings up the **Language Selector** dialog. You can choose a language in the dialog. It only shows you the languages you have configured in the Language Configurator. Once you select a language in the upper half of the dialog it, will show you the language subset on the lower half it there is any. Once you have selected the desired language, you can click on "OK" to exit all of the dialogs and return to the GraphWorX32 display.

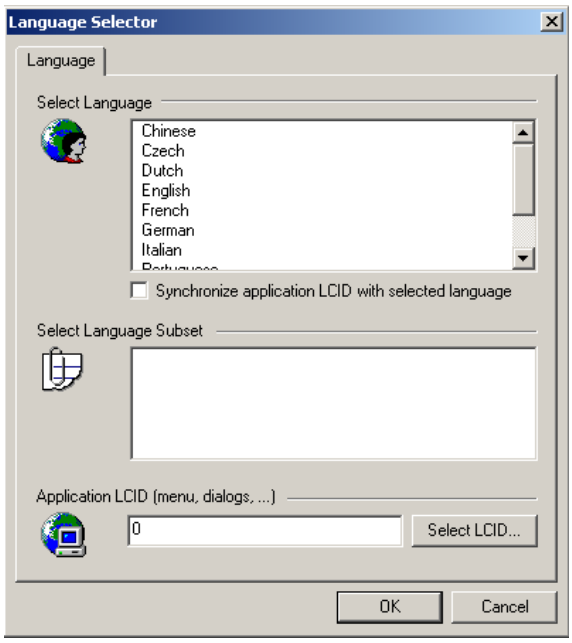


Figure 14.10 - Language Selector Dialog

NOTE: For demonstration and testing purposes, choose a language other than the language that will be active on the machine where you are testing.

When you click on the button in runtime the pick action for the button changes the GENESIS32 localization (current language/culture) setting. The display shows the translated value stored in the language alias database for that alias in that particular language.

Figure 14.11 shows how language aliasing works in runtime of a GraphWorX32 display. With a click of a button, you can set the text string to display in different languages as well as convert the values to the units that the local user understands.

Figure 14.11 - Language Alias in Runtime: English (left), Chinese (middle), Czech (Right)

Remember that language aliases are applied to the client based on the active SQL database. Because only one SQL database is active at any one time, and that database can use only one language setting for displaying data on the client, only one language is used in any given display. This means that when you use a pick action to change the language/locale in a display all of the language aliases defined in that display change to the currently selected language.

Lab: Multilingual Displays L14

Many companies have plants around the world and operators in other countries may not be comfortable operating in English. By using Language Aliasing, we can easily change the language of an application, saving time to recreate completely new applications whenever a plant opens in a new country. Acmeseutical would like to enhance their currently existing displays with Language aliasing, in preparation for their worldwide system deployment.

Estimated time to complete this lab: 30 minutes

14.1. Task Checklist

In this lab, you will complete the following tasks:

- Create a new Language Aliasing configuration database
- Insert languages
- Add language aliases and translations
- Add translation expressions

14.2. Support Files

You will need the following file(s) to complete this lab:

- PainReliefWorldMap.gdf
- ACMEPIllMachine.gdf
- ACMEMixers.gdf
- LanguageAliasingTranslation.txt

14.3. Language Aliasing Configurator

In order to use Language aliasing, you will need to define the language and associated translations in a system-wide database. We will configure that in this section of the lab.

1. Start the Language Configurator by going to Start → ICONICS Tools → Languages → Language Configurator.

2. Click on the "New" button in the toolbar to create a new configuration database.

3. In the Welcome screen of the Database Wizard, click on “Next” to continue.

4. Leave the default setting to “Create [a] new configuration in MS Access database” and click on “Next.”

5. In the Create MS Access Database page, click on the ellipse button and browse C:\Training and change the file name to ACMELanguageCnfg.mdb. Click on “Save” to close the dialog.

6. The Database wizard should look similar to Figure L14.1.

Figure L14.1 - Creating MS Access Database

7. Click on “Finish” to complete the database creation process.

8. You should now have a blank language alias configuration.

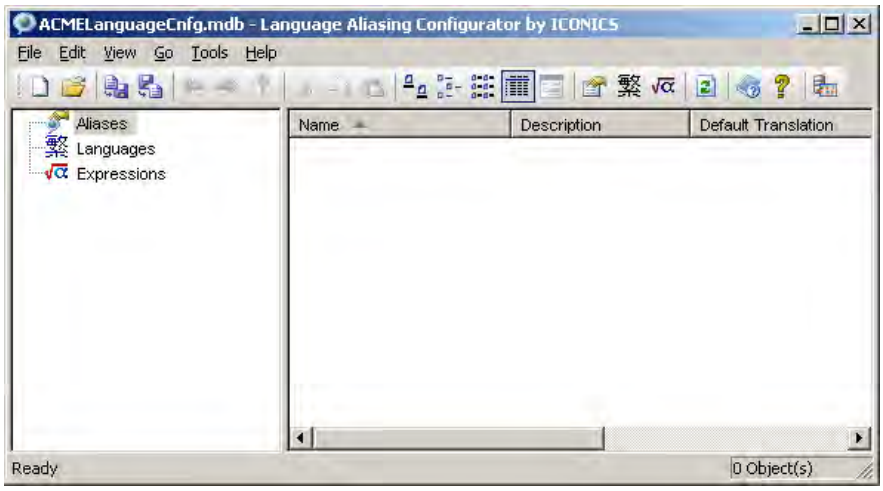


Figure L14.2 - Empty Language Aliasing Configuration

14.3.1. Adding Languages

1. Right-click on Languages in the tree control on the left-hand pane and select New → Language.

2. In the Language configuration form that shows in the right-hand pane, click on the "Browse" button.

3. Select "English – Unitied States" in the **Select Language** dialog and click on "OK."

4. Notice that the name of the language is automatically filled in once you select a language.

5. Click on "Apply" to save the changes.

6. Repeat the above steps and add the following 'new' languages:

   - Chinese – People's Republic of China
   - Czech – Czech Republic
   - German – Germany
   - Italian – Italy

7. Your configuration should look similar to Figure L14.3 once you have added the above languages.

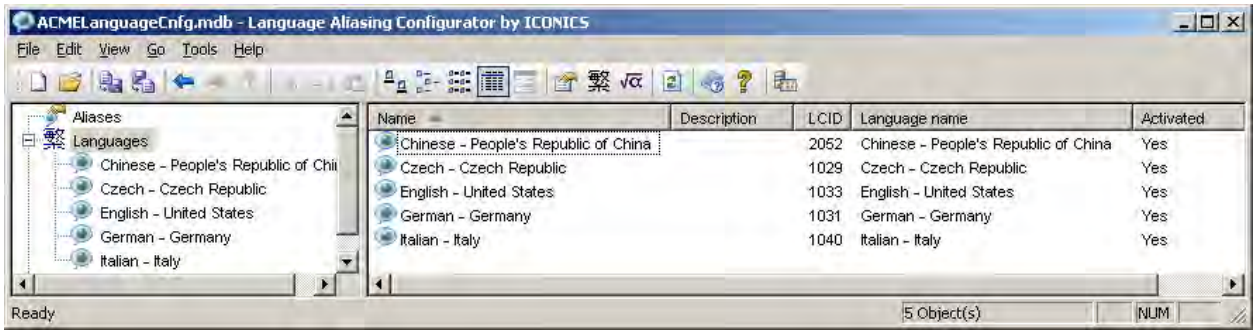


Figure L14.3 - Language Configurator with Languages Added

8. Click on the "English – United States" language in the tree control on the left and check the "Default Language" option in the configuration form on the right-hand pane.

9. Click on "Apply" to save the changes. Notice that once you click on Apply, the icon for this language changes to one with a checkmark to indicate that it is the default language.

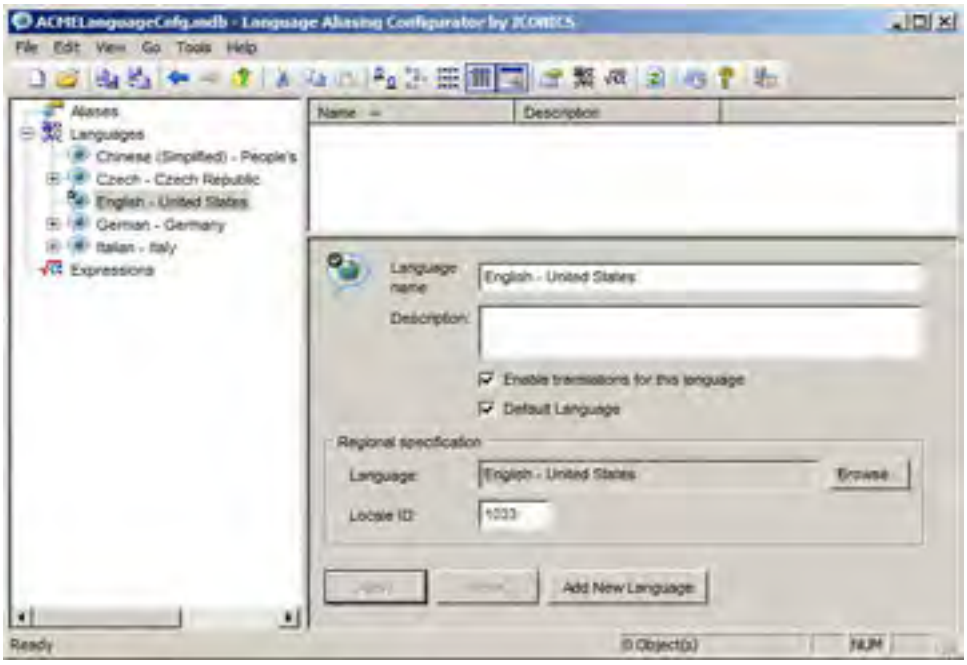


Figure L14.4 - Making English the Default Language

14.3.2. Adding Expressions

Since the different cultures we have added in the previous step use different measurement units, we will add expressions in this section to convert the unit measurements into one that the culture uses.

1. Right-click on Expressions in the tree control on the left-hand side and select New → Expression.
2. On the right-hand side, name the expression "kg to kg."
3. In the Read Expression section, click on the "Exp" button to bring up the **Expression Editor**.
4. Click on Tags → Input Variable as shown in Figure L14.5.

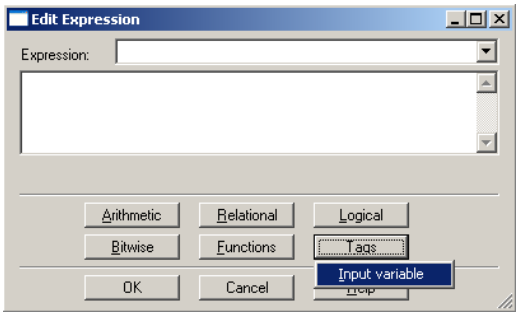


Figure L14.5 - Adding Input Variable to Expression

5. Click on “OK” to close the **Expression Editor**.

6. Repeat steps 4-5 in the Write Expression section. Click on “Apply” to save the changes.

7. Your finished “kg to kg” expression should look similar to Figure L14.6. Since the units coming from the OPC servers are in kilograms we do not need to do any translation and our configuration of this expression does not allow for conversion.

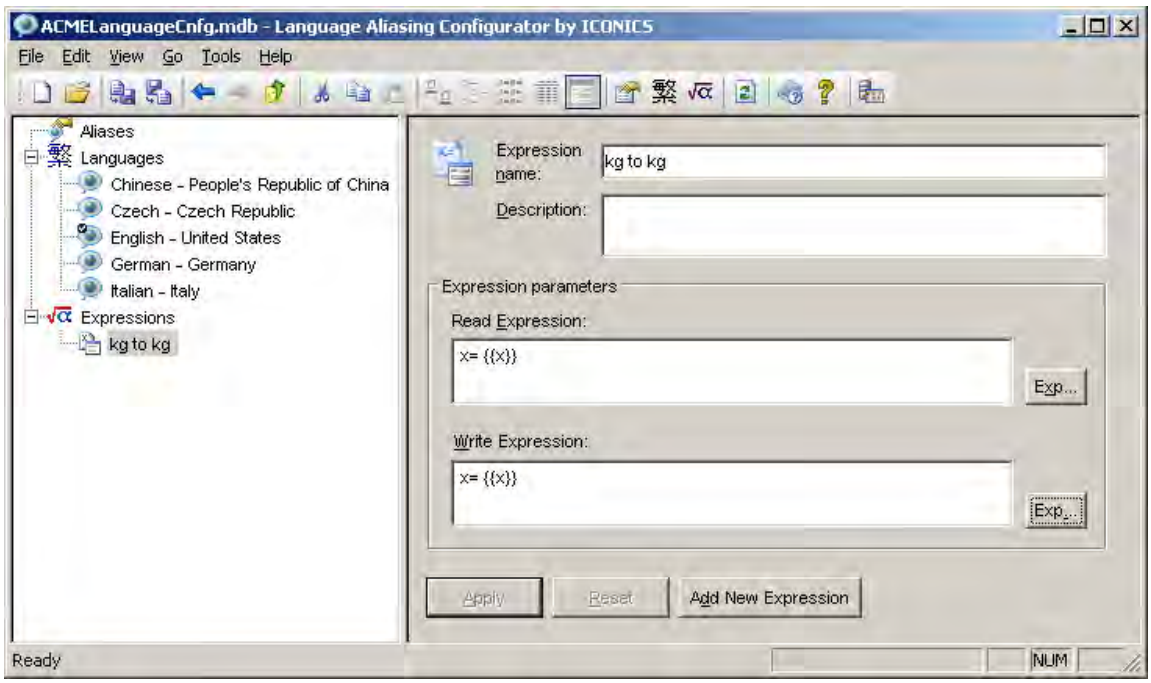


Figure L14.6 - kg to kg Expression

8. Right-click on Expressions and select New → Expression to create another expression.

9. Name this configuration “kg to lb.”

10. In the Read Expression, click on the “Exp” button to bring up the **Expression Editor**.

11. In the Read Expression, we need to provide the conversion from kilograms to pounds. The value in kilograms is the input value we inserted before. Select Tags → Input variable.

12. Click on Arithmetic → * Multiplication and type in 2.205. Your expression should look like x= {{x}} * 2.205.

13. Click on “OK” to close the Expression editor.

14. Copy the expression in the Read Expression section and paste it into the Write Expression section.

15. In the Write Expression, we need to perform the reverse calculation to write the value back into the data source in kilograms. Simply change the multiplication sign to a division sign.

16. Click on "Apply" to save the changes. Your "kg to lb" expression should look similar to Figure L14.7.

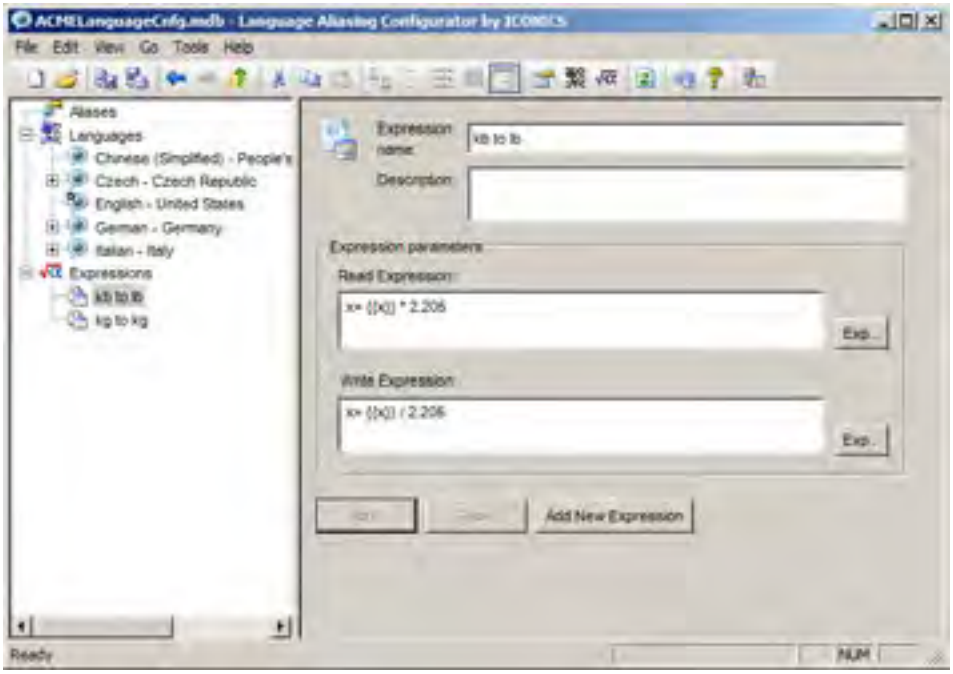


Figure L14.7 - kg to lb Expression

14.3.3. Adding Aliases

1. Right-click on Aliases in the tree control on the left-hand side and select New → Alias.

2. Name this Alias "Welcome" and click on "Apply."

3. Notice that the alias in the tree control contains an exclamation mark. This indicates that the alias is incomplete.

4. Expand the Welcome alias and notice that all of the languages we have added earlier are listed there with a question mark as the icon. This is because no text translation or expression has been configured for any of the languages.

5. Click on "English – United States" in the tree control.

6. In the Text tab, type in "Welcome" and click on Apply. Notice that the icon changes from a question mark to one that says "abc" . This indicates that this is a text translation. As you will see later there are different icons for expression translations and both text and expression translations.

7. Go through each of the languages listed under the "Welcome" alias and provide the translation according to Table L14.1.

Table L14.1 - Welcome Text Translation

Language	Text Translation
Chinese – People's Republic of China	□迎
Czech – Czech Republic	Vítejte
German – Germany	Willkommen
Italian – Italy	Benvenuto

NOTE: You can copy and paste the text in LanguageAliasingTranslation.txt located in C:\Training for the characters that you cannot type.

8. Create another Alias by right-clicking on Aliases in the tree control and select New → Alias.

9. Name this alias "Weight" and click on "Apply."

10. Edit each language under the alias according to Table L14.2.

Table L14.2 - weight Text Translation and Expression Conversion

Language	Text Translation	Expression	Format String
Chinese	千克 or kg	kg to kg	TYPE=<FLOAT> FORMAT=<xxx.xx>
Czech	kg	kg to kg	TYPE=<FLOAT> FORMAT=<xxx,xx>
English	lb	kg to lb	TYPE=<FLOAT> FORMAT=<xxx.xx>
German	kg	kg to kg	TYPE=<FLOAT> FORMAT=<xxx.xx>
Italian	kg	kg to kg	TYPE=<FLOAT> FORMAT=<xxx,xx>

NOTE: Expression and Format String are on the Expression tab. You can also type in the comma in place of the period in the format string to give you the correct symbol for a decimal point.

11. Your finished configuration should look similar to Figure L14.8.

12. Make sure your current configuration is the active one by going to File → Make Active. If the Make Active option is grayed out, then your current configuration is the active one. You can now close the Language Alias Configurator.

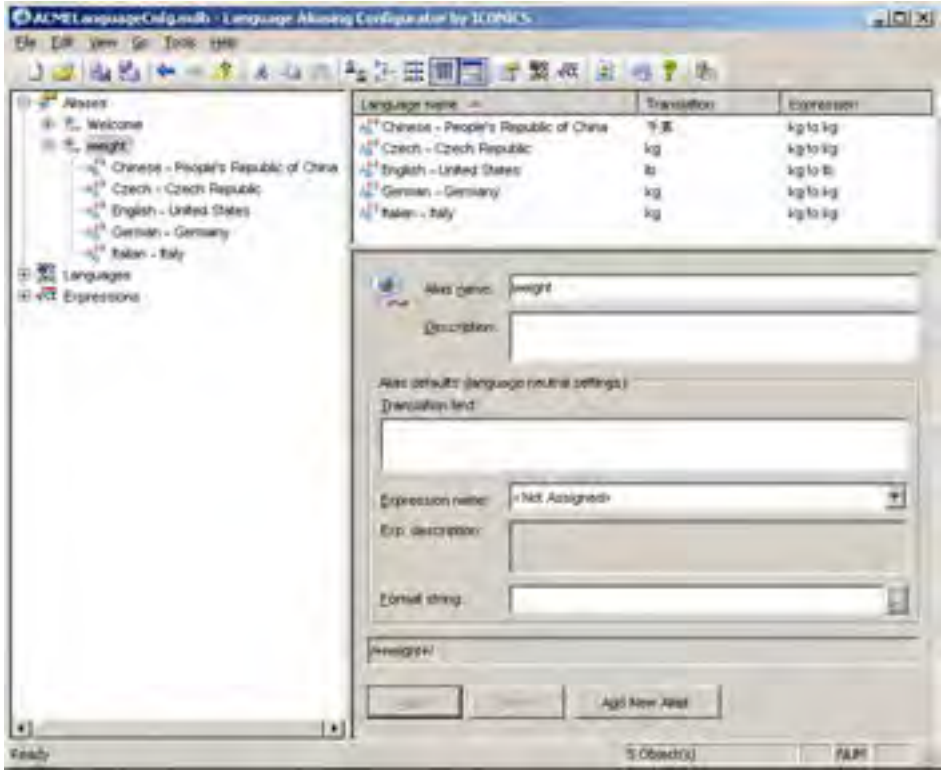


Figure L14.8 - Weight Alias Configuration Summary

14.4. Language Aliases in GraphWorX32

Now that we have configured language aliases, we will need to add them into our GraphWorX32 displays in order to take advantage of them.

1. Open the PainReliefWorldMap display and add a label to it below the map.
2. Type in /+Welcome+/ in the text label. Resize the text of the label.
3. Add a button to the display below the text label.
4. Change the Action dropdown list to “Select Language.”
5. Click on the “Browse” button to bring up the **Language Selector**.
6. Choose “Chinese – People’s Republic of China” and click on “OK” to close the dialog.
7. Your Pick action Property Inspector should look similar to Figure L14.9.

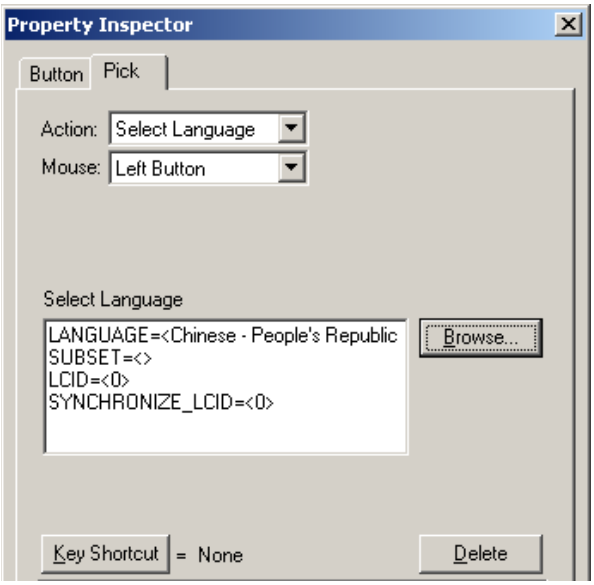


Figure L14.9 - Select Language Pick Action

8. Click on the Button tab and change the button text to say “Chinese.” Click on “OK” to close the Property Inspector.

9. Repeat steps 3-8 for the languages Czech, English, German and Italian.

10. Resize and align the buttons so that the display looks similar to Figure L14.10. Remember that you can use actions such as “Make Same Size,” “Align Tops,” and “Distribute Evenly Across” in the Arrange toolbar.

Figure L14.10 - PainReliefWorldMap.gdf with Language Aliasing

11. Go into runtime and click on the different buttons to see that the phrase “Welcome” translates to the different languages. Make sure you change the language back to “English” as the last language. Since language settings are applied system-wide, it will remember the last language you worked with; in this case English. Exit runtime.
12. Save the PainReliefWorldMap.gdf display and open the ACMEMixers.gdf display.
13. Double-click on the process point below the tank to open the Property Inspector.
14. Select the PPT/DE tab.
15. Check the “Show Unit” option and click on the ellipse button next to the “Unit Text” field.
16. Choose the Language Aliases option to open the UDB.
17. Select the “weight” alias and click on “OK” to return to the Property Inspector.
18. Check the “Lang Alias” option at the bottom.
19. Click on the ellipse button to bring up the UDB and choose the “weight” alias again.
20. Click on “OK” to close the UDB.
21. Click on “OK” again to close the Property Inspector.
22. Save and close the display.
23. Open the ACMEPillMachine.gdf display and go into runtime.
24. Click on a Mixer to bring up the ACMEMixers display.
25. Notice that the unit is “lb” in the process point below the mixer.
26. Close the ACMEMixers Display.
27. On the ACMEPillMachine display, go to View → Select Language.
28. In the **Language Selector** dialog, choose Italian and click on “OK.”
29. Notice that the date at the top of the display changes to the corresponding language.
30. Bring up the ACMEMixers Display again. Notice that the unit for the process point under the mixer now reads “kg” and the values correspond to the unit.
31. Try changing to other languages and see what happens.

Review: Language Aliasing R14

14.1. Chapter Objectives

Globalization is the key to a successful business in today's world and as a result, many companies have expanded, including production facilities around the world. While the facilities may be similar, the operators in these facilities speak different languages and maintaining HMI displays for different languages could be a tedious task. ICONICS Language Aliasing System can expedite the HMI creation with different languages, keeping the system easy to maintain yet allowing operators to control the facility in their native language. By now, you should know how to do the following:

- Configure a language
- Configure a Language Alias
- Associate Expressions to a Language Alias
- Switch language in an application

14.2. Summary

The Language Aliasing System provides a simple way to use translated text strings from different languages. Using the Language Configurator, you can add as many languages as necessary and create an alias that contains the translation of text strings to that language.

You can also create expressions to convert between the different measuring units that different countries use as a standard. For example, operators in the US may be used to the imperial measuring system, while operators in Mexico are more familiar with the metric measuring system. Using Expressions within Language Aliasing, the operator can view values in units that they are familiar with as they switch the language.

14.3. Terms

Language Alias Language Aliasing System Language Configurator

14.4. Test Your Knowledge

In this chapter, we discussed the Language Aliasing System in details. By now, you should be able to answer the following questions.

1. What should you do when configuring a language in the Language Configurator?

    a. You do not need to do anything; languages are predefined.
    b. Specify a name for the language, click on the “Browse” button in the Regional specification section, and select the appropriate language.
    c. Simply click on the “Browse” button in the Regional specification section and select the appropriate language.
    d. Find out the Locale ID for the appropriate language for the operating system and enter it in the Regional Specification section.

2. What is the syntax to use the original value in the read or write expression? (i.e. If the process point where you are applying a language alias has a value of 10, how do you get that value in the expression?)

    a. {{x}}
    b. {{X}}
    c. {{value}}
    d. {{<value>}}

3. True or False: The icons in the tree control of the Language Configurator indicate if the status of the alias and the languages associate with it.

4. True or False: You can define a default language to the system.

5. How many expressions should you have if your OPC tag provides data in degrees Celsius and your operators are from the U.S. and Spain?

    a. 1
    b. 2
    c. 4
    d. 6

6. Which of the following is the correct syntax for a language alias?

    a. <<LanguageAlias>>
    b. <#LanguageAlias#>
    c. <+LanguageAlias+>
    d. /+LanguageAlias+/

Unified Data Manager 15

We have introduced the Unified Data Manager (UDM) in version 9 as a tool to manage expressions, data connectivity, and schedules in a centralized location. Everything configured in the UDM is globally available across any ICONICS products via the Unified Data Browser. This will aid in the process of developing your project by reducing development time as well as ease configuration, logistics, and deployment.

You can configure the following in the UDM:

- Expressions
- Groups
- Value Sets
- Recipes
- Registers
- Event Triggers
- Alarm Subscriptions
- Alarm Filters

This class will cover all of these except Event Triggers and Registers, which will be covered in the Expanded GENESIS32 class or in a standard BizViz class. If you would like more information on these parameters, please refer to the Help file for the Unified Data Manager Manual and/or attend the mentioned classes.

By the end of this chapter you will be familiar with the following topics:

- How the Unified Data Manager can ease development
- How the Unified Data Manager works
- Configuring a Read Expression with parameters
- Configuring a Write Expression that takes user inputs
- How Global Expressions work with other ICONICS Applications
- Configuring a Group in the Unified Data Manager
- Using Group tag to change multiple tags to a single value
- Configuring a Value Set in the Unified Data Manager
- Using Value Set tags to change multiple tags to different values
- How recipes are setup in the Unified Data Manager
- The difference between Groups, Value Sets, and Recipes
- How to configure a Recipe in the Unified Data Manager configuring alarm subscriptions and filters in the Unified Data Manager
- Using global subscription and filters

15.1. The Unified Data Manager Configurator

You can setup Global Expressions, Global Alarm Subscriptions, Global Alarm Filters, Groups, Value Sets, Recipes, and Event Triggers in the UDM as shown in Figure 15.1. In keeping with the open standards employed throughout the GENESIS32 architecture, the UDM Configuration is saved to a Microsoft Access or Microsoft SQL Server database.

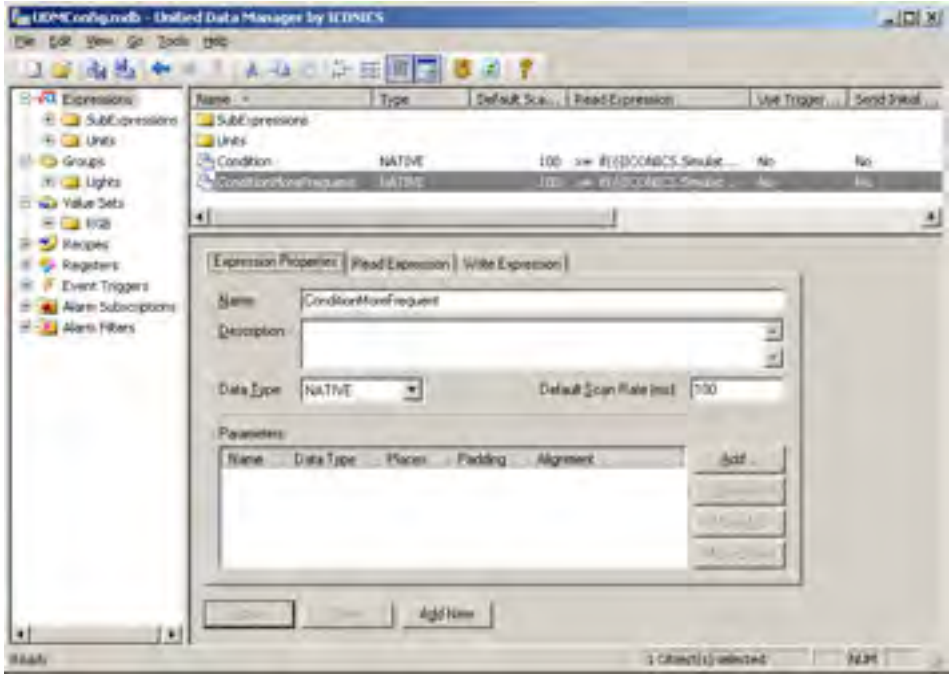


Figure 15.1 - Unified Data Manager Configurator

Once the different tags are configured, the UDM takes care of getting data from OPC servers, reading/writing to OPC tags, as well as monitoring events to execute triggers using the Gen-Broker plug-in technology.

