



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

ICONICS

100 Foxborough Blvd.

Foxborough, MA 02035

Tel: 508-543-8600

Fax: 508-543-1503

E-Mail: support@iconics.com

Web: www.iconics.com

Contents

TrendWorX32 Viewer ActiveX
Using TrendWorX32 Viewer ActiveX
Language-Aliasing Support
Global Aliasing Support
Internal Architecture of ActiveX
User Interface
Trend Viewer Toolbar

**Historical Replay
OLE Automation
Code Samples**

Introduction

The TrendWorX32 Viewer ActiveX is an OPC client application that provides both real-time trending as well as historical data trending within the GENESIS32 product family. It can also be used with third-party OPC server applications to provide excellent trending features.

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.

TrendWorX32 Viewer is interfaced to display memory-based historical data from the ICONICS 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 ICONICS TrendWorX32 SQL Server.

TrendWorX32 Viewer ActiveX Component

TrendWorX32 Viewer ActiveX includes the following features:

- Highly customizable component
- Fast data replay
- Independent data collection/screen refresh mechanism
- Full exposure through OLE Automation objects, such as:
 - * Title
 - * X-grids
 - * Y-grids
 - * Ranges/x-axis ranges (for XY plots)
 - * Details
 - * Time settings
 - * Statistics
 - * Pens
- Extensive runtime configuration
- Built-in buffering system
- Integration with the ICONICS Security Server
- Support for custom imported data
- Multilingual support
- Interface to OPC Real-Time servers
- Easy Visual Basic integration
- Drag-and-drop support during runtime
- Extensive real-time signal information (including signal quality)
- Expression handling
- Real-time statistical information with .csv file support
- User-configured alarm levels for signals
- User-defined global or relative signal ranges

-
- Fixed location or scrolling grids
 - Embedded signal simulator
 - Support for multiple plot types, including:
 - * Time plots
 - * XY-plots
 - * Bar plots
 - * Logarithmic plots
 - * Strip charts
 - * Circular charts
 - Custom data displaying
 - Global aliasing with browsing support
 - Language aliasing with browsing support
 - Latest HDA compliance support

New Features

The TrendWorX32 Viewer ActiveX has many new features, including:

- Global aliasing with browsing support
- Language aliasing with browsing support
- Latest HDA compliance support
- Enhanced daylight savings time support for historical replay
- Reordering on Trend pens
- TraceWorX debug tracing support
- Enhanced expression support
- Progress indicator dialogs for potentially time consuming pen connections or historical updates

Note: TrendWorX32 up to and including Version 6.1x is using the Windows-based APIs for converting a time from local time to UTC or from UTC to local time. The Windows APIs do not correctly adjust the conversion for the daylight savings time if the timestamp to be converted was obtained from a different time period (i.e. it was obtained in "summer" time but now we have "winter" time). TrendWorX32 Version 7.0 has new time conversion utilities that account for this discrepancy. There are, however, some things to consider:

- These functions are used by default (no user enabling is required).
- This is what they do in order to convert each time stamp:
 - They check Windows for the PC Time Zone Settings and to see if automatic daylight savings is enabled.
 - They also obtain from Windows the times/dates for switching over. (These dates do not change; they are in the form of, e.g., the first Sunday of April.)
 - Depending on the previous information, the new functions properly adjust the conversion times to compensate for daylight savings time.

Using the TrendWorX32 Viewer ActiveX

The TrendWorX32 Viewer ActiveX component integrates easily into Visual C++ or Visual Basic container applications. The ICONICS GENESIS32 containers, such as GraphWorX32, provide Toolbar support for inserting the TrendWorX32 Viewer ActiveX. Once inserted into a document or a form, the TrendWorX32 Viewer ActiveX can be resized and configured.

Within the ICONICS GENESIS32 containers and in Visual Basic forms, the component is initially placed in design mode. In design mode, you can configure the TrendWorX32 Viewer ActiveX using the **TWXView32 ActiveX Properties** dialog box. Once the application is committed to runtime mode, the **Trend Viewer** toolbar is activated. In addition, it becomes a drop target for text and therefore can establish signal connections through drag-and-drop.

The TrendWorX32 Viewer ActiveX configuration data are saved directly in the container form or within its own streamed file format with the file extension .v32. You must provide a valid file name before the data are serialized and saved. TrendWorX32 Viewer uses the concept of a “working directory,” defined in the ICONICS GENESIS32 containers (such as GraphWorX32). This represents the working directory where project files and configuration files will be stored. TrendWorX32 Viewer will use this directory as a starting point for saving and loading its configuration files.

Connections

The TrendWorX32 Viewer can display data from real-time signals connected to OPC servers, calculated variables (using expressions), ICONICS simulated data and custom (user-fed) data through OLE Automation, or historical data connected to OPC HDA servers (such as ICONICS TrendWorX32 SQL Server).

Signal connections can be established in various ways:

- In configuration mode, through the **Pens** tab of the **TWXView32 ActiveX Properties** dialog box
- In runtime mode, through the **Edit Pens** button on the **Trend Viewer** toolbar
- Using OLE Automation methods (such as "AddRealTimePen" and "AddHistoryPen") and VB Script/VBA
- Using drag-and-drop (text format), in runtime mode only

Once a signal is connected to a TrendWorX32 Viewer pen, the TrendWorX32 Viewer adds the pen to its buffering system manipulation and starts data collection and data display. To configure a pen to display a calculated variable, click **Tags Menu** and select **Expression Editor**. This opens the **Edit Expression** dialog box. In the **Edit Expression** dialog box, click **Tags** to open the **OPC Universal Tag Browser**, which enables you to connect pens to real-time OPC server signals.

To connect a real-time pen to a signal using drag-and-drop, the TrendWorX32 Viewer must be activated and in runtime mode. The drag-and-drop object should be of text type.

The TrendWorX32 Viewer ActiveX utilizes the ICONICS thin wrapper OLEExpress to communicate to OPC servers, therefore providing a very fast and efficient mechanism for real-time data acquisition and display.

To display only historical data, select **History Only** in the **Pens** tab of the **TWXView32 ActiveX Properties** dialog box. Connections to the historical signals can be made by selecting **History Tag** from the **Tags Menu** to view all of the available OPC HDA servers. In addition to real-time and historical data, the TrendWorX32Viewer also supports the custom pen type through

OLE Automation. This pen can be fed with user data using OLE Automation methods.

Runtime Mode Operations

The **Trend Viewer** toolbar, shown below, enables you to modify the trend display during runtime mode.



Trend Viewer Toolbar

During runtime mode, the TrendWorX32 Viewer provides support for the following functions:

- Adding, deleting, and editing a pen.
- Freezing or unfreezing the Trend display (does not halt data collection).
- Editing trend options, such as grids, ranges, cursor step, and language resources.
- Editing the trend period or data-collection rates. Note that by changing data period, TrendWorX32 Viewer may modify the data-collection rate in order not to exceed the configured number of samples.
- Displaying statistical information for all the pens regarding the displayed values. This statistical information can be stored in a .csv file. You first need to provide a valid file name for the function to work.
- Zooming on a specific region of a trend when the trend is in **Freeze Mode**. You can zoom by pressing the left mouse button while holding down the **SHIFT** key and selecting the area you on which you would like to zoom. Clicking the left mouse button on the trend display undoes the zoom. In addition, there is Point and Click support for zooming by clicking the Zoom Trend button on the **Trend Viewer** toolbar.
- Scanning the range of a trend by using the cursor to view specific sample values.
- Selecting the focus pen, which is the pen whose details and (if desired) range information will be displayed first in the TrendWorX32 Viewer.

-
- Retrieving and replaying from the TrendWorX32 Persistent Trending server.
 - Paging forward and backward in time.
 - Time shifting for comparison of pens.

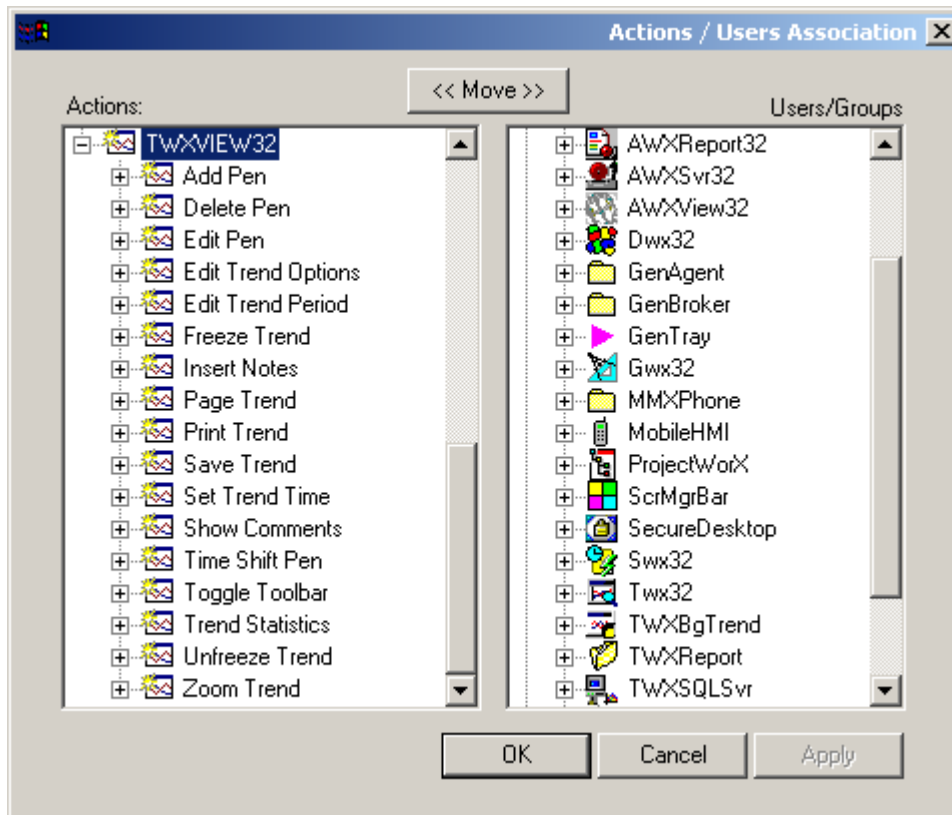
Functions Not Supported in Runtime Mode

The following functions are not supported in runtime mode:

- Replacing point names (You should use delete/add, or within the VBA environment, use the "FormLoad" event for replacing point names.)
- Changing the trend plot type
- Getting to the Time Settings object through the OLE interface
- Modifying the x-axis of XY plots
- Modifying file name for saving statistics
- Modifying preference for global ranges
- Modifying preference for shifting grids
- Modifying preference for startup value
- Modifying preference for restoring range values

Security

The TrendWorX32 Viewer ActiveX is currently interfaced to the ICONICS Security Server through OLEExpress. The security system is enabled within runtime mode to provide a very flexible security mechanism. Currently the TrendWorX32 Viewer supports the security actions shown in the dialog box below.



ICONICS Security Server

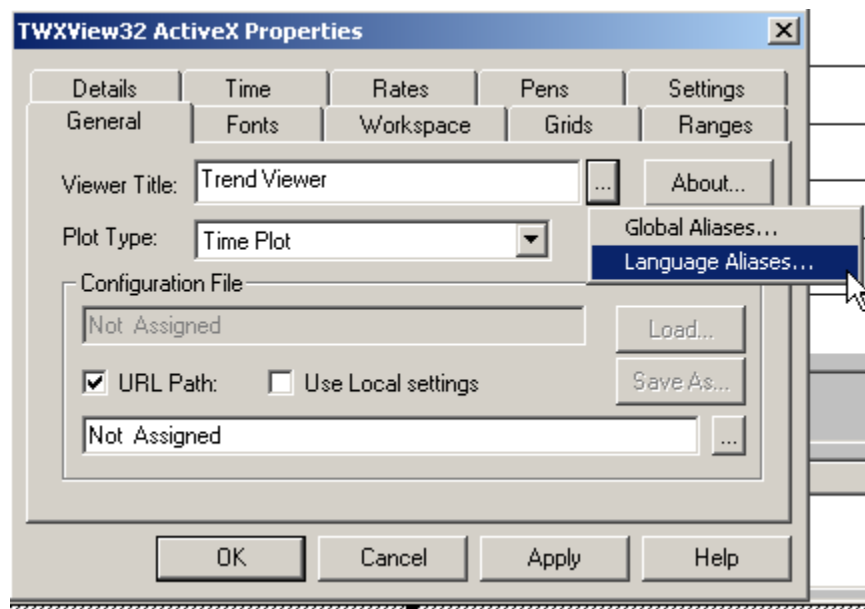
If you do not have the appropriate security actions for the functions you want to perform, an error message box will appear indicating that you do not have the security level required by the action.

Asynchronous Downloading Features

The TrendWorX32 Viewer supports asynchronous downloading of configuration data when used inside ICONICS ActiveX document servers, such as GraphWorX32, to produce documents, which can be viewed using Microsoft Internet Explorer.

To enable this feature, select **URL Path** from the **General** tab of the **TWXView32 ActiveX Properties** dialog box, shown below, and supply a URL path. In the case of viewing ActiveX documents within an Intranet setup, you can supply a server name and directory specifying the location of the TrendWorX32 Viewer configuration data (must be a shared directory), by using the convention "file:\\Server Name\\Directory\\File Name.v32."

You can also enable the **Use Local Settings** option, which, if checked, stores runtime changes to a local copy of the remote configuration file. It will use this local file every time the HTML page that hosts the Viewer is loaded inside the Web browser.



TrendWorX32 Viewer ActiveX Properties: General Tab

Language-Aliasing Support

The TrendWorX32 Viewer ActiveX component has all of its strings saved in the resource file, which can be modified to provide custom versions. In addition, it provides real-time support for loading a resource-only .dll to support international languages. This .dll file is compiled as a standard Win32 .dll, and it is made up of all the resources and the .odl file of the TrendWorX32 Viewer module. Any time the ICONICS applications settings are switched to a different language, TrendWorX32 Viewer will load the corresponding resource .dll.

The resource-only .dll file name uses the TrendWorX32 Viewer component, followed by the three characters used in Windows NT as the abbreviation for a language (e.g., the abbreviation "deu" stands for German, and the .dll should be named "TWXViewer_deu.dll"). In addition, using the OLE automation property language ID, you can control this functionality through OLE automation.

In addition, the Unicode version of the TrendWorX32 Viewer ActiveX also supports language aliasing with automatic range and value scaling, as well as unit conversions, formatting, and font selection. For more information about configuring Unicode version language-aliasing support refer to the GENESIS32 **Language Configurator** documentation. The TrendWorX32 Viewer ActiveX supports the following language-aliased strings:

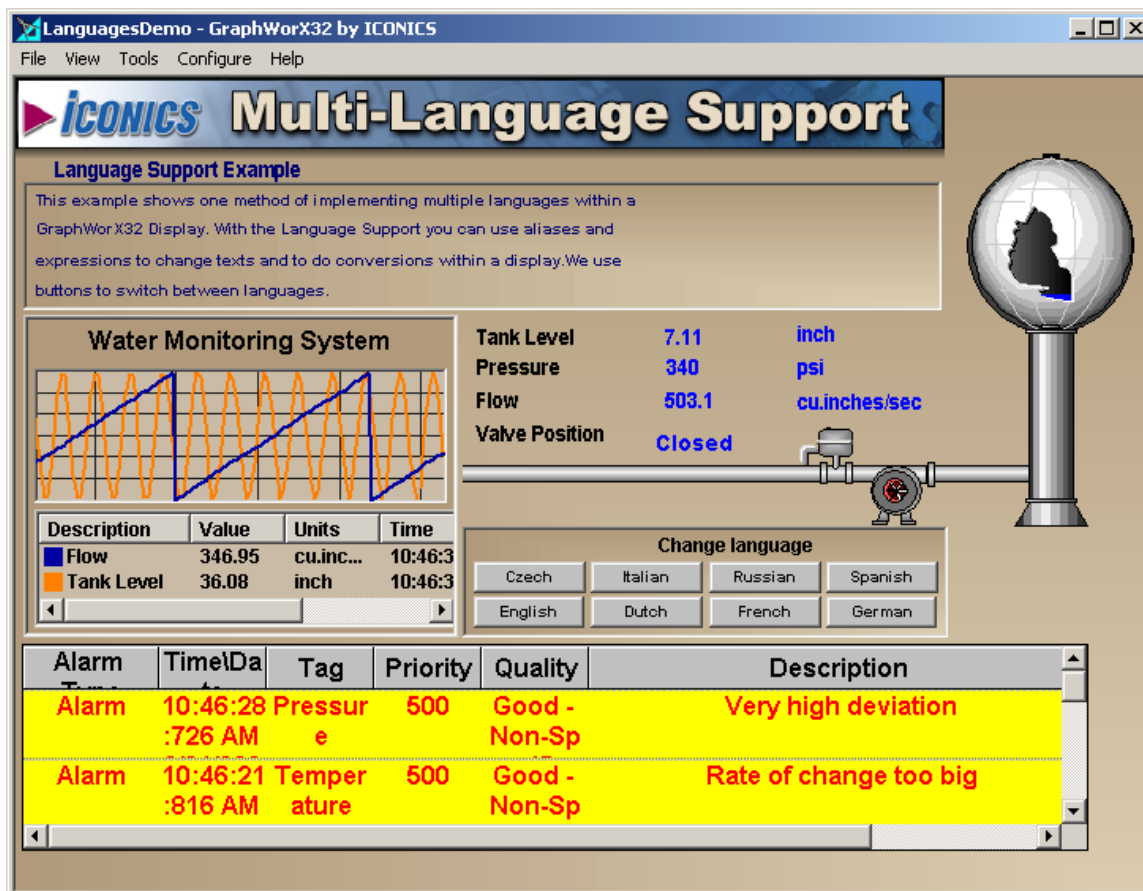
- Pen description
- Pen engineering units (with scaling support)
- Pen details format
- Title string of viewer
- Format strings of details
- Detail window column names

Note: The TrendWorX32 Viewer requires the following additional language-aliasing configuration work. If any expression is to be used for range, alarm limits, and or unit scaling, you need to define both a "Read Expression" and a "Write Expression" in the Language Configurator. By doing so, the Viewer will be able to function properly between language switches.