GenBroker Plug-ins are light-weight modules that provide real time data, alarms and events. They function as a subset of an OPC DA Server. Unlike OPC DA servers, the GenBroker plug-ins are bound to the Unified Data Browser on the client side. The GenClient name syntax supports them which allows ICONICS client applications to access them seamlessly.

When you try to use tags created in the UDM, they will appear in the general form of

TagType:TagName

We will go into more detail about the syntax for each type of tags in the corresponding sections that follows.

15.2. Introduction to Global Expression

A global expression allows you to modify the data from the PLC by creating expressions that you can use throughout all ICONICS applications. Each global expression is implemented as a single expression data tag. The application calculates the result of the tag based on input values or on trigger data tags. Client applications can access these expression data tags as read/write data tags with a configured data type.

The Expressions tree control in the Unified Data Manager enables you to set up global expressions. Global expressions are organized in a tree structure of folders and expression items. Features of global expressions include:

- Create and edit expressions
- Create folders for expressions
- Expressions are available as data tags
- Parameters are supported
- Specify read expressions or write expressions
- Expressions within expressions are calculated together

NOTE: For more information about expressions in general, please refer to the Expression Editor Help documentation.

Once you have configured global expressions, they can be accessed by many applications via the appropriate tab in the Unified Data Browser (UDB).

15.2.1. Configuring Expression

You can configure Expression Folders and Expression Items by right-clicking on Expressions in the tree-control and the select the desire action.

Once you have created an expression item there are three tabs:

- **Expression Properties**: configures properties of the expression item
- **Read Expression**: configures how the expression item reads and shows value
- **Write Expression**: manipulates the value to write to a tag using the expression.

15.2.1.1. Expression Properties

You can configure the following in the Expressions Properties tab:

- **Name**: The name of the expression item
- **Description**: The description for the expression item

- **Data Type**: The data type of the expression; it's of any OPC Data type, the default is NATIVE
- **Data Scan Rate (ms)**: The rate that it will scan input OPC tags (if defined) in milliseconds. This rate can be overwritten by using extended point syntax.
- **Parameters**: Variables or placeholders that you would like to use within the expression

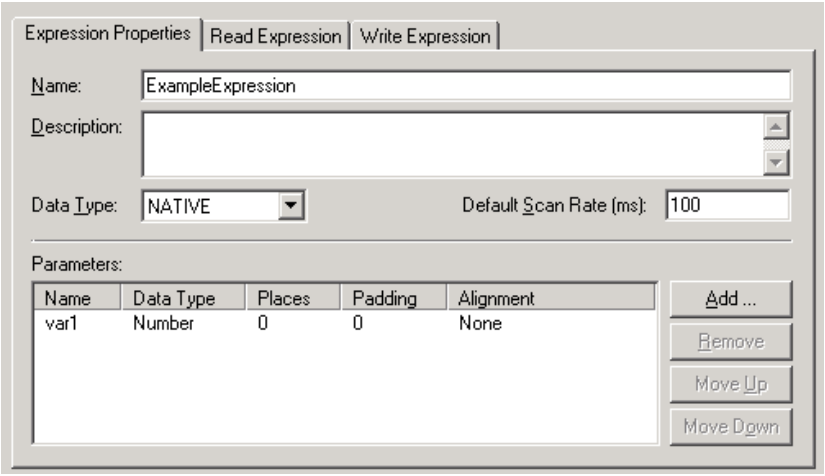


Figure 15.2 - Expression Item: Expression Properties

Parameters can be either integers or strings. You can optionally configure the Data Type, Places, Padding, and Alignment for each parameter. The Places column allows you to configure the minimum number of characters you want to show, while the Padding allows you to choose which character to use for the leading places.

> **NOTE:** If you use "0" as your Padding Character and the result is numeric, you will not see the leading zeros.

The Alignment setting is only available when the Data Type is String. You can configure the string to align Left (default) or Right. Table 15.1 shows some configuration and the expected result of the parameter string in runtime.

Table 15.1 - Parameter Configuration and Results

Original Value	Places	Padding	Alignment	Results
123	0	1	None	123
123	5	8	None	88123
123	5	0	None	123
Test	6	@	Left	@@Test
Test	6	$	Right	Test$$
Test	6	0	Left	00Test

15.2.1.2. Read Expression

The Read Expression tab allows you to configure your expression. You can write your expression so that when you look at the tag in a GraphWorX32 display, it will show you the desired value (e.g. displaying values in a different measuring unit).

You can click on the Expression Editor Button to bring up the Expression Editor Dialog. This dialog is the same Expression Editor used through the GENESIS32 products. The editor is designed to help you create your expressions. It contains buttons where you can select functions like arithmetic, string compare, logical, relational, conditional, bitwise, and etc. If you simply want to insert an OPC point, you can click on the Insert Tag… button, which will bring up the Unified Data Browser for you to select an available OPC point.

It is important to know that when writing an expression in the Read Expression section, although the tag will display a different value (the result of the expression); the process point value has not changed. It works much the same was as writing an expression inside a GraphWorX32 process point. You will see the result of the expression, but the input value is not changed.

You can also use a trigger to activate the read expression. Once you check the use Trigger Tags, the trigger section will become available. These triggers are different from the ones listed in the UDM tree-control.

In the Trigger Tags section, you can add your trigger. In the Method column, there are three options that you can choose from:

- **On Change:** Activates the read expression whenever the associated process point changes its value.
- **On True:** Triggers read expression when the associated process point changes to true (1).
- **On False:** Initiates the read expression when the associated process point changes to false (0).

You can change the data source of the trigger tag by clicking on it and selecting the Edit button to the right. You can also add and remove the tags by clicking on the respective buttons.

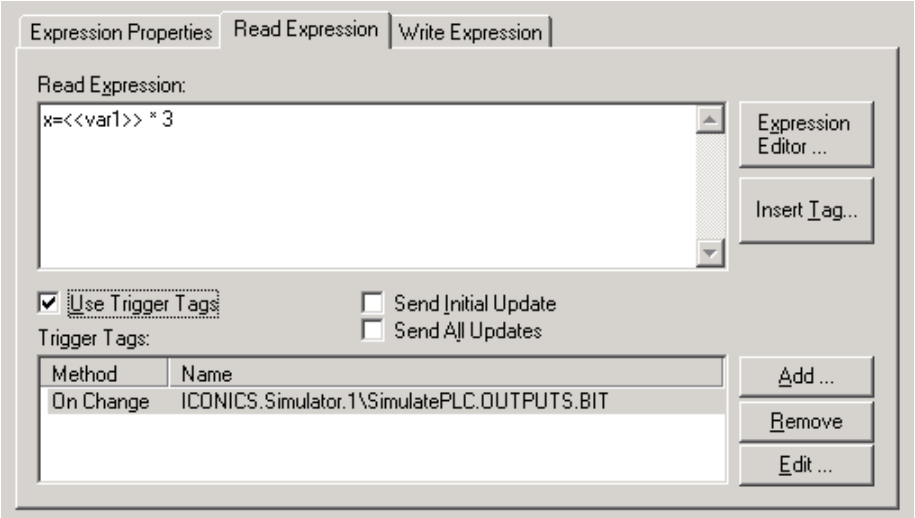


Figure 15.3 - Expression Item: Read Expression

Two checkboxes become available when you select to use a trigger tag: Send Initial Update and Send All Updates.

When you check the **Send Initial Update** checkbox, the value of the tag will be sent to the client whenever there is a client that tries to connect to the tag regardless if the trigger tag conditions are met. If this box is not checked you may not see a value until one of the next trigger condition becomes true.

The **Send All Updates** checkbox, when checked, will update the tag whenever there is a satisfied trigger condition regardless if the value has changed since the last update. Since OPC is changed based, values are not sent if the new value is the same as the previous value; checking this option will force an update despite the fact that the value is still the same.

15.2.1.3. Write Expression

When you only enable the Read Expression the tag is read-only and you are prohibited from writing to the expression tag. You can enable the write wxpression by checking the **Enable Write Expression** checkbox as show in Figure 15.4.

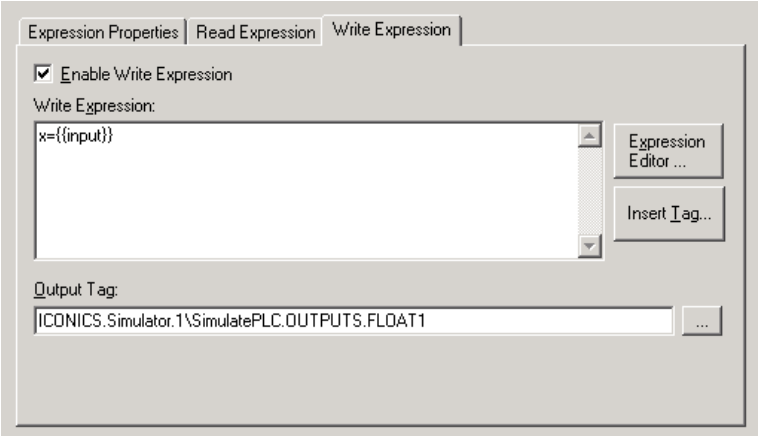


Figure 15.4 - Expression Item: Write Expression

Once you have enabled this option, the rest of the tab becomes active. You can either enter your expression in the white space or you can click on the **Expression Editor**… button, you will get the familiar dialog where you can use the preset functions. The **Insert Tag**… button will bring up the Unified Data Browser for you to input a tag.

Specific to the Expressions Editor that you bring up in the Write Expression, you will be able to click on Tags → Input Value and it will insert {{Input}} into the expression. {{Input}} is a special tag for the Write Expression which is a placeholder in the expression. When a value is written to the expression tag, {{Input}} takes on this value for the calculation.

On the bottom of the tab is an **Output Tag**. If you click on the ellipsis button to the right of this field, you will get the unified data browser. The result of the Write Expression will be passed to the Output tag.

15.2.2. Using Global Expressions

Once you have configured your Global Expression, you can use it with any ICONICS OPC client (i.e. GraphWorX32, DataSpy, etc). The data source of the tag always starts with “exp” to indicate that this is a tag from the Unified Data Manager and that it’s an expression. The exp is followed by a colon and the name of your Global Expression Item. If you are using parameters in parenthesis following the name of the Global Expression Item, this is where you would pass in the parameter values. Your expression may look similar to the one shown in Figure 15.5.

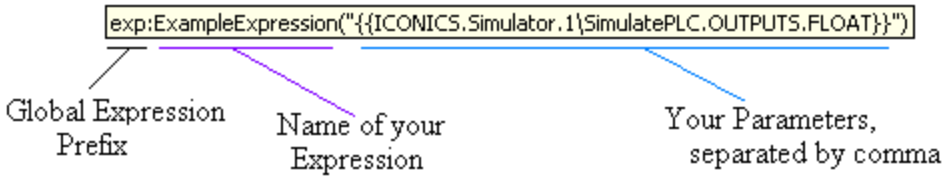


Figure 15.5 - Global Expression tag Syntax

In this case, the name of the Global Expression Item is “ExampleExpression” and it is giving the parameter in the expression the value “{{ICONICS.Simulator.1\SimulatedPLC.OUTPUTS.FLOAT}}.” As shown in the Read Expression in Figure 15.3, everything within the quotation marks inside the parenthesis will replace the <<var1>>. Because the OPC tag has the double curly brackets ({{ }}) around it, the server will treat it as an OPC tag instead of a simple string and will multiply the tag value by three as defined in the expression.

As the value of the parameter changes the expression it will reevaluate and update the changes in real time. If you write to the expression, you will notice that the tag you specified in the Output Tag in the Write Expression tag will change each time you actively write to the tag.

15.3. Introduction to Groups

Grouping allows you to write a single variant value to a collection of data tags thus allowing an operator to change many tags to one value at once instead of changing the value of tags individually. Since each group is treated like a data tag you can use it within other groups allowing for even simpler control.

You can configure a group tag in the Groups tree-control in the UDM. A Group data tag represents a collection of data tags and they are organized in a tree structure of folders and group items. Features of UDM groups include:

- Create and edit groups
- Create folders for groups
- Groups are available as read/write data tags
- Parameters are supported
- Groups within groups are allowed
- Group values are written synchronously to target OPC servers

Any write to a group data tag will distribute written values to all data tags included in the group; all included data tags will have the same value as the group data tag.

Once you have configured groups, they can be accessed from many applications via the appropriate tab in the Unified Data Browser (UDB).

> **NOTE:** For more information about Groups please refer to the Expression Editor Help documentation.

15.3.1. Configuring Groups

You can configure Group Folders and Group Items by right-clicking on Groups in the tree-control and the select the desire action.

Once you have created a group item, there are two tabs:

- **Group Properties**: configure properties of the group item
- **Grouped Tags**: allows you to add, remove, and edit tags that belong to the group

15.3.1.1. Group Properties

You can configure the following in the Group Properties tab:

- **Name**: the name of the group item
- **Description**: the description for the group item
- **Data Type**: the data type of the group item; of any OPC Data type, the default is Float
- **Parameters**: variables or placeholders that you would like to use within the Group

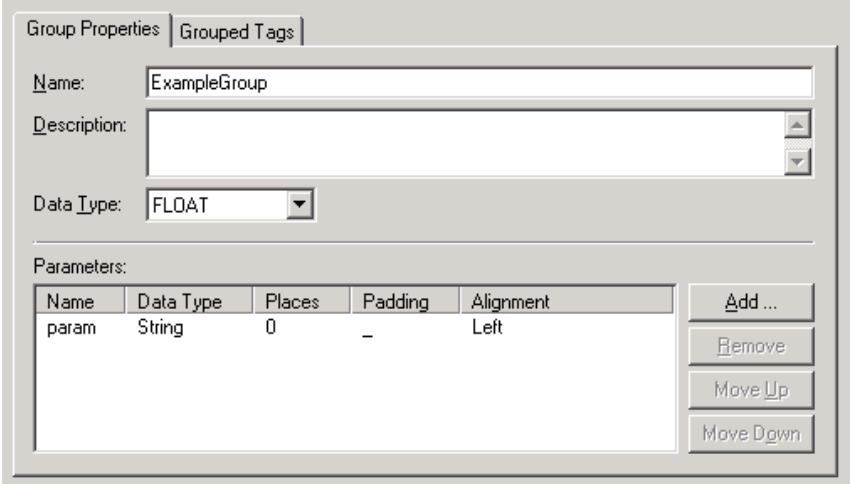


Figure 15.6 - Group Item: Group Properties

The Parameters field in a group item is similar to that in an expression item. It can be either integers or strings. You can optionally configure the Data Type, Places, Padding, and Alignment for each parameter. The Places column allows you to configure the minimum number of characters you want to show while the Padding allows you to choose which character to use for the leading places. If you use "0" as your Padding Character and the result is numeric, you will not see the leading zeros. The Alignment setting is only available when the Data Type is String. You can configure the string to align Left (default) or Right. For examples of your parameter configuration and predicted results, please look at Table 15.1 in the "Expression Properties" section.

15.3.1.2. Grouped Tags

The **Grouped Tags** tab allows you to configure tags that you would like to group. From here you can insert the OPC tags and edit them (i.e. replace certain words with parameters) so that when you change the value of the group item. All tags listed in the group will take on this value.

You can click on the "Add" Button to bring up the Unified Data Browser to select the tags you need. Once you have the tag in the Grouped Tags section, you can select the tag, click on "Edit" to change the value in the Tag Name Column. Clicking on the "Edit" button will bring up the Unified Data Browser where you can select another tag. Notice that in the Unified Data Browser the tag path is already in the textbox near the top. If you would like to add in a parameter, you can simply replace certain text with the parameter in the textbox and click on "OK."

There are three more fields in the Grouped Tags section that you can edit once you have inserted a tag into the group:

- **Array:** Takes a Boolean value to indicate if the tag is an array; the default value is false.
- **Lo Bound:** This field only applies if the Array column is true. This indicates the lower boundary of the array (e.g. if the first array index is 0, then you would input 0).
- **Length:** This field is also only valid if the array column is true; it takes on an integer value of the total number of items in the array.

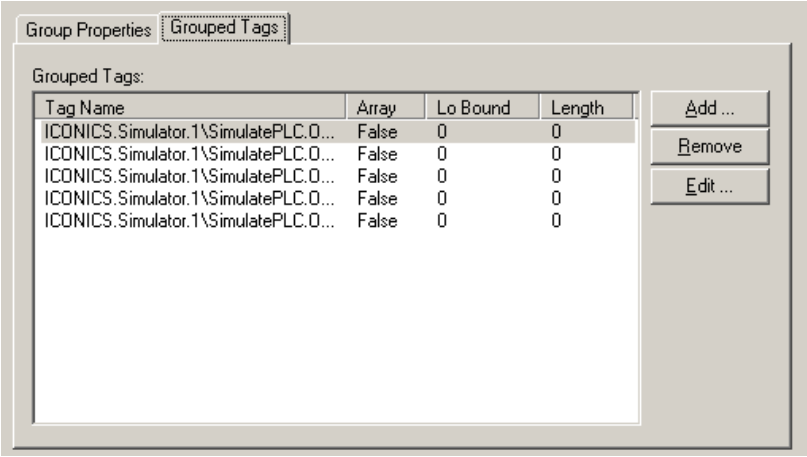


Figure 15.7 - Group Item: Grouped Tags

15.3.2. Using Grouped Tags

Once you have configured your group tag, you can use it with any ICONICS OPC client (i.e. GraphWorX32, DataSpy, etc). The data source of the tag always starts with "grp" to indicate that this is tag from the Unified Data Manager and it's a group tag. The grp is followed by a colon and the name of your Group Tag Item. If you are using parameters in parenthesis following the name of the Global Expression Item, this is where you would pass in the parameter values. Your expression may look similar to the one show in Figure 15.5.

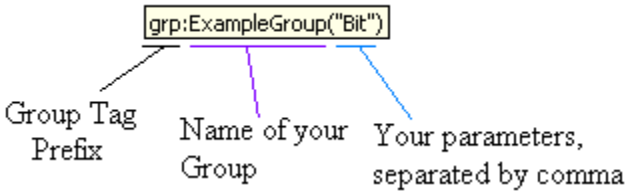


Figure 15.8 - Group tag Syntax

In this case, the name of the Group Item is "ExampleGroup" and it is giving the parameter in the tag the value "Bit." The tag in the configuration is "ICONICS.Simulator.1\SimulatePLC.OUTPUTS.<<param>>1" and everything within the quotation marks inside the parenthesis will replace <<param>>. In runtime, the tag will become "ICONICS.Simulator.1\SimulatePLC.OUTPUTS.Bit1" since the value of the parameter in Figure 15.8 is "Bit."

As the value of the parameter changes, it will change the tags that the group includes. If you write to the group tag, you will notice that the tags within the group change to the value you wrote to the group tag.

15.4. Introduction to Value Sets

A Value Set allows you to write different preset values to a collection of data tags, enabling operators to change many tags to their default values at once instead of changing the value of each tag individually. Since each Value Set is considered a data tag, you can use it within other Value Sets allowing for even simpler control.

You can configure a Value Set tag in the Value Sets tree-control in the UDM. A Value Set data tag represents a collection of data tags and they are organized in a tree structure of folders and value set items. Features of value sets include:

- Create and edit value sets
- Create folders for value sets
- Value sets are available as read/write data tags with string data type
- Parameters are supported
- Values are written synchronously to target OPC servers
- Import/export value set item configurations from/to XML files

A value set data tag represents a set of tags. Any write to a value set data tag will distribute written values to all data tags included in the value set; all included data tags will have the same value as the value set data tag. Once you have configured a Value Set, it can be accessed from many applications via the appropriate tab in the Unified Data Browser.

15.4.1. Configuring Value Sets

You can configure Value Set Folders and Value Set Items by right-clicking on Value Sets in the tree-control and the select the desire action.

Once you have created a Value Set item, there are four tabs:

- **ValueSet Properties** – configure properties of the Value Set item (which becomes an OPC tag when used within GENESIS32 Applications).
- **ValueSet Tags** – allow you to add, remove, and edit data tags that belong to the Value Set.
- **ValueSet Values** – allow you to define the value for the Value Set tag and the values for the associated data tag. When you write a predefined value to the ValueSet, all associated tags will take on the values defined here.
- **Import/Export** – enable you to export or import value set item configurations to or from XML files.

15.4.1.1. ValueSet Properties

You can configure the following in the ValueSet Properties tab:

- **Name** – the name of the Value Set item
- **Description** – the description for the Value Set item
- **Parameters** – variables or placeholders that you would like to use within the expression

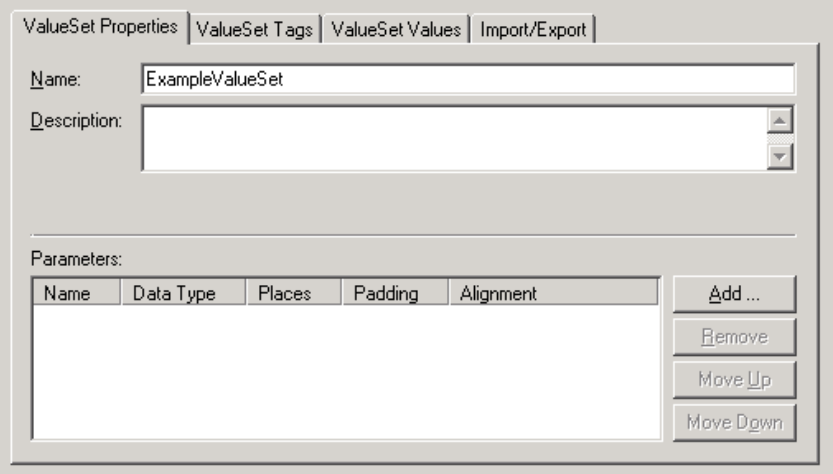


Figure 15.9 - Value Set Item: ValueSet Properties

The Parameters field in a ValueSet item is similar to that in an expression item. It can be either integers or strings. You can optionally configure the Data Type, Places, Padding, and Alignment for each parameter. The Places column allows you to configure the minimum number of character you want to show, while the Padding allows you to choose which character to use for the leading places. If you use "0" as your Padding Character and the result is numeric, you will not see the leading zeros. The Alignment setting is only available when the Data Type is String. You can configure the string to align Left (default) or Right. For examples of your parameter configuration and predicted results, please look at Table 15.1 in the "Expression Properties" section.

NOTE: If you use "0" as your Padding Character and the result is numeric, you will not see the leading zeros.

15.4.1.2. ValueSet Tags

The ValueSet Tags tab is similar to the Grouped Tags tab in a Group item. In this tab, you can configure the tags that you would like to be in the Value Set. In this location, you can insert the OPC tags and edit them (i.e. replace certain words with parameters) so that, when you change the value of the group item, all tags listed in the group will take on this value.

You can click on the "Add" Button to bring up the Unified Data Browser to select that tags you need. Once you have the tag in the ValueSet Tags section, you can select the tag and click on "Edit" to change the Tag Name Column. Clicking on the "Edit" button will bring up the Unified Data Browser where you can select another tag. Notice that in this Unified Data Browser, the tag path is already in the textbox near the top. If you would like to add in a parameter, you can simply replace certain text with the parameter in the textbox and click on "OK."

There are a few more fields in the ValueSet Tags section that you can edit once you have inserted a tag into the group:

- **Alias:** allows you to enter an alias for the tag. If you enter an alias you will see the alias being the table header in the ValueSet Values tab, otherwise, the short tag name is used in that table.
- **Array:** takes a Boolean value to indicate if the tag is an array; the default value is false.
- **Lo Bound:** this field only applies if the Array column is true. This indicates the lower boundary of the array (e.g. if the first array index is 0, then you would input 0).
- **Length:** this field is also only valid if the array column is true; it takes on an integer value of the total number of items in the array.

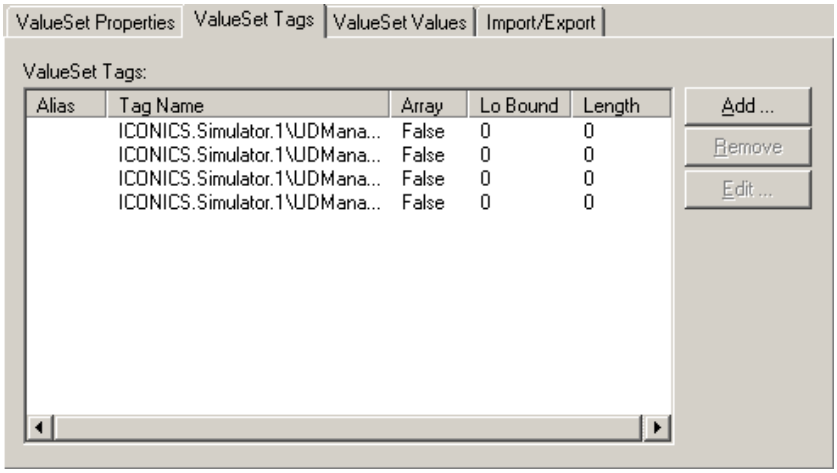


Figure 15.10 - Value Set Item: ValueSet Tags

15.4.1.3. ValueSet Values

Notice that all the tags you have added in the ValueSet Tags tab are now imported into the ValueSet Values table; each tag is listed in as a column with the tag name or specified alias as the column header. Each row in this table shows a single ValueSet Value. Once you have added a ValueSet Value, you can copy it, delete it, or edit by clicking on the respective buttons. The copy action will copy the values of the Data Tag already set for the Value Set.

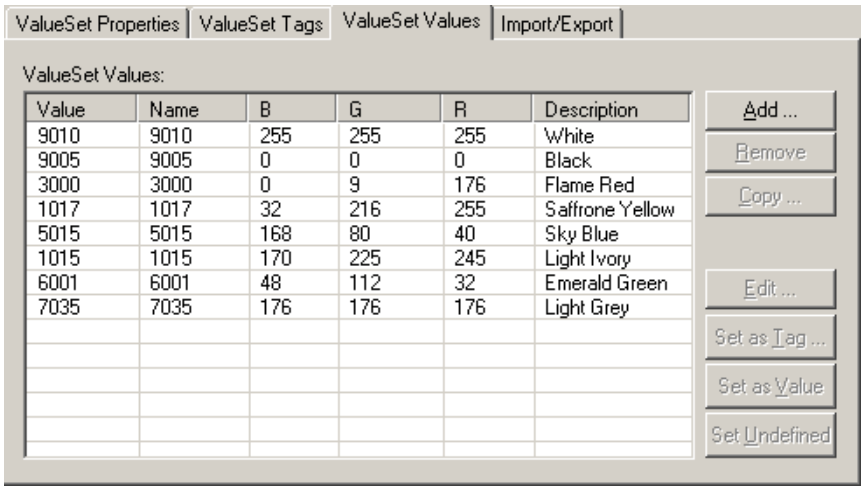


Figure 15.11 - Value Set item: ValueSet Values

After inserting a ValueSet Value, you may notice that the different tag columns show a value of "Undefined." The value of the different columns can be a tag, a manually entered value, or Undefined.

If you would like a particular cell in the table to read a value from another data tag, you can select the cell and click on the "Set as Tag…" button. This will bring up the Unified Data Browser for you to select any available data tags in the system or on the network. You can also click on the "Set as Value" button to bring up the **Enter Value** dialog as shown in Figure 15.12.

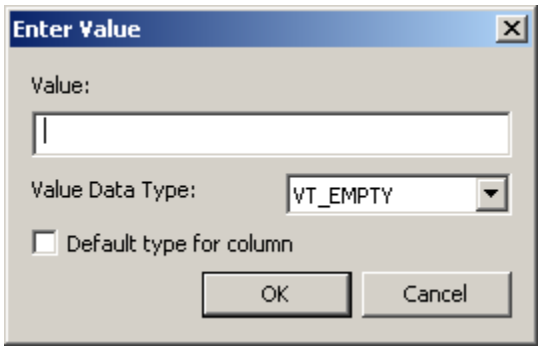


Figure 15.12 - Enter Value Dialog

You can enter any value for the data tag in the textbox with an appropriate data type. Notice that the Value Data Type is by defaultVT_EMPTY which means that there is no value. You should always change the Value Data Type after you have entered the value; otherwise the value entered will be lost. If you wish to set your currently selected data type as the default one, also check the “Default type for column” checkbox.

Once you have given a data tag a value (manual or from a data tag), you can edit it by clicking on the “Edit” button. Depending on the Mode of the tag, you will either get the Unified Data Browser or the Enter Value dialog. If you decide that you need to reset the tag, you can always click on the “Set Undefined” button. If the tag is “undefined,” nothing will be written to the data tag and the tag will keep its last known value.

15.4.1.4. Import/Export

In the Value Set Import/Export tab, you can import or export value set item configurations to or from XML files.

When you click the “Import from XML” button, the value set item configuration will be imported from the selected XML file. The existing value set item configuration will be replaced by the imported configuration.

When you click the Export to XML button, the value set item configuration will be exported to a specified XML file.

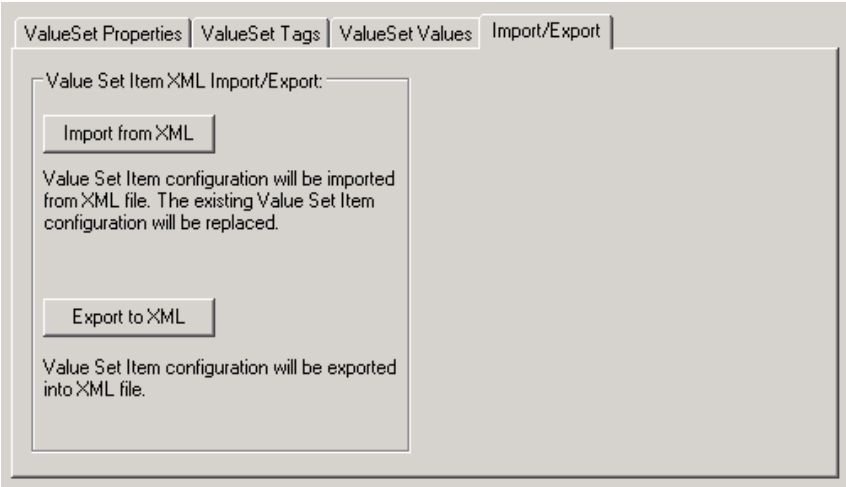


Figure 15.13 - Value Set Item: Import/Export

15.4.2. Using Value Set Tags

Once you have configured your Value Set tag, you can use it with any ICONICS OPC client (i.e. GraphWorX32, DataSpy, etc). The data source of the tag always starts with "vset" to indicate that this is tag from the Unified Data Manager and it's a Value Set tag. The vset is followed by a colon and the name of your Value Set Tag Item. If you are using parameters in parenthesis following the name of the Global Expression Item, this is where you would pass in the parameter values. Your expression may look similar to the one show in Figure 15.14.

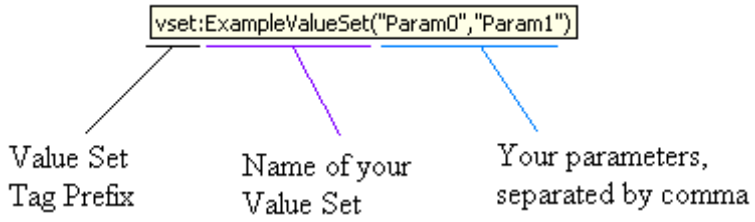


Figure 15.14 - Value Set Tag Syntax

In this case, the name of the Value Set Item is "ExampleValueSet" and it is giving the parameter in the tag the value "Param0" and "Param1." The tags that use these parameters will resolve to these values as part of the data source.

When you write to the Value Set tag, the values in the associated data tag will change according to the value you have set in the UDM. Figure 15.15 is a variation of the Colour_Chart Value Set in the demo example. When you write "white" to the Value Set tag, the other tags: description, R, G, and B also takes on new values.

Figure 15.15 - Using a Value Set Tag

15.5. Introduction to Recipes in the UDM

A recipe is a formula that is used to create a product. It consists of a list of ingredients called parameters and the amount or values for those ingredients. A recipe can also contain variations called recipe items which have different values for the same ingredients.

The ICONICS recipe solution is made up of the recipe configuration in Unified Data Manager which allows you to save these recipes and its corresponding parameter values and deploy them quickly and efficiently. The UDM also exposes the recipe symbols and recipe grid in GraphWorX32 to view, edit, add and download the recipes during runtime.

15.5.1. Configuring Recipes

Recipe configuration in the UDM consists of the recipe which contains the parameters and the recipe items that contain the values for the recipe parameters. You can configure Recipe Folders and Recipes by right-clicking on Recipes in the tree-control and the select the desire action.

Once you have created a Recipe, there are four configuration tabs:

- **Recipe Properties**: Allows you to configure the general properties of the recipe.
- **Recipe Parameters**: Allows you to add the different data sources you wish to use as part of the recipe.
- **Parameter Details**: Allows you to configure the details to each parameter you have added in the Recipe Parameters tab.
- **Import/Export**: Allows you to import and/or export recipe data.

15.5.1.1. Recipe Properties

You can configure the following in the Recipe Properties tab:

- **Enable**: Activates the recipe in the recipe server configuration. Enabled recipes are available to client applications via the Tree Explorer ActiveX and the Recipe ActiveX. For more information, please see the Tree Explorer and Recipe ActiveX help documentation.
- **Name**: The name of this recipe. Clicking the '…' button allows you to browse for global aliases and language aliases. Recipe names cannot contain periods (.).
- **ID #**: A unique identification number for this recipe.

- **Description**: A description of this recipe. Clicking the '…' button allows you to browse for global aliases and language aliases.
- **Percentage**: allows you to download only the specified percentage of each parameter into the data source. The default is 100%, which will download the parameter value as configured. You can specify the percentage in two ways:
  - **Absolute**: allows you to set a static percentage.
  - **Use Tag**: allows you to set a dynamic percentage based on a tag. You can click on the ellipse button to the right of the field to bring up the UDB for tag selection.

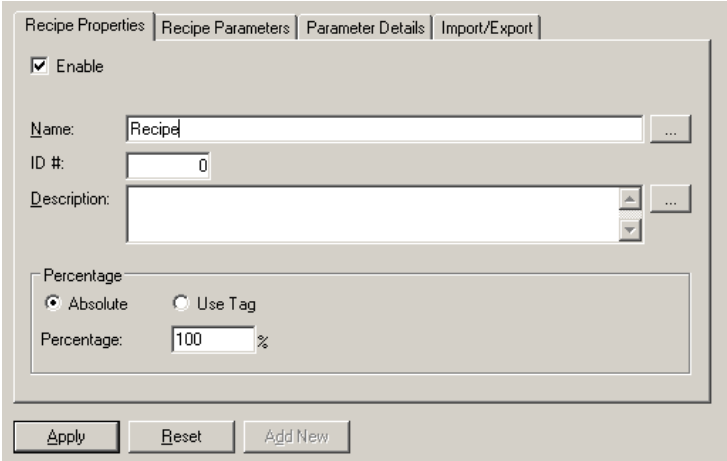


Figure 15.16 - Recipe Properties

15.5.1.2. Recipe Parameters

The next tab is the **Recipe Parameters** tab as shown in Figure 15.17. This tab is where the ingredients or parameters of the recipe are configured. Click on the "Add" button to add OPC tags that correspond to the ingredients used for this recipe. Each parameter is given an index number. You can click on "Remove" to delete the selected tag from the list.

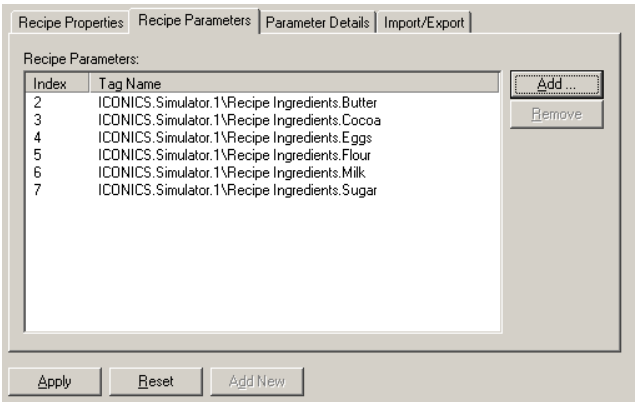


Figure 15.17 - Recipe Parameters

15.5.1.3. Parameter Details

The **Parameter Details** tab as shown in Figure 15.18 is where a name is assigned to each recipe parameter and to edit the properties of the parameter.

You can configure the following in the Parameter Details tab:

- **Parameter #**: The index number that was assigned to the parameter in the Recipe Parameters tab. Select one from the dropdown list or click on the right and left arrow buttons to scroll through available parameters.
- **Name**: A meaningful name for the parameter. Clicking the ellipse button allows you to browse for global aliases and language aliases.
- **OPC Tag**: The OPC tag that is associated to the parameter index. Click on the ellipse button to change this.
- **Data Type**: The data type of the OPC tag.
- **Unit**: The engineering unit that is used for this parameter.
- **Minimum**: The minimum value that can be set for this parameter.
- **Maximum**: The maximum value that can be set for this parameter.
- **Use Value Expression**: Enable this to recalculate the input value before the value is sent to the OPC tag.
- **Value Expression**: The expression that defines the calculations that is to be done. Use the {{value}} syntax to do calculations on the input value. Click on the ellipsis button to launch the Expression Editor which is available to assist you in creating your value expression.

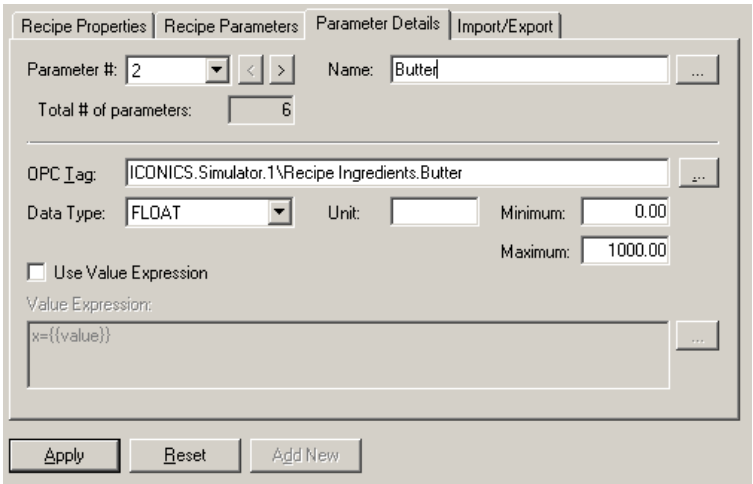


Figure 15.18 - Parameter Details

15.5.1.4. Import/Export

The Import/Export tab as shown in Figure 15.19 is used to import and/or export recipes from and to an XML or CSV file.

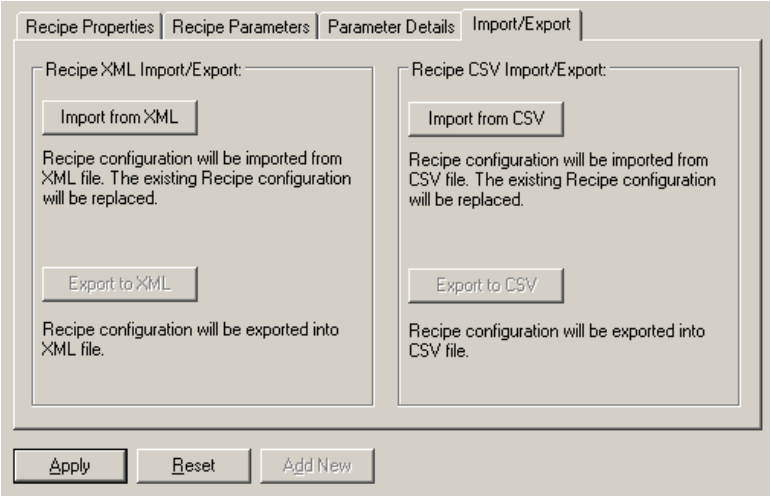


Figure 15.19 - Import/Export

15.5.2. Configuring Recipe Items

Each Recipe contains Recipe Items that would contain the corresponding values for the Recipe parameters. A new recipe item consists of three tabs. Each Recipe Item contains the following configuration tabs:

- **Recipe Item Properties**: Allows you to configure the general properties of the recipe item.
- **Recipe Item values**: Allows you to define the values for the recipe parameters for the particular recipe item.
- **Import/Export**: Allows you to import or export a recipe item using CSV and XML files.

15.5.2.1. Recipe Item Properties

You can add recipe item properties by selecting a recipe in the tree control, right mouse click and select new. You can configure the following in the Recipe Item Properties tab:

- **Enable**: Check the checkbox to enable or disable this recipe item.
- **Name**: The name of this recipe item. Clicking on the ellipse button allows you to browse for global aliases and language aliases.
- **ID #**: A unique identifier for the recipe item.

- **Description**: A description of the recipe item. Clicking the on the ellipsis button allows you to browse for global aliases and language aliases.
- **Author**: The author of this recipe item.
- **Version**: The version of this recipe.
- **Help**: Help instructions for this recipe item that can be displayed as a tooltip during a mouse-over of the item in a client such as GraphWorX32. Clicking the ellipse button allows you to browse for global aliases and language aliases.

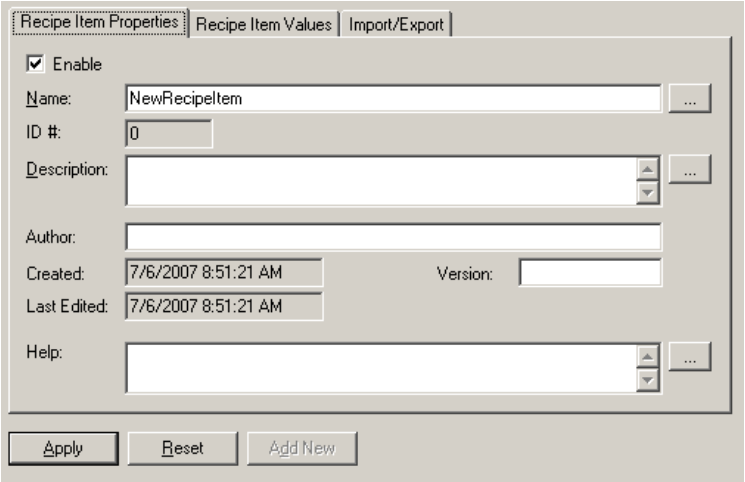


Figure 15.20 - Recipe Item Properties

15.5.2.2. Recipe Item Values

The Recipe Item Values tab as shown in Figure 15.21 is where the values for each recipe parameter is defined. Double-click on the value column for any recipe parameter to set a value.

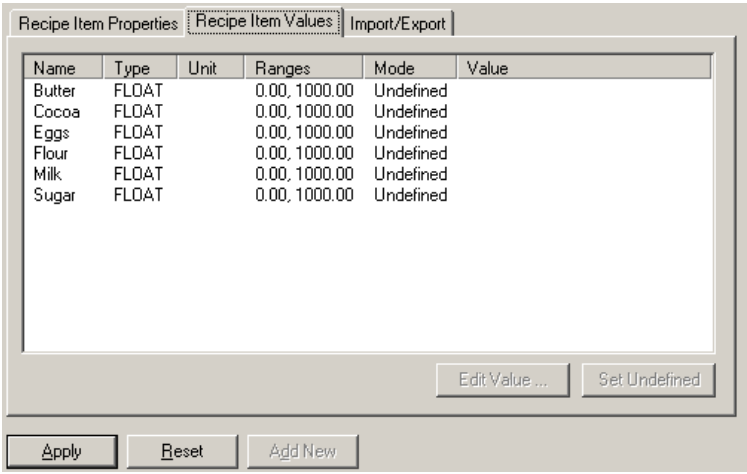


Figure 15.21 - Recipe Item Values

You can also use the “Edit Value” button to change the value of a parameter. Clicking on the “Set Undefined” button will set the value as “Undefined.” If you set a parameter as undefined, that parameter will not contain any value when you set the recipe.

15.5.2.3. Import/Export

The Import/Export tab as in Figure 15.22 is used to import and/or export the recipe item from and to an XML or CSV file.

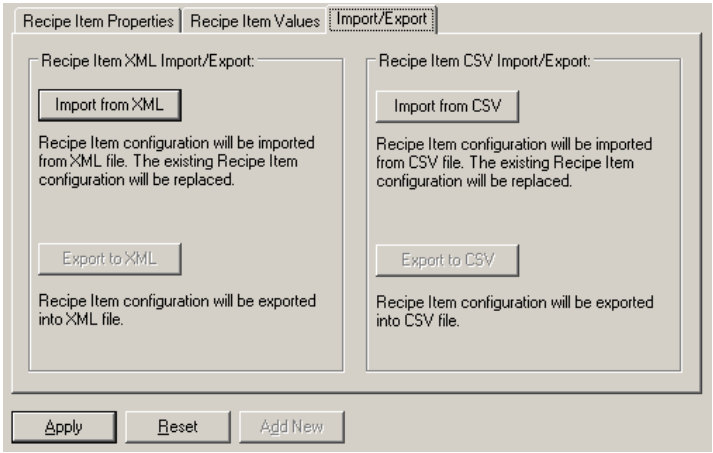


Figure 15.22 - Import/Export

15.5.3. Using Recipes

There are a two ActiveX controls that you need to be familiar with in order to visualize your recipes. Without going into too much detail about these ActiveX Controls, you can use the Symbols Library to generate an extensive display that allows you to see and control your recipes.

From the Symbols Library you can browse to the Recipe folder and select the Recipe.sdf library. The first symbol, “Default Recipe,” contains everything you need to view and control what you have configured in the UDM. Simply drag the symbol out from the library and drop it on the canvas of the GraphWorX32 display.

In your GraphWorX32 session, go into runtime. You can choose the different recipes you have configured in the Tree Explorer ActiveX control on the top right of the display. Once you select the recipe item in the Tree Explorer, the Recipe Grid and Recipe Settings populate with the data from that recipe.

You can change the recipe by clicking on the different cells in the Recipe Grid and typing in a new value. If you want to save the changes, you can click on the “Save” button in the Recipe Buttons section. You can also download the values to your data source by clicking on the “Download” button.

There are other buttons in the Recipe Buttons section. The function of these buttons is intuitive.

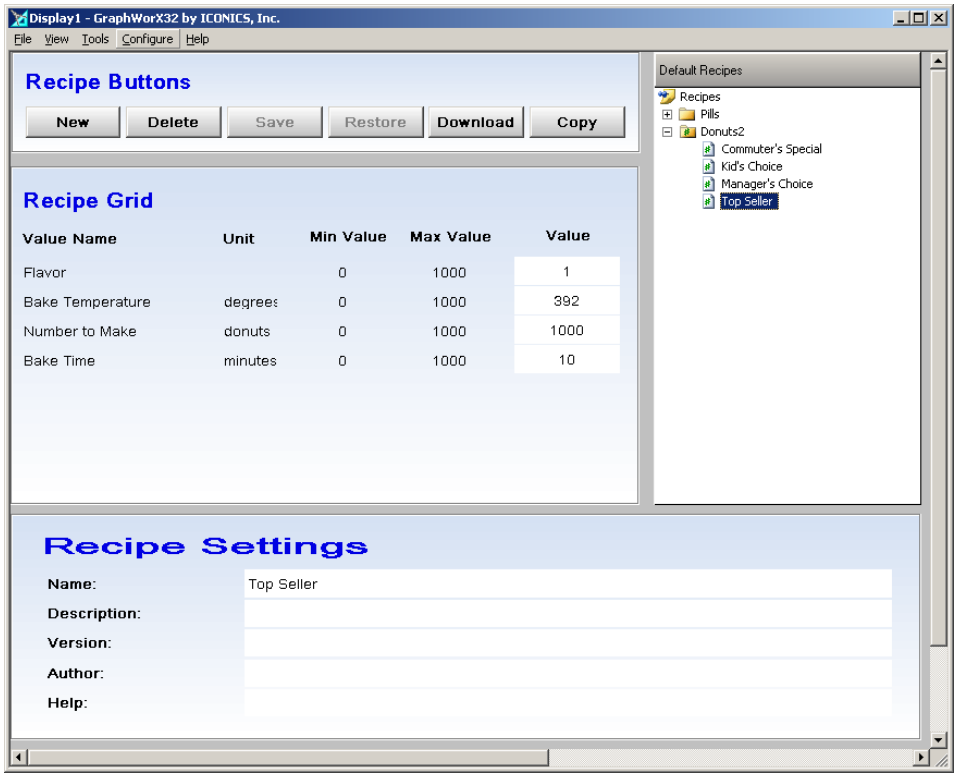


Figure 15.23 - Recipe Symbol in Runtime

15.6. Introduction to Global Alarm Subscriptions and Filters

We have discussed local alarm server subscriptions and filters in the “AlarmWorX32 Viewer” chapter. In this chapter, we will introduce you to global alarm subscription and filters. The configuration of both is similar to that of the local ones. The difference is that to gain global access we will configure it in the UDM. This will allow you to use common subscriptions and filters (some maybe very complicated) with a click of the mouse instead of having to configure it each time you want to use it.

In the UDM, you can configure Alarm Subscriptions and Filters in their respective node in the tree-control. They are organized in subscription/filter folders and subscription/filter items. Features of alarm subscription and filter include:

- Create and edit alarm subscription/filters.
- Create folders for alarm subscription/filters.
- Provides global management of alarm subscription/filters.
- Choose between local or global alarm subscription/filters.

Once you have configured a global alarm subscription or filter, then you can use them wherever you need to subscribe to an alarm server or use alarm filters.

15.6.1. Configuring Global Alarm Subscriptions

You can configure alarm subscription Folders and Alarm Subscription Items by right-clicking on Alarm Subscription in the tree-control and the select the desire action.

Once you have created an Alarm Subscription item, you will see the properties of the subscription as shown in Figure 15.24. You can give the subscription a name and a description.

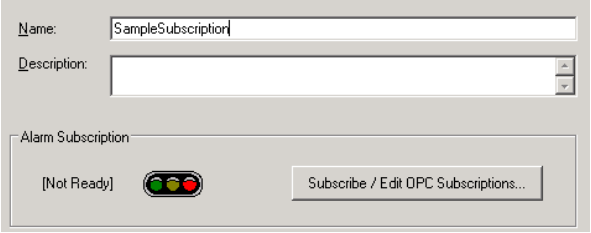


Figure 15.24 - Alarm Subscription Item Properties

To subscribe to an alarm or event server, you can click on the “Subscribe/Edit OPC Subscriptions…” button. Once you click on the button, you will see the Subscription Properties dialog as show in Figure 15.25. This dialog is used throughout any applications that require a subscription to an alarm or event server (i.e. Alarm Logger, Alarm Viewer ActiveX, etc). You can then add in subscribe to any OPC compliant alarms or events server. You can subscribe to as many A&E servers as you wish.

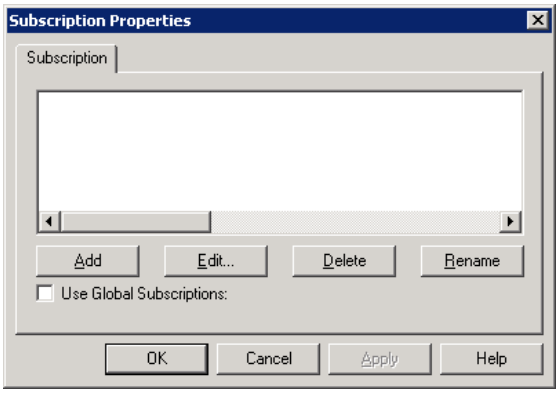


Figure 15.25 - Alarm Subscription Item - Subscription Properties Dialog

Notice that in Figure 15.25 toward the bottom, there is a checkbox for using Global Subscriptions. You can create a Global Subscription that includes other global subscriptions.

For example, if you have a global subscription for the ICONICS Alarm Server and another one for the GenEvent server, you can create a third global subscription that combines the two.

Once you have created a valid subscription to an A&E server the traffic light in the Alarm Subscriptions area will turn green indicating that the subscription is now ready for you to use.

15.6.2. Configuring Global Alarm Filters

You can configure alarm filter Folders and Alarm Filter Items by right-clicking on Alarm Subscriptions in the tree-control and selecting the desired action. Once you have created an alarm filter you will see the alarm filter properties as show in Figure 15.26. You can give the filter a name, a description, and the filter string. By default, the filtering string is “x=1.” This guarantees that you will always have a filter but the default filter will always be true.

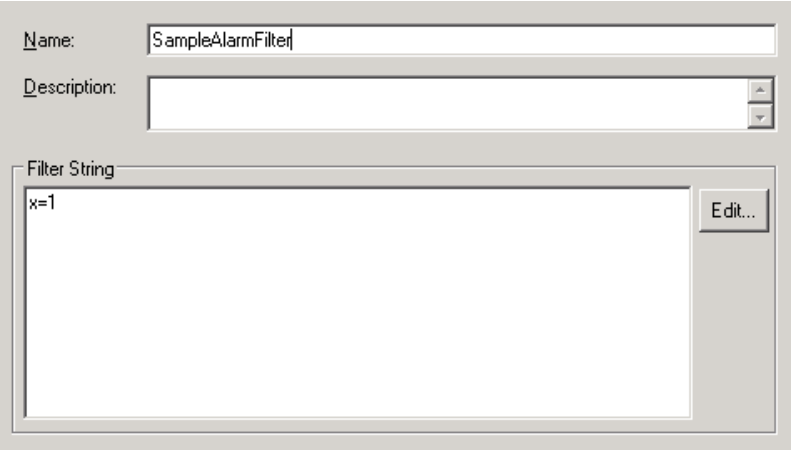


Figure 15.26 - Global Alarm Filter Properties

If you click on the "Edit" button when the Filter String field is blank, you will get the Filter Wizard dialog. This is the same dialog used throughout all application that allows you to use an alarm filter. You can use this wizard to configure your filter.

You can also use the Expression Editor to configure your wizard. To get to it, you can click on "Advanced…" in the Filter Wizard dialog. If you have a filter string already, you can click on "Edit" and in the dialog that comes up click on "Yes" to preserve the filter. This will also bring up the Expression Editor as well.

You can use either the Filter Wizard or the Expression Editor to build your filter. You can also type in your own Filtering string in Filter String section. It is however, recommended that you try to use the Expression Editor to try to catch any syntax errors.

15.6.3. Using Global Alarm Subscriptions and Filters

Once you have configured your global Alarm Subscriptions and/or Filters, you can use it in any ICONICS applications that allow you to make a subscription or filter (i.e. Alarm Logger Configurator, Alarm Viewer, etc).

For the global alarm subscription, the data source always starts with "awxs" to indicate that this is a global subscription. The "awxs" is followed by a colon and the name of your alarm subscription as shown in Figure 15.5.

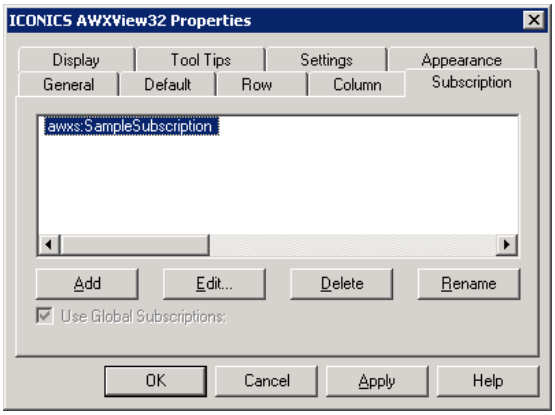


Figure 15.27 - Global Alarm Subscription Syntax

Notice that the Use Global Subscriptions checkbox is disabled when there is a subscription in the list. To enable or disable Use Global Subscriptions: the list of alarms subscription must be empty. Furthermore, *you may not mix global and local subscriptions in any one subscription dialog in an application*. Once you start using a global subscription, you will need to use global subscriptions for the application.

To add a global subscription, check off the **Use Global Subscriptions**, option and click on add. This will launch the Unified Data Browser for you to select the Global Subscription that you would like to use.

For the global alarm filter, the data source will start with “awxf” to indicate that it is a global alarm filter. The prefix is follow by the usual colon and then the name of the filter as shown in Figure 15.28.

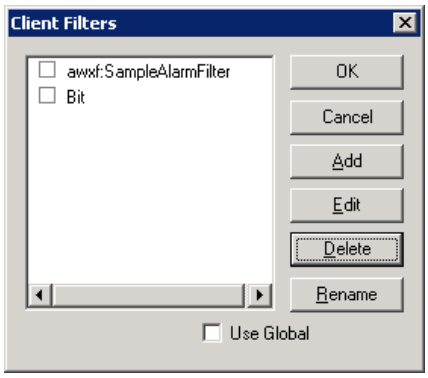


Figure 15.28 - Global Alarm Filter Syntax

Depending on the application that you are using, the way to setup Global Filter is different. For example, Figure 15.28 is an example of the client filter setup in the Alarm viewer ActiveX. To insert a global filter here you would add a filter, then select the new filter, check the Use Global Filter option, then click on Edit. It will then bring up the Unified Data Browser for you to choose the global filter you would like to use. Unlike global subscriptions, you can use a mixture of local or global filters here.

Lab: Recipe Management L15

Thusfar in the project we have configured alarms and trends for the Acmeseutical pill making process. However, a crucial part of this project is to allow the operator to select a recipe on the HMI. We will need to set up a recipe management system for Acmeseutical to complete the project requirement.

Estimated time to complete this lab: 45 minutes

15.1. Task Checklist

In this lab, you will complete the following tasks:

- Create a Recipe
- Create Recipe items
- Create a HMI interface

15.2. Support Files

You will need the following file(s) to complete this lab:

- ACMEPillMachine.gdf
- AcmeseuticalTemplate.tdf

15.3. Recipe Configuration

We will first need to create a set of Recipes before we can make the HMI that allows the operators to change it.

1. Open the Unified Data Manager (UDM) by going to Start → Programs → ICONICS Tools → Unified Data Manager.

2. Since there are other example configurations here, we will not create our own configuration database. Instead, we will add to the existing configuration.

3. In the tree control of the UDM on the left-hand side, expand Recipes.

4. Expand "Acmeseutical Recipes."

5. Double-click on the “Pain Killers” folder on the Recipe Properties tab.

6. Change the Recipe ID to ten (10).

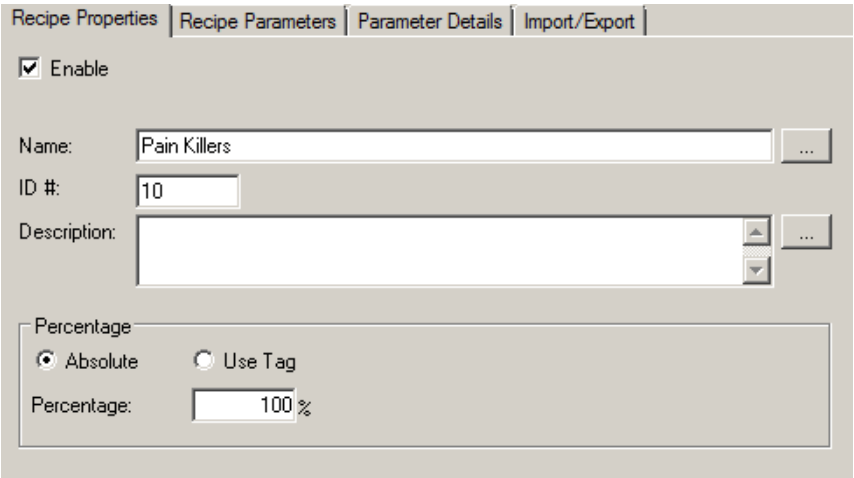


Figure L15.1 - Recipe: Recipe Properties

7. Select the Recipe Parameters tab and click on the “Add” button.

8. In the Unified Data Browser (UDB), select the OPC DA tab if it is not already selected.

9. On the tree control in the bottom right-hand pane, browse to ICONICS.DataWorX32.8 → Registers → USA → Foxborough → Machine1.