Note: To enhance language-aliasing performance, select the Microsoft Arial Unicode font, which contains all Unicode characters. The Arial Unicode MICROSOFT font must be selected separately within the Properties dialog box of each ActiveX component.

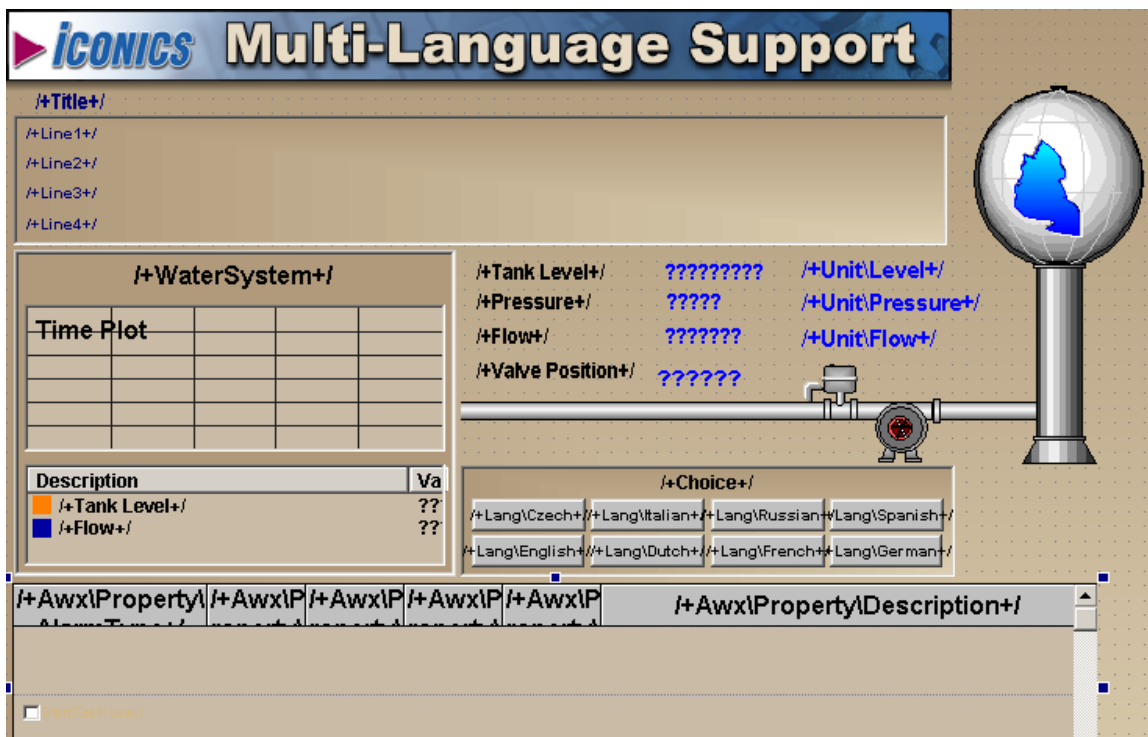
Sample Language Configuration

The Unicode Installation of GENESIS32 installs a sample Language Server configuration under the "Languages" folder of the GENESIS32 product installation "tree." In addition, a sample display "languagesDemo.gdf," which includes a sample Viewer configuration, is provided under the Examples/Languages folder. The figure below shows this sample during runtime mode. Notice that you can switch between languages in this display by simply clicking the buttons in the Change Language field.



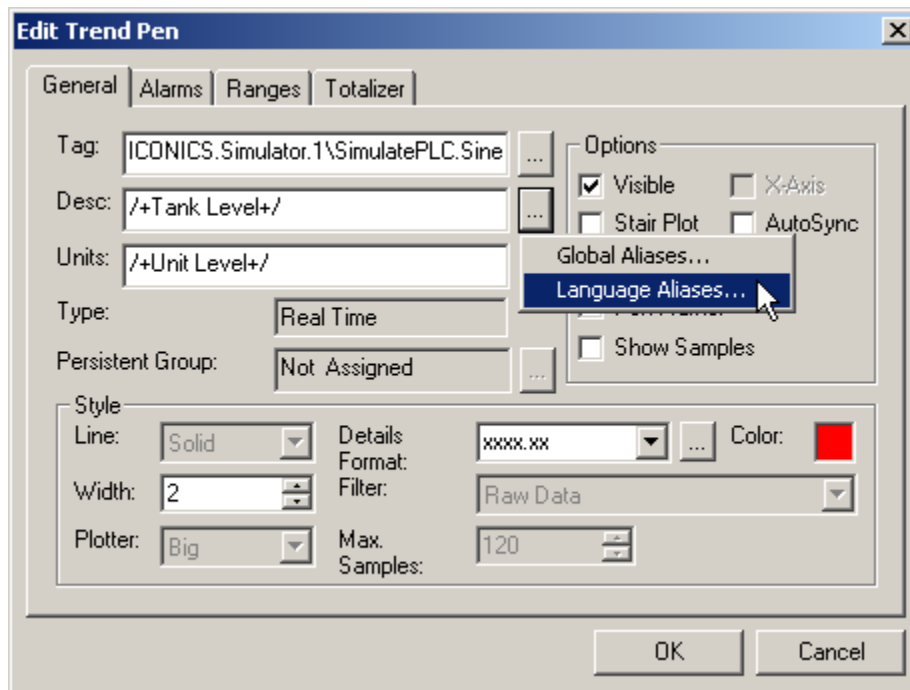
Language-Aliasing Demo in Runtime Mode

The figure below shows this same language-aliasing demo in configuration mode. This display incorporates ActiveX objects from GraphWorX32, TrendWorX32, and AlarmWorX32.



Language-Aliasing Demo in Configuration Mode

For example, if you click on the "Water System" time plot trend in the language-aliasing configuration shown above, you can view the properties of the trend in the TrendWorX32 Viewer Active X. The **Edit Trend Pen** dialog box, shown below, displays the language-aliasing configuration of a pen in the Viewer.

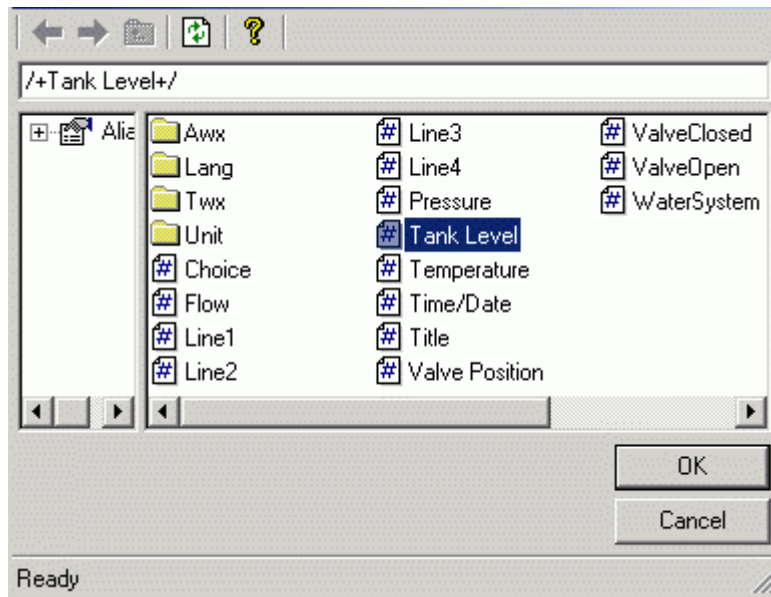


Language Aliasing Pen Configuration

For the example above, the Trend display appears as shown in the figure below. Note that the strings within a "/" and "/" delimiter pair define a language-aliased string.

When specifying a language alias in the **Edit Trend Pen** dialog box, you can also select a language alias from the Language Alias Browser, which includes all language aliases in the language database. This eliminates the need to manually type in the alias name. Click the ... button next to the **Lang Alias** box and select **Language Aliases**, as shown in the figure above.

The **Language Alias Browser** opens, as shown in the figure below. The browser includes all languages aliases in the language database. All language aliases that are configured in the Language Configurator are conveniently available to choose from inside the browser. The tree control of the Language Configurator is mimicked in the tree control of the Language Alias Browser. Select a language alias by double-clicking the alias name. The alias name appears at the top of the browser, which automatically adds the /+ and +/ delimiters to the alias name. Click the **OK** button.



Selecting an Alias From the Language Alias Browser

During animation (runtime) mode, the Viewer ActiveX will interface to the ICONICS Language Server, and it will try to resolve the language-aliased strings. If successful, it will display the translated strings in the trend screen configuration, as shown below. In addition, if the **Units** language alias string has an equation defined in the Language Configurator, it will autoscale the pen values based on the scaling equation supplied by the Language Server.

/+WaterSystem+/					
Time Plot					
Description	Value	Units	Time	Date	
 /+Tank Level+/	????	/+Unit\level+/	###:###:##	###:###:##	
 /+Flow+/	????	/+Unit\Flow+/	###:###:##	###:###:##	

Language-Aliased Trend Display

Note: The Unicode version language aliasing is independent of the resource .dll selection, which relates mostly to the text in the dialog boxes and other user interface elements.

Global Aliasing Support

The TrendWorX32 Viewer supports Global Aliasing by interfacing to the ICONICS Global Aliasing Server. The following strings can be aliased:

- ☐ **Title**
 - Title
- ☐ **Ranges**
 - Hi Tag, Lo Tag, Format, Format XY

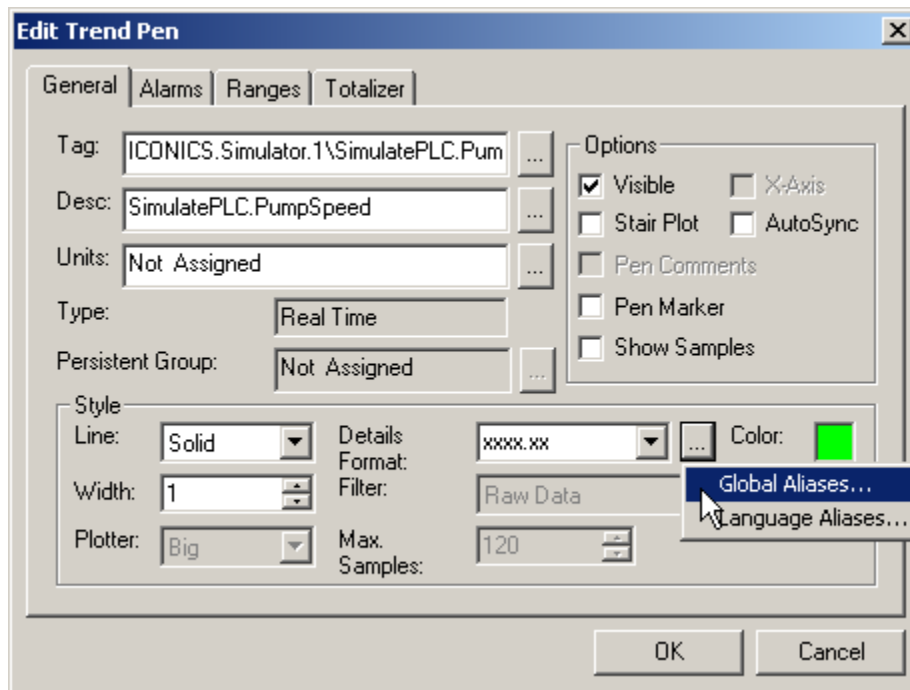
-
- ❑ **Details**
 - Format
 - ❑ **Pen**
 - Point Name, Hi Tag, Low Tag, Description, Units, Details Format, Ranges Format, Persistent Group Name
 - ❑ **URL Path**
 - Configured in the **General** tab of the **TrendWorX32 Viewer ActiveX Properties** dialog box.
 - ❑ **Expressions**
 - In the **Expression Editor**, click the **Tags** button and select **Global Aliases** from the pop-up menu to open the **Global Alias Browser**.

Currently the TrendWorX32 Viewer supports machine-level and process-level aliasing. All fields, which support aliasing, have been interfaced to the Global Alias Browser dialog, which is evoked by clicking on the ... button.

In addition, the TrendWorX32 Viewer now has Language and Global Aliasing browsers integrated.

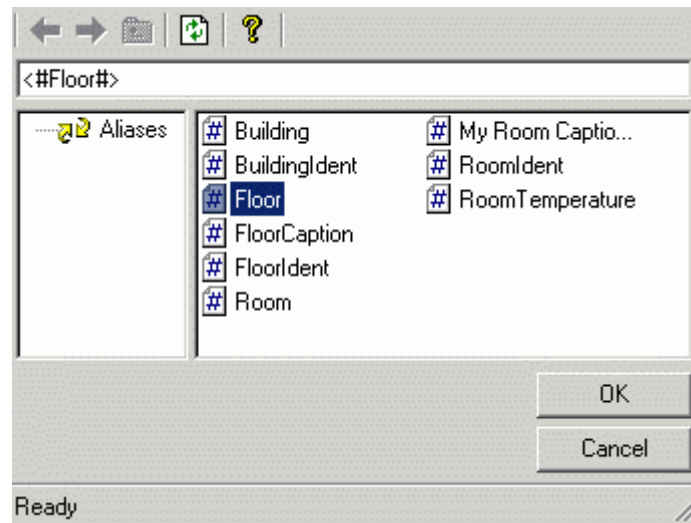
The user first connects a real time, persistent, or historical tag and then has the option to alias the entire pen string or part of it. Global aliasing is also supported also for drag-and-drop operations as well as related OLE automation methods.

- ❑ What is NOT supported:
 - Runtime editing of aliased fields
 - Aliases resulting in new global aliases (ie a resolved alias is also an alias)
 - Local settings saving if Global Aliases are used.



Global Aliasing Pen Configuration

Clicking the ... button and selecting **Global Aliases** from the pop-up menu opens the Global Alias Browser, as shown in the figure below. Select a global alias from the Global Alias Browser, which includes all global aliases in the global alias database. This eliminates the need to manually type in the alias name. All global aliases that are configured in the Global Alias Engine Configurator are conveniently available to choose from inside the browser. The tree control of the Global Alias Engine Configurator is mimicked in the tree control of the Global Alias Browser. Select a global alias by double-clicking the alias name (e.g. "Floor" in the figure below). The alias name appears at the top of the browser, which automatically adds the <# and #> delimiters to the alias name. Click the **OK** button.



Selecting an Alias From the Global Alias Browser

Internal Architecture of ActiveX

The internal architecture of the TrendWorX32 Viewer ActiveX is composed of three individual object managers:

- A **Point** manager (used to store point name information)
- A **Variable** manager (used to store data buffers and server updates for each point)
- A **Pen** manager (used to represent the visual aspect of the collected data points)

In addition, TrendWorX32 Viewer has the following threading mechanism:

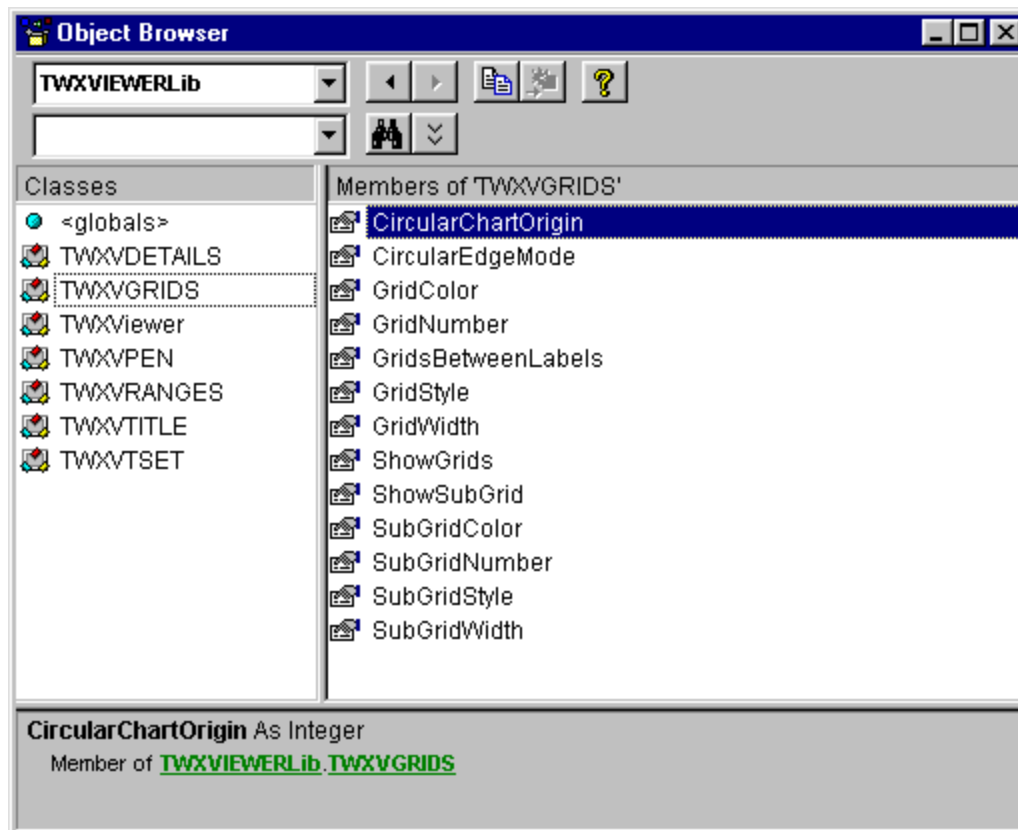
- A **Main** (primary) thread
- A **Worker** thread used for data updates and buffer management

The internal architecture of the TrendWorX32 Viewer allows the data to be collected at a different rate and graphically drawn at a multiple of the data-collection rate in order to minimize overall system load. Currently the minimal scan rate is set to be 10 milliseconds, and it is required that all data-collection rates that you set be an integer multiple of 10 milliseconds.

In addition, you can configure the desired number of samples to view on the screen (up to a maximum of 7200) and the desired window time period. Furthermore, all objects exposed by the TrendWorX32 Viewer ActiveX can be accessed directly through OLE Automation, and their associated properties can be manipulated directly through C++ clients (which support OLE containment) or Visual Basic applications. The fastest display refresh (redraw) rate is 50 milliseconds.

TrendWorX32 Object Capability

This section describes the objects exposed through OLE Automation used in TrendWorX32 Viewer ActiveX. These can be easily viewed in Visual Basic's **Object Browser**, shown below.



Object Browser

Using Visual Basic To View Title Objects

Title manages title information, such as the title name, font options, and drawing of the title. The **Title Object** serializes all title-specific information.

Using Visual Basic To View Grid Objects

Grid manages grid options and drawing of the grid. The **Grids Object** serializes all grid-specific information. There are individual objects for X-grids and Y-grids.

Using Visual Basic To View Ranges Objects

Ranges manages range information, such as border options, and drawing of the ranges according to plot type. The **Ranges Object** serializes all range-specific information.

Using Visual Basic To View Time Settings Objects

Time Settings manages time information, such as border options, font options, and drawing of the time field. The **Time Settings Object** serializes all time field-specific information.

Using Visual Basic To View Details Objects

Details manages detail information, such as border options, and drawing of the Details field. The **Details Object** serializes all detail-specific

information.

Using Visual Basic To View Pen Objects

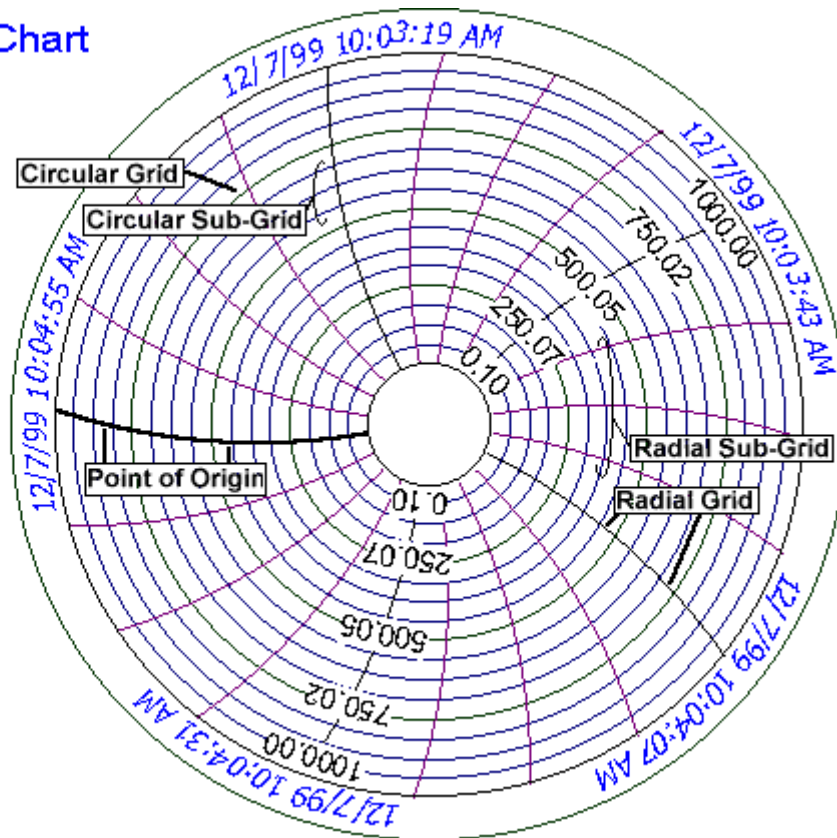
Pen manages all visual aspects of the signal being trended, including display management, statistics, and curve style. The **Pen Object** serializes all pen-specific information.

Circular Charts

A **circular chart** is a trend type that plots signal values in a circular field. The **signal value** is represented by the distance from the center of the chart. The angular position of the points on the line represents **time**.

To create a circular chart ActiveX, select **Circular Chart** from the **Add Trend** submenu of the **Edit** menu, or press the shortcut keys **CTRL+6**. This will display the circular chart ActiveX in the TrendWorX32 Container, as shown below.

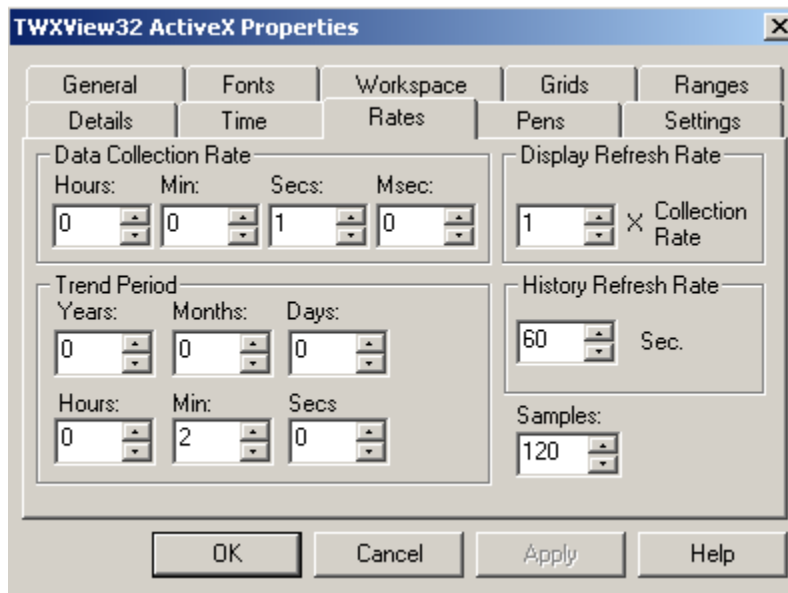
Circular Chart



Circular Chart

Open the **TWXView32 ActiveX Properties** dialog box by double-clicking the circular chart ActiveX, or by selecting **Properties** from the **Edit** menu. This will enable you to set the properties for the circular chart.

Set the trend period and the data-collection rate using the **Rates** tab in the **TWXView32 ActiveX Properties** dialog box, shown in the figure below. The thickest line on the circular chart, prominent because of its bold appearance, represents the point of origin for the trend period. Notice that the time labels go counter-clockwise in conformity to the industry standards.



Properties Dialog Box: Rates Tab

Once all the data points are plotted within the trend period during runtime mode, one of the following will occur:

- All data points will disappear, and a new trend period will begin.
- All data points will remain and continuously scroll as old data points are replaced by new data points.

You can control these conditions using the **Settings** tab of the **TWXView32 ActiveX Properties** dialog box. When **Reset Circular Chart** is checked in

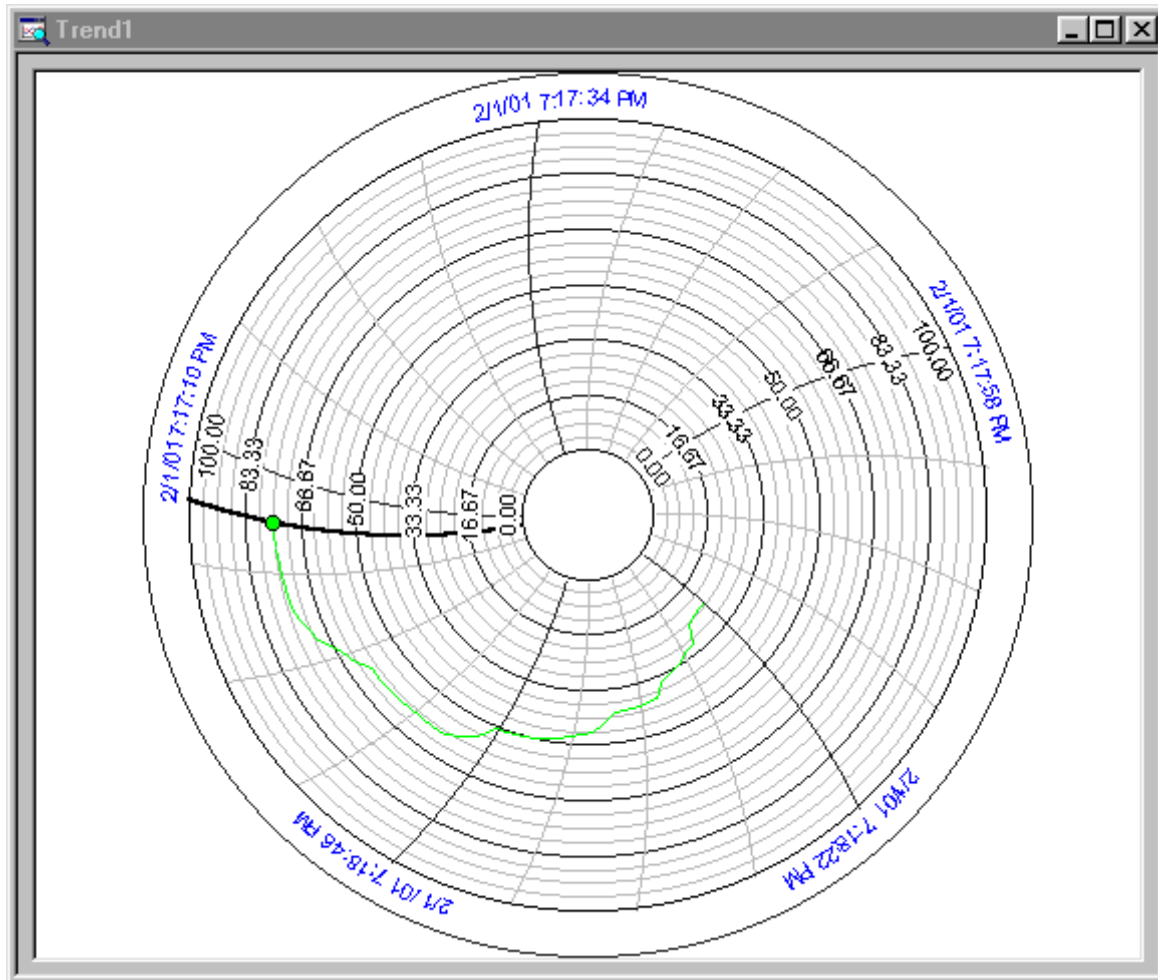
the **Settings** tab, all data points will disappear at the beginning of each new trend period. In addition, you can choose to have the circular chart auto print upon resetting after a full trend period has elapsed.

Altering the Appearance of a Circular Chart

You can change the appearance of a circular chart using the tabs in the **TWXView32 ActiveX Properties** dialog box. Particularly relevant to the appearance of the circular charts are the **Grids** tab, **Fonts** tab, and to a lesser extent, the **Ranges** tab. For more information on how to use these tabs, please see the appropriate sections.

Circular Charts in Runtime

The following figure is an example of how a circular chart looks during runtime mode.



Circular Chart in Runtime Mode

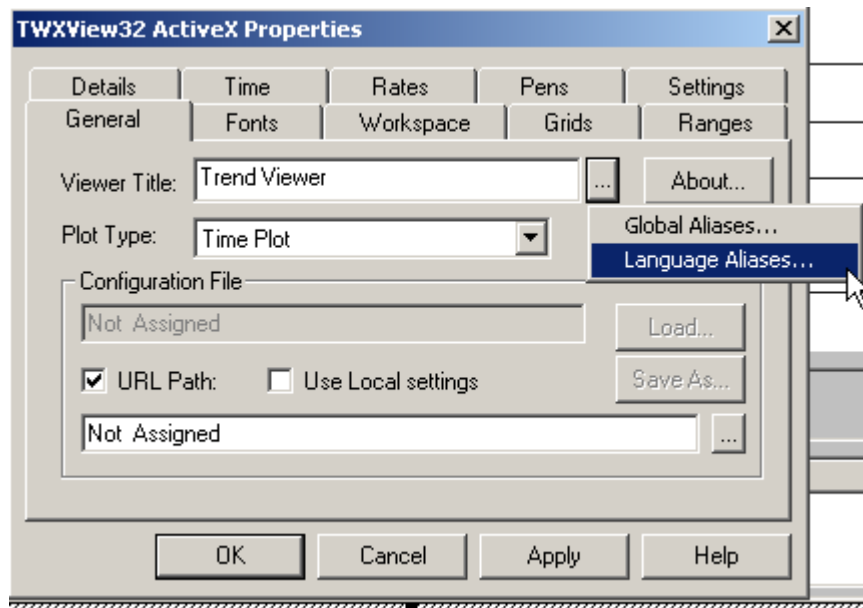
User Interface

This section describes the user interface for TrendWorX32 Viewer ActiveX.