10. Double-click on the NumberOfPills and RecipeName tags to add them.

11. In the tree-control on the right-hand pane, right-click on Bottling Machine and select “Enter Up” to add the two tags that are in this folder.

12. Expand Mixers → MixerA. Using the Crtl key select the Contents, Level_Setpoint, and RequireInRecipe tags. With these three tags selected, right-click on one of the highlighted tags and select Enter Up.

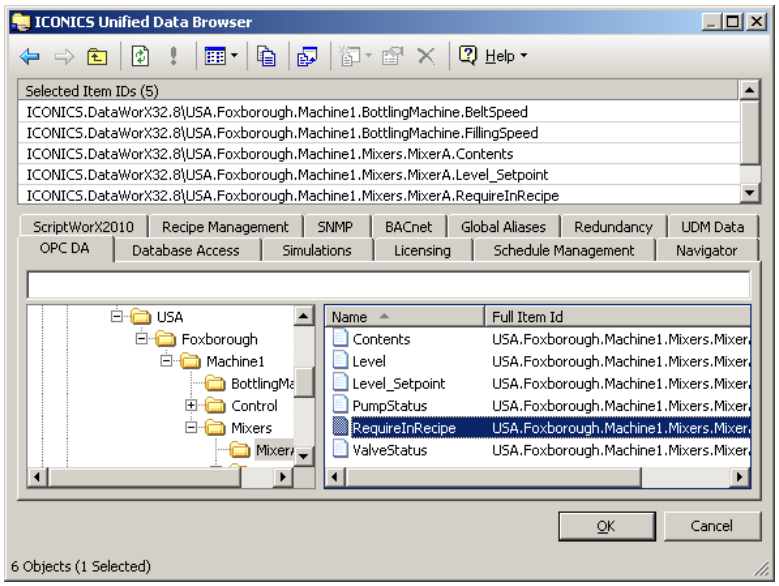


Figure L15.2 - Selected Tags in Unified Data Browser

13. Equally, add the Contents, Level_Setpoint, and the RequireInRecipe tags for MixerB, MixerC, and MixerD.

14. Click on “OK” to add all the tags into the Recipe Parameters tab.

15. Select the Parameter Details tab. Notice that the Parameter # is currently at 0 and the OPC Tag is the one for the BeltSpeed.

16. Name this parameter “Belt Speed.” Click Apply to commit the changes.

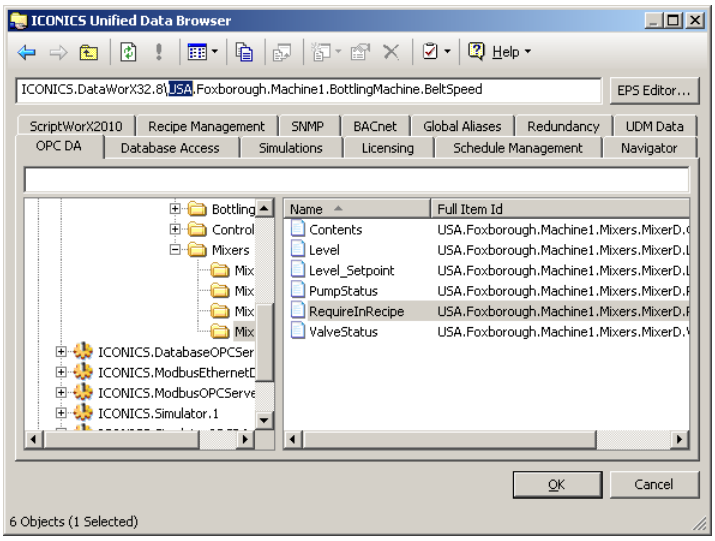


Figure L15.3 - Highlighting Part of the Tag

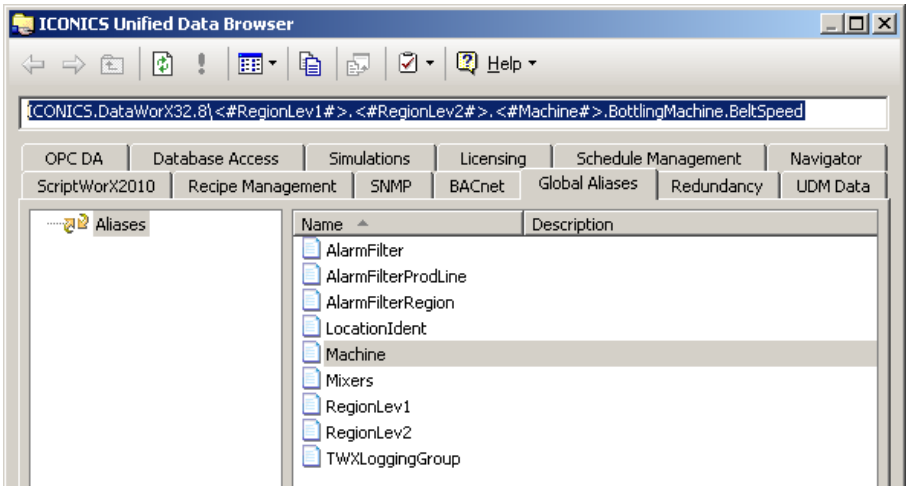


Figure L15.4 - Tag with Global Aliases

17. The “Total # of parameters” field should indicate sixteen (16) parameters.

Note: If you select the Recipe Parameters tab, you will notice that, although there are 16 parameters, the values are 0 thru 15.

18. Right click on the “Pain Killers” folder and select New → Recipe Item.

19. Name the Recipe Item “MigraineAwayClassic” and set the ID # to 5.

20. Enter a Description (show your creativity), Author, (your name), Version (e.g. 1.00) and “See data sheet” in the Help field.

21. Click Apply and select the Recipe Item Values tab.

22. By either selecting the tag and clicking on the Edit Value button, or by double clicking in the Value column, enter the values from the table below.

Name	**Value**
Belt Speed	15
Filling Speed	750
Mixer A Contents	Soya
Mixer A Level_Setpoint	75
Mixer A Required	1
Mixer B Contents	CaCl
Mixer B Level_Setpoint	44
Mixer B Required	1
Mixer C Contents	Migraine Away Classic
Mixer C Level_Setpoint	60
Mixer C Required	1
Mixer D Contents	PainKiller A
Mixer D Level_Setpoint	58
Mixer D Required	1
Number of Pills	40
Recipe Name	MigraineAwayClassic

23. You should have ten (10) recipes when you are done. Your UDM Recipes should look similar to Figure L15.5.

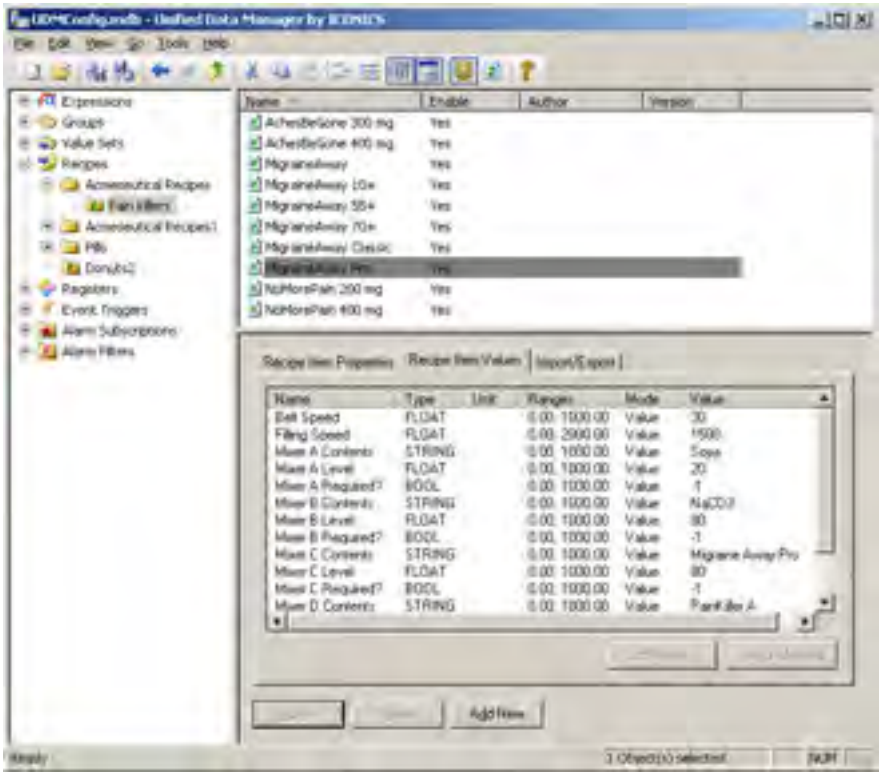


Figure L15.5 - All Recipes Created

15.4. Recipes Display

Now that we have completed the recipe configuration, we are ready to make a display for the operators to use.

1. From the Windows Explorer, double-click on your AcmeseuticalTemplate.tdf to create a new display.
2. Save this display as ACMERecipes.gdf in the C:\Training folder. If this file name already exists, you may replace it.
3. Open the Symbols Library and browse to Recipe → Recipe.sdf
4. Drag the "RecipeButtons" and "RecipeSettings" symbol onto the display.
5. Place the RecipeSettings near the bottom.
6. Drill into the RecipeButtons symbol and change the buttons to align vertically instead.

> ***Tip!*** You can use tools such as align left and distribute evenly down in the align toolbar.

7. Resize the background rectangle to fit the new button layout.
8. Move the edited Recipe Buttons to the right of the display.
9. Click on the "Tree Explorer" icon in the ActiveX Toolbar.
10. Place the control to the left of the display and resize it.
11. Double-click on the control to bring up the properties dialog.
12. Click on the Tab Configuration tab.
13. Click on the "Insert Tab" button. You should now see a new item in the area below the "Insert Tab" button.
14. Select the "New Item". The options in the Tab Properties section should become available.
15. In the Selected Tab Name: field, type in "Acmeseutical Recipes."
16. In the Tab Type: dropdown list, choose Recipe Explorer.
17. Click on the "Advanced" button to bring up the **Additional Node Path** dialog.

18. Click on the ellipse button to bring up the UDB.

19. Select the “Acmeseutical Recipes” item and click on “OK.”

20. In the Initial Selection field, type in “Pain Killer\MigraineAway” (without quotes).

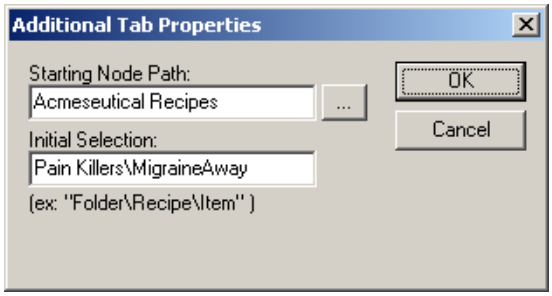


Figure L15.6 - Additional Tab Properties

21. Click on “OK” twice to return to the GraphWorX32 display.

22. Click on the “Recipe Grid” icon in the ActiveX toolbar.

23. Resize the control to fit in the remaining empty space of the display.

24. Select Format → Template → Edit Applied Template. In the template header, remove the global alias text and “Navigate To” button from the display. Exit out of the template, Format → Template → Exit Edit Applied Template.

25. Your display may look similar to Figure L15.7 now.

Figure L15.7 - Complete Recipe Display

26. Save the display and go into runtime.

27. Notice that the MigraineAway recipe is the default one that is selected.

28. Click on the different recipes in the tree explorer and see that the information about that recipe shows in the Recipe Grid.

29. Once you have the information loaded into the Recipe Grid, you can change the values of the different parameters in the Values column (this is the only column you can edit).

30. Click on the "Download" button to push the recipe values onto the OPC Tags. You can verify that the recipe is in use by checking the values in your ACMEPillMachine display.

    NOTE: If you change a value in the recipe, when you click on the "Download" button, the recipe will save with the new value and then write the appropriate recipe information into the data sources.

31. Feel free to click on other buttons to add or delete a recipe.

32. To make getting to the ACMERecipes display easier in the future, we will add a button to the ACMEPillMachine display.

33. Open the ACMEPillMachine.gdf display and add a button by clicking on the "Button" icon in the Dynamics toolbar.

34. Click on the display next to the funnel.

35. In the **Properties Inspector**, check that the Action is set to "Load Display."

36. In the Filename field, type in "ACMERecipes.gdf."

37. Click on the Button tab and change the button text to read "Select Recipe."

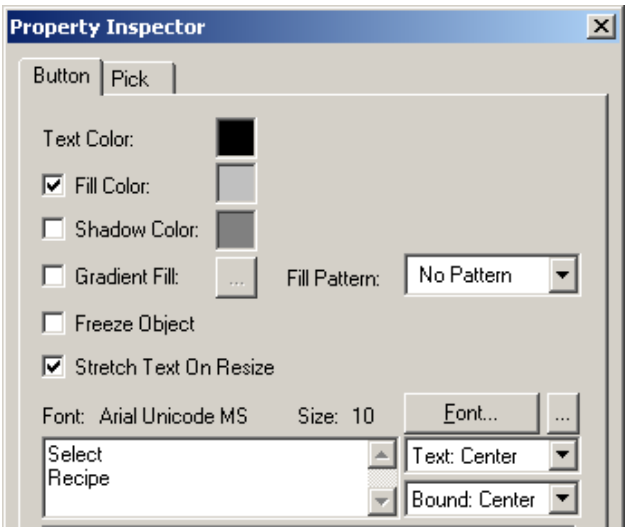


Figure L15.8 - Button Property

38. Click on “OK” to close the **Property Inspector**.

39. Save the ACMEPillMachine.gdf display.

40. Go into runtime and make sure that the Select Recipe button works accordingly.

15.5. AcmePillMachine Display

Now that we have completed the recipe display, we will add process point fields to the AcmePillMachine display to display the contents, belt speed and filling speed values.

1. From within GraphWorX, open the AcmePillMachine.gdf file.

2. Note the additional fields in the figure below. One **Contents PPT** for each mixer. And one **PPT** each for **Belt Speed** and **Filling Speed**. Let's proceed.

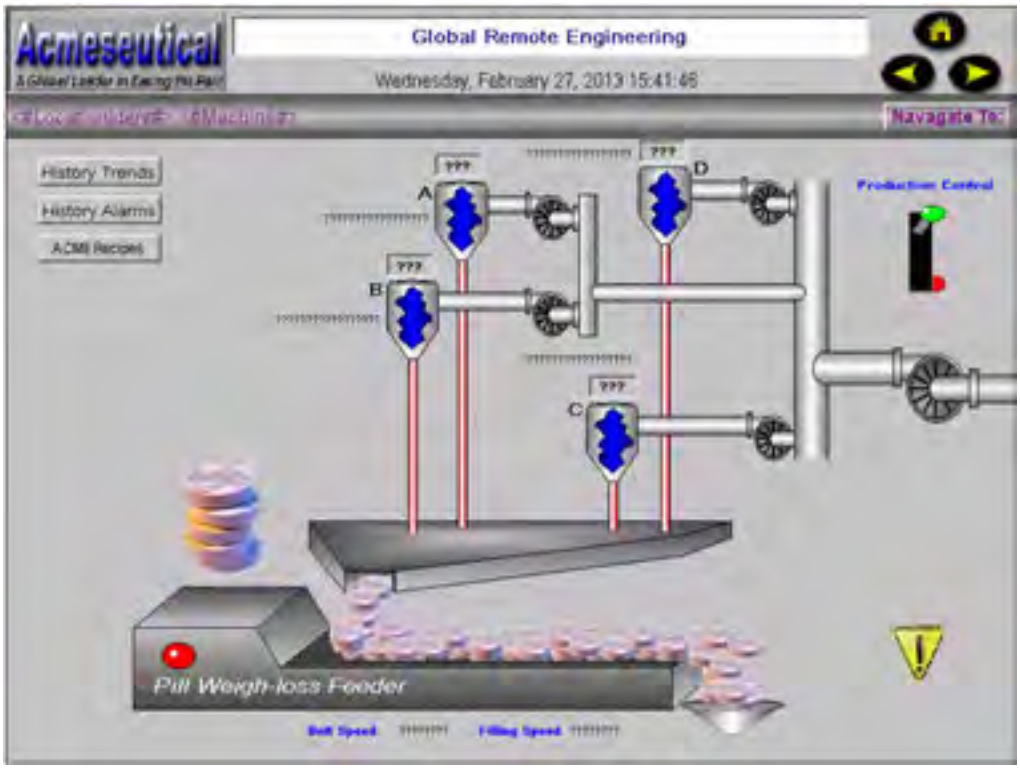


Figure L15.9 – Content, Belt Speed and Filling Speed PPTs Added

3. Click on the "Process Point" button in the Dynamics toolbar and click somewhere near Mixer A in the display.

4. In the Property Inspector's "PPT/DE" tab, click on the "Data Tags…" button to bring up the Unified Data Browser (UDB).

5. On the OPC DA tab, expand ICONICS.DataWorX32.8 → Registers → USA → Foxborough → Machine1 → Mixers → MixerA and select the Contents tag.

6. Set the Format field to 17 to limit the string to display only 17 characters.

7. Select the "Text" tab and change the 3D Edge filed to "Sunken."

8. Set the alignment to Center and bounding to Left. Click on "OK" to save settings and close the Property Inspector.

9. Copy this PPT and place a copy near each of the other three mixers. Edit the data source to be MixerB, MixerC or MixerD accordingly. Use the figure above as a reference.

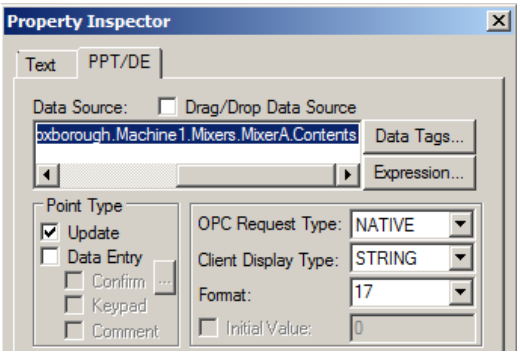


Figure L15.10 - MixerA Contents Tag

10. Next we will add the Belt and Filling Speed text and process points. We will position the text and process points as seen in the previous figure.

11. Select the "Text" button from the Draw toolbar. Double-click on the text to open the property inspector. Rename the text to Belt Speed. Perform the same procedure for the Filling Speed text.

12. Click on the "Process Point" button in the Dynamics toolbar and click beside the corresponding text.

13. In the Property Inspector's "PPT/DE" tab, click on the "Data Tags…" button to bring up the Unified Data Browser (UDB).

14. On the OPC DA tab, connect one Process Point to ICONICS.DataWorX32.8 → Registers → USA → Foxborough → Machine1 → BottlingMachine →BeltSpeed.

15. Connect the other Filling Speed Process Point to ICONICS.DataWorX32.8 → Registers → USA → Foxborough → Machine1 → BottlingMachine →FillingSpeed.

16. Save the display.

Review: Acmeseutical Pill Recipes

R15

15.1. Chapter Objectives

You should know and understand the following topics:

- How the Unified Data Manager can ease development
- How the Unified Data Manager works
- Configuring a Read Expression with parameters
- Configuring a Write Expression that takes user inputs
- How Global Expressions work with other ICONICS Applications
- Configuring a Group in the Unified Data Manager
- Using Group tag to change multiple tags to a single value
- Configuring a Value Set in the Unified Data Manager
- Using Value Set tags to change multiple tags to different values
- How recipes are setup in the Unified Data Manager
- The difference between Groups, Value Sets, and Recipes
- How to configure a Recipe in the Unified Data Manager Configuring alarm subscriptions and filters in the Unified Data Manager
- Using global subscription and filters

15.2. Summary

The Unified Data Manager (UDM) is a centralized application for you to configure expressions, alarm subscriptions, and alarm filters. You can also use the UDM to create useful data sources such as Groups, Value Sets, and Recipes. Using the different parts available in the UDM, you can reduce your development time.

For example, you can create global expressions in the UDM Expressions section. In the data source you can then dynamically pass on that which you need to perform the expression. This way, instead of creating the expression for each tag in the client application, you can create the expression once and simply provide the data source when you want to use the expression. This also makes maintaining of the expression simpler.

If you use a similar expression in multiple places, you will need to edit each instance of the expression on the client. However, if you configured the expression in the UDM, you only need to edit the expression in the UDM once.

The UDM also provides other global parameters such as Alarm Subscriptions and Alarm Filters. You can also use the UDM to create Groups, Value Sets, and Recipes. While these items behave similarly, they are very different. A Group tag allows you to broadcast a single value to multiple predefined data sources while a Value Set allows you to write different predefined values to a defined set of tags. Recipes work similar to a Value Set, but it includes many more options on the client side.

15.3. Terms

UDM Expression	UDM Group	UDM Recipe
UDM Trigger	UDM Value Set	Unified Data Manager

15.4. Test Your Knowledge

In this chapter, we discussed the Unified Data Manager being a centralized location to configure expressions, recipes, etc. Before you continue to the next chapter, make sure you can answer the following questions.

1. The following data type is NOT configured in the UDM:

    a. Sets of values
    b. Groups of users
    c. Expressions and calculations
    d. Triggers that are based on conditions

2. True or False: You can use a UDM group in another UDM group?

3. True or False: You can make use of parameters in value sets, groups, and expressions?

4. In which of the following application can you use a global filter?

    a. AlarmWorX32 Viewer
    b. AlarmWorX32 Logger
    c. AlarmWorX32 Indicator
    d. All of the Above
    e. Only A and B
    f. None of the above

5. True or False: You can use both local subscriptions and global subscriptions together in a single alarm viewer.

6. True or False: When you select a recipe in the Tree Explorer, the information about the recipe shows in the connected recipe viewer, but the values associated in the recipe are not downloaded to the data source until you click on a Pick Action to download the value.

7. Which of the following can you use to replace a section of a data source in an expression?

   a. Variable
   b. Alias
   c. Parameter
   d. None of the above; you cannot have a dynamic data source

8. What is the difference between a Group and a Value Set?

   a. A Group allows you to write a single value to all predefined tags at the same time while a Value Set writes predefined values for each tag based on its input value.
   b. A Group allows you to writes predefined values for each tag based on its input value while a Value Set write a single value to all predefined tags at the same time.
   c. A Group allows you to define many tags in a single data item for writing to while a Value Set defines many tags in a single data item to read.
   d. There is no difference between the two.

Security 16

The security of a system is crucial to many production environments. In general, you want to protect the system from unauthorized access to your production data. Sometimes you may want to go so far as preventing your operators from exiting a screen or editing configurations.

The GENESIS32 security system provides restricted access to GENESIS32 functions based on the concept of a logged-in user. A security system administrator configures the system by adding users and assigning them specific GENESIS32 privileges. In addition, administrators may associate users with certain administrator-defined groups that also have assigned privileges. Thus, a user has the effective rights of all the groups to which he or she belong plus his or her own private rights.

By the end of this chapter you should be familiar with the following:

- Setting up security groups and users
- Distinguishing between advanced, basic, and NT integrated security modes
- Importing domain accounts
- Using the Security Login Dialog
- Configuring the Security Indicator ActiveX Control

16.1. Installation

Every GENESIS32 installation comes with a security system which includes a Security Server and a Security Login client. After installation, the system runs without restriction until you apply a security configuration on the system. Security protection applies to the following items within the GENESIS32 system:

- Application actions
- Process output points
- Critical points
- Alarms
- Critical Alarms
- Files
- Custom strings
- Stations

The security configuration also includes protection against unauthorized access. If you configure a security administrator you will have to enter a password to run the Security Configurator.

16.2. Security System Components

The security system, shown in Figure 16.1, consists of a Security Server and several security clients. The clients communicate with the server via Microsoft COM/DCOM and therefore can optionally execute on network nodes other than the Security Server Node.

The security system provides two special purpose security clients: one for user login (the Security Login application) and another for administration of the Security Server (the Security Configurator). The rest of the security system clients are the other applications in the GENESIS32 product family (e.g., GraphWorX32, TrendWorX32, AlarmWorX32, etc.). Any stimulus (e.g., a user login or logout) that causes a change in security status will immediately post to the affected clients.

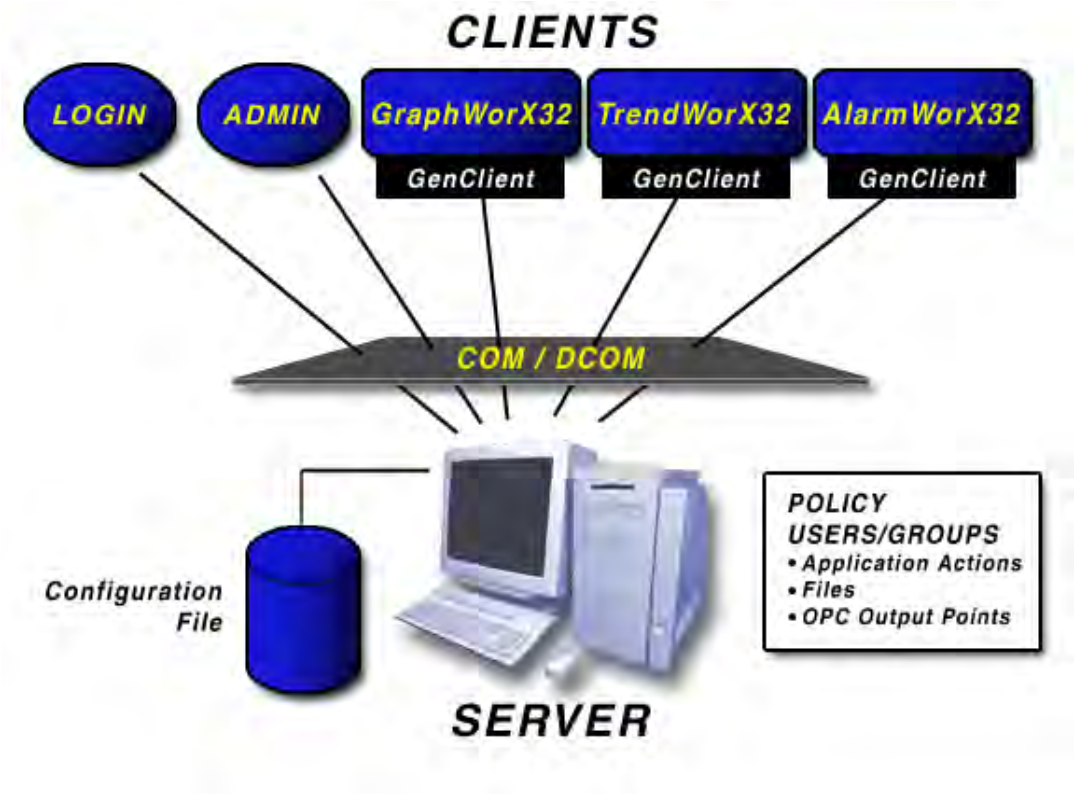


Figure 16.1 - Security System Components

16.3. Security Server Configuration

The **Security Configurator** allows Security Server administrators to configure security settings for users and groups. Configuration of the security system is accomplished by running the Security Server Configurator in interactive mode. The Security Server may be launched in interactive mode from the GENESIS32 program group by going to **Start → Programs → ICONICS GENESIS32 → Tools → Security Configurator**. You can also launch the configurator from other GENESIS32 applications while they are in configuration mode.

16.3.1. Launching the Security Configurator for the First Time

Since the system installs without security in place, the first time you launch the security configurator you will need to create a security file. ICONICS Security comes in different modes and when you create a security file, you will need to choose in which mode to create the file. The dialog shown in Figure 16.2 allows you to select the mode for your security configuration.

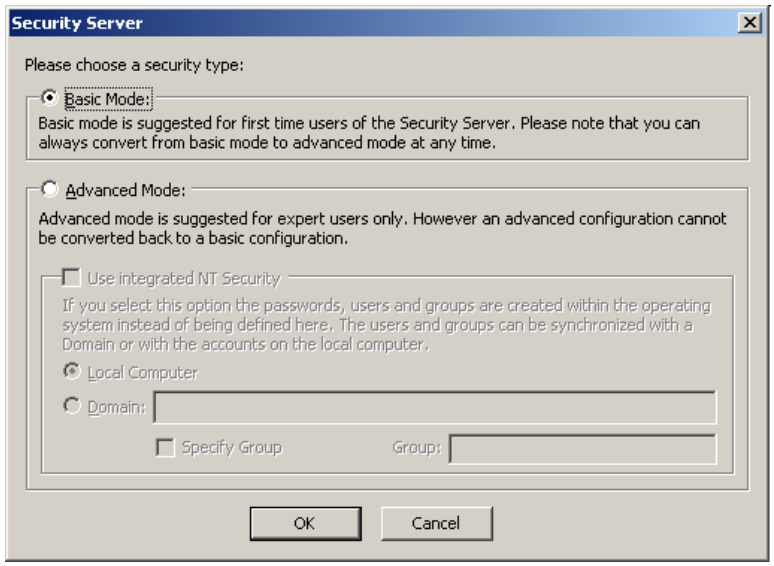


Figure 16.2 - Security Server: Security Modes

You will see this dialog immediately after you launch the Security Configurator for the first time as well as each time you want to create a new security configuration. We will discuss the differences between each available security mode in the "Security Configuration Modes" section.

Once you choose the security mode, you will be asked to specify a file name and location for your security configuration file as shown in the **Save As** dialog in Figure 16.3. The default location for this file is in the \Program Files\ICONICS\GENESIS32 \GEN32DEMO folder with the name "Untitled." You can save this file to any other location with a name that you prefer. To change the name and/or location of the file after you have created it, choose **Save As** from the **File** menu in the security configurator. Once you save the file, this will become the active security configuration and opening the configurator in the future will automatically load this file on startup.

Figure 16.3 - Saving a Security Configuration

NOTE: The most recently saved .sec file is always the currently active security configuration.

Once you save the security configuration file, the Security Configurator will open an empty configuration as shown in Figure 16.4. Notice that the configurator is divided into two sections. We will discuss the reason for this in the "Group and User Permissions" section.

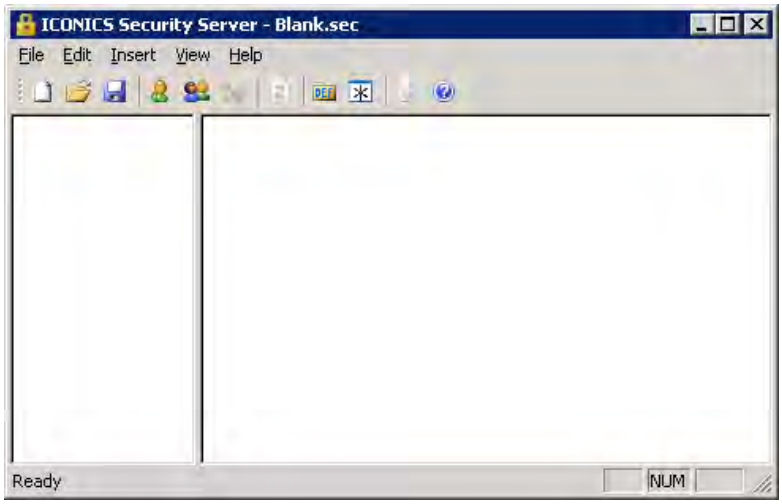


Figure 16.4 - Blank Security Configuration

> ***Tip!*** Because each time you create a security configuration it starts a configuration file with no security applied, the last saved file is the active configuration. You can always create a new security configuration file to revert to a state without security on the system.

16.3.2. Security Configuration Modes

The Security Server supports three general modes of security configuration. The security mode is specified in the Security Configurator:

- Basic security mode
- Advanced security mode
- Integrated NT security mode

The Security Server can run in "basic" or "advanced" mode. We suggest using the **Basic mode** if this is the first time you are using the Security Server. The **Advanced mode** is equivalent to the only security mode in version 7.x and prior to GENESIS32.

> **NOTE:** You can always convert a basic mode configuration to an advanced mode configuration at any time. However, you cannot reverse the conversion from basic mode to advanced mode (i.e., an advanced configuration cannot be converted to a basic configuration).

The **integrated NT security mode** automatically synchronizes users and groups with the Windows NT security database. You can synchronize with users on the local computer or to your organization's domain. If you choose to synchronize with your organization's domain, you must run the Security Server and Configurator with a user that has permissions to query the Active Directory on the domain. If the domain is large, you can provide a specific group in the domain to query. This will result in a faster query returned by the domain controller.

16.3.3. Group and User Permissions

The Security Configurator consists of two separate panes. Both panes of the view will be empty when you first create a security configuration. Each pane has a tree control. The left tree is the **Group View**. Here the root nodes are groups, and the child nodes are the users that belong to the group. The right tree is the **User View**, in which the root nodes are users, and the child nodes are the groups to which each user associates.

The example security configuration in Figure 16.5 shows sample users and groups for a factory. The personnel groups for the factory are:

- General Manager
- Supervisors
- Operators

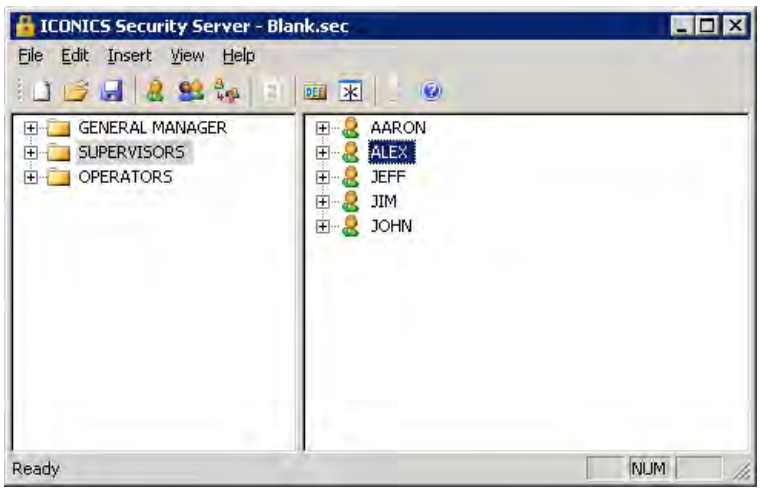


Figure 16.5 - Example Security Configuration

Each of these groups has one or more users, all of whom need to have access to factory data. There are five different users:

- Aaron (Operator)
- Alex (Supervisor)
- Jeff (General Manager and Supervisor)
- Jim (Operator)
- John (Supervisor)

Figure 16.6 - Configuring Advanced Security for Users and Groups

As you can see in Figure 16.6, you can associate users with various groups to help simplify and organize security management. This way, all users associated with a particular group are bound to the restrictions or properties for that group. For example, both Jeff and John are supervisors, associated with the **Supervisors** group.

> **NOTE:** Jeff is associated with both the **General Manager** group and the **Supervisors** group. This association of one user with two different groups is possible only in advanced security mode.

If there are certain files, for example, that only the general manager and supervisors are allowed to view but the operators may not view, the security administrator can limit the operators from accessing those files by configuring the **Operators** group properties.

You can also configure properties for each user within a group. While it is true that all users in a group inherit the permissions of the group, a user can also have unique security permissions. For example, both Aaron and Jim are operators and are therefore associated with the **Operators** group, inheriting the Operators group permissions. However, Aaron's user properties may be configured separately from those of Jim such that Aaron can view a subset of the files that the Supervisors can view.

16.3.3.1. The Default Group

The Advanced security mode has a special group of users: the Default Group. When configuring permissions for the Default Group, you should treat it similar to a “guest” user. All users will inherit any permission configured for the group. Furthermore, anyone NOT logged into the system will use permissions specified in the Default Group.

When you first create a security configuration in Advanced mode, the Default Group is set to allow everything. That means, without logging into the security system, anyone can view a file, change a value of a process point, acknowledge an alarm, and open any GENESIS32 application. Because of this, we recommend that after you create a security configuration in Advanced mode, you should first limit the permission on the Default Group and then create a Security Administrator (see section “Security Administrator” for more details) with all permission so that you have one backup user in case you went too far in restricting access.

Also note that the Account Policy configured for the Default Group is the default account policy used when creating a Group or User. If there is a certain policy that you use repeatedly, you should configure it in the Default Group.

16.3.3.2. Security Administrator

When you create a user in ICONICS security, you can make the user a Security System Administrator. To make a user the security system administrator is as simple as checking the "Security System Administrator" option at the bottom of the User Properties dialog shown in Figure 16.7.

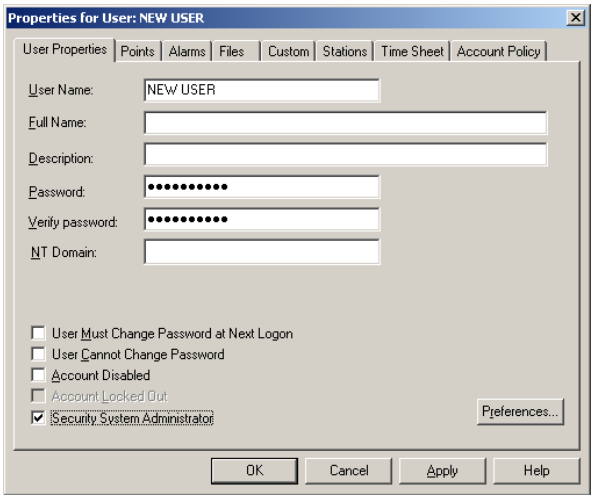


Figure 16.7 - Security System Administrator

Being the Security System Administrator does not mean that this user has all permissions regardless of your configuration. In fact, you can give the user no permissions in viewing files and interacting with alarms and process points. However, this user will have permission to open the Security Configurator.

If there is a lack of a Security System Administrator, anyone can open the Security Configurator to modify user permissions. As soon as the system detects that there is one Security System Administrator, then you must enter an administrator username and password to gain access to the Security Configurator.

When there is a configured Security System Administrator, launching the security configurator will open the **Security Server Administrator Login** dialog, shown in Figure 16.8. You must enter one of the following to proceed to configuration:

- A **User Name** and **Password** for a user that has previously been configured as a Security Administrator.
- An emergency password you received from technical support based on the challenge code shown in the login dialog.

Figure 16.8 - Security Server Administrator Login

16.3.3.3. Configuring Permissions

The Group and User properties dialog are very similar. The only difference between the two is that when you are configuring permissions for a user, you have a User Properties tab and when you are configuring permissions for a group, you have a Groups Properties tab.

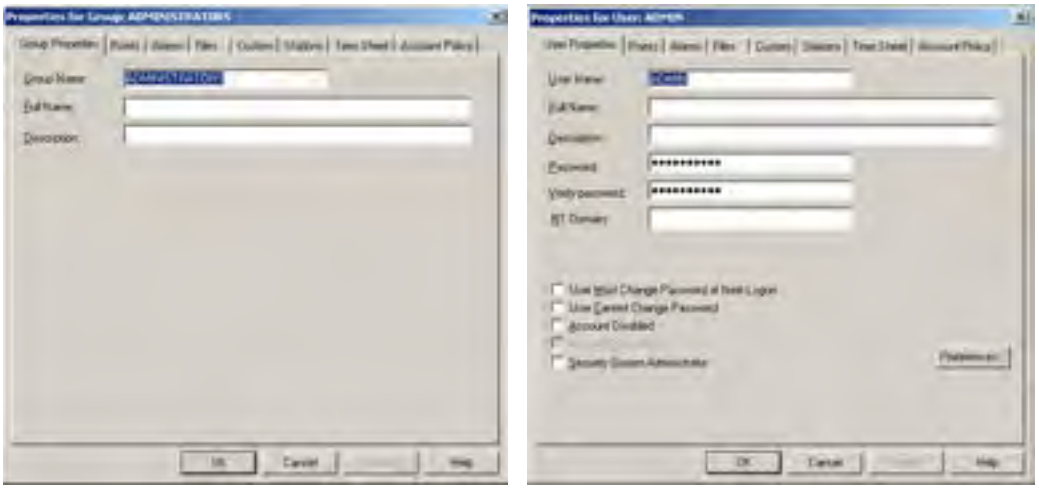


Figure 16.9 - Group Properties (left) v. User Properties (right)

On the Points, Alarms, and Files tab, you can add the respective objects in the Allow or Deny list. You should know that Deny permissions always have precedence over Allow permissions. The proper way to achieve security in this situation is to exclude the point from both the Allow and Deny lists and add the point for the user permission in the Allow list.

In situations where you want to include everything, you can use the wildcard character * (asterisk). For both Points and Alarms, you can simply add this character in the Allow list to allow writing to and acknowledging process points and alarms respectively.

For files, you can add *.* into the allow list, which means that you are allowing access to any filename with any extension.

The Properties dialog also supports Regular Expressions. When you put your mouse pointer in the Allow or Deny textbox, the Regular Expression help tooltip will display. You can use this to help you write a Regular Expression to match the name of a point, alarm, or file. You can always test the expression by entering some text in the **Text String** textbox at the bottom of the respective tabs. If the test string matches those in the Allow list, then the system will check the **Access Granted** checkbox to indicate the security status.

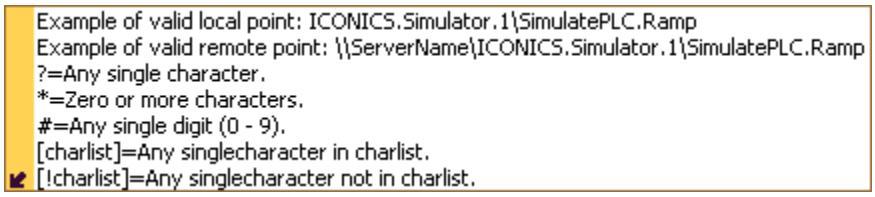


Figure 16.10 - Regular Expression Tooltip

> ***Tip!*** Regular Expression can be a powerful tool. It can help you reduce the number or points, alarms, or files you need to add specifically. For example, if all of your files for the operators group start with the word "operator" you can use an expression "operator*.*" in the allow list. This way you only need to add one line in the allow list instead of adding specific files. Teaching the syntax for Regular Expression is out of the scope of this class; you can find much online help if you wish to learn more about Regular Expressions. We suggest the site http://www.regular-expressions.info/.

You will find an Account Policy tab on both the Group and User properties. You can configure account properties such as Password Age, Length, Uniqueness, Complexity, Account Lockout, and Auto Logout period on this tab. The configuration for each option is intuitive and self-explanatory. You should know, however, that the auto logout period starts counting as soon as the user logs into the system.

Since you can configure an Account Policy in both the Group and User properties, we recommend that you only configure it for a user. This way, you can avoid configuring contradicting policies.

16.3.4. Application Actions

Each GENESIS32 application may supply a static list of securable functions. For example, functions such as adding trend pens in TrendWorX32 or entering configuration mode in GraphWorX32 are commonly disallowed for operators via the security system.

Each GENESIS32 client provides a list of application functions that can be protected through the security system. To configure which users and groups have access to specific application actions, select **Application Actions** from the **Edit** menu in the Security Configurator. This opens the **Actions/User Association** dialog, shown in Figure 16.11.

Figure 16.11 - Assigning Application Actions

The dialog has two tree controls. The parent items in the **Actions** (left) tree control are the GENESIS32 application names. The child items of the application names are the application functions that can be protected. The child items of the application functions are the users and groups that are granted access to the function.

The parent items in the **Users/Groups** tree control on the right are the users and groups defined in the security system. The child items of the users and groups are the GENESIS32 application names. The child items of the application names are the application functions that are allowed for the parent user or group.

To grant access to a single application function to a user or group, only three simple steps are required:

- In the Actions tree control on the left, select the application function (listed under the application folder) to be assigned.
- In the Users/Groups tree control on the right, select the user or group that should have access to the application function selected in the Actions tree.
- Click the "<< Move >>" button near the top of the dialog to add the permission.

You can use a similar procedure to grant access to all application functions of a GENESIS32 client application. Instead of selecting the application function in first step, select application name (the entire application folder). This way, when you click on the "Move" button, it will add the entire application tree, including all functions.

If you changed your mind later on and you want to remove some of the application permissions, you can select the user or group name under that application action in the Actions tree and then press the **Delete** key. You can also select the application name or function under that particular user or group in the Users/Group tree and press the **Delete** key to remove the application functions.

> **NOTE:** This operation never deletes the user, group or application function. Only their association is removed.

Sometimes it may be easier to remove permissions from a user than to individually add all of the required ones. In this case, you can first add all of the available application actions, and then remove the few that you do not want. To do that, simply right-click on a user or group in the right-hand pane of the **Applications Actions** dialog, which will shows a popup menu with two entries:

- **Add All Actions:** Associates all actions with the selected user or group.
- **Remove All Actions:** Deletes the selected user or group from all actions.

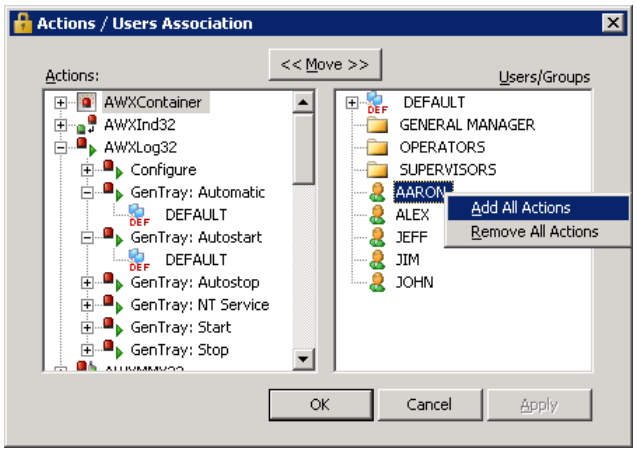


Figure 16.12 - Options to Add or Remove All Application Actions

16.3.5. Global Settings

You can configure some behaviors of the Security Login client (discussed in more detail the "Security Login Client" section) in the Security Configurator. Clicking on the **Edit** menu and selecting the **Global Settings** option in the Security Configurator will launch the **Global Settings** dialog as shown in Figure 16.13.

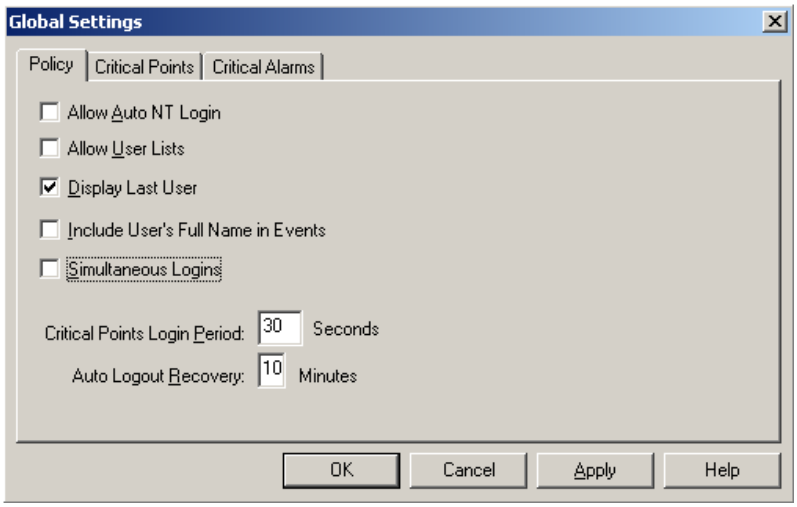


Figure 16.13 - Global Settings Dialog

There are three tabs on in this dialog: Policy, Critical Points, and Critical Alarms. The Critical Points and Critical Alarms tab are similar to the Points and Alarms tab in a User's properties dialog, except when you add a point or an alarm as critical, you will have to re-login each time you want to change the value of the point or acknowledge the alarm. Furthermore, you have to perform the action within the specified "Critical Points Login Period"; otherwise, you will need to login again to perform the actions.

On the Policy tab is where you can configure Security Login Client behaviors as well as the security server behaviors. You can configure the following options on the Policy tab.

- **Allow Auto NT Login**: when checked, the windows user will be automatically logged into ICONICS Security when the user logs into Windows. This option only applies when you are using the Integrated NT security mode.
- **Allow User Lists**: when checked will show a list of configured user in the Security Login client. The user can choose a user from a dropdown list to login with; this can prevent some user errors during the security login process.
- **Display Last User**: This option is checked by default, which will show the last logged-in user on a machine in the Security Login client.
- **Include User's Full Name in Events**: will show the user's full name (as opposed to just the username) in the GenEvent when checked. All security actions are posted to the GenEvent server.
- **Simultaneous Login**: when enabled, will allow multiple users to log into the security server at the same time. The default behavior is such that it will only allow one user to log in. When a second user tries to log in, the system will log out the first user and then log in the second user. When you have more than one user logged into the system, the system will allow you to perform actions available to both users.

- **Critical Point Login Period (Seconds)**: indicates the number of seconds in which you have after logging in to change a process point's value or acknowledge an alarm for those points and alarms listed as critical.
- **Auto Logout Recovery (Minutes)**: indicates the amount of time after all security requests from a computer have ceased to log out users logged in from that computer.
- **NT Synchronization Period**: indicates how often the security system should query the Active Directory to receive and update. This field is only visible when using the Integrated NT Security mode.

16.4. Security Login Client

Once you have setup a security configuration, you can test it with the respective application. You can launch the Security Login client from the start menu or within a viewer in runtime mode such as GraphWorX32. The option to launch the login dialog in a viewer is generally in the Tools menu; you can choose the "Security Login…" option to launch it.

You can also use the start menu option to log in by going to Start → Programs → ICONICS Tools → Security Login to launch the dialog. You can type in the username or select it from the dropdown list if you enabled the "Allow user Lists" option in the security configuration.

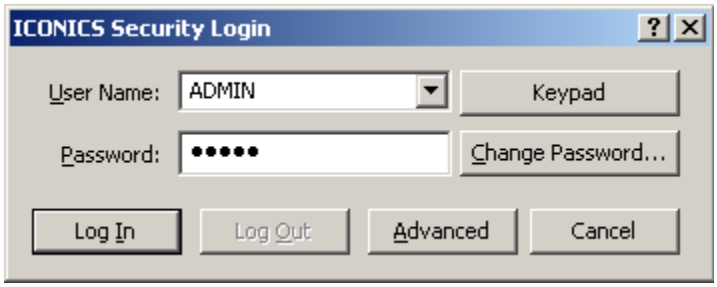


Figure 16.14 - ICONICS Security Login

You can launch a keypad from this dialog if the computer does not have a keyboard plug into it. Aside from logging users in and out of the system, you can also use the dialog to change a user's password and enter the Advanced Security dialog.

> **NOTE:** The user name will always be capitalized; therefore, the keyboard will not show you a "CapsLock" key. The Password filed is case-sensitive, however, and launching the keypad from with the cursor in the Password field will show the "CapsLock" button.

Figure 16.15 shows the Advanced ICONICS Security Login dialog. From this dialog, you can see the location of your security server, when it started, and where the location of the security server configuration file. The dialog can further tell you who is logged into the system, the name of the machine where the user logged into the system with, when the user logged in, and when the user will be logged-out based on the auto logout configuration.

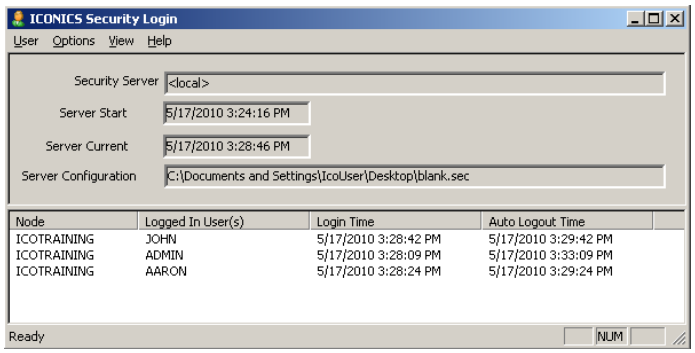


Figure 16.15 - Advanced ICONICS Security Login

16.5. The Security Indicator ActiveX Control

While the Advanced ICONICS Security Login dialog can show you much information about your security system, you may not always be on a system with GENESIS32 installed to gain the necessary information. In such cases, you can use the Security Indicator ActiveX Control.

You can find this control in the ActiveX Toolbar in GraphWorX32. Clicking on the "ICONICS Security Indicator ActiveX" button will put the control on the display. You can double-click on the control to bring up the properties dialog of the control. By default, it is configured to show you a list of logged in users and the Security Server Node. You can change this in the properties dialog as well as the font and color used by the control. When you go into runtime, the control will display the logged in user information as shown in Figure 16.16.

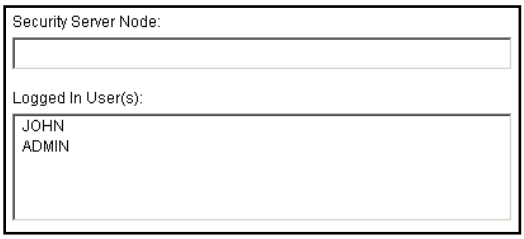


Figure 16.16 - Security Indicator ActiveX Control in Runtime

NOTE: You will only see the security server node name if the security server is on a remote machine.

Lab: Security L16

For many projects, it is necessary to add security to the application. The project we have been building for Acmeseutical is no exception. Acmeseutical would like to restrict configuration access to their operators and only allow a few privileged individuals to access the History Trends and History Alarms displays. In this lab, we will add security permissions to our project.

Estimated time to complete this lab: 30 minutes

16.1. Task Checklist

In this lab, you will complete the following tasks:

- Create a new Security Server configuration
- Modify the Default security group
- Create new groups and users
- Assign user and group permissions
- Assign application rights to users and groups
- Test the newly configured security in runtime

16.2. Support Files

You will need the following file(s) to complete this lab:

- ACMEPillMachine.gdf
- ACMEHistoryAlarms.gdf
- ACMEHistoryTrends.gdf
- ACMEMixers.gdf
- Acmeseutical_DWXConfig.mdb

16.3. Security Configuration

1. Open the ICONICS Security Configurator from Start → Programs → ICONICS Tools → Security Configurator.

2. Since this is the first time you will open the Security Configurator, you will see the **Security Server** dialog to help you create a new configuration.

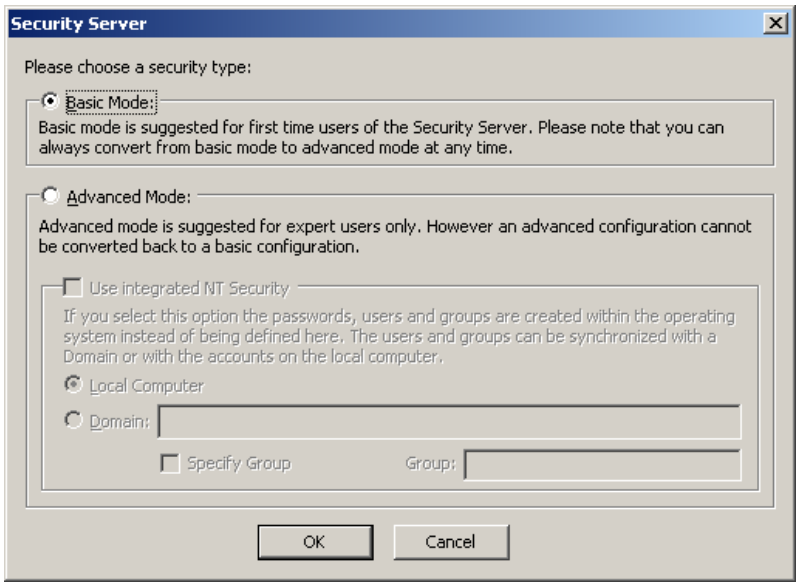


Figure L16.1 - Security Server Dialog to Create a New Configuration

3. We will create an advanced configuration with specific GENESIS32 security users.

4. Choose the “Advanced Mode” radio option and click on “OK.”

5. The **Save As** dialog appears. Save the file in C:\Training and call this file “ACMESecurity.sec.” Click on “Save” to create the new security file.

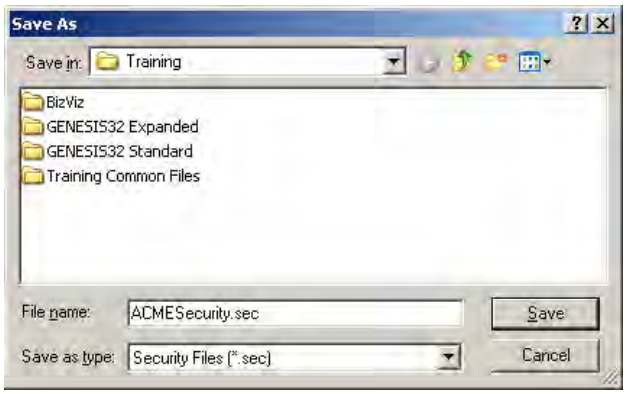


Figure L16.2 - Security File

6. The **ICONICS Security Server** Configuration window will open**,** as depicted in Figure L16.3.

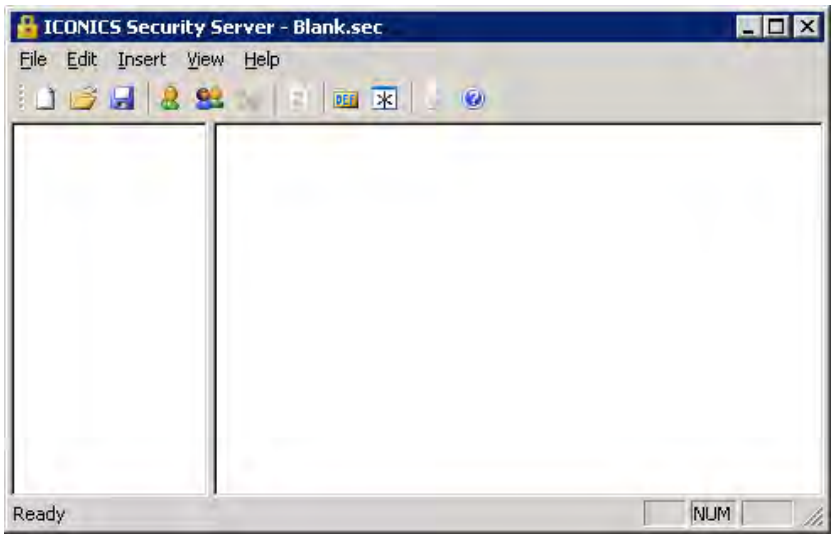


Figure L16.3 - Empty ICONICS Security Server Configuration

16.3.1. Modifying the Default Group

1. Select Edit → Default Group to open the **Properties for Group: DEFAULT** dialog.

2. Select the Points tab. Select the ***** in the Allow writing list and click on "Delete," as depicted in Figure L16.4, to remove rights to write to OPC tags when nobody is logged on to the system.

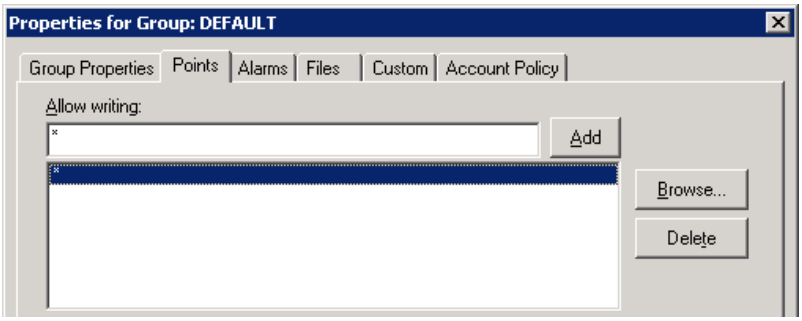


Figure L16.4 - Removing Write Access for the Default Group

3. Select the Alarms tab. Select the ***** in the Allow writing list and click on "Delete" to remove alarm acknowledgement rights when nobody is logged on to the system.