ActiveX Properties Dialog Box

The **TwxView32 ActiveX Properties** dialog box (also called the **Properties** dialog box), shown below, is the configuration user interface for the ActiveX. Double-clicking the ActiveX during configuration mode opens the **Properties** dialog box, which as you can see contains the following tabs:

- **General tab**
- **Fonts tab**
- **Workspace tab**
- **Grids tab**
- **Ranges tab**
- **Details tab**
- **Time tab**
- **Rates tab**
- **Settings tab**
- **Pens tab**

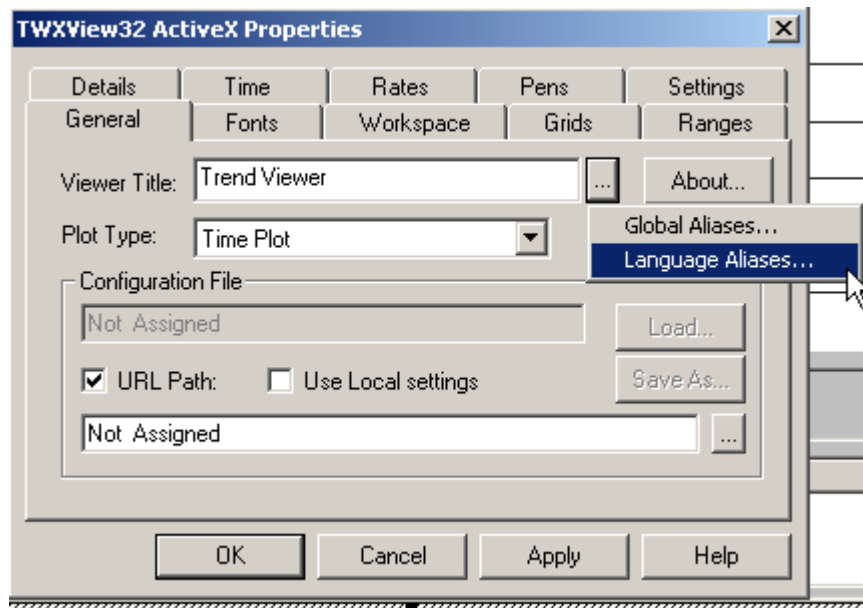


TWXView32 ActiveX Properties Dialog Box

General Tab

The **General** tab, shown below, is the default tab of the **Properties** dialog box. The **General** tab handles the following functions:

- Viewer title
- File selection
- Web configuration
- Plot type



Properties Dialog Box: General Tab

To select an alias to use for the **Viewer Title**, click the ... button to the right of the text box and select either **Global Aliases** or **Language Aliases** from the pop-up menu, as shown in the figure above. This opens the respective alias browsers, allowing you to choose from a list of available aliases.

You should always choose the **Plot Type** first because this choice determines which options will also be available later on. The TrendWorX32 Viewer component can save its configuration data on a serialized file using the file extension .v32. By default, its configuration is saved in the container. You must configure this parameter (by clicking **Save As**) for all configuration data to be saved.

Clicking **About** launches the **ICONICS About Box**, which displays information about the product and ICONICS.

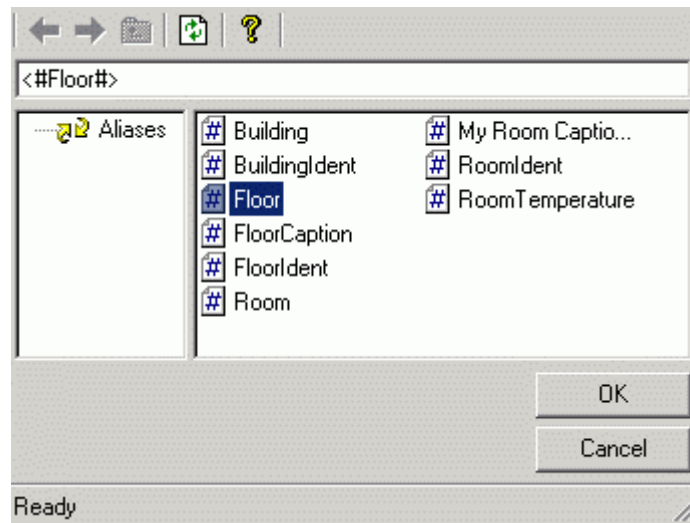
By default, information is stored as part of the container application's document file. You can also store information as part of a separate (.v32) file. To save the .v32 file, under **Configuration File** click **Save As** to open the **Save TWXViewer Display** dialog box. Enter a name for the .v32 file in

the **File Name** field, and then click **Save**.

You can also open an existing configuration file by clicking the **Load** button on the **General** tab and browsing for the file. The ActiveX will attempt to load the configuration file. If **URL Path** is checked, the ActiveX will use the specified network URL path upon loading the file.

Note: The ActiveX cannot be saved to a URL path. To create a URL file, save a report configuration to a local file and copy the file to the desired network location. You can also enable the **Use Local Settings** option, which, if checked, stores runtime changes to a local copy of the remote configuration file. It will use this local file every time the HTML page that hosts the Viewer is loaded inside the Web browser.

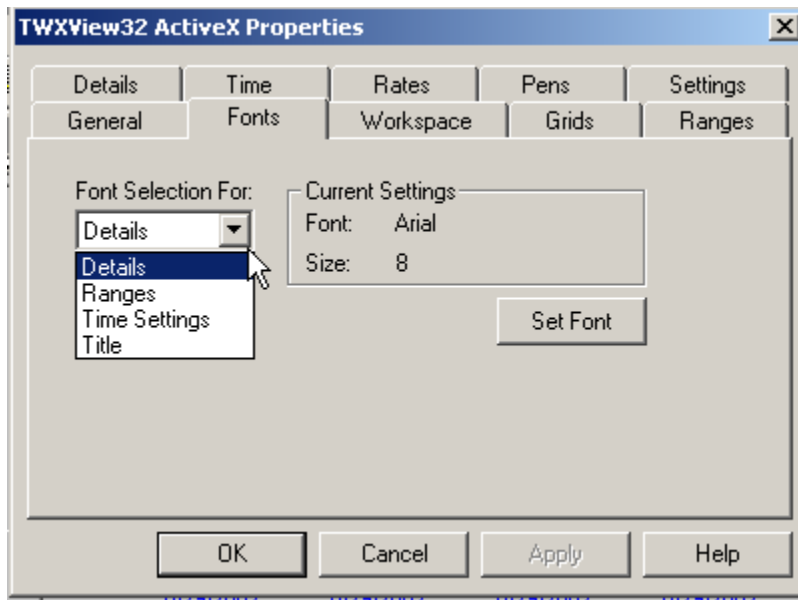
You can also select aliases to use for the configuration file. Clicking the ... button and selecting **Global Alias Browser** from the pop-up menu opens the Global Alias Browser, as shown in the figure below. Select a global alias from the Global Alias Browser, which includes all global aliases in the global alias database. This eliminates the need to manually type in the alias name. All global aliases that are configured in the Global Alias Engine Configurator are conveniently available to choose from inside the browser. The tree control of the Global Alias Engine Configurator is mimicked in the tree control of the Global Alias Browser. Select a global alias by double-clicking the alias name (e.g. "Floor" in the figure below). The alias name appears at the top of the browser, which automatically adds the <# and #> delimiters to the alias name. Click the **OK** button.



Selecting an Alias From the Global Alias Browser

Fonts Tab

The **Fonts** tab of the **Properties** dialog box, shown below, sets the font type for the **Details**, **Ranges**, **Time Settings**, and **Title** sections of the **Trend Selection For** drop-down list and click **Set Font** to choose a font. The standard Windows **Font** dialog box appears, and you can select the font type, size, and style.



Properties Dialog Box: Fonts Tab

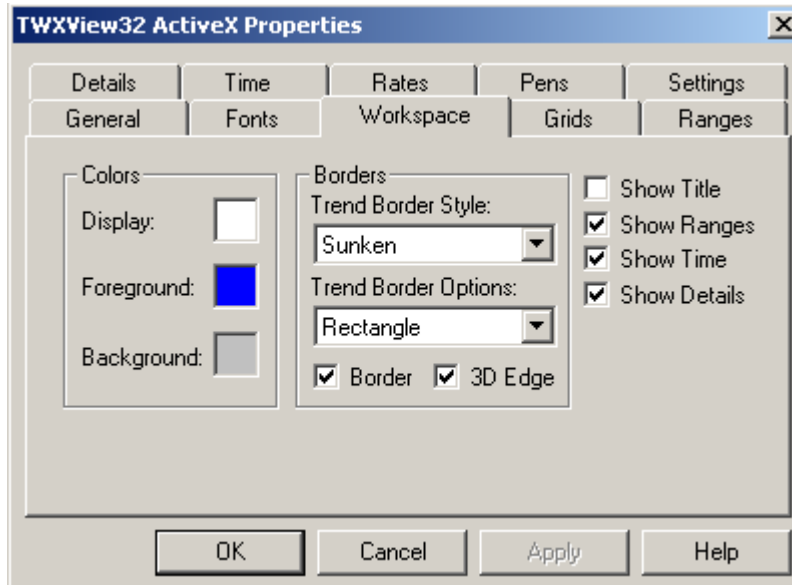
Circular Charts

You can also use the **Fonts** tab to change the fonts used in circular charts. Simply select the proper option in the **Font Selection For** drop-down list. Selecting **Ranges** from the drop-down list allows you to set the font for data labels on radial grids. You can also set the font and font size for **Time Settings** and **Title**.

Workspace Tab

The **Workspace** tab of the **Properties** dialog box, shown below, configures various cosmetic aspects of the trend display, such as showing range information and details about the trend.

You can select which components of the trend you want to appear on the display. Additionally, you can select **Trend Border Style** and the **Trend Border Options**. You can also give the border a three-dimensional look.



Properties Dialog Box: Workspace Tab

Colors

In the **Colors** section of the **Workspace** tab, you can choose the colors for the **Background**, **Foreground** (the title and text displaying the type of trend), and **Display** (the actual chart). To change the color, simply click the box corresponding to the area that you want to change. This opens a **Color** dialog box, which enables you to select a predetermined color. You may also select the **Define Custom Colors** button to create a color.

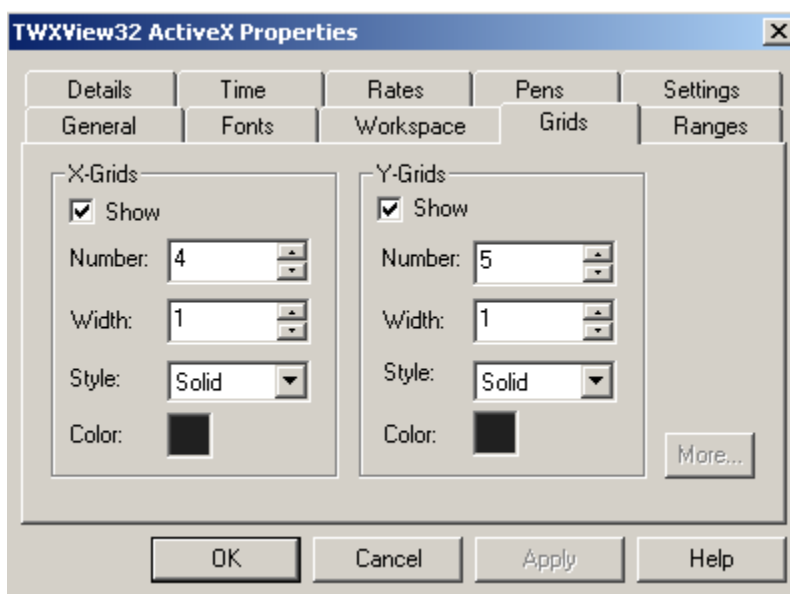
Borders

The **Borders** section of the **Workspace** tab defines the type of border and placement the display will have. There are four options for the **Trend Border Style**: Bumped, Etched, Raised, and Sunken. In the bumped style, only the border looks raised, while the etched style is mostly flat. In the raised style, the border looks as if it is coming out of the window. In the sunken style, the border looks as if it is going into the window. The **Trend Border Options** list enables you to select where you want the border to be placed: Bottom, Bottom Left, Bottom Right, Left, Rectangle, Right, Top, Top Left, or Top Right.

Grids Tab

The **Grids** tab of the **Properties** dialog box, shown below, manages the grid configuration for the trend display. Depending on the selected plot type, some options may or may not be available.

The **Grids** tab configures the appearance of the grid on the display. If the **Show** box is checked, the grid will appear on the display. The **Number** field specifies the number of lines you want to appear for each axis, and the **Width** field specifies how wide the lines are. You can select a style for the line from the following options: **Dashed**, **Dotted**, **Dotted Dash**, or **Solid**. As in the **Workspace** tab, you can select a color for the grid by clicking the color box and choosing a color. TrendWorX32 conveniently provides for separate configuration for the x-axis and y-axis, with the exception of the bar plot, which does not support x-axis grids.



Properties Dialog Box: Grids Tab

Circular Charts

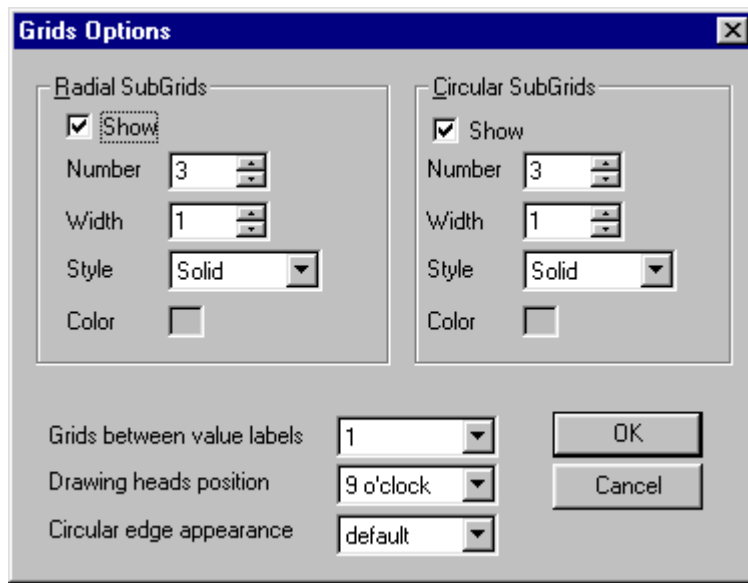
If you are using a circular chart, the following dialog box appears. Select the

Grids tab to modify the color and style of the radial grids and circular grids. Using this tab, you can elect to show or not to show radial and circular grids. You can also choose the **Number**, **Width**, **Style**, and **Color** of the grids.



Grids Tab for Circular Charts

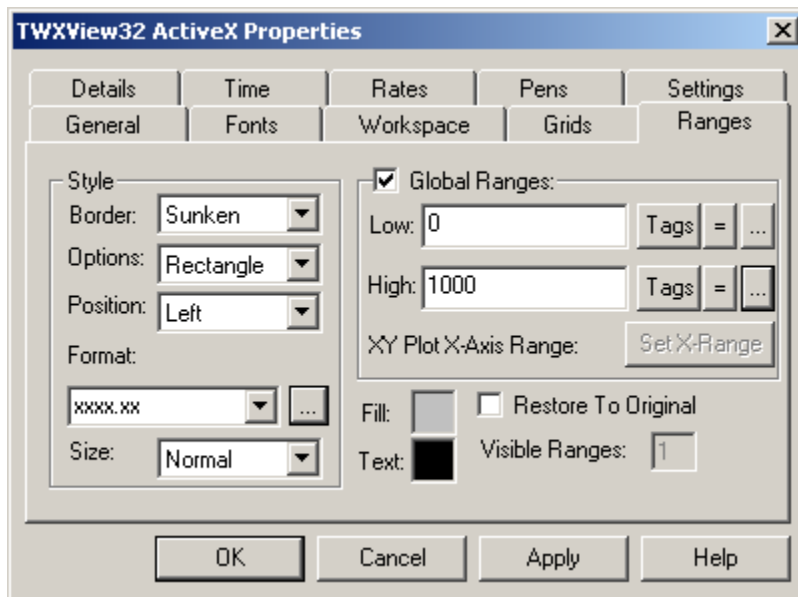
Click **More** to open the **Grids Options** dialog box, shown below, which enables you to change the number, width, style, and color of the radial and circular subgrids. You can elect to show or not to show (enable or disable) the radial and circular subgrids. You can choose the number, width, style and color of the radial and circular subgrids. You can also set the number of grids to be shown between value labels, whether they are grids or subgrids. You can set the position of the drawing heads by selecting one of the options from the **Drawing Heads Position** drop-down list. You can also select the edge thickness of the circle.



Grids Options Dialog Box for Circular Charts

Ranges Tab

The **Ranges** tab of the **Properties** dialog box, shown below, manages the Ranges Configuration for the trend display. Depending on the selected plot type, some options may or may not be available.



Properties Dialog Box: Ranges Tab

Style

The **Style** section of the **Ranges** tab defines the appearance of the ranges in the trend display. As in the **Workspace** tab, the **Border** and **Options** fields allow you to choose the border style and placement for ranges. The **Position** field determines where the ranges field will be placed in the **Trend** window (right or left), with the exception of the Strip Chart, in which the only range values are x-ranges that cannot be moved from left to right. The **Format** field determines how the numbers representing the ranges will appear (decimals or integers). The **Size** field indicates the size of the entire ranges filed in the trend display.

To select an alias to use for the **Format**, click the ... button to the right of the text box and select either **Global Aliases** or **Language Aliases** from the pop-up menu, as shown in the figure below. This opens the respective alias browsers, allowing you to choose from a list of available aliases.



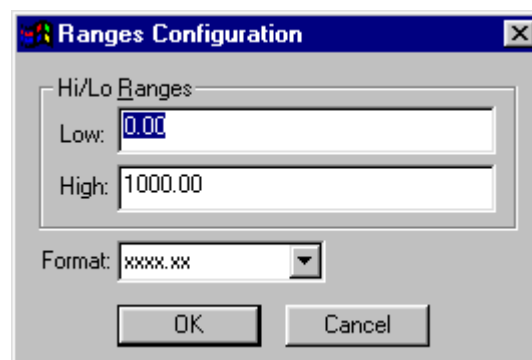
Selecting Aliases for the Format Field

Global Ranges

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. To select an OPC tag for the low or high ranges, click the **Tags** button. To select an expression, click the **=** button to open the Expression Editor. To select a global alias, click the **...** button and select **Global Aliases** from the pop-up menu. This opens the Global Alias browser, allowing you to choose from a list of available aliases.

In the Unicode version of the TrendWorX32 Viewer, if **Global Ranges** are enabled, the **Format** field can be language-aliased to produce the desired number formatting, font, and scaling (if a scaling expression is available in the Language Configurator).

If you are using an XY plot, the **Set X-Range** button will be available. Clicking this button opens the **Ranges Configuration** dialog box, which specifies the **High** and **Low** ranges, as well as the **Format**, for the x-axis.



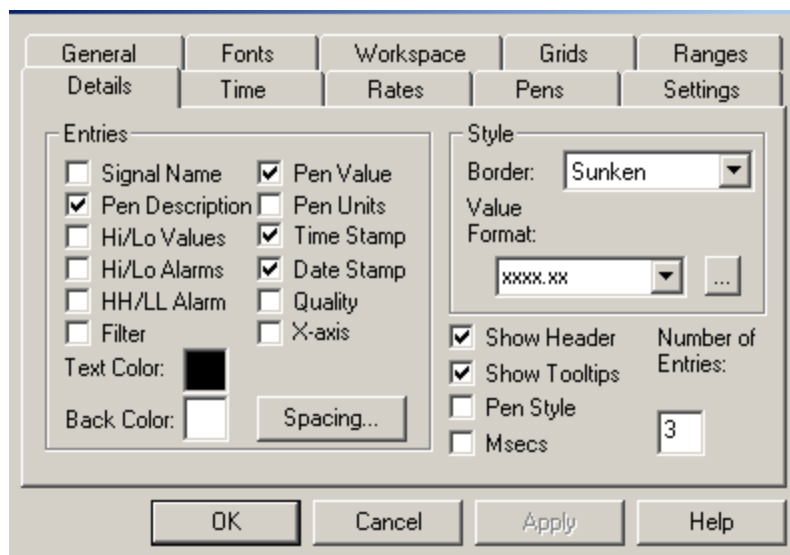
Ranges Configuration for XY Plots

Other Options

You can choose the **Fill** and **Text** color by clicking the corresponding box and selecting the color you want. The **Global Ranges** box is checked by default, but if you clear the selection, you will be able to change the number of visible pens. If **Restore To Original** is checked, TrendWorX32 maintains the ranges even if you change something in runtime. The **Visible Ranges** sets how many range scales will be visible in the trend display. When the **Global Ranges** option is enabled, you can define constant values for the global ranges or have them evaluated from OPC real-time tags or expressions.

Details Tab

The **Details** tab of the **Properties** dialog box, shown below, manages the detail display information area, and the enabled choices closely match the selected plotting type.



Properties Dialog Box: Details Tab

Entries

The **Entries** section of the **Details** tab provides the following list of items that may appear in the details section of the **Trend** window: **Signal Name**, **Pen Description**, **Hi/Lo Values**, **Hi/Lo Alarms**, **HH/LL Alarm**, **x-Axis**, **Pen Value**, **Pen Unit**, **Time Stamp**, **Date Stamp**, and **Quality**. The items will appear in the order that they are listed. For example, if **Pen Value** and **Time Stamp** are checked, **Pen Value** will always appear before **Time Stamp** in the window.

The **Show Header** item is listed in addition to the options above. If this box is checked, the header for the details section will appear, and you will be able to see what all the data in the display refer to. Otherwise the data will be displayed without any indication of what they correspond to. The **Filter** attribute can be used during historical replay to look at the averaging method used for data retrieval. The options enabled here are automatically used in the tool tip support of the viewer.

The **Spacing** button allows you to configure a default format for the **Details** window to be recalled during runtime mode.

Style

The **Style** section of the **Details** tab is similar to that of the **Ranges** tab, with a few additions. The time and date format is established using the Windows Registry settings for time and date, based on the International Language Resource .dll currently loaded in the TrendWorX32 Viewer ActiveX. You can also choose the **Border** attributes for the **Details** window.

In the Unicode version of the TrendWorX32 Viewer, the **Format** field can be language-aliased to produce the font for the **Details** window. Note that the formatting of the values of each pen is performed on a pen-by-pen basis, independently of the details formatting. To select an alias to use for the **Value Format**, click the ... button to the left of the text box and select either **Global Aliases** or **Language Aliases** from the pop-up menu, as shown in the figure below. This opens the respective alias browsers, allowing you to choose from a list of available aliases.

Other Options

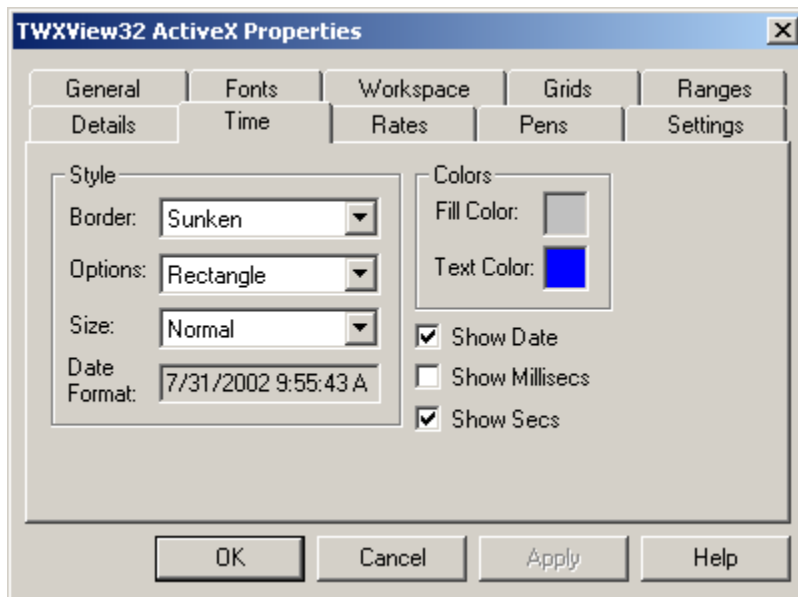
The remaining options allow you to change the number of visible pens, the fill color, and the text color and to choose whether to show tool tips.

The **Number of Entries** field displays the number of pens in the **Details** window of the Viewer. If more pens exist, the **Details** window will show a scroll bar for scrolling up or down. If **Show Header** is checked, the **Details** window header will be displayed. If **Show Tool Tips** is checked, tool tips indicating the sample values will be displayed as the cursor moves across the display area in freeze mode.

If **Msecs** is checked, the **Details** window will append the milliseconds value to each time stamp displayed. If **Pen Style** is checked, the **Details** window will draw a line in front of the pen description text in the same style, width, and color as the pen curve. This way, signals drawn with different pen styles can be easily identified in monochrome printouts.

Time Tab

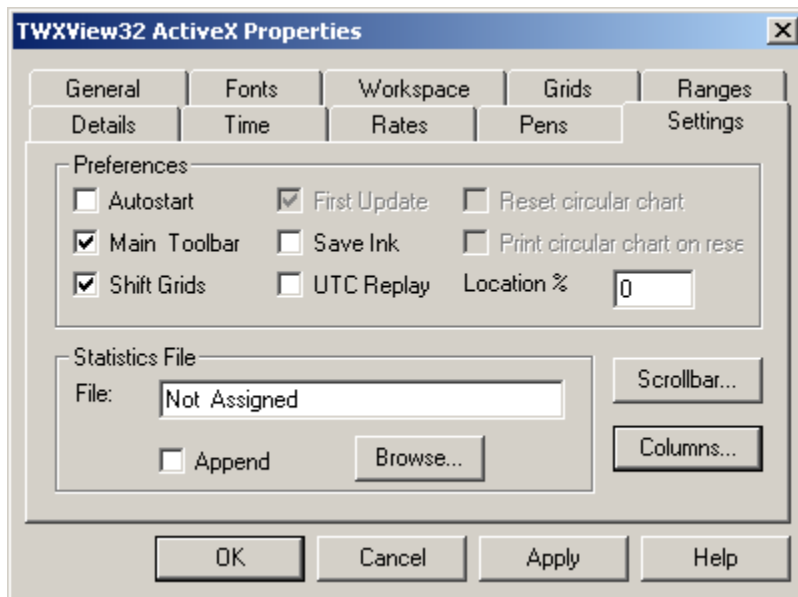
The **Time** tab of the **Properties** dialog box, shown below, configures the style of the time display inside the trend. You can select the border and size parameters, as well as the text and fill colors. You can also choose to show or not to show the date, seconds, and milliseconds during runtime mode.



Properties Dialog Box: Time Tab

Settings Tab

The **Settings** tab of the **Properties** dialog box, shown below, manages various aspects of the trend configuration.



Properties Dialog Box: Settings Tab

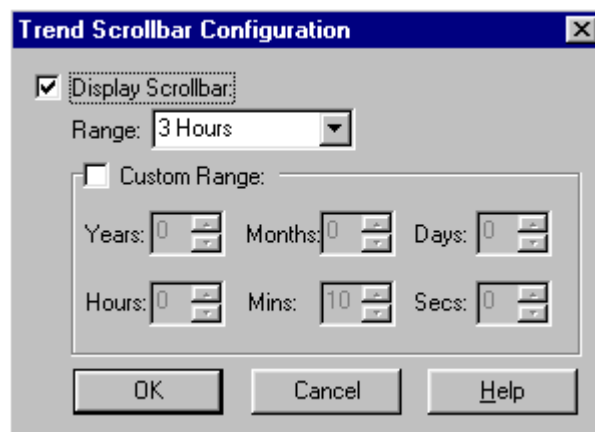
If **Autostart** is checked, the TrendWorX32 Viewer ActiveX will automatically enter animation mode when the container application switches from design mode to use mode. Note that in the TrendWorX32 and AlarmWorX32 containers, this option is disabled, and both applications will start the ActiveX Viewers through OLE Automation.

If **Main Toolbar** is checked, you can access the **Trend Viewer** toolbar during runtime mode. When **Shift Grids** is checked, the relative location of the grids on the axis is time-scale based. You can also enable the restoration of range information that may have been changed during runtime mode. Choosing startup determines a relative plot of each signal on the display based on the **Location %** value. The default value is 0, and the pen curve will begin plotting from the far right side of the trend display.

Under the **Statistics File** section, you can click the Browse button and select a .csv file for saving statistical information during runtime. Check **Append** if you want to append the statistical information to the .csv file or overwrite the existing one.

Selecting **Save Ink** will conserve ink while printing by not using the configured background color fill of the individual areas within the trend display. The curves and text, however, will be printed in the configured colors. Also, if you select **Print Circular Chart on Reset**, and the **Reset Circular Chart** after full period option is enabled, the TrendWorX32 Viewer will automatically print the circular chart display.

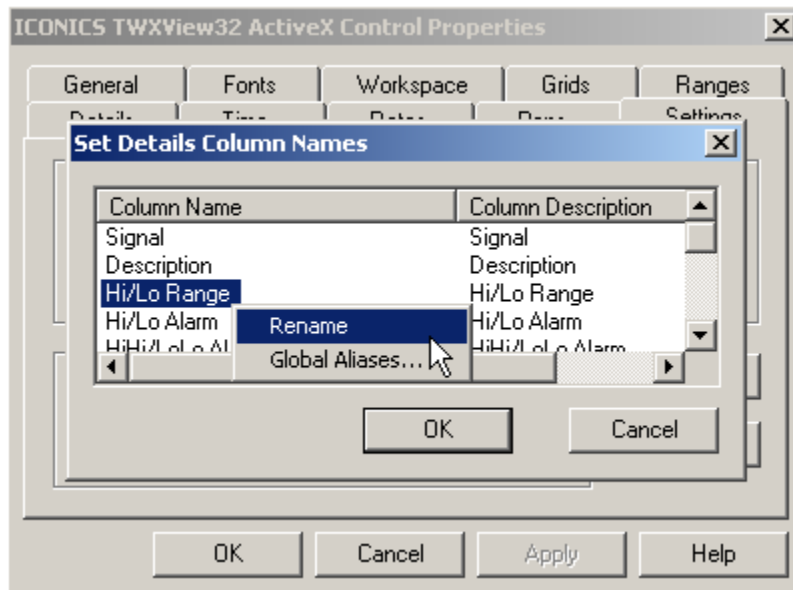
Click the **Scroll Bar** button to set the time range for how far back you will be able to scroll while in freeze mode. This opens the **Trend Scroll Bar Configuration** dialog box is displayed as shown below. If **Display Scrollbar** is checked, you must select a range for how far back you can scroll. By default, the scroll range is set at 3 hours, but the options of 10 seconds, 30 seconds, 1 minute, 2 minutes, 10 minutes, 30 minutes, 1 hour, 8 hours, 1 day, 5 days, 1 week, 1 month, 3 months, 6 months and 1 year are also available from this list. If you wish to enter a more-specific range you can check **Custom Range** and select the range in years, months, days, hours, minutes, and seconds.



Trend Scroll Bar Configuration Dialog Box

Clicking the **Columns** button on the **Settings** tab opens the **Set Details Column Names** dialog box, as shown in the figure below. This dialog specifies the column names for the details section of the Trend Viewer. Both the column names and descriptions are listed, but you can only modify column names. To change a column name, right-click on the name and select **Rename** from the pop-up menu, as shown in the figure below. Your changes are reflected in the Viewer.

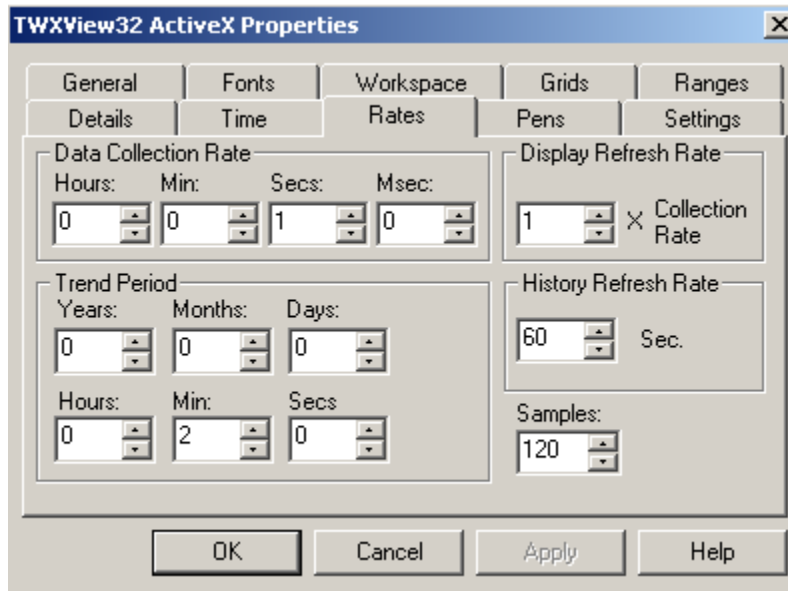
Note: When you right-click on a column name, you can also select a global alias from the Global Alias browser, or a language alias (Unicode version only) from the Language Alias browser.



Setting the Column Names for the Viewer

Rates Tab

In the **Rates** tab of the **Properties** dialog box, shown below, 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.