4. Select the Files tab. Enter "ACMEMixers.gdf" in the Deny Access list and then click on "Add" to deny access rights to this display when nobody is logged on to the system.

5. Enter "*.*|Secured" in the Deny Access list, and click on "Add" to deny access rights to any layer called Secured on all displays when nobody is logged on, as shown in Figure L16.5.

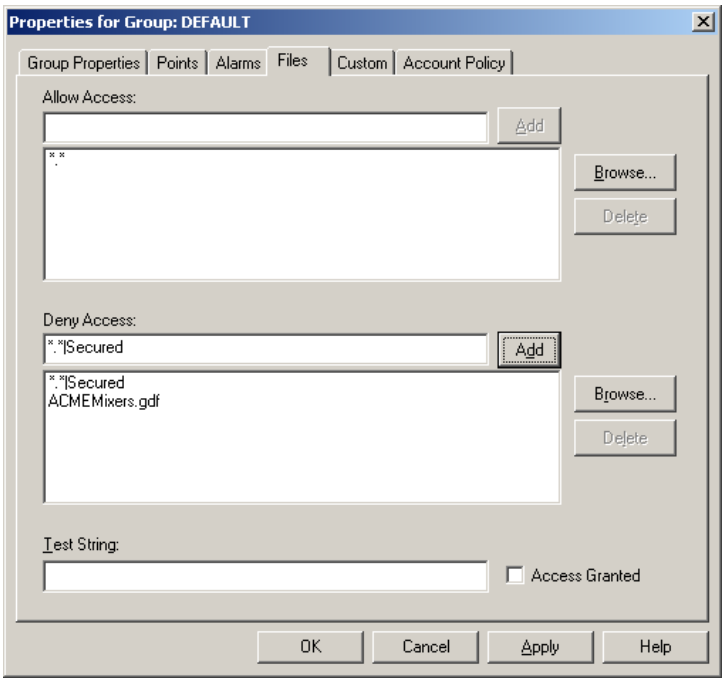


Figure L16.5 - Denying Display Access for the Default Group

6. Select the **Account Policy** tab, as depicted in Figure L16.6.
7. Set the Maximum Password Age to "Password Never Expires".
8. Set Minimum Password Age to "Allow Changes Immediately".
9. Set Minimum Password Length to "At least 1 Character".
10. Set Password Uniqueness to "Do Not Keep Password History".
11. Set Account Lockout to "No Account Lockout".
12. Set Auto Logout to "Never Logout".

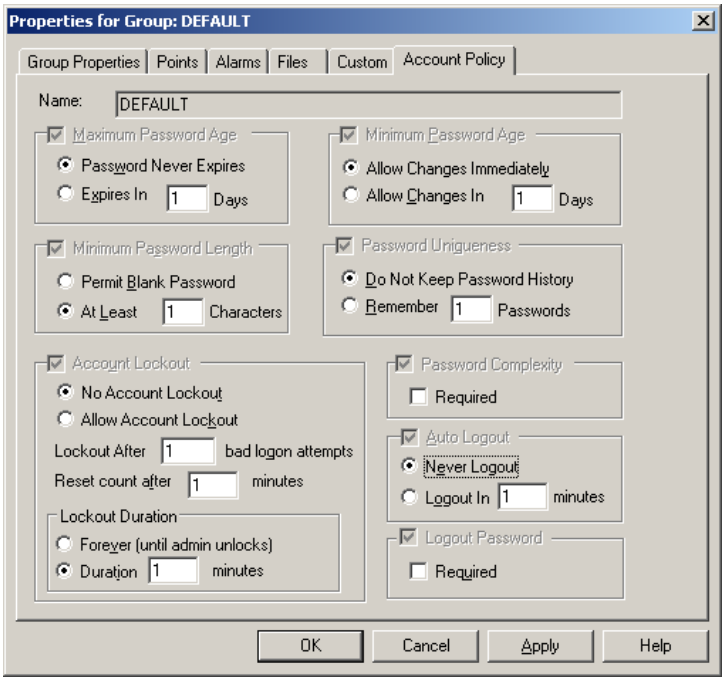


Figure L16.6 - Account Policy for the Default Group

13. Click on "OK" to commit the changes to the default group.

16.3.2. Creating New Groups and Users

1. Click on the "New Group" button on the toolbar, or select Insert → New Group, to insert a new group into the configuration and to open the **Properties for Group: NEW GROUP** dialog.

2. Set the Group Name to "OPERATORS" and click on "OK" to save the group.

3. Repeat steps 1 and 2 to create two more groups called "ENGINEERS" and "ADMINISTRATORS." When finished, your configuration should look like Figure L16.7.

Figure L16.7 - Groups Created

4. Click on the "New User" button on the toolbar, or select Insert → New User, to insert a new user into the configuration and to open the **Properties for User: NEW USER** dialog.

5. Set the User Name to "FRED," and set the Password to "fred" (without quotes). Retype the password in the Verify Password textbox.

6. Uncheck the "Account Disabled" option as shown in Figure L16.8, and click on "OK" to save the user.

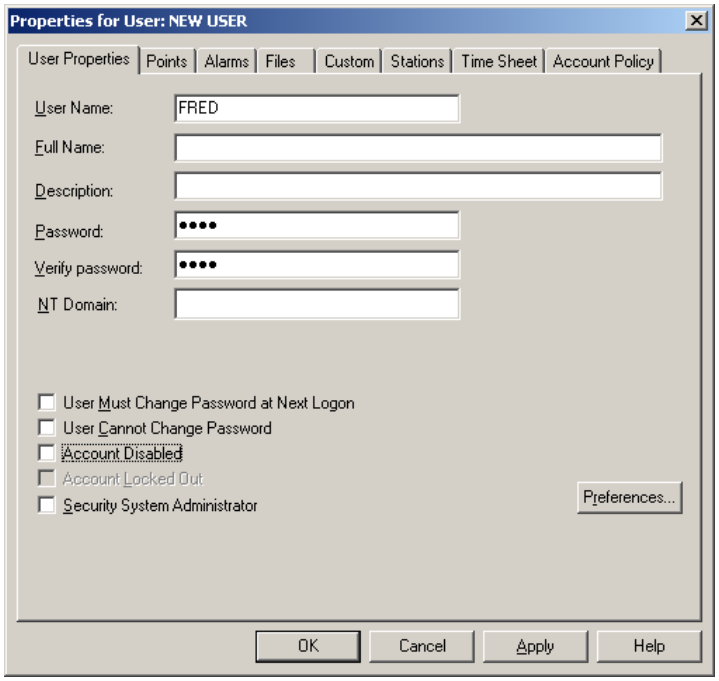


Figure L16.8 - Creating a New User

7. Follow these same steps to create the following users with the following properties shown in Table L16.1.

Table L16.1 - Users and Properties

User	Password	Account Disabled	Security System Administrator
FRED	fred	Unchecked	Unchecked
BETTY	betty	Unchecked	Unchecked
BARNEY	barney	Unchecked	Unchecked
WILMA	wilma	Unchecked	Checked

8. Click on the "OPERATORS" group in the left-hand pane, and then click on the user "FRED" on the right-hand pane.

9. Click on the "Associate Users with Group" button, or select Insert → Associate Users and Group, to assign the user FRED to the group OPERATORS. This is shown in Figure L16.9.

Figure L16.9 - Associating Users and Groups

10. Repeat Step 8-9 to associate the following users with the groups listed in Table L16.2.

Table L16.2 - User and Group Association

User	Group
BETTY	OPERATORS
BARNEY	OPERATORS
BARNEY	ENGINEERS
WILMA	OPERATORS
WILMA	ENGINEERS
WILMA	ADMINISTRATORS

11. The final user and group configuration should now look like the one shown in Figure L16.10.

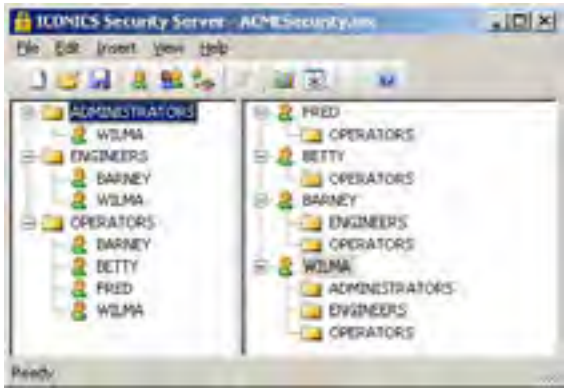


Figure L16.10 - User and Group Associations

16.3.3. Assigning Permissions to Groups

1. In the tree control, double-click on the "OPERATORS" group to open its properties dialog.

2. Select the Points tab. Enter ***** in the Allow writing list and click on "Add" to provide write access permissions to all OPC points when a user from the OPERATORS group is logged on, as depicted in Figure L16.11.

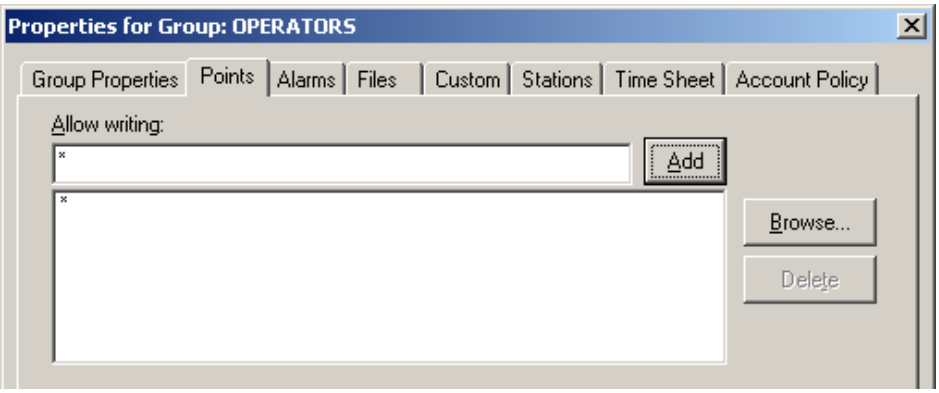


Figure L16.11 - Granting Write Access to OPERATORS Group

3. Select the Alarms tab. Enter ***** in the Allow Ack list and click on "Add" to provide alarm acknowledge permissions when a user from the OPERATORS group is logged on to the system.

4. Select the Files tab. Enter "ACMEMixers.gdf" (without quotes) in the Allow Access list as shown in Figure L16.12, and click on "Add" to provide access permissions to this display when a user from the OPERATORS group is logged on to the system.

5. Click on "OK" to commit the changes.

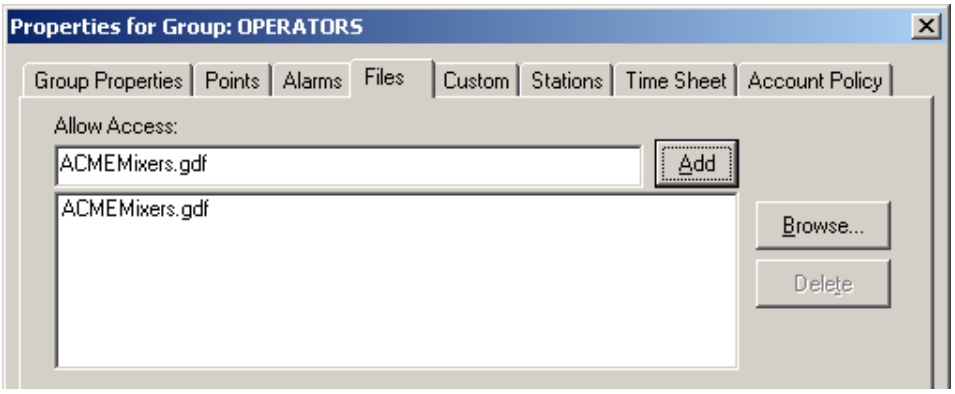


Figure L16.12 - Granting Display Access to the OPERATORS Group

6. Double-click on the “ENGINEERS” group to open its properties dialog.

7. Select the Files tab. Enter “*.*|Secured” (without quotes) in the Allow Access list as shown in Figure L16.13, and click on “Add” to provide access permissions to this layer on all displays when someone from the ENGINEERS group is logged on to the system.

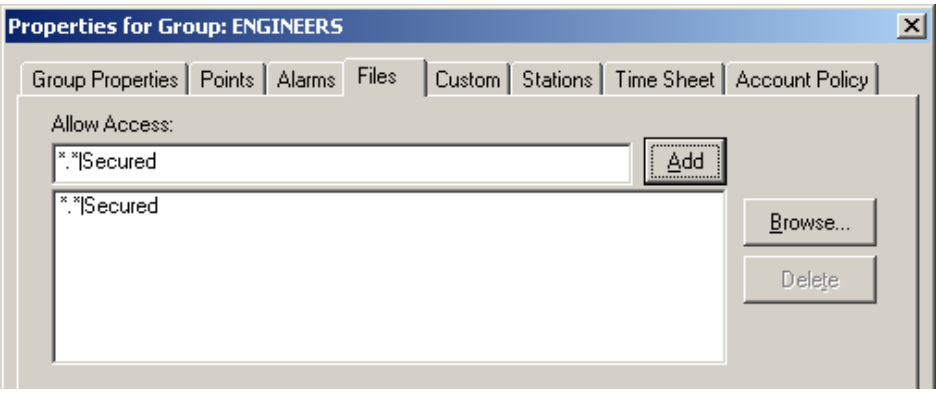


Figure L16.13 - Granting Layer Access for the ENGINEERS Group

8. Click on “OK” to apply the changes close the dialog.

16.3.4. Assigning Application Rights to Groups

1. Select Edit → Application Actions to open the **Actions/Users Association** dialog.

2. In the Users/Groups pane on the right, expand DEFAULT → GWX32.

3. Select “Exit Application” and then press the “Delete” key on your keyboard. Then select “Menu: Exit Runtime,” and press the “Delete” key on your keyboard.

4. Now these actions have been denied from the DEFAULT group, as depicted in Figure L16.14.

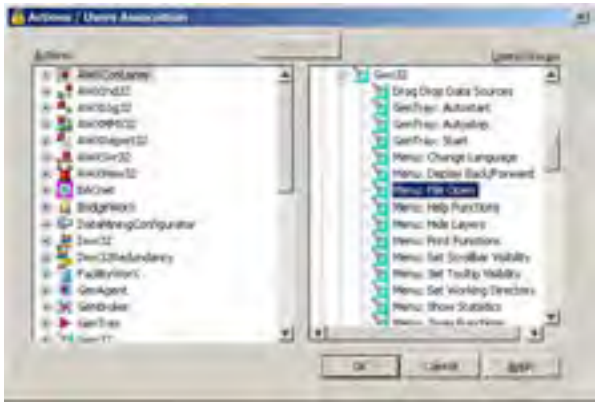


Figure L16.14 - DEFAULT GROUP Action Associations

5. In the Users/Groups pane on the right, collapse DEFAULT and select ADMINISTRATORS.

6. In the Actions pane on the left, expand GWX32.

7. Select “Exit Application” and click the “<< Move >>” button at the top of the dialog. Then select “Menu: Exit Runtime” and click the “<< Move >>” button.

8. Now these actions have been assigned to the ADMINISTRATORS group, as depicted in Figure L16.15.

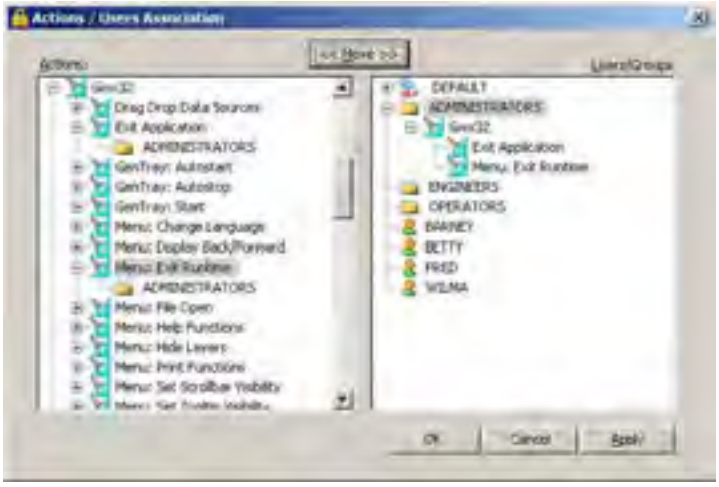


Figure L16.15 - ADMINISTRATORS Actions Associations

9. Click on “OK” to close the dialog.

10. Click on the “Save” icon in the toolbar to save the configuration.

11. You may close the security configuration if you wish.

16.3.5. Using Security During Runtime

1. In Windows Explorer, browse to your C:\Training directory, and double-click on ACMEPillMachine.gdf to open the display.

2. Choose **Runtime** from the menu to show the display in runtime mode.

3. Mouse over the Production Control and see that, in the tooltip, it says “Security Write Denied.”

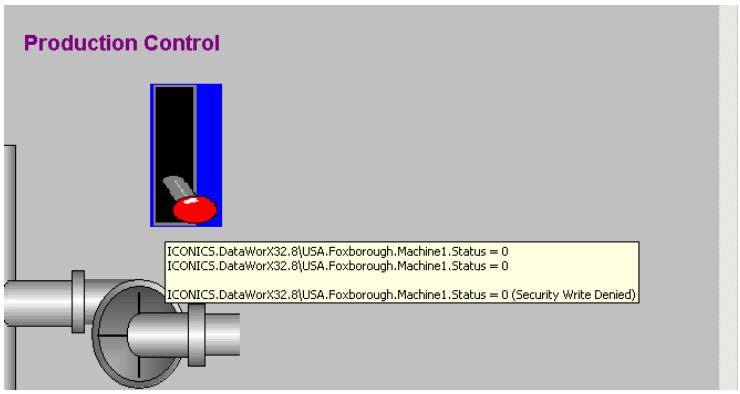


Figure L16.16: Security Write Denied

4. Click on one of the Mixers to open the popup window. The security warning appears, as shown in Figure L16.17.

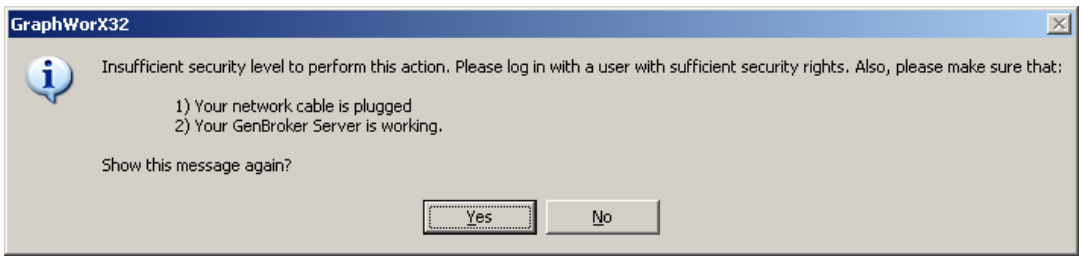


Figure L16.17 - Insufficient Security Level

5. Click on "Yes" to show the message again. The **ICONICS Security Login** dialog opens as a result.

6. For the User Name, enter "BETTY," and for the Password, enter "betty" (both without quotes).

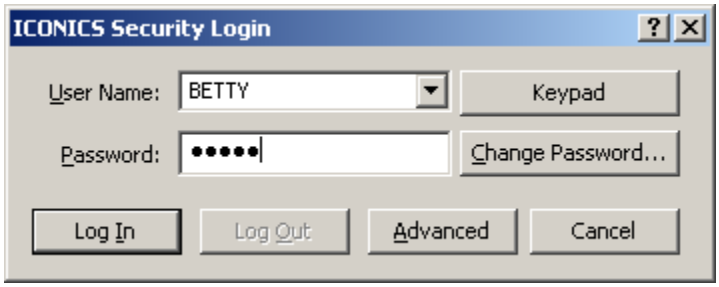


Figure L16.18 - ICONICS Security Login Dialog

7. Click the "Log In" button.

8. After logging in, it becomes possible again for the user to start/stop the production and open the ACMEMixers display.

9. Select Tools → Security Login from the GraphWorX32 Runtime menu to open the **ICONICS Security Login** dialog.

10. For the User Name, enter "BARNEY," and for the Password, enter "barney" (both without quotes). Click on the "Log In" button.

11. After logging in, the "History Trend" and "History Alarms" buttons should become visible on the display, because the user BARNEY has access to the Secured layer.

12. Select "Configure" from the GraphWorX32 Runtime menu. The message in Figure L16.17 will become visible again, because the user BARNEY does not have permissions to return to configuration mode.

13. Click on "Yes" to show the message in the future. This should also open the **ICONICS Security Login** dialog.

14. For the User Name, enter "WILMA," and for the Password, enter "Wilma" (both without quotes). Click the "Log In" button.

15. After logging in, the "Configure" menu has once again become useable.

16. Since we do not want to have the menu bars on our display, we can add a button to log into without going to Tools → Security login.

17. In configuration mode, select Format → Display Properties, then select the Window tab, and uncheck "Show Runtime Menubar," then click on "OK" to close the dialog.

18. Open the Symbols library if it is not already opened (View → Symbol Toolbar), and look for the Login Button in 3_VBScriptSymbols →WebHMI SecurityLogin.sdf. Drag the button into the display and place it where you would like it to go.

19. Save the display and go into runtime. Click on the login button to log in as other users.

Review: Security R16

16.1. Chapter Objectives

In this chapter, we discussed how ICONICS protects your system. You can use ICONICS Security to restrict user access to process points, alarms, files, and much more. By now, you should know how to do the following:

- Create a security configuration file
- Configure Groups and Users in the Security Configurator
- Allow or Deny permissions to points, alarms, and files
- Configure Account Policies
- Add or remove application actions from a group or user
- Restrict access of a "guest" user
- Login into the ICONICS Security Server
- Retrieve a list of logged in user
- Allow multiple users to log into the system simultaneously
- Configure the Login Client to show a list of users

16.2. Summary

ICONICS understands that it is very important to protect data within your organization. Using the ICONICS security, you can restrict access to your system. ICONICS security uses a server-client model, where you configure the security server and all ICONICS applications are security clients that look to the server when a user tries to interact with an application.

When you configure Security, your configuration can be in one of three modes: basic, advanced, and Integrated NT. If this is the first time you are configuring security, you may choose to create a basic configuration, otherwise, you may choose to create an advanced security configuration. If you want to import security users from Windows (either local machine users or domain users), you will need to create the security configuration in Advanced mode and choose the "Use integrated NT Security" option.

Once you create a security configuration, you can add permissions to groups and users. Users within a group will inherit permissions from the group. ICONICS security uses the idea of a Default Group to allow guest users to view information.

If you created an advanced security configuration, you should first limit the access on the Default Group. By default, this group is configured to allow access to everything, which means everyone has access to the entire system.

You can also configure application actions in security. Application Actions are functions provided by a particular application. For example, an application action could be to open the AlarmWorX32 Server Configurator. You can add or remove these actions from a user or group.

You can also protect the security configuration by creating a security administrator. As soon as you create such a user, you will need to provide a username and password when you try to launch the security Configurator. Failure in providing valid information will prohibit you from opening the application.

Once you configure security on a system, you can log into the security server by using the Security Login client. You can also view a list of logged-in users by using the Security Indicator ActiveX Control in GraphWorX32.

16.3. Terms

Application Actions	Default Group	Regular Expression
Security Administrator	Security Client	Security Configurator
Security Indicator	Security Login Client	Security Modes
Security Server		

16.4. Test Your Knowledge

In this chapter, we discuss ICONICS Security in details. You should now be able to answer the following questions.

1. Which of the following is NOT a valid security-mode that you can create in a security configuration?

    a. Basic
    b. Advanced
    c. NTLM
    d. Integrated NT

2. True or False: A user can log into ICONICS security when they log into Windows.

3. True or False: You can configure guest user permissions by creating a user name “Guest”.

4. Which of the following can you customize in a security configuration?

    a. Define Account Policy
    b. Define Critical Points
    c. Configure Access to Files
    d. Restrict which computer has access
    e. All of the Above
    f. None of the Above

5. How can you protect your security configuration from unauthorized access?

    a. Use the "Set Configuration Mode Password" option protect it with a password
    b. Create at least one Security System Administrator such that one will need to provide the administrator username and password to open the configuration
    c. Use windows security to restrict access to the security configuration file
    d. Zip up the security configuration file and password protect the zip file

6. True or False: The security auto-logout period starts counting as soon as the user logs into the system.

7. True or False: A user can change their password using the ICONICS Security Login dialog.

For questions 8-10, please refer to security configuration shown in Figure R16.1. Further, assume that Opeartor1 is part of the Operators group.

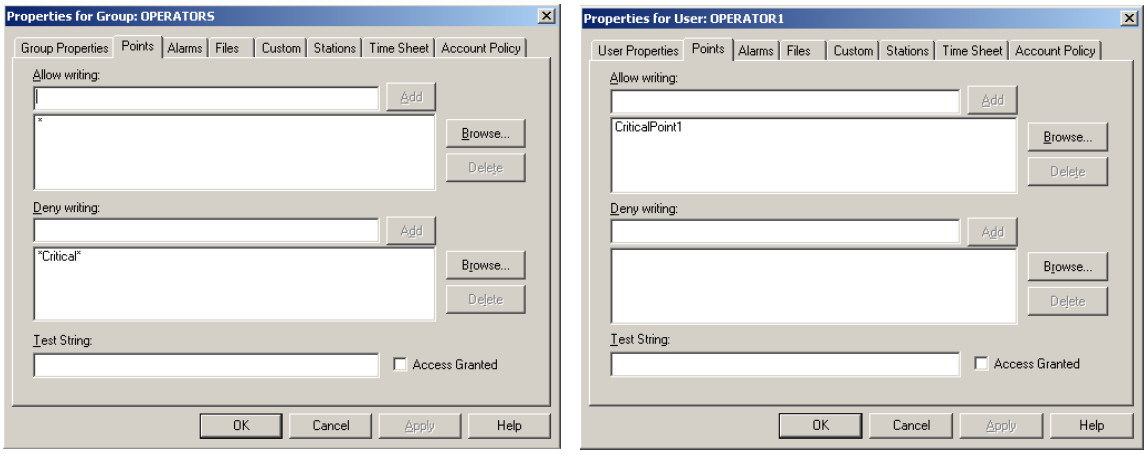


Figure R16.1 - Operators Group (left) and Operator1 (right) Security Permissions

8. True or False: Opeartor1 will be able to write values to a process point called "CriticalPoint1".

9. Which process points will Opeartor1 be able to access?

    a. Only CriticalPoint1
    b. Every point but CriticalPoint1
    c. Every point accept those containing “Critical” in the point name
    d. Every point with no exception

10. What does *Critical* in the Group permission Deny writing section represent?

    a. Deny access to any point containing the string “Critical”
    b. Deny access to any point with exactly one character before and after the string “Critical.”
    c. Deny access to any point with exactly one digit before and after the string “Critical.”
    d. Deny access to the point called “*Critical*”

Additional Applications 17

In addition to all the already-configured applications such as GraphWorX32, TrendWorX32 SQL Data Logger, and the AlarmWorX32 Server and Logger, even more applications are provided with GENESIS32. Each of these applications has their own significance. We will discuss a few of these applications in this chapter. By the end of the chapter you should be familiar with the following applications:

- GenBroker
- GenTray
- MonitorWorX
- ProjectWorX32

17.1. GenBroker

DCOM is the method of communication for GENESIS32. There are however, certain disadvantages to using DCOM. To overcome DCOM timeout problems, for example, it was necessary to eliminate Windows callbacks and reduce the number of different function calls over Windows DCOM communications.

DCOM's major disadvantages are:

- Its complex settings often make its configuration ineffective when the client is remote.
- It does not work well across NT domains.
- Its connection over slow and unreliable networks (e.g., the Internet) is practically impossible.
- It is not easy to configure to pass through firewalls.

In response to these difficulties, ICONICS has enhanced GENESIS32 with GenBroker, a proven, high-speed communications utility. Employing a network of applications GenBroker uses TCP/IP and SOAP/XML channels to achieve real-time and secure communications between Web browser clients and WebHMI servers. Everything except the method of communication remains the same as before.

17.1.1. Communication Types

GenBroker uses the following basic methods of networked communications:

- OPC Direct Channels
- OPC over DCOM Direct Channels
- OPC over TCP/IP Direct Channels
- OPC over SOAP/XML Direct Channels
- Indirect channel via a mediator node

Most channels, either direct or indirect, translate the OPC COM calls into one of two protocols and transmit them to the remote node where GenBroker can execute them. (The OPC Direct channel is the only exception; it uses OPC calls.) Protocols that every channel can use for these transmissions are:

- **DCOM:** Suitable for local networks enabled with DCOM. Not recommended for Internet use.
- **TCP/IP:** Communicates over intranet/Internet through routers and firewalls.
- **SOAP/XML:** Communicates over intranet/Internet through routers and firewalls.

NOTE: In order to use SOAP/XML channels in GenBroker, you must first install Microsoft Internet Information Server (IIS) on the GenBroker Server node. For more information, please refer to the GenBroker Help documentation.

17.1.2. Administrative Servers

Aside from providing a superior communication method compared to DCOM, GenBroker also allows you to configure Administrative Servers. Any GENESIS32 system contains five administrative servers:

- Events
- Global Aliasing
- Language
- Licensing
- Security

By default, these servers are configured on the machine where you installed GENESIS32. However, you can point these servers to a remote location. For example, if you have many different GENESIS32 Browser machines running you will need to have a separate machine to serve the browser stations license information.

In such cases you can point the Licensing server on a particular browser machine to a remote licensing server machine. Similarly, you can centralize your security, Global Aliasing, Language Aliasing, and GenEvent servers on one machine.

To configure a remote administrative server, simply open the GenBroker Configurator by going to **Start → Programs → ICONICS Tools → GenBroker Configurator**. The system comes with a default GenBroker configuration; you can either edit this one or create a new one of your own.

The GenBroker Configurator opens to the Administrative Servers page. The default configuration for all administrative servers is set to use "<local>", meaning that all of these servers are on the local machine.