Properties Dialog Box: Rates Tab

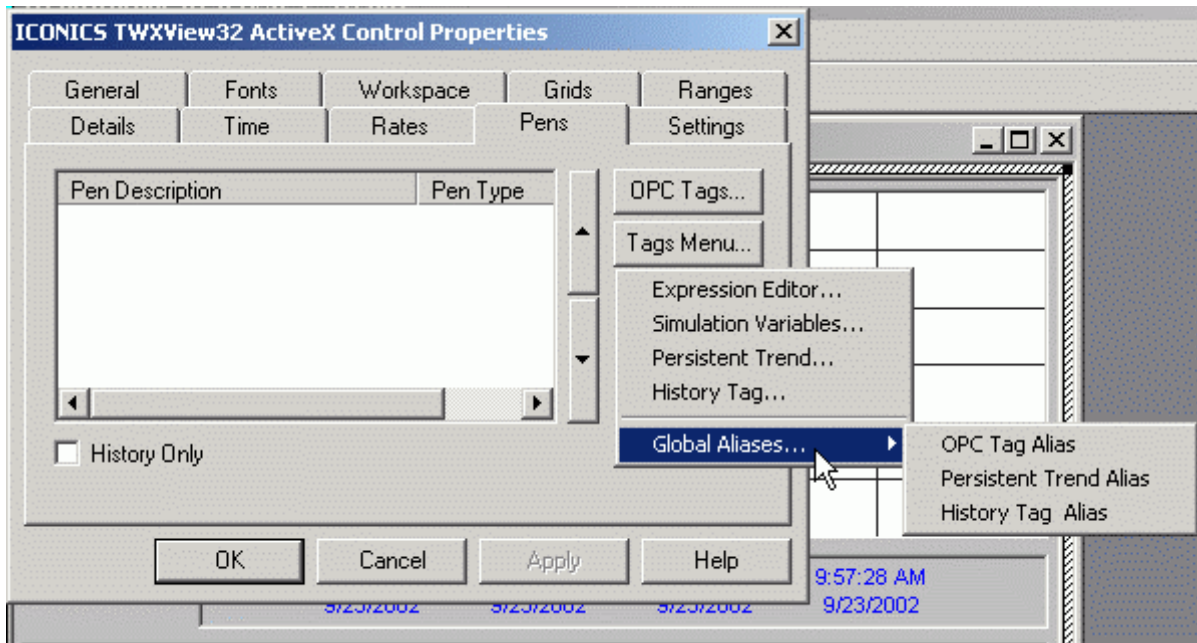
The **Display Refresh Rate** controls the speed at which the signal curves will be drawn. If, for example, the data-collection rate is set to 250 ms and the **Display Refresh Rate** is set to 4, the samples will be collected at 250 ms, but the signal curve will be updated every second without loss of 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 min., then the History Refresh Rate should be set to a maximum of 1 min.

Pens Tab

The **Pens** tab of the **Properties** dialog box, shown below, handles the signal connection, editing, and configuration for the data displayed in the

trend. Clicking **OPC Tags** opens the **OPC Universal Tag Browser**.



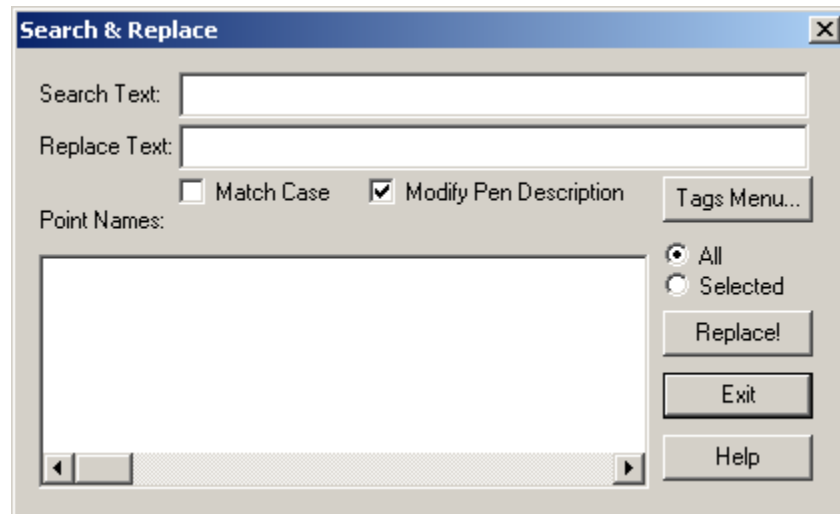
Properties Dialog Box: Pens Tab

To display only historical data, in freeze mode you must check **History Only**. Then you can make connections to OPC HDA servers. If this option is not set, you can make both real-time and historical tag connections on the same TrendWorX32 ActiveX Viewer. You can connect signals to pens, which are coming from an OPC server or are the result of a calculated expression to show the real-time data. This is illustrated in the following sections:

- **OPC Tags**
- **Tags Menu**
- **Edit Button**

You can also delete, modify, and replace pens. To delete one or more pens, select the pen and then click **Delete** on the **Pens** tab. Then click **Yes** to delete the pen. To replace or modify the pen, click **Replace** on the **Pens** tab. This opens the **Search and Replace** dialog box, shown below. You can

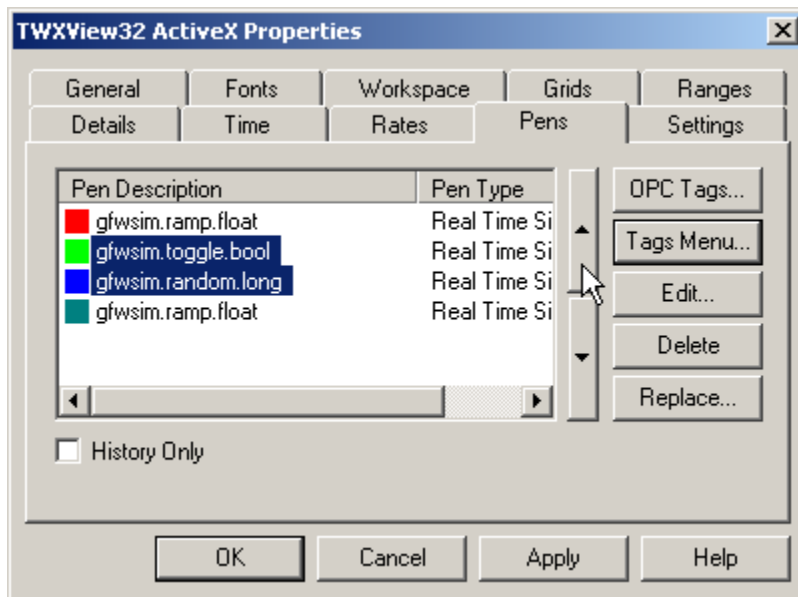
search for and replace text as needed to modify the pen.



Search and Replace Dialog Box

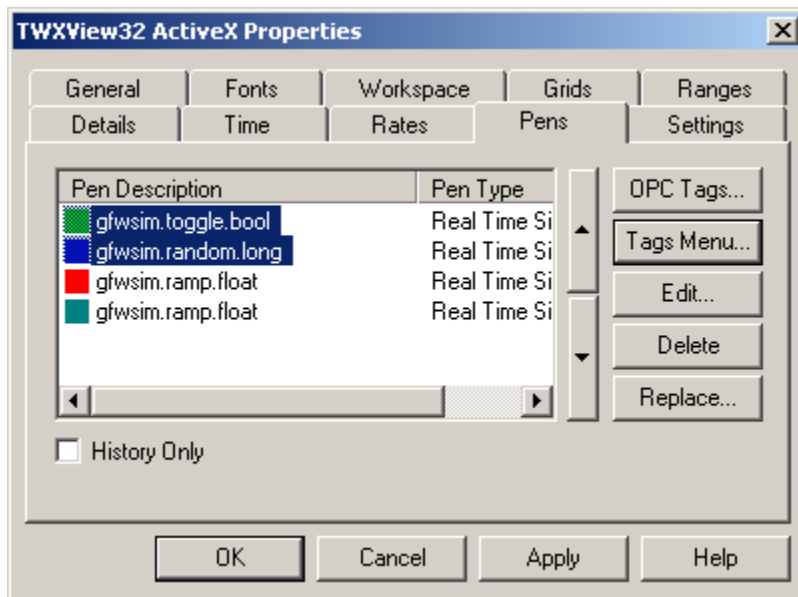
Reordering Trend Pens

The TrendWorX32 Viewer ActiveX now supports reordering of pens in configuration mode. In the **Pens** tab, select the trend pen(s) you would like to reposition, as shown in the figure below.



Reordering Trend Pens

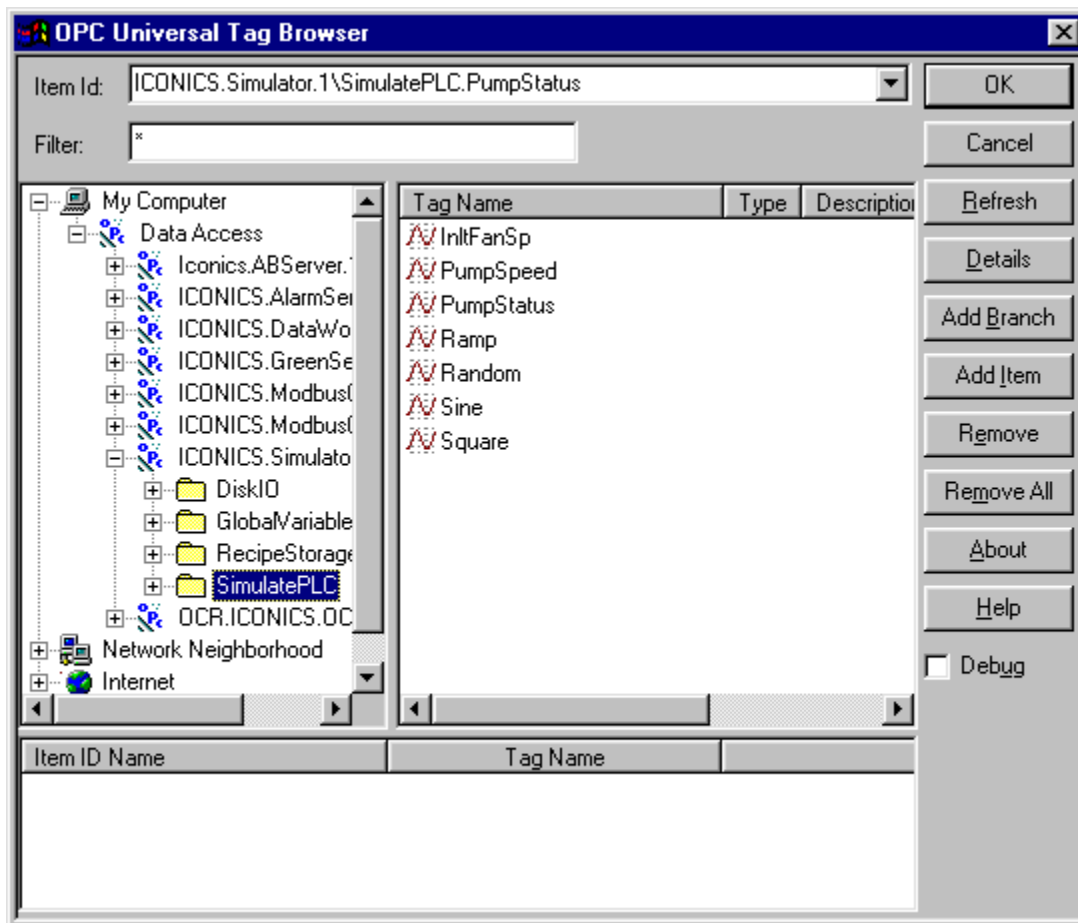
Then click the up or down arrow buttons to reposition the selected pen(s). The pens move up or down, as shown in the figure below.



Trend Pens Repositioned

OPC Tags

To make real-time connections from the **Pens** tab of the **Properties** dialog box, click the **OPC Tags** button. This opens the **OPC Universal Tag Browser**, which enables you to browse all available OPC servers and their signals, as shown below.



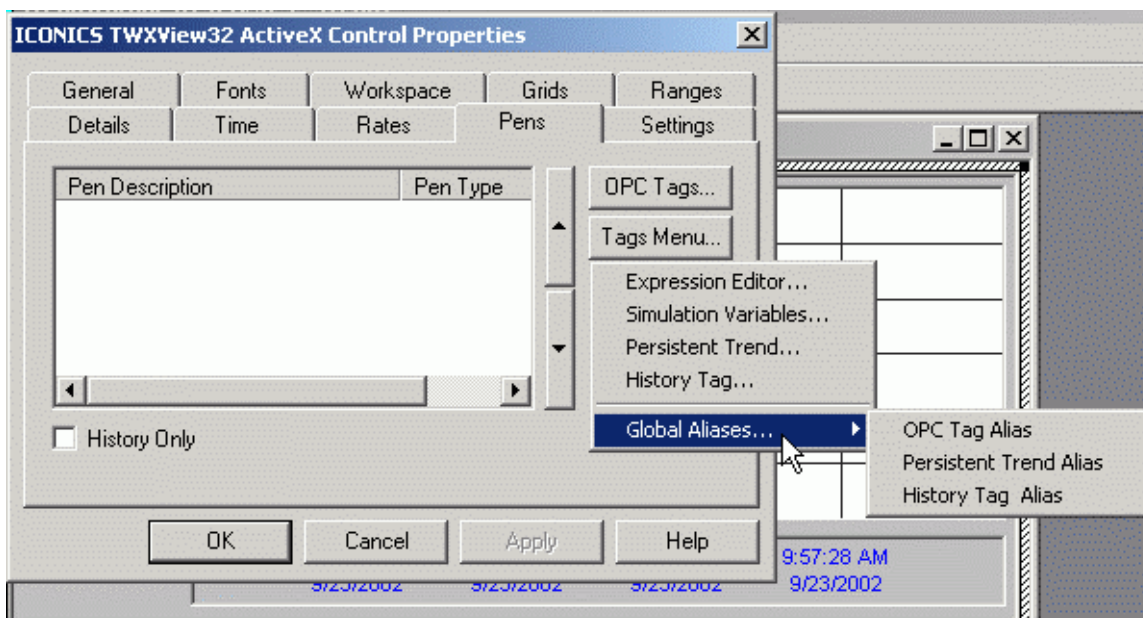
OPC Universal Tag Browser

You can make multiple tag connections through the OPC Universal Tag Browser. To do so, first expand the server "tree" and then select the tags you wish to connect from the expanded tree view. Then, select **Add Item** to

place the selected tags on the bottom list control. Once all selections are made, click **OK** to establish the connections.

Tags Menu

Clicking **Tags Menu** in the **Pens** tab of the **Properties** dialog box opens the following menu, as shown in the figure below.



Tags Menu

The **Tags** menu contains the following basic options:

- **Expression editor**
- **Simulation variables**
- **Persistent trend**
- **History tag**

In addition to these options, the following new global alias options have been added to allow for global aliasing connections:

- ☐ **OPC Tag Alias**
- ☐ **Persistent Trend Alias**

□ **History Tag Alias**

You can now select these options to connect to various alias types.

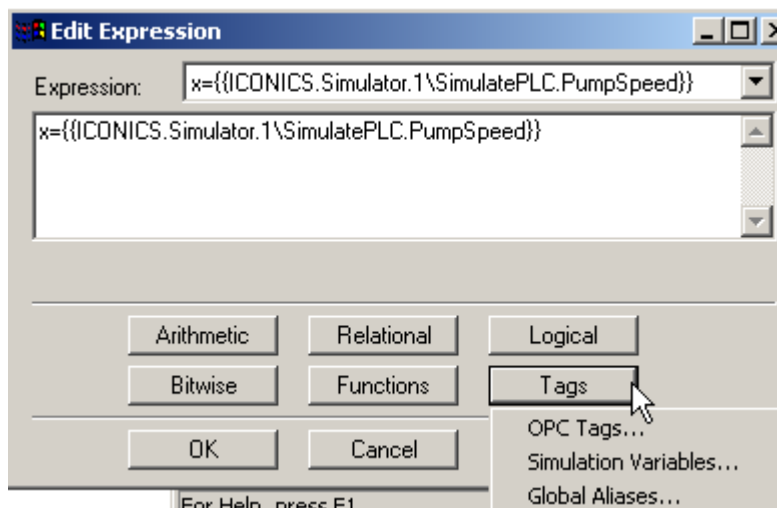
Selecting **OPC Tag Alias** opens the **Global Alias Browser**, where you can select from a list of configured global aliases.

If you select **Persistent Trend Alias**, you will be asked to edit the pen, after the alias connection is made, and alias the Persistent Trend group for proper operation.

If you select a **History Tag Alias** pen, you will be prompted to first pick a historical tag from an OPC HDA server and then select the alias to use. After the connection is made, you will be asked to edit the pen and select the remaining aliases.

Expression Editor

Selecting **Expression Editor** from the **Tags Menu** in the **Pens** tab of the **Properties** dialog box launches the **Edit Expression** dialog box, shown in the figure below. The Expression Editor is available to assist you in creating expressions for your GENESIS32 applications. The window is resizable and can be stretched as well as maximized or minimized. The drop-down list at the top of the **Edit Expression** dialog box keeps track of the last 50 expressions you have entered. The expression entered most recently is the first one in the drop-down list.



Edit Expression Dialog Box

Writing Expressions

An **expression** is a string that defines and evaluates a data connection between a client and an OPC server. During runtime mode, OPC servers resolve the data value for the expression. To indicate that a data connection is an expression, precede the string with the "**x=**" token, as shown below:

```
x={{ICONICS.Simulator.1\SimulatePLC.PumpSpeed}}
```

You can either type your expressions directly into the text box of the **Edit Expression** dialog box, or you can use the symbols and functions provided that help you use the proper string syntax when writing expressions. The following categories are available:

-
- **Arithmetic**
 - **Relational**
 - **Logical**
 - **Bitwise**
 - **Functions**
 - **Tags**

OPC Tags

An **OPC tag**, or data point, is a data connection between a client and an OPC server. OPC tags can be used in expressions when the tag is embedded between double brackets, as shown below:

`{{tag_name}}`

Example:

`x={{ICONICS.Simulator.1\SimulatePLC.PumpSpeed}}`

You can use the Tag Browser to select OPC tags.

Aliases

An **alias** is a string that represents or describes an object or data point in a display. Both local and global aliases can be used in expressions.

Local Aliases

For local aliases within the expression, use the following syntax:

`<<local_alias_name>>`

Example:

x=<<TankLevel>>

Global Aliases

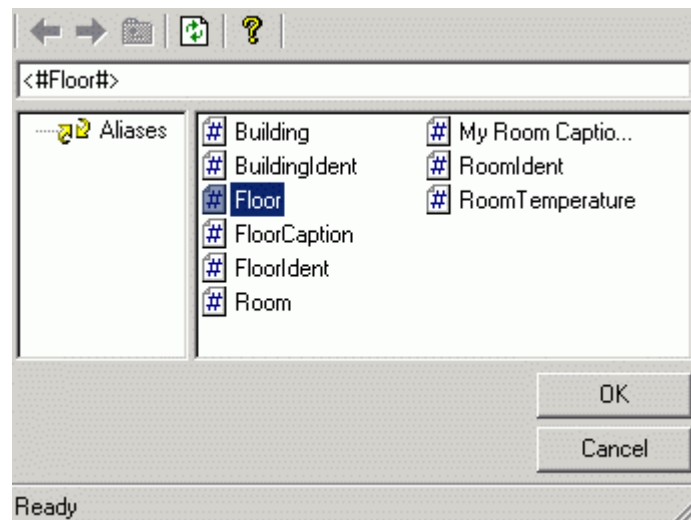
For global aliases within the expression, use the following syntax:

<#global_alias_name#>

Example:

x=<#RoomTemperature#>

Selecting **Global Alias Browser** opens the Global Alias Browser, as shown in the figure below. Select a global alias from the Global Alias Browser, which includes all global aliases in the global alias database. This eliminates the need to manually type in the alias name. All global aliases that are configured in the Global Alias Engine Configurator are conveniently available to choose from inside the browser. The tree control of the Global Alias Engine Configurator is mimicked in the tree control of the Global Alias Browser. Select a global alias by double-clicking the alias name (e.g. "Floor" in the figure below). The alias name appears at the top of the browser, which automatically adds the <# and #> delimiters to the alias name. Click the **OK** button.



Selecting an Alias From the Global Alias Browser

Language Aliases

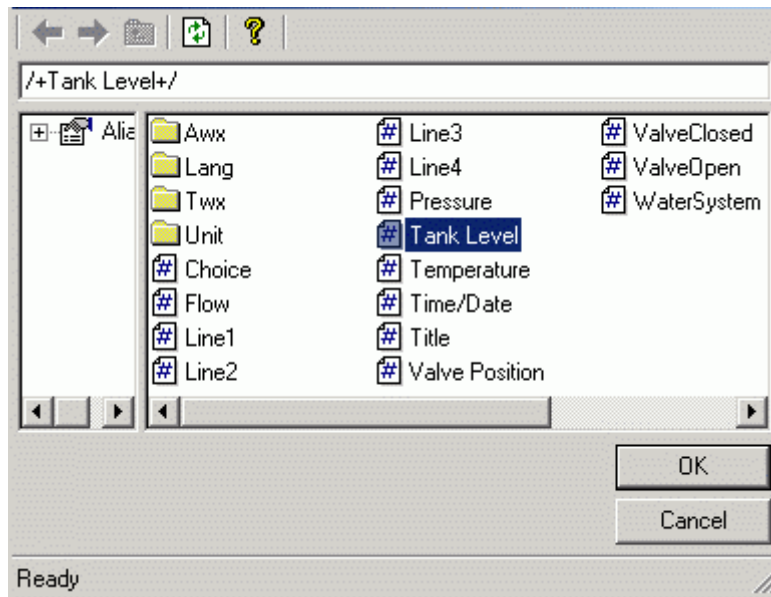
For language aliases within the expression, use the following syntax:

`<#language_alias_name#>`

Example:

`x=/+WaterSystem+/-`

Selecting **Language Alias Browser** from the pop-up menu opens the Language Alias Browser, as shown in the figure below. The browser includes all languages aliases in the language database. All language aliases that are configured in the Language Configurator are conveniently available to choose from inside the browser. The tree control of the Language Configurator is mimicked in the tree control of the Language Alias Browser. Select a language alias by double-clicking the alias name. The alias name appears at the top of the browser, which automatically adds the `/+` and `+/-` delimiters to the alias name. Click the **OK** button.



Selecting an Alias From the Language Alias Browser

Variables

Variables can be used in expressions. How the variable needs to be referred to depends on the type of variable. A local variable can be used in expressions when the variable is embedded between double tildes.

Local Variables

For local variables within the expression, use the following syntax:

~~local_variable_name~~

Example:

x=~~Setpoint~~

Simulation Variables

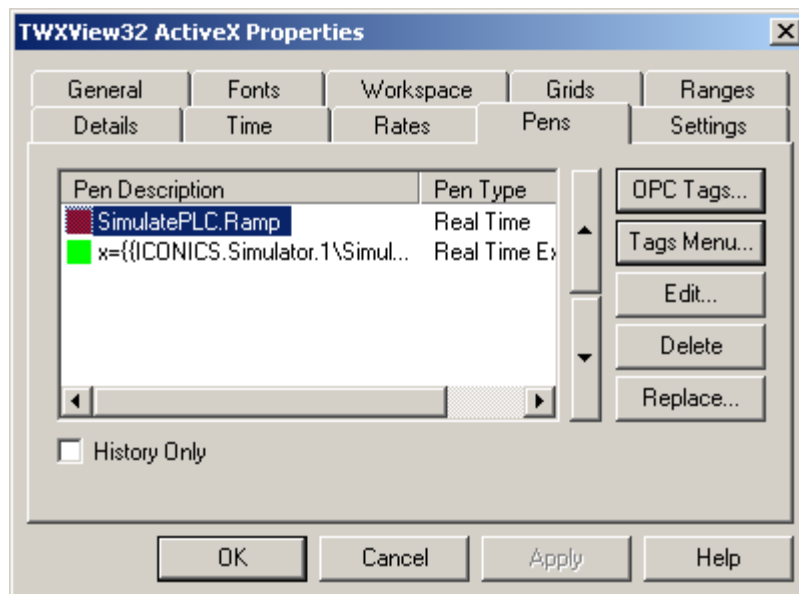
For simulation variables within the expression, use the following syntax:

`{{simulation_variable_name}}`

`x={{gfwsim.random.long}}`

Editing the Trend Pen

Once you have configured a pen, you can double-click the pen in the **Pens** tab list, as shown below. This opens the **Edit Trend Pen** dialog box, which allows you to edit many options specific to that pen.



Selecting Pens To Edit

Arithmetic

The symbols '+', '-', '*', '/' and '%' use the following format:

expression :: parameter **symbol** parameter

Where

Parameter	A local variable, an OPC tag, a constant, or another expression
Symbol	+ or - or * or / or %

Result

The expression results in a number of any type (float, long, etc.).

Examples

Symbol	Description	Example	Result
+	Addition	~~var1~~ + ~~var2~~	8+3 = 11
-	Subtraction	~~var1~~ - ~~var2~~	8-3 = 5
*	Multiplication	~~var1~~ * ~~var2~~	8*3 = 24
/	Division	~~var1~~ / ~~var2~~	8/3 = 2.66667
%	Calculates the remainder after division	~~var1~~ % ~~var2~~	8%3 = 2
(and)	Gives precedence to parts of the calculation	~~var1~~ / (~~var2~~ + ~~var3~~)	8/(3+2) = 1.6

Relational

Clicking **Relational** in the **Edit Expression** dialog box allows you to edit expressions using the following relational functions:

The symbols '<', '>', '<=', '>=', '==' and '!=' use the following format:

expression :: parameter **symbol** parameter

Where

Parameter	A local variable, an OPC tag, a constant, or another expression
Symbol	< or > or <= or >= or == or !=

Result

The expression results in a Boolean value (0 or 1).

Examples

Symbol	Description	Example	Result
<	Less than	~~var1~~ < ~~var2~~	8<3 = 0
>	Greater than	~~var1~~ > ~~var2~~	8>3 = 1
<=	Less than or equal to	~~var1~~ <= ~~var2~~	8<=3 = 0
>=	Greater than or equal to	~~var1~~ >= ~~var2~~	8>=3 = 1
==	Equal to	~~var1~~ == ~~var2~~	8==3 = 0
!=	Not equal to	~~var1~~ != ~~var2~~	8!=3 = 1

Logical

The symbols '&&' and '||' use the following format:

expression :: parameter **symbol** parameter

The symbol '!' uses the following format:

expression :: **symbol** parameter

Where

Parameter	A local variable, an OPC tag, a constant, or another expression
Symbol	&& or or !

Result

The expression results in a Boolean value (0 or 1).

Truth table

~~var1~~ ~~var2~~	0		not 0	
	0	not 0	0	not 0
~~var1~~ && ~~var2~~	0	0	0	1
~~var1~~ ~~var2~~	0	1	1	1
!~~var1~~	1	1	0	0

Examples

Symbol	Description	Example	Result
--------	-------------	---------	--------

Symbol	Description	Example	Result
&&	And	~~var1~~ && ~~var2~~	8 && 3 = 1
	Or	~~var1~~ ~~var2~~	8 3 = 1
!	Not	!~~var1~~	!8 = 0

Bitwise

The symbols '&', '|', and '^' of the bitwise group use the following format:

expression :: parameter **symbol** parameter

The symbol '~' of the logical group uses the following format:

expression :: **symbol** parameter

The symbols 'shl' and 'shr' of the bitwise group use the following format:

expression :: **symbol** (value, shift by)

Where

Parameter	A local variable, an OPC tag, a constant, or another expression
Symbol	&& or or ^ or shl or shr or ~

Result

The expression results in a number when the parameters used contain numbers.

Bit Table

	Binary (Decimal)	Binary (Decimal)
~~var1~~	0000.0000.0000.1000 (8)	0000.0000.0110.0000 (96)
~~var2~~	0000.0000.0000.1010 (10)	0000.0000.0000.1000 (8)
~~var1~~ & ~~var2~~	0000.0000.0000.1000 (8)	0000.0000.0000.0000 (0)
~~var1~~ ~~var2~~	0000.0000.0000.1010 (10)	0000.0000.0110.1000 (104)
~~var1~~ ^ ~~var2~~	0000.0000.0000.0010 (2)	0000.0000.0110.1000 (104)
shl (~~var1~~,3)	0000.0000.0100.0000 (64)	0000.0011.0000.0000 (768)
shr (~~var1~~,3)	0000.0000.0000.0001 (1)	0000.0000.0000.1100 (12)
~(~~var1~~)	1111.1111.1111.0111 (-9)	1111.1100.1111.1111 (-97)
bittest(~~var1~~,3)	0000.0000.0000.0001 (1)	0000.0000.0000.0000 (0)

Examples

Symbol	Description	Example	Result
&	Bit And	~~var1~~ & ~~var2~~	8 & 3 = 0
	Bit Or	~~var1~~ ~~var2~~	8 3 = 11
^	Bit eXclusive Or	~~var1~~ ^ ~~var2~~	8 ^ 3 = 11
shl	Bit shift left	shl (~~var1~~,3)	8 << 3 = 64

Symbol	Description	Example	Result
shr	Bit shift right	shr (~~var1~~, 3)	8>>3=1
~	Not (two's complement)	~ (~~var1~~)	! 8 = -9
bittest	Bit Test	bittest (5 , 0)	1

Note: The bittest function requires you to specify the position of the bit to be tested. You must indicate that it starts from 0. In other words, a bit position of "0" indicates the "less significant" bit.

Functions

The symbols 'sin', 'asin', 'cos', 'acos', 'tan', 'atan', 'log', 'ln', 'exp', 'sqrt', 'abs', 'ceil', and 'floor' use the following format:

expression :: **symbol** (parameter)

The symbols 'pow', 'min', and 'max' use the following format:

expression :: **symbol** (parameter,parameter)

The symbol 'if' uses the following format:

expression :: **symbol** (parameter,parameter,parameter)

Where

Parameter	A local variable, an OPC tag, a constant, or another expression
------------------	---

Symbol	sin, asin, cos, acos, tan, atan, log, ln, exp, sqrt, abs, ceil, floor, min, max, pow, or if
---------------	---

Result

The expression results in a number.

Examples

Symbol	Description	Example	Result
sin	sine of an angle in radians	sin(~~var1~~)	sin(0.785)=0.71
cos	cosine of an angle in radians	cos(~~var1~~)	cos(0.785)=0.71
tan	tangent of an angle in radians	tan(~~var1~~)	tan(0.785)=1.0
asin	arc sine returns an angle in radians	asin(~~var1~~)	asin(0.5)=0.52
acos	arc cosine returns an angle in radians	acos(~~var1~~)	acos(0.5)=1.05
atan	arc tangent returns an angle in radians	atan(~~var1~~)	atan(1)=0.785
sqrt	Returns the square root	sqrt(~~var1~~)	sqrt(100)=10
pow	Returns value 1 raised to the power value 2	pow(~~var1~~,~~var2~~)	pow(100,1.5)=1000
log	10 based logarithm	log(~~var1~~)	log(100)=2

Symbol	Description	Example	Result
ln	e based logarithm	ln(~~var1~~)	ln(7.389)=2
exp	Exponential	exp(~~var1~~)	exp(2)=7.389
abs	Absolute value	abs(~~var1~~)	abs(-1)=1
ceil	Integer ceiling	ceil(~~var1~~)	ceil(7.39)=8
floor	Integer floor	floor(~~var1~~)	floor(7.39)=7
min	Lowest value of two	min(~~var1~~,~~var2~~)	min(10,5)=5
max	Highest value of two	max(~~var1~~,~~var2~~)	min(10,5)=10
if	Conditional statement	if(~~var1~~<~~var2~~,~~var1~~,~~var2~~)	if(5<8,5,8)=5
like	Wildcard string compare	Like(string, pattern, casesensitive')	
quality	Quality of tag or expression	See below.	See below.
tostring	Type conversion	See below.	See below.
0x	Hexadecimal constant	x=0x11	17
0t	Octal constant	x=0t11	9
0b	Binary constant	x=0b11	3

Note: For the like operator: "string" equals the string to search in; "pattern" equals the string to search for (can include wildcards); nonzero for case-sensitive search; zero for case-insensitive search. String syntax is \$"string"\$.

You can use these special characters in pattern matches in string:

-
- ? Any single character.
 - Zero or more characters.
 - # Any single digit (0-9).
 - [charlist] Any single character in charlist.
 - [!charlist] Any single character not in charlist.

Quality

The **quality** option on the **Functions** menu of the **Expression Editor** is used to evaluate the quality of an OPC tag or an expression.

The following general syntax is used for quality expressions:

x=quality(expression)

Note: The "(expression)" can also be a simple expression composed of a single tag.

The **quality** function returns the OPC quality of the string between parentheses as one of the following results:

- ☐ 192: quality is GOOD
- ☐ 64: quality UNCERTAIN
- ☐ 0: quality BAD

Note: The OPC Foundation establishes the value ranges for quality. There are actually varying degrees of quality:

- ☐ GOOD: 192-252
- ☐ UNCERTAIN: 64-191
- ☐ BAD: 0-63

For more information, refer to the *OPC Data Access Custom Interface Standard* available for download at the OPC Foundation's Web site, www.opcfoundation.org/.

Example Quality Expression

Expression	Result
x=quality({{ICONICS.Simulator.1\SimulatePLC.PumpStatus}})	192 (Quality GOOD)

The quality of an expression is determined through the evaluation of each single tag in the expression. Thus, if you have multiple tags in an expression (and each tag has a different quality), the result of the expression (i.e. 192 [GOOD], 64 [BAD], or 0 [UNCERTAIN]) corresponds to the quality of the tag with the lowest quality. If an expression contains a conditional statement (e.g. if, then, or else), then the result of the expression is affected only by the quality of the branch being executed.

Consider the following sample expression:

x= if (quality(tag1) == 192, tag1, tag2)

This expression can be read as follows:

"If the quality of tag 1 is GOOD (i.e. 192), then the expression result (x) is the value of tag 1. In all other cases (i.e. the quality of tag 1 is UNCERTAIN or BAD), the expression result (x) is the value of tag 2."

We can calculate the results for this expression using different qualities for tag 1 and tag 2, as shown in the figure below.

Case	Tag1 quality	Tag2 quality	Result	Result quality
1	GOOD	GOOD	Tag1	192 (GOOD)
2	GOOD	UNCERTAIN	Tag1	192 (GOOD)
3	GOOD	BAD	Tag1	192 (GOOD)
4	UNCERTAIN	GOOD	Tag2	192 (GOOD)

5	UNCERTAIN	UNCERTAIN	Tag2	64 (UNCERTAIN)
6	UNCERTAIN	BAD	Tag2	0 (BAD)
7	BAD	GOOD	Tag2	192 (GOOD)
8	BAD	UNCERTAIN	Tag2	64 (UNCERTAIN)
9	BAD	BAD	Tag2	0 (BAD)

In cases 1-3 above, the quality of tag1 is GOOD, and therefore the result of the expression is GOOD. Thus, the result of the expression is not affected by the quality of tag2 (the “else” branch of the expression), which is ignored.

In cases 4-6, the quality of tag1 is UNCERTAIN, and therefore the result of the expression is the quality of tag2.

In cases 7-9, the quality of tag1 is BAD, and therefore the result of the expression is the quality of tag2.

Note: The “quality()” function returns a value that represents the quality of the expression within the parentheses but is always GOOD_QUALITY. For example, if TAG1 is BAD_QUALITY then the expression “x=quality(TAG1)” will return 0 with GOOD_QUALITY.

The result of an expression is the minimum quality of the evaluated tag in the expression and is affected only by the quality of the conditional (if, then, or else) branch that is executed.

Consider the following sample expression:

x= if (TAG_01>0,TAG_02,TAG_03)

This expression can be read as follows:

"If the value of TAG_01 is greater than 0, then the expression result (x) is TAG_02. If the value of TAG_01 is less than or equal to 0, then the expression result (x) is TAG_03."

Let's assume that the following values and qualities for these tags:

TAG_01=5 with quality GOOD

TAG_02=6 with quality UNCERTAIN

TAG_03=7 with quality BAD

Because the value of TAG_01 is 5 (greater than 0), the expression result is TAG_02. Thus, the final expression result is 6, and the final expression quality is UNCERTAIN.

Type Conversion

The **tostring** option on the **Functions** menu of the **Expression Editor** takes the value of whatever item is in parentheses and converts it into a string as follows:

The value is +(value)+unit

It can be used to convert from number to string, and it can be very useful for string concatenation.

The proper syntax for the **tostring** option is:

x="\$The value is "\$ + tostring(value) + \$" unit"\$

Note: In the expression above, the word "unit" is placeholder text for a user-specified unit of measurement or variable (e.g. Watt, inches, meters, etc.).

Example Expressions Type Conversion

Expression	Result
x="\$The value is "\$ + tostring({{gfwsim.ramp.float}}) + \$" Watt"\$	"The value is 543.2345152 Watt"

Constants

The **Functions** menu of the **Expression Editor** supports constant values, including hexadecimal, octal, and binary formats.

Example Expressions Using Constants

Expression	Result
------------	--------

x=0x11	17
x=0t11	9
x=0b11	3

The **Expression Editor** conveniently inserts the 0x and 0t and 0b prefixes for you so do not have to recall them.

Interpreting and Translating Constants

The examples below show how values are calculated for each type of constant.

- **Hexadecimal:** $0x20A = 2 * (16^2) + 0 * (16^1) + 10 * (16^0) = 2 * 256 + 0 * 16 + 10 * 1 = 512 + 0 + 10 = 522$
- **Octal:** $0t36 = 3 * (7^1) + 6 * (7^0) = 3 * 7 + 6 * 1 = 21 + 6 = 27$
- **Binary:** $0b110 = 1 * (2^2) + 1 * (2^1) + 0 * (2^0) = 1 * 4 + 1 * 2 + 0 * 1 = 4 + 2 + 0 = 6$

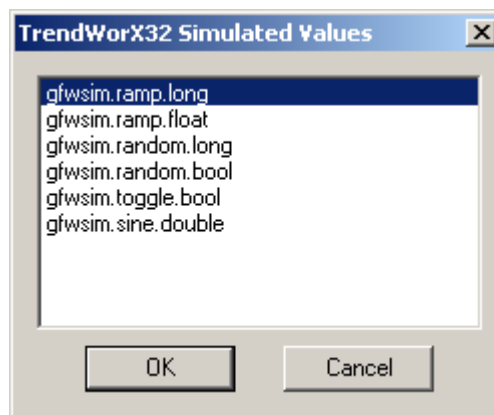
Simulation Variables

Selecting **Simulation Variables** from the **Tags** menu in the **Pens** tab of the **Properties** dialog box launches the **TrendWorX32 Simulated Values** dialog box, shown below, which lists all available simulated variables. Selecting one of these variables will create a pen attached to the simulated variable. These variables are helpful in testing the setup of your pens.

For simulation variables within an expression, use the following syntax:

`{{simulation_variable_name}}`

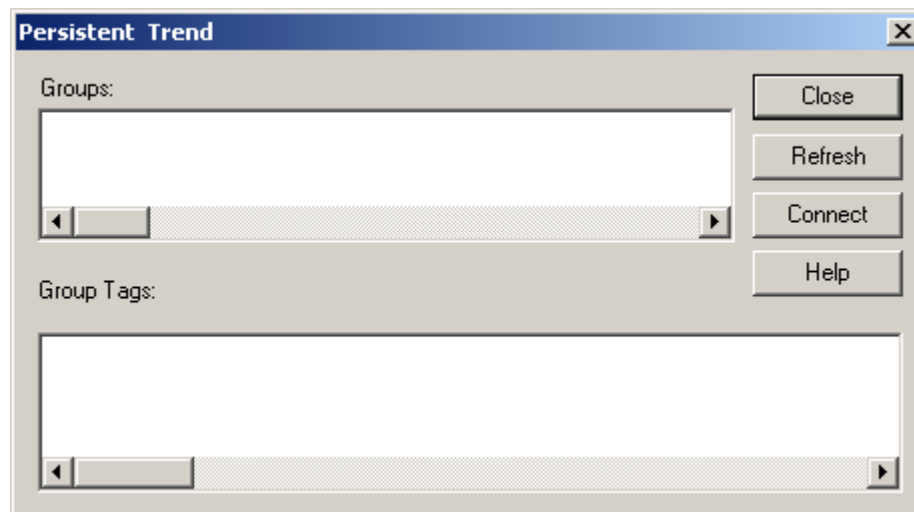
`x={{gfwsim.random.long}}`



Simulated Values Dialog Box

Persistent Trend

If the ICONICS Persistent Trending Automation Server is installed, configured, and running, selecting **Persistent Trend** from the **Tags** menu in the **Pens** tab of the **Properties** dialog box enables direct connections to the Persistent Trending server. A new dialog box appears, as shown below, showing all groups configured in the Persistent Trending server.



Persistent Trend Dialog Box

Double-clicking a group displays all signals configured for that specific group. You can select which signals to connect from the Persistent Trending server and click **Connect**. TrendWorX32 Viewer fills its buffer from the data existing in Persistent Trending prior to drawing.

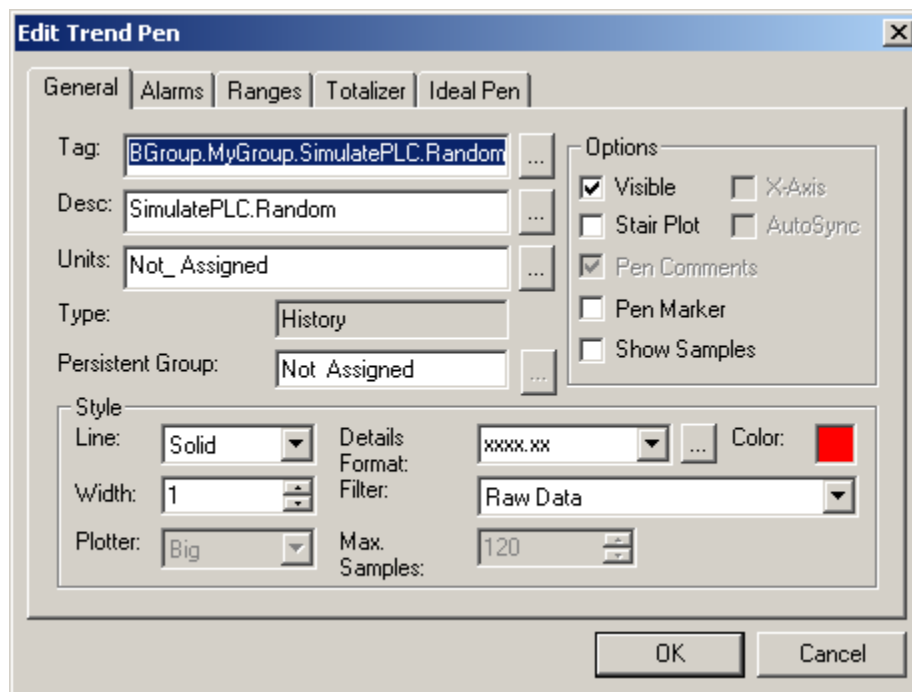
History Tag

Selecting **History Tag** from the **Tags** menu in the **Pens** tab of the **Properties** dialog box opens the **OPC Universal Tag Browser**. Then you can browse for OPC HDA tags.

Editing Pens

Clicking **Edit** in the **Pens** tab of the **Properties** dialog box opens the **Edit Trend Pen** dialog box, shown below, which contains the following tabs:

- **General tab**
- **Alarms tab**
- **Ranges tab**
- **Totalizer tab**
- **Ideal Pens tab**



Edit Trend Pen Dialog Box

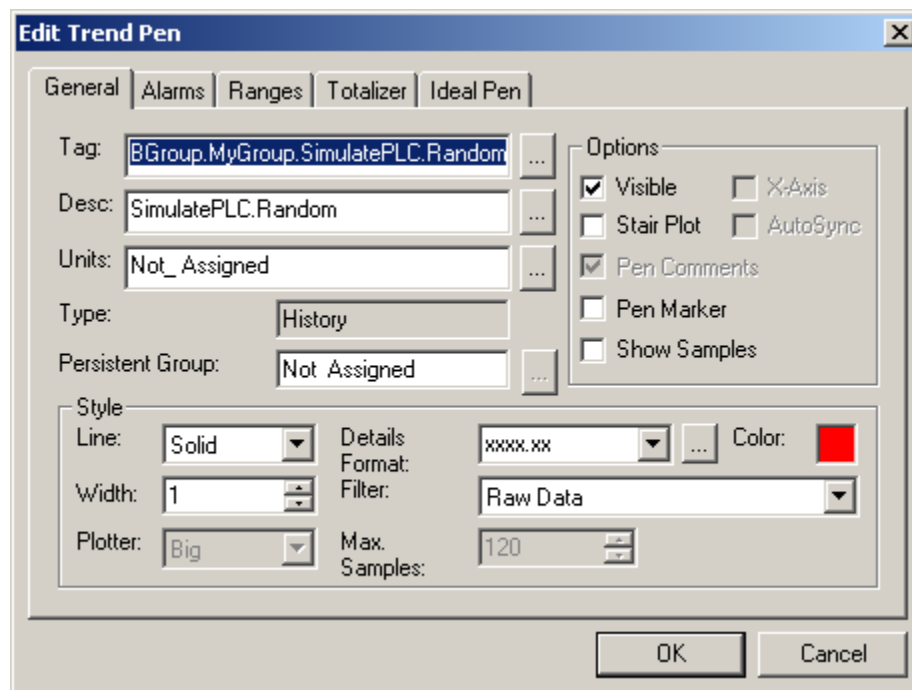
General Tab

The **General** tab, shown below, is the default tab of the **Edit Trend Pen**

dialog box. This tab allows you to change the tag with which the selected is associated, the description of the pen, and the units assigned to the pen. It also provides various options and style settings for the pen.

In the Unicode version of TrendWorX32 Viewer, the **Description (Desc)**, **Units**, and **Details Format** fields can be language-aliased. If the language alias for the units also has a related equation, this will result in the automatic range and value scaling for the pen upon switching languages.

To select an alias to use, click the ... button to the right of the **Tag**, **Desc**, **Units**, **Persistent Group**, or **Details Format** fields and select either **Global Aliases** or **Language Aliases** from the pop-up menu. This opens the respective alias browsers, allowing you to choose from a list of available aliases.



Edit Trend Pen Dialog Box: General Tab

General Tab Functions

Function	Description
Options - Visible	Indicates whether the plot of the selected pen will be visible. If Visible is not checked, the Viewer will still buffer data for the pen, but it will not draw it.
Options - Stair Plot	Describes how the plot will appear. The stair plot uses a stepwise draw method to plot the trend. It is ideal for plotting slowly changing signals or setpoints.
Options - Pen Comments	If the pen being edited is a historical pen, then the Pen Comments check box will be enabled. If checked, when the TrendWorX32 Viewer ActiveX is in freeze mode, it will retrieve and display all related historical comments and batch information (if any).
Options - Pen Marker	Provides a marker for the pen on the right-hand side of the trend display indicating the last data point drawn on the graph.
Options - Show Samples	Inserts a circle around every point on the graph that represents an actual data point.
Options - X-Axis	Indicates the horizontal axis selection for XY plots.
Options - Auto Sync	This option applies to real-time pens. If enabled, it will shift the time axis of the pen so that the curve will be plotted within the current time frame of the Viewer. This option should be used when real-time pens are connected to remote OPC Data servers and there no network synchronization facility is applied.
Style - Line	Select a line style for the plot from the drop-down list: Solid , Dashed , Dotted , or Dotted Dash .
Style - Width	Determines the width of the plot line. If the line width is greater than 1, the Line field will be unavailable, and the Solid line style will be applied.
Style - Plotter	Used only for circular charts, determines the size of the plotter for the selected pen: small, medium, big, and huge.
Style - Color	Clicking the Color box will open the Color dialog box, from which you can select the color of the pen.

Details Format	Specifies how data will be displayed in the Details window of the Trend Viewer.
Filter	The Filter selection is mostly used for historical replay and corresponds to the averaging method used in retrieving historical data. For example, if Filter is set to Last Sample, the last sample of the historical data within each subinterval will be retrieved. The TrendWorX32 Viewer supports the Raw Data filter for historical replay as follows: • Select the Raw Data filter for a historical pen. • Select the Max. Samples to retrieve. • The upper value of the Max. Samples is taken from the registry setting: "HKEY_LOCAL_MACHINE\SOFTWARE\ICONICS\TWXView32\Data Properties\Max Pen Samples" As a result, the TrendWorX32 Viewer will ask the TrendWorX32 SQL Server to retrieve the Max. Samples number of raw samples (no interpolated/extrapolated) from the historical data. If more data exist for the time interval for which the data were requested, an