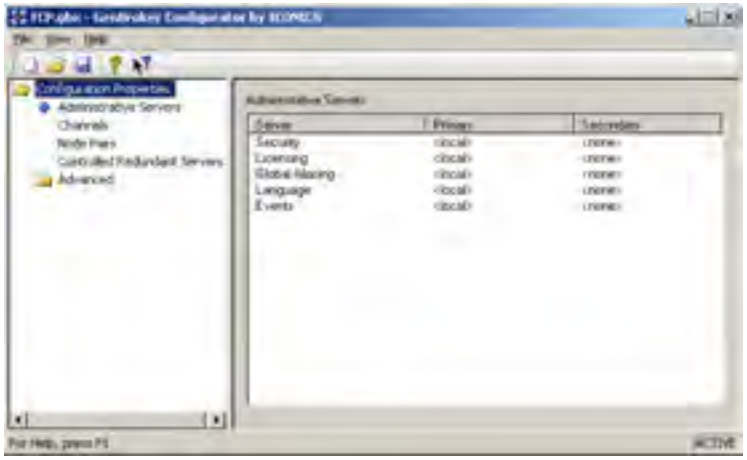


Figure 17.1 - GenBroker Configurator: Administrative Servers

You can double-click on any of the servers listed on the right-hand pane to bring up a configuration dialog for that server. Then in the popup dialog choose the appropriate remote machine name from the dropdown list. If your network contains many machines, it may be faster to type in the computer name or IP Address instead of selecting from the dropdown list. If you do choose to type in value, make sure that there are no spelling errors.

> **NOTE:** If the administrative server resides on the local machine you should leave the value as "<local>". Changing this value to the local machine name or IP address could result in high CPU use by GenBroker.

Notice also that you can provide a secondary node for each of the administrative server. Indeed, all of these servers have built-in redundancy capability such that if the primary server fails it will try to contact the secondary server to obtain the information. If you do not have a secondary server machine, simply leave it as "<none>".

17.1.3. Serving and Retrieving Data

Regardless of whether you are trying to serve licensing information or OPC data to a client computer, you will need to prepare the server machine and the client machines. Figure 17.2 shows a basic network diagram for a server and two clients. The **Server Node** is usually the machine where the administrative servers and OPC servers reside. The **Client Nodes** are the computers where you want to receive information from the administrative servers and OPC Servers.

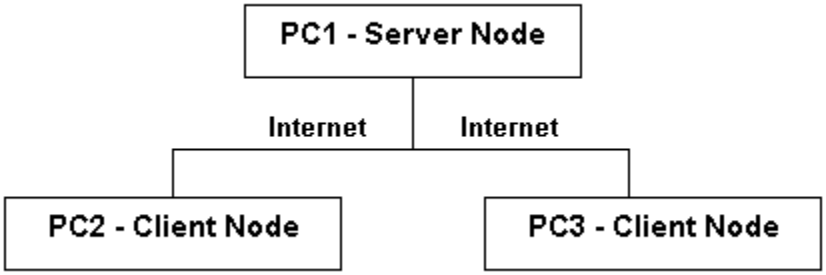


Figure 17.2 - Basic Network Configuration

In general, no configuration is required on the server machine. You can simply use the default GenBroker configuration settings. However, you must make sure that GenBroker is running on the server machine. In recent versions (version 9.13 and above), GenBroker runs as an automatic service so that it starts up when the server node starts. You can manually start and stop the GenBroker server in GenTray, MonitorWorX, or the Windows Services Manager.

You should configure GenBroker on client nodes. The GenBroker Configuration contains information on how to connect to the server machine and which machines are your administrative servers; therefore, we always need to configure GenBroker on a client. As with GenBroker on the server machine, the client comes with a default GenBroker Configuration that sets all the Administrative servers to the local machine and the default communication channel to **OPC over TCP/IP Direct Channels**. You should change the administrative server locations and the default communication channel accordingly on the client machine.

17.2. GenTray and MonitorWorX

GENESIS32 provides applications for you to control the startup of different applications and monitor their status. There are two applications that can help you achieve this.

First, GenTray (which is sometimes referred to as the GENESIS32 Tool Tray) allows you to configure your application to run as services as well as provide the order to start and stop applications. In addition to allowing a specific order for applications to startup, you can also add-in delays for each application, giving the applications ample time to complete the startup process before launching a different application.

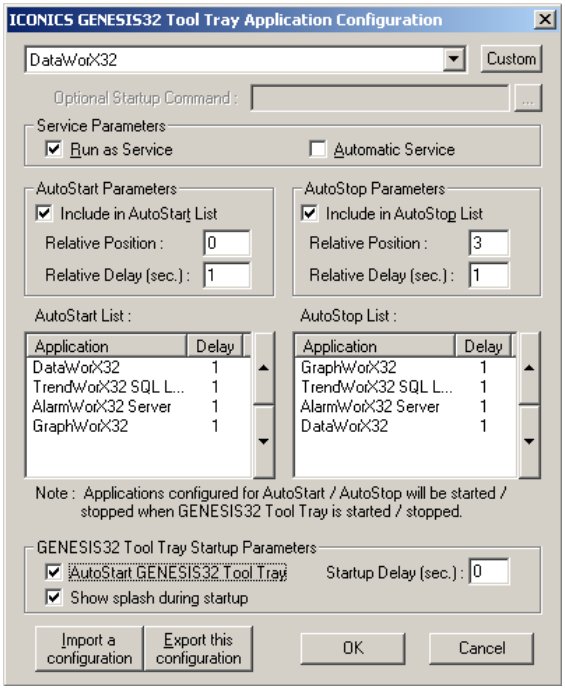


Figure 17.3 - ICONICS GENESIS32 Tool Tray

If you want GenTray to automatically launch applications for you after you log into the machine, you will need to check the “AutoStart GENESIS32 Tool Tray” option at the bottom. This will start GenTray after a user logs into the computer, which in turn starts the applications that you have configured to start.

On the GenTray popup menu in the system tray, you can view and set the status of your applications. In Figure 17.4, you can see that GenBroker is set as an automatic NT Service. Moreover, since the “Start” option is grayed-out and the “Stop” option is enabled, we know that GenBroker is running.

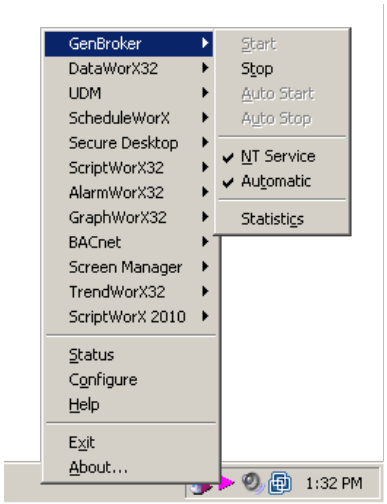


Figure 17.4 - GenTray Context Menu

You can obtain similar application information from MonitorWorX. Figure 17.5 shows the MonitorWorX dialog. On the Runtime tab, you can view the different applications or services that are running on the machine. If you select one of the applications in the left-hand pane, more information about that application displays on the right.

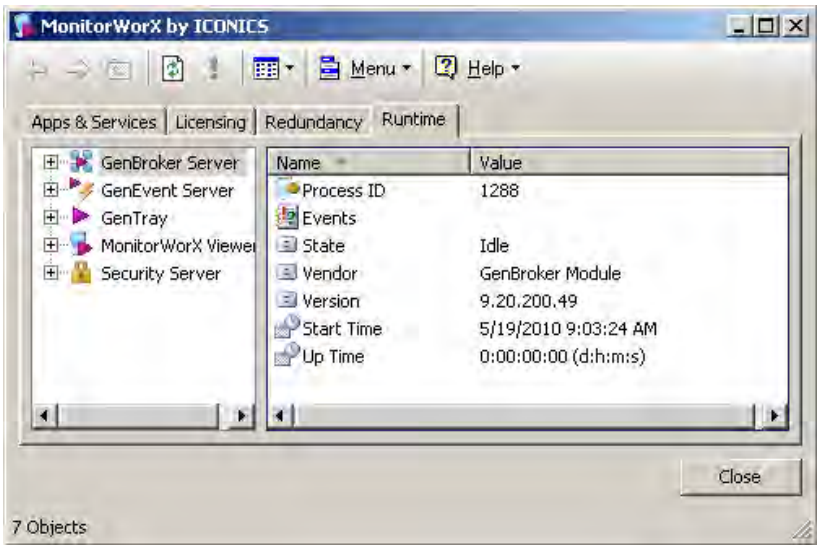


Figure 17.5 – MonitorWorX

Since MonitorWorX is designed to monitor the status of ICONICS applications, it naturally became the application that keeps track of the active server in cases of redundant setups. If you have configured ICONICS Professional Redundancy, you will use the options on the Redundancy tab to monitor the different servers and use it to delegate the active server.

17.3. ProjectWorX32

GENESIS32 contains numerous applications with multiple components requiring a well-trained user in order to successfully create, integrate, deploy, and manage projects. Each of these components requires complex setup of many different types of configuration files, databases, and OPC servers, as well as runtime and security settings. To simplify the management and deployment of the broad spectrum of GENESIS32 applications, ICONICS has introduced ProjectWorX32, a sort of "super configurator" that integrates all GENESIS32 applications into a single, easy-to-manage format. The ProjectWorX32 user interface is basically a container embedded in the Microsoft Management Console (MMC). The ProjectWorX32 console conveniently consolidates all of your GENESIS32 files in one easily accessible location, enabling you to:

- Organize all GENESIS32 application files into separate projects.
- Create new GENESIS32 application files and OPC server configurations.
- Import application files into projects.
- Start and stop GENESIS32 applications.
- Pack project files into a single, compressed file, and unpack project files for easy deployment to multiple computers.
- Activate entire projects into runtime mode.
- Control the layout of windows using the Screen Manager.
- Configure security settings to password-protect projects.
- Back up project files to Microsoft Visual SourceSafe.
- Search for and replace machine names and OPC tags throughout all project files using a global find/replace utility.
- Control and monitor runtime functions for all applications using GenTray.
- Launch GENESIS32 system tools.
- Import and manage third-party applications.
- Publish projects to the Web.
- Generate comprehensive reports for all files and projects.

NOTE: Version 9.20 introduced Workbench32, which provides similar functionality along with the ability to configure your project over the Web. For more information about Workbench32, please refer to the help files.

17.3.1. System Tools

You can open ProjectWorX32 by going to **Start → Programs → ICONICS GENESIS32 → ProjectWorX32**.

The System Tools tree of the ProjectWorX32 console is a snap-in from which you can conveniently launch all of the available GENESIS32 tools including:

- GenTray
- License Monitor
- OPC DataSpy
- OPC Simulator
- Screen Manager
- Secure Desktop Configurator
- Security Configurator
- Security Login
- TraceWorX32

Figure 17.6 shows the user interface of the System Tools tree of the ProjectWorX32 console. To launch a tool, double-click on the tool in the right-hand pane. We have discussed some of these tools in this class; for information on those tools not mentioned here refer to the ProjectWorX32 Help files or to the another training classes offered by ICONICS.

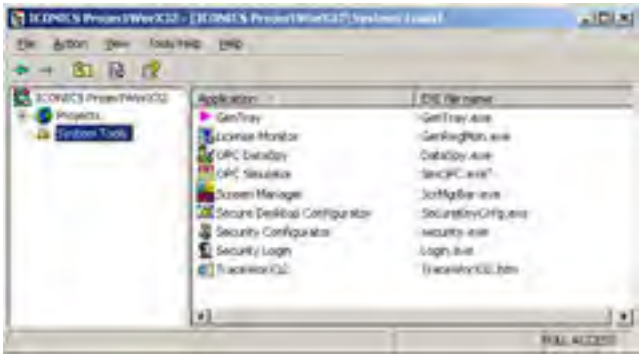


Figure 17.6 - System Tools in ProjectWorX32

17.3.2. Managing and Running Applications

You will always start by creating a new project in ProjectWorX32 from which you can create and manage all of your ICONICS applications directly. In the tree control of ProjectWorX32 (left-hand pane), you can right-click on **Projects** and select **New Project**, or go to **Action → New Project**. Once a new project is created, management capabilities for all ICONICS applications are automatically added.

The Applications tree for each project, shown in Figure 17.7, contains all GENESIS32 client applications and components, including GraphWorX32, AlarmWorX32, TrendWorX32, and GenBroker just to name a few. Each of these applications has several additional modules or components (e.g. AlarmWorX32 Container, Logger Configurator, Server Configurator, and Multimedia Configurator).

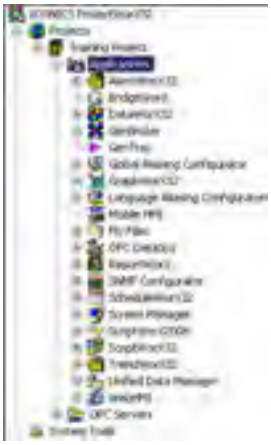


Figure 17.7 - ProjectWorX32 Applications Tree

For each component you can create a new file, add a file to the project, add a file in the project to Microsoft SourceSafe, or generate a report.

17.3.2.1. Creating a New Application File

Each time you create a new application file inside a project in the Applications tree of the ProjectWorX32 console, the new file is saved to the main project directory. Although the file types vary with each GENESIS32 application (e.g. display files, Microsoft Access configuration databases, etc.), the method for creating new application files is generally the same for each application in the ProjectWorX32 console. To create a new application file, you can follow the steps below:

1. Right-click on the application folder
2. Choose “New [File Type]”
3. Give the file a name and click on “OK”

Figure 17.8 - Creating a New Application File

NOTE: Microsoft SQL Server configuration files are handled differently.

17.3.2.2. Adding an Application File to a Project

When you add an existing application file to a project in the Applications tree of the ProjectWorX32 console, you have the option of copying the file to the project folder or running the file from its current location. Although the file types vary with each GENESIS32 application, the method for adding existing application files is generally the same for each application in the ProjectWorX32 console. To add an existing application file to your project, you can follow the steps below:

1. Right-click on the application folder
2. Choose “Add [File Type]”
3. Browse to the file and click on “OK”
4. Choose to make a local copy of the file in the project folder or create a reference to the original

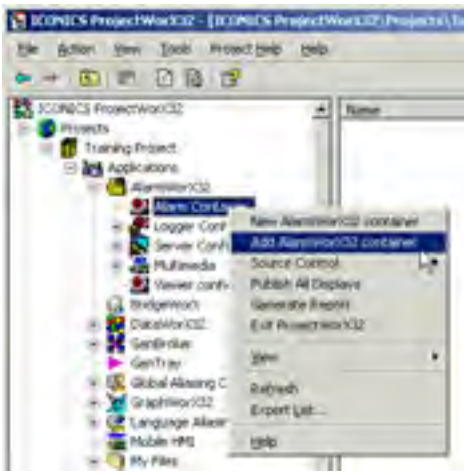


Figure 17.9 - Adding an Application File

If you manually copy a file to a project folder in Windows Explorer (i.e. C:\Program Files\ICONICS\GENESIS_32\Projects\Training Project), the next time you open that project in the ProjectWorX32 console it will detect the file and ask you whether to import the file into the project. This is shown in Figure 17.10. If you click "Yes," the file is added to the project tree and is displayed in the right-hand pane of the ProjectWorX32 console. If you click No, the file remains in the project folder but is not imported into to the project and is not displayed in the ProjectWorX32 console. However, this does not apply to unpacking_log.txt, packing_log.txt and all .mdb and .udl files.

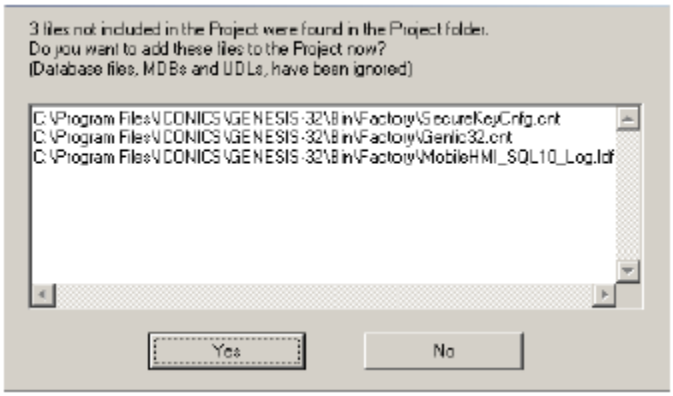


Figure 17.10 - Automatic Import of Files

17.3.2.3. Activating Configuration Databases

Some application configurators (e.g. Alarm Server, Alarm Logger, and Trend Logger) provide an option to specify an active configuration database.

> **NOTE:** For active Microsoft SQL Server configuration databases, information is stored inside the ProjectWorX32 database and is not subject to continuous polling. If the active configuration is changed from outside ProjectWorX32 (e.g. with the AlarmWorX32 Server Configurator) you will have to refresh the view by right-clicking and selecting **Refresh** from the pop-up menu. This updates ProjectWorX32 with the new database status.

To activate a configuration database:

1. Right-click the configuration database and select "Make Configuration Active" from the pop-up menu.
2. The icon next to the configuration file changes from the standard icon to a globe icon to indicate that it is active.

17.3.3. OPC Tag Verification

The ProjectWorX32 Console includes a feature that lists all of the OPC tags that belong to certain files. For example, if you right-click a GraphWorX32 display file in the ProjectWorX32 Console and select "List OPC Tags" from the popup menu, a **Tag Verification Utility** will open. This window lists all OPC tags contained within the display file. You can verify the status of tags in one of four ways:

- Double-click a tag under **OPC Tag Name**
- Select one or more tags under **OPC Tag Name** and click on the "Verify Selected" button
- Click on the "Verify All" button to check the status of all tags in the display file
- Click on "Export List" to export the data to a text file

Each of the tags selected for verification shows an indicator of good (green), bad (red), or uncertain (yellow) as a form of verification, as shown in Figure 17.11.

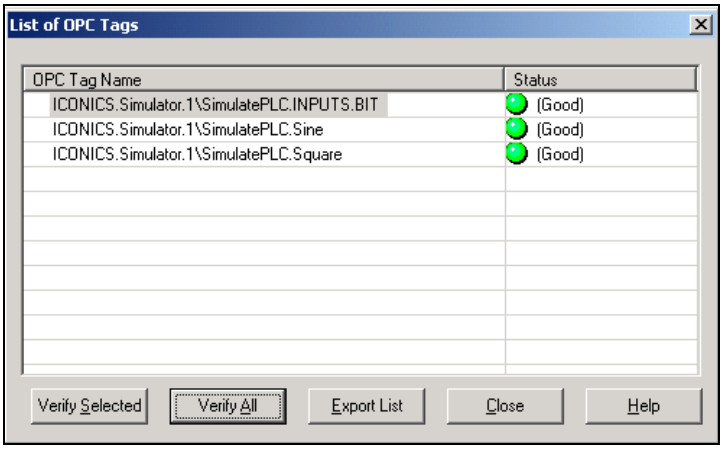


Figure 17.11 - List of OPC Tags

17.3.4. Using GenTray in ProjectWorX32

The **GenTray** tree of the ProjectWorX32 console, shown in Figure 17.12, includes all the capabilities of the GENESIS32 GenTray application. When the GenTray tree is selected, a list of the GENESIS32 applications on the machine is provided. From this list you can check to see whether a server or an application is running and browse their settings. You can also start and stop applications from here or set auto-start, auto-stop, and other properties for each application.

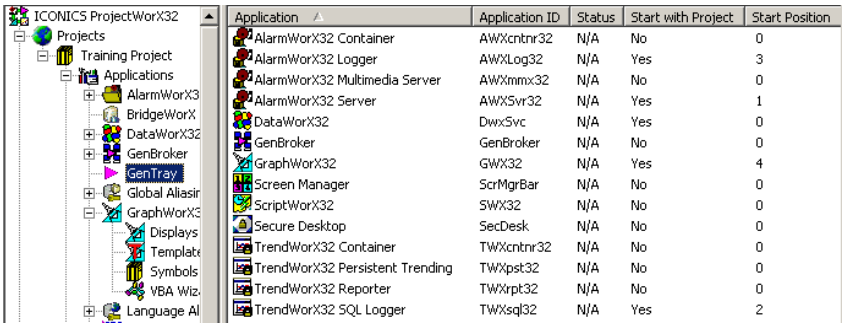


Figure 17.12 - GenTray in ProjectWorX32

When a project is put into Runtime the GenTray configuration will take effect. Auto-start applications will be started.

17.3.5. Packing and Unpacking a Project

The pack/unpack projects feature of the ProjectWorX32 console allows you to pack up an entire project (with all included application files) into a single, compressed file. This is especially useful for system administrators who want to configure a project and then send or copy the project to several different computers (e.g. nodes on a network). Then the packed file can be unpacked and run on each computer.

Packing a project is very simple: select a project, right-click and select “Pack Project.” In addition, you can select the same option from the **Action** menu. When the packing is complete a log file is displayed showing any errors or warnings, as shown in Figure 17.13. This log file will become a part of the project and is updated each time packing is executed.

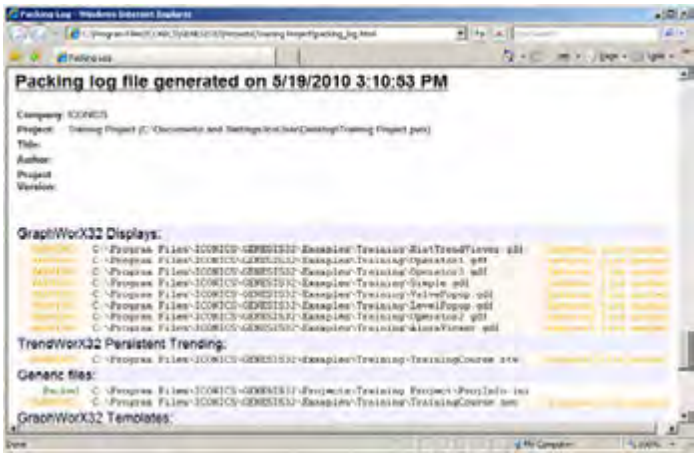


Figure 17.13 - Packing Log File

NOTE: The warnings in this packing log indicate the packed file was a referenced one. When you unpack the project they will reside in the project folder instead of the folder where it originally resided.

A packed file can be unpacked on any destination machine by taking the packed project file (.pwx) and copying it to the destination machine. On the destination machine, you can simply right-click on the **Projects** branch of ProjectWorX32 and choose the "Unpack Project" option. You will need to point to the packed file and the system can automatically update referenced files to work in the current projects folder on deployment machine during unpacking if necessary. A log file is again created and displayed.

After you unpack, you can use the **Global Find and Replace** to replace the node name throughout the application. This utility can be launched by selecting the project, right-clicking and selecting "Global Find\Replace."

A project at this point simply needs to be made Active, by right clicking on the Project and selecting the option "Make Project Active."

17.3.6. Overwriting Existing Projects

ProjectWorX32 includes a feature that allows you to overwrite an existing project by unpacking a newer version of that project. For example, if a system administrator makes changes to a project and needs to deploy an updated version of the project to multiple users on different network nodes, users on each node can unpack and automatically update the project.

When you unpack a packaged (updated) version of an existing project (a project with the same name) in the ProjectWorX32 Console, ProjectWorX32 scans each file in the updated project. ProjectWorX32 automatically unpacks any new files in the updated project (i.e. files that have been modified or added to the project since the last version).

For packaged files that already exist (i.e. with the same name) inside the project, ProjectWorX32 also checks the date/time of the files. If the date/time of a packaged file is the same as the date/time of the existing file of the same name nothing happens. If a packaged file is newer (i.e. more recent date/time) than the existing file of the same name, ProjectWorX32 automatically unpacks the newer file and overwrites the existing file. If a packaged file is older than the existing file of the same name, ProjectWorX32 warns you of this fact and gives you the option to overwrite the file.

> **NOTE: Difference Unpacking** of projects will not work for Microsoft SQL Server databases. When unpacking SQL-based configuration files the files will not be overwritten even if the packed project contains newer files of the same name.

17.3.7. Activating and Running Projects

The ProjectWorX32 console enables you to create and manage multiple GENESIS32 projects each of which may contain many different types of application files. Only one project can be put into runtime mode at a time. In order to enter a project into runtime mode, you must first designate the project as the currently active project. When you make a project active in the ProjectWorX32 console, all the connected included databases that are active for the project and all the settings to properly set up the GenTray are "made active" on the current machine. If there are multiple configuration databases for a single application within the active project, only the active database for that application will enter runtime mode.

You can make a project Active by right-clicking on the project in the tree control and selecting **Activation → Make Project Active** from the popup menu, as shown in Figure 17.14.

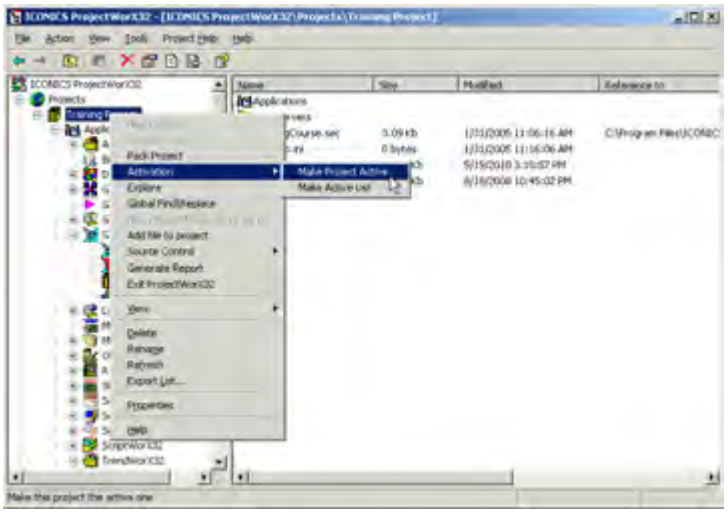


Figure 17.14 - Making a Project Active

You can also view the activation report at any time by choosing **Activation → Make Active List** from the popup menu. Click on "Print" if you want to print out the activation report.

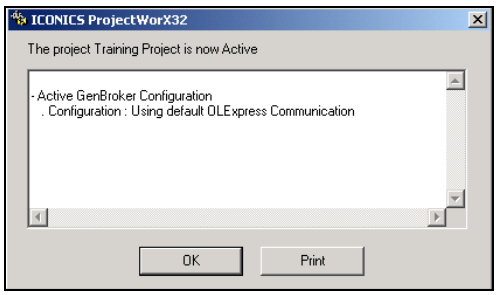


Figure 17.15 - Make Active List

When a project is active, the icon for the project will be green to indicate that it is the currently active project. Since activating a project does NOT set the active project into runtime mode you will have to enter the project into runtime once you make it active. You can enter the entire project into runtime mode by selecting “Put [Project Name] in Runtime” from the **Tools** menu, as shown in Figure 17.16. The name of the currently active project appears in the Tools menu command.

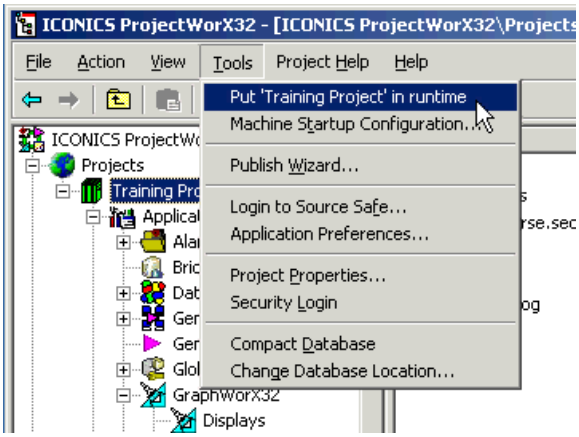


Figure 17.16 - Putting a Project into Runtime

To take the active project out of runtime mode, select “Shut Down [Project Name]” from the **Tools** menu, as shown in Figure 17.17. The name of the currently running, active project appears in the Tools menu command.

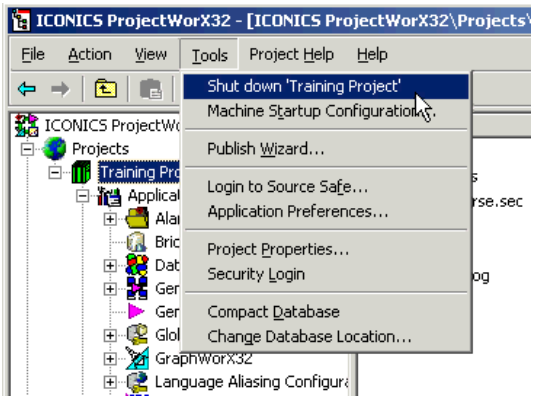


Figure 17.17 - Shutting Down a Project

Review: Additional Applications

R17

17.1. Chapter Objectives

In this chapter, we discussed a few different helpful applications in GENESIS32. You should now have a general understanding of the following applications:

- GenBroker
- GENESIS32 Tool Tray
- MonitorWorX
- ProjectWorX32

17.2. Summary

GENESIS32 comes with a few helpful applications. First, we created GenBroker to overcome DCOM communications problems. Using GenBroker, you can communicate with remote machines using channels such as TCP/IP and SOAP/XML. This allows us to easily obtain remote OPC data within a network as well over the internet. You can further use GenBroker to configure centralized administrative servers such security and licensing.

Using GenTray and MonitorWorX, you can configure applications to run as a service and provide a startup sequence to make sure applications start in the correct order. You can further provide a delay between starting individual applications and services to ensure that there is plenty of time to start each. These two applications can also tell you if the application or service is currently in runtime. Using MonitorWorX, you can further examine the application to find out information such as when it was started and how long it has been running.

ProjectWorX32 provides a centralized project management utility. Similar to Workbench32, you can create a project by adding different application files. You can further use ProjectWorX32 to pack a project and deploy it on a different machine.

Below are ten easy steps to having a running project using ProjectWorX32:

1. Add all your files to a newly created project, or create the files directly from within the project.
2. Mark your primary application files as the active files from these applications.
3. Set up an appropriate GenTray configuration, which will determine the applications that are executed or put into Runtime when the project itself is put into Runtime.
4. Activate Project and run if desire. You should make sure that no application is running when you activate the project.
5. If the project is running, you should shut it down, and then pack the project to move it from the development machine to the final destination machine (if necessary).
6. Move the .pwx file from the development machine to the destination machine.
7. Unpack the project on the destination machine (if necessary).
8. Globally find and replace the Node Name/IP Address if necessary.
9. Make the project active on the destination machine.
10. Put the project into Runtime.

17.3. Terms

GenBroker	GENESIS32 Tool Tray	GenTray
MonitorWorX	ProjectWorX32	

17.4. Test Your Knowledge

In this chapter, we discussed a few applications within the GENESIS32 suite. You should now be able to answer the following questions.

1. Which of the following is NOT a communication protocol used by GenBroker?

    a. TCP/IP
    b. DCOM
    c. SOAP/XML
    d. UDP

2. Which of the following is an Administrative Server in a GENESIS32 system?

    a. Licensing
    b. Security
    c. Events
    d. Global Alias
    e. Language
    f. All of the above
    g. Only A and B

3. True or False: You can use ProjectWorX32 to publish your GraphWorX32 displays all at once.

4. If machine A is running your OPC server and machine B is trying to access data from the remote OPC server, how should you setup GenBroker?

    a. Configure GenBroker and make sure that GenBroker is running on both machines.
    b. Configure GenBroker on both machines and make sure that it is running on machine A.
    c. Configure GenBroker on both machines and make sure that it is running on machine B.
    d. Run GenBroker on machine A with the default configuration and configure GenBroker on machine B to use the desire communication channel.

5. True or False: You can configure application startup sequence using MonitorWorX.

GraphWorX32 Shortcuts A

Shortcut Key	Function	Config Mode	Runtime Mode
F1	Help	✓	
F4	Properties window	✓	
Alt + F4	Close a display	✓	
Alt + F11	Visual Basic editor	✓	
Alt + A	Arrange menu	✓	
Alt + C	Configure mode		✓
Alt + D	Draw menu	✓	
Alt + E	Edit menu	✓	
Alt + F	File menu	✓	✓
Alt + H	Help menu	✓	✓
Alt + O	Format menu	✓	
Alt + R	Runtime mode	✓	
Alt + T	Tools menu	✓	✓
Alt + V	View menu	✓	✓
Alt + Y	Dynamics menu	✓	
Ctrl + 0	Custom Zoom	✓	✓
Ctrl + 1	Zoom in 50%	✓	✓
Ctrl + 2	Zoom in 75%	✓	✓
Ctrl + 3	Zoom in 100%	✓	✓
Ctrl + 4	Zoom in 150%	✓	✓
Ctrl + 5	Zoom in 250%	✓	✓
Ctrl + A	Select all objects	✓	
Ctrl + B	Toggle status bar	✓	
Ctrl + C	Copy	✓	
Ctrl + D	Duplicate selected object(s)	✓	
Ctrl + E	Replace	✓	
Ctrl + F	Find	✓	
Ctrl + G	Group into symbol	✓	
Ctrl + H	Home View	✓	✓
Ctrl + L	Toggle scroll bars	✓	✓
Ctrl + N	New file	✓	
Ctrl + M	Toggle between runtime mode and configuration mode	✓	✓
Ctrl + O	Open file	✓	✓
Ctrl + P	Print file	✓	✓
Ctrl + R	Runtime Window Properties Mode	✓	
Ctrl + S	Save file	✓	

Shortcut Key	Function	Config Mode	Runtime Mode
Ctrl + T	Toggle toolbars	✓	
Ctrl + U	Ungroup symbol	✓	
Ctrl + V	Paste object	✓	
Ctrl + W	Show whole display	✓	✓
Ctrl + X	Cut object	✓	
Ctrl + Z	Undo	✓	
Ctrl + PgUp	Bring forward	✓	
Ctrl + PgDn	Send backward	✓	
Ctrl + Numpad +	Zoom in	✓	✓
Ctrl + Numpad –	Zoom out	✓	✓
Ctrl + Alt + A	Hide layers above current layer	✓	
Ctrl + Alt + B	Hide layers bellow current layer	✓	
Ctrl + Alt + C	Set current active layer	✓	
Ctrl + Alt + J	Toggle script toolbar	✓	
Ctrl + Alt + S	Toggle symbol toolbar	✓	
Ctrl + Alt + U	Select language	✓	✓
Ctrl + Alt + PgUp	Set next layer active	✓	
Ctrl + Alt + PgDn	Set previous layer active	✓	
Ctrl + Shift + 0	Hide Layers	✓	✓
Ctrl + Shift + A	Space evenly across	✓	
Ctrl + Shift + B	Align bottoms	✓	
Ctrl + Shift + C	Align centers	✓	
Ctrl + Shift + D	Space evenly down	✓	
Ctrl + Shift + H	Make same height	✓	
Ctrl + Shift + L	Align lefts	✓	
Ctrl + Shift + M	Align middles	✓	
Ctrl + Shift + N	Decluttering zoom	✓	✓
Ctrl + Shift + O	Make same height and width	✓	
Ctrl + Shift + F	Fit to window	✓	✓
Ctrl + Shift + G	Snap to grid	✓	
Ctrl + Shift + R	Align rights	✓	
Ctrl + Shift + S	Zoom selection	✓	
Ctrl + Shift + T	Align tops	✓	
Ctrl + Shift + U	Undo zoom	✓	✓
Ctrl + Shift + W	Make same width	✓	
Ctrl + Shift + Z	Box zoom	✓	✓
Ctrl + Shift + PgUp	Bring to front	✓	
Ctrl + Shift + PgDn	Bring to back	✓	
Shift + resize	Preserve aspect ratio while resizing an object	✓	

Shortcut Key	**Function**	**Config Mode**	**Runtime Mode**
Shift + object creation	Create an object with equal width and height	✓	
Shift + right-click on a grouped object	Gain access to items in the group	✓	

TrendWorX32 SQL Query Engine Syntax

B

Table B.1 - Supported DML Syntax

Syntax	NOTE
SELECT	
Now()	
NowUTC()	
AND	
OR	
<=	
<	
>	
>=	
<>	
=	
TOP	Requires a number to return the top x-number of records
MIN	Requires a column name as a parameter
MAX	Requires a column name as a parameter
AVG	Requires a column name as a parameter
SUM	Requires a column name as a parameter
COUNT	Requires a column name as a parameter
DateAdd(Datepart, value, BaseTimestamp)	Datepart uses the abbreviations such as mi for minutes, hh for hours, yy for years, etc. BaseTimestamp is in the format of 'yyyy-mm-dd hh:mi;ss'

Table B.2 - Syntax NOT Supported

Syntax	Description
Between…and…	
GetDate()	
DateDiff()	
Order By	
Group By	
Like	

Table B.3 - TrendWorX32 SQL Query Engine Functions

Function	Description
QualityToStr(Quality)	This function translates the Quality column in the RawData table from a DWORD (number) to a string. The resulting values could be Good, Bad, and Uncertain.
DataTypeToStr(DataType)	This function translates the DataType Column in the Tags table from a DWORD (number) to a string. The resulting values could be VT_BOOL, VT_UI2, VT_UI4, and VT_R8.
AccessRightsToStr(AccessRights)	This function translates the AccessRights Column in the Tags table from a DWORD (number) to a string. The resulting values could be TAG_ACCESS_RIGHTS_NONE, TAG_ACCESS_RIGHTS_READ, TAG_ACCESS_RIGHTS_WRITE, and TAG_ACCESS_RIGHTS_READ_WRITE
IsGood(Quality)	This function allows you to return only good quality data. EXAMPLE: SELECT * FROM RawData WHERE IsGood(Quality)=1
IsBad(Quality)	This function allows you to return only bad quality data. EXAMPLE: SELECT * FROM RawData WHERE IsBad(Quality)=1
IsUncertain(Quality)	This function allows you to return only data with uncertain quality. EXAMPLE: SELECT * FROM RawData WHERE IsGood(Quality)=1

For Example, here the query for data collected today for the Sinus simulated signal:

```
SELECT * FROM RawData
WHERE TAGNAME = 'SimulatePLC.Sine' AND
TIMESTAMP >= DateAdd(dd, -1, NowUTC) AND
TIMESTAMP <= NowUTC
```

Delimiters and Syntaxes *C*

Beginning Delimiter	**End Delimiter**	**Function**
{{	}}	OPC Tag delimiters in an expression
~~	~~	Local Variable delimiters
lic:		License GenBroker Plug-in tags
sim:		Simulation GenBroker Plug-in tags
schwxm:		ScheduleWorX tags
swx:		ScriptWorX2010 GenBroker Plug-in tags
rdcy:		Redundancy GenBroker Plug-in tags
bacnet:		ICONICS Bacnet OPC Server GenBroker Plug-in tags
snmp:		ICONICS SNMP Server GenBroker Plug-in tags
rcpm:		Recipe management GenBroker Plug-in tags
exp:		Global Expression GenBroker Plug-in tags
grp:		Group GenBroker Plug-in tags
vset:		Value Set GenBroker Plug-in tags
rcp:		Recipe GenBroker Plug-in tags
reg:		UDM Register GenBroker Plug-in tags
awxs:		Global Alarm Subscription GenBroker Plug-in tags
awxf:		Global Alarm Filter GenBroker Plug-in tags
<#	#>	Global Aliasing delimiters
/+	+/	Language Aliasing delimiters
$“	”$	String literal in a process point
<<	>>	Local aliasing delimiters
A:AliasName		Allows dynamic global aliasing definition. The Complete syntax is *A:AliasName=“AliasValue”/ThemeScope;*. ThemeScope can be, but not limited to the following: Not defined – Relative to current document level 0 0 – Machine Level 1 – Process Level 2 – Document Level 0 3 – Document Level 1 +1 – Relative to current document level 1 +2 – Relative to current document level 2

Glossary *D*

Active Configuration refers to the currently used configuration. Some ICONICS applications allow for multiple configurations to be active simultaneously, but most ICONICS applications only allows for one Active Configuration at a time.

ActiveX Controls are mini-programs that contain a specific functionality. These controls can be embedded into other programs, or ActiveX containers, gaining the functionality of the controls.

Alarm Area is a way to organize your alarms. You can organize alarms into areas; you can also use areas to filter alarms.

Alarm Attributes are characteristics about a certain alarm provided by the alarm server. OPC defines certain attributes that each alarm server must provide. The server can also provide additional attributes on top of the OPC specification.

Alarm Subscription is used by alarm clients. It is a connection to an *OPC A&E* server detailing specific categories, areas, and alarm tags on which to receive notifications.

Alarm Tag is an OPC A&E tag that monitors an OPC DA tag for alarms configured in your Alarm Server.

Alarm Template contains information to configure many similar alarms. Using an Alarm Template, you can easily configure the information common to your alarms and then create alarms based on the template. Using this in conjunction with the *Tag Import Wizard* will save much development time.

AlarmWorX32 is a suite of products in the GENESIS32 product family. It includes application such as the *AlarmWorX32 Server*, *AlarmWorX32 Logger*, *AlarmWorX32 Container*, *AlarmWorX32 Viewer*, and *AlarmWorX32 Reporter*.

AlarmWorX32 Container is an application within the AlarmWorX32 application suite. It is an ActiveX container created for the *AlarmWorX32 Viewer* and *AlarmWorX32 Reporter*.

AlarmWorX32 Logger is the ICONICS application that allows you to record alarm and event history. It can subscribe to any OPC A&E compliant OPC servers and write relevant information into a MS Access or SQL database.

AlarmWorX32 Reporter is the ICONICS A&E client that allows you to visualize your historic alarms and events. Through the AlarmWorX32 Reporter, you can see alarms and events as well as generate reports on the logged alarm information.

AlarmWorX32 Server is the ICONICS alarm notification application. It comes in two parts: a configurator and a server. The configurator allows you to add and edit alarm tags while the server reads the configured information and monitors OPC tags to generate real-time alarms.

AlarmWorX32 Viewer is the ICONICS A&E client that allows you to visualize your real-time alarms and events. Through the AlarmWorX32 Viewer, you can see alarms and events as well as interact with the alarms by acknowledging it.

Alias Resolution is the ability to define a string in configuration mode that represents another string in runtime mode. The alias can represent multiple strings in runtime depends on the type of alias and how you set its definition.

Application Actions refers to the functions available in a particular application to ICONICS Security. For example, you can limit a user from exiting *GraphWorX32* runtime by removing the "Exit Runtime" application action under GraphWorX32 from a particular user.

Application Setup Utility is an ICONICS utility that helps configure DCOM so that different applications in the GENESIS32 Suite can communicate to each other.

Application Preferences refers to system wide preference settings for GraphWorX32. Application preferences are similar to *Display Properties* accept that application preferences will apply to newly created displays when you are done with the configuration.

BizViz a family of ICONICS products created to bridge the gap between manufacturing and corporate business information systems. The product family includes application such as BridgeWorX, ReportWorX, PortalWorX, MobileHMI, Alarm Analytics, Productivity Analytics, and Hyper Historian.

Calculation Period defines how often the *TrendWorX32 Logger* calculates an aggregate base on the *Data Collection Rate*. When you are using a Calculation Period, the only data logged to database is the calculated aggregate unless you select to log all samples.

Client-side Filter is a filter created on an *OPC A&E* client to obtain a subset of alarms sent to client from the server.

Close Window is a *Pick Action* available in GraphWorX32. It allows you to close the currently opened display in runtime.

Configuration refers to an object that contains information about your system. The configured information is stored in individual *Configuration Database* for the application.

Configuration Database refers to the database that stores your configured system. A Configuration Database can contain many *Configurations*, but only one of these *Configurations* will be marked as the *Active Configuration*. For most GENESIS32 applications, the Configuration Database can take the form of either MS Access or MS SQL.

Configuration Mode is the mode in an application that allows you to define and customize settings specific to your project.

Contents Area is the portion in an application where it provides you the workspace to perform configuration.

Data Collection Rate is the rate at which *TrendWorX32 Logger* or *TrendWorX32* Viewer samples a tag for value.

Data Link Properties are properties to make a connection to a database. This is similar to the *ODBC Data Source* accept that instead of creating a source object that stores the connection information, you fill out the information asked in the Data Link Properties, which will allow the application to connect to the database directly.

Data Mining Configurator is the user interface used to setup the Data Mining OPC Server. You can create Data Sources and Data Manipulators in the configurator.

Data Source is the source of data that drives the behavior of the dynamic. In general, it refers to the OPC tag or expression used in a GraphWorX32 dynamic.

Database Connection is a connection that the *AlarmWorX32 Logger*, *AlarmWorX32 Reporter* and *TrendWorX32* configuration uses to connect to a database where the system logs the information. See *ODBC Data Source* or *Data Link Properties* for more information.

Database Group is a group defined in the *TrendWorX32 Logger* where you specify the *ODBC Data Source* to connect to the logging database. In the hierarchy of the TrendWorX32 Logging Configuration, it is directly below the configuration.

Database Logging is the type of logging that *AlarmWorX32 Logger* uses to log information into a database. Alternatively, it can use *Printer Logging*.

Default Group contains permissions for the “guest” user in ICONICS Security. Any user, including those that are not logged into the system, will have permissions configured in the Default Group.

Display Back is a *Pick Action* available in GrpahWorX32. It allows you to browse back to a display that you have shown previously.

Display Forward is a *Pick Action* available in GrpahWorX32. Similar to the *Display Back* action, it allows you to browse forward to a more recently view display after using the *Display Back* action.

Display Properties are properties of a display. It is similar to *Application Preferences* but the settings only applied to the currently opened display.

Display Refresh Rate is the rate at which the *TrendWorX32 Viewer* will update with the collected samples.

Download Value is a *Pick Action* available in GrpahWorX32. It allows you to write a value to an OPC tag when you trigger the action. The action contains three execution triggers: On Down, While Down, and On Up. You can assign different values to write to the OPC tag for each of the execution trigger.

Dynamic Objects in GraphWorX32 are static objects with applied GraphWorX32 dynamics. GraphWorX32 dynamics includes Size, Location, Color, Animation, and many more.

Embedded Window is a *Pick Action* available in GrpahWorX32. It works similar to the *Popup Window* action, except that the open window is embedded in the parent display that called it.

Expression Editor is the dialog window that allows you to enter an expression for a *Data Source*. The Expression Editor contains many different functions as well as checks for the syntax of the expression when you are done entering it.

Filter Wizard is the wizard dialog that comes up the first time you are trying to create an alarm filter. You can use it to create useful *client-side filters* easily and quickly.

Freeze Mode is a mode in *TrendWorX32 Viewer* during runtime. You can stop the trending by entering Freeze Mode. This allows you to scroll back in time to look at history data.

GAS Theme refers to a Global Aliasing theme. In runtime, a Global Aliasing theme governs the Global Alias values that the system resolves.

GenBroker is the tool that ICONICS provide to provide communication between two networked machines. GenBroker helps bypass shortcomings of DCOM and allowing data to pass over the internet.

GENESIS32 is the ICONICS' suite of HMI/SCADA applications for 32-bit systems.

GENESIS32 Tool Tray is synonymous with GenTray. It is an application that allows you to configure how ICONICS application should run. The application also lets you configure which applications on the computers should start on system startup and provide a delay between each startup applications.

GenTray please see *GENESIS32 Tool Tray*.

Global Alias is a string representation of a different string in runtime. Using a Global Alias can help you quickly develop HMI screens that use similar data sources.

Global Aliasing System is the aliasing system that *GENESIS32* uses through the system. There are two pieces to Global Alias: a Global Alias Engine and a Global Alias Client. The Engine takes care of talking to the *Global Aliasing Configurator* to find the defined alias and the Client talks to the engine to resolve the alias values.

Global Aliasing Configurator is the centralized location where you can configure your Global Alias Themes (*GAS Theme)*, *Theme Item*, the alias, and the alias definitions. It can be either a Microsoft Access or SQL Server database.

GraphWorX32 is the ICONICS 32-bit HMI/SCADA graphical creation application. It allows you to create dynamic displays for operators to control the production floor. A GraphWorX32 display has a file extension of .gdf.

GraphWorX32 Layers can be used to show different information on the display based on user selection or current active security level. You can also use layers to clutter or de-clutter the display based on zoom levels.

GraphWorX32 Template takes the file extension .tdf. You can use template files to enforce company standard. When you double-click on template file, you are creating a new GraphWorX32 display based on the define standard in the template.

History Refresh Rate is the rate at which the *TrendWorX32 Viewer* goes to the database to retrieve history data. For this reason, the History Refresh Rate should be set to the Data Logging Rate specified in the *TrendWorX32 Logger* Configuration or larger.

HMI stands for Human Machine Interface. It is usually a graphic user interface (GUI) that an operator uses to control the production floor.

Language Alias is a string representation of another string in runtime. Language Alias allows you to switch the text on a display to different languages.

Language Aliasing System is an aliasing system that works throughout GENESIS32. It contains a server that is configurable through the *Language Configurator* and all GENESIS32 applications can be clients to this server during runtime to log or display translated text.

Language Configurator is the application to use when defining the different languages in the system, the aliases, text translations, and expressions.

Linked Server is Microsoft SQL Server objects. This allows you to make a connection to external (third party) database and interact with it using the Microsoft SQL Server Management Studio.

Load Display is a *Pick Action* available in GrpahWorX32. It allows you to open a specific GraphWorX32 display into the same application window during runtime.

Local Alias is an aliasing system that GraphWorX32 uses. Unlike *Global Alias*, you can only use Local Alias within GraphWorX32 and you may not use it within ActiveX Controls in a GraphWorX32 display.

Local Variable is a variable available only in GraphWorX32. It is a temporary placeholder for a value. Each time you exit GraphWorX32, including going out of runtime mode and browsing to another display, the variable will lose the current value and revert to the default value assigned.

Logging Group is a group that you define under the *Database Group* in the TrendWorX32 Logging Configuration hierarchy. You can define the *Data Collection Rate, Logging Rate,* and table management settings in the Logging Group.

Logging Rate is the rate at which the *TrendWorX32 Logger* writes collected values into the database.

Logging Tag is a tag defined in the *Logging Group* for data collection. This is the lowest on the TrendWorX32 Logging Configuration hierarchy.

Modal Popup is a parameter to the *Popup Window* pick action. When you check this option, the user will not be able to access the parent display until child (the popup window) is closed.

MonitorWorX is a new application introduced in version 9. The main task of this application is to provide statistics about ICONICS application. You can also use the application to monitor status between two redundant servers.

Object Level Aliasing refers to a *Local Alias* that is applied to a particular object. In GraphWorX32, you can use the same *Local Alias* in many objects and define the resolution of the alias base on the object.

ODBC Data Source is a Data Source used by *AlarmWorX32 Logger* and *TrendWorX32*. It is Window's way to connect to a database. You can get to the Data Source (ODBC) by going into the Control Panel, choosing Administrative Tool and select Data Source (ODBC). You should be careful when using an x64 system since the default is x64, but it also has an x86 counterpart; since GENESIS32 is x86, it uses the counterpart.

OPC used to stand for OLE for Process Control. Currently, it is no longer an acronym. It simply means the technology governed and standardized by the *OPC Foundation*.

OPC Alarm and Events or OPC A&E is a protocol defined by the *OPC Foundation*. It primarily governs the communication of Alarms and Events between OPC A&E server and clients.

OPC Data Access or OPC DA is a protocol defined by the *OPC Foundation* containing information on how OPC DA server should provide real-time data to OPC DA clients such as GraphWorX32.

OPC Foundation is the organization that maintains and continues to develop OPC standards for the automation industry. For more information about the OPC Foundation, please visit http://opcfoundation.org.

OPC Historic Data Access or OPC HDA is a protocol defined by the *OPC Foundation.* It specifies how the server should give data to historic data access clients.

OPC Server is an application server that connects to data sources or devices such as a *PLC* to bring data into your *HMI* or *SCADA* system. ICONICS application can communicate to all OPC compliant OPC Servers to visualize your data.

Pack and Go Wizard is the wizard inside *Workbench32* that helps you compact a project from one machine and deploy it in another machine.

Pick Action is a dynamic in GraphWorX32 that a user can trigger. You can assign Pick Actions to static objects, buttons, checkboxes, and radio buttons.

PLC stands for Programmable Logic Control. It is an example of a real time system that contains instruction sets that you can program to control your automation environment. *OPC Servers* can connect to these devices to obtain human readable data.

Popup Menu is a *Pick Action* available in GrpahWorX32. It allows you to create a customer user menu to use in runtime.

Popup Window is a *Pick Action* available in GrpahWorX32. Similar to the *Load Display* pick action, it allows you to open a specific display in runtime. The difference is that the Popup Window action will open the display in a separate window than the one the parent display is using.

PPT see *Process Point*.

Printer Logging is a type of logging available in the *AlarmWorX32 Logger*. It allows you to print the alarm information directly, creating a paper trail. You can enable Printer Logging alongside *Database Logging* or you can use it individually.

Process Point or sometimes referred to as *PPT*, refers to dynamic object in GraphWorX32 that can show you the value of an OPC tag.

ProjectWorX32 is ICONICS' application that allows you to consolidate all applications files in one when you need to move applications from a development machine to a production machine. Starting in version 9.2, you can also use *Workbench32* to do the same task.

Provider is an application in *Workbench32* that allows you to configure options for the application.

PPT see *Process Point*.

Referenced Object is the object to base changes on amongst all the selected objects. For example, if you want to make all the objects the same size, after the action is performed, all objects will be the same size as the referenced object. This object has active selection squares (blue) in the group while the others are in active (white with blue outline).

Regular Expression is a special string that describes a search pattern. You can use Regular Expressions in your security configuration to allow or deny access to Points, Alarms, and files.

ReportWorX is part of the *BizViz* family of products. It takes advantage of Microsoft Excel to provide decision makers analyzed information in a standard reporting format.

Rotation Pivot is the point around which an object will rotate for the Rotation dynamic. When the pivot point is inside the object, it will spin itself; if the point is outside of the object, it will orbit around the point.

Runtime Mode is the mode in which you can see live data and control equipments.

Runtime Toolbar is available in most *ActiveX Controls*. The runtime toolbar gives you options to control the appearance and behavior of the *ActiveX Control* when you are in *Runtime Mode*.

SCADA stands for Supervisory Control and Data Acquisition.

Second Level Aliasing refers to passing an alias value from a parent display to a child display. You can do this by using the Set Aliases command in a *Load Display* or *Popup Window* pick action.

Security Administrator is the user who can access the security configuration. If the system does not recognize any security administrator, anyone can open the security configuration.

Security Client is an application that uses ICONICS Security. In the case of GENESIS32, it includes runtime components such as GraphWorX32 and configuration components such as Alarm Server Configurator.

Security Configurator is the application you use to setup security.

Security Indicator is an ActiveX Control that allows you to see your security server node as well as the users logged into the security server.

Security Login Client is the user interface from ICONICS that you can use to log into ICONICS Security.

Security Modes refers to the different types of security configuration you can create. There are three modes: Basic, Advanced, and Integrated NT.

Security Server uses the last saved security configuration to provide protection for your ICONICS system.

Server-side Filter is alarm filters created in the *Alarm Subscription*. This type of filter limits the server from sending alarms to the client that fails to meet the filter criteria.

Set Right Time is a function available in *TrendWorX32 Viewer* when it is in *Freeze Mode*. This option allows you to set the right-hand time (end time) of the trend viewer, allowing you to display data during a particular period with a click of a button.

Severity or sometimes referred to as Priority can be any value between 0-1000. This is used to indicate the urgency of an alarm tag.

Shadow Copy is a copy of an object that we use to indicate the end location of an object when using a Location dynamic.

Startup Squelch is delay that the *AlarmWorX32 Server* uses during system startup to give OPC servers a chance to start properly before sending out alarms.

Symbols Library contains a set of previously configure static and dynamic objects. These objects can help you quickly create attractive displays.

Tag Import Wizard is used in the *AlarmWorX32 Server* configuration in *Workbench32*. Using the Tag Import Wizard, you can create many alarm tags based on an *Alarm Template*. It is a simple way to create many similar alarms with little effort.

Theme Item refers to an item in a *GAS Theme*. The value the Global Alias resolves to during runtime is the item that you configured.

Toggle Value is a *Pick Action* available in GrpahWorX32. Similar to the *Download Value* pick action, it allows you to write values to an OPC tag. However, the toggle value would write two configured values alternating between clicks as opposed to writing just one value.

Tree Explorer is part of *Workbench32*. It contains a list of providers toward the bottom and provides a tree structure to the selected provider.

Trend Period is the length of time (or period) for which the TrendWorX32 Viewer will display data.

TrendWorX32 is a suite of products used for logging OPC data and replaying it with an *OPC HDA* client. It is part of the *GENESIS32* product family and it contains application such as *TrendWorX32 Logger*, *TrendWorX32 Viewer*, *TrendWorX32 Reporter*, and a few other less use applications.

TrendWorX32 HDA Server is usually transparent to the user. It is an *OPC HDA* compliant server used to retrieve data from the logging database and replay history data to an *OPC HDA* compliant client.

TrendWorX32 Logger is the application that logs *OPC DA* data into a database for trending purpose. The Logger comes with a user interface, the TrendWorX32 Configurator, and you can define the how to sample data, where to log the data to, how often to write the data to the database, and which tag should be track using the Configurator.

TrendWorX32 Reporting is the application that can create reports with data log by *TrendWorX32 Logger*. It can export data to a text file (CSV), Microsoft Excel, and a database table.

TrendWorX32 SQL Query Engine provides an SQL interface for you to query for history data. You can retrieve data using the TrendWorX32 SQL Query Engine using the *Data Mining Configurator*, *ReportWorX*, Microsoft Excel, and any other applications that can run SQL Queries.

TrendWorX32 SQL Tool ActiveX is an ActiveX control that allows you to retrieve data logged by *TrendWorX32 Logger* in conjunction with automation methods using any ActiveX container such as GraphWorX32 and the TrendWorX32 container.

TrendWorX32 Viewer is the ICONICS OPC DA and HDA client that allows you to visualize your real-time and history trends. The TrendWorX32 Viewer supports different plot type including time plots, XY plots, bar plots, strip charts, logarithmic plots, and circular charts.

UDM Expression is a function that takes an input, performs a specified calculation, and returns an output.

UDM Group is a collection of data tags that are manipulated as a single item, which are represented by a single group tag.

UDM Recipe is a collection of data tags that represents the different ingredients in a recipe. The recipe item stores the different value for each ingredient.

UDM Trigger is a rule that defines conditions under which an event occurs. There are two types of triggers: data triggers and time triggers.

UDM Value Set allows you to write pre-configured sets of values into a group of data tags.

Unified Data Browser (UDB) is the application you can use to look for OPC tags. The Unified Data Browser also allows you to browse for *Global Aliases* and GenBroker plug-in tags.

Unified Data Manager (UDM) is the application that allows you to manage and configure data that may be use globally in many different applications of GENESIS32. For example, if you would like to write one value to many tags at the same time, you can create a Group tag in the Unified Data Manager. You can create Global Expressions, Groups, Value Sets, Recipes, Registers, Event Triggers, Global Alarm Subscriptions, and Global Alarm Filters in the Unified Data Manager.

Workbench32 is a centralized location for configuration and runtime operations.

Answers *E*

Questions / Chapters	1	2	3	4	5	6	7	8	9	10
Chapter 1	A	D	True	False	False					
Chapter 2	True	False	D	B	D	False	C	False	True	A
Chapter 3	E	E	True	B	True	False	True	C	E	True
Chapter 4	D	A	True	True	C	A	True	True	G	B
Chapter 5	B	C	True	True	True					
Chapter 6	D	A	True	G	B	A	B	A	True	True
Chapter 7	A	C	True	False	B	D	True	D	False	True
Chapter 8	E	F	A	True	True	True	True	D	B	A
Chapter 9	B	False	E	False	False	C	True			
Chapter 10	B	C	A	D	G	True	True	D	False	True
Chapter 11	D	True	E	C	B	False	B	A	C	True
Chapter 12	E	B	True	True	True	A	B	False		
Chapter 13	D	B	True	True	D	G	E	True	True	
Chapter 14	C	A	True	True	B	D				
Chapter 15	B	True	True	D	False	True	C	A		
Chapter 16	C	True	False	E	B	True	True	False	C	A
Chapter 17	D	F	True	E	True					

Visualize Your Enterprise™

100 Foxborough Blvd., Foxborough, MA 02035
508.543.8600 • FAX: 508.543.1503
www.iconics.com • E-mail: info@iconics.com

© 2013 ICONICS, Inc. All Rights Reserved.
Specifications subject to change without notice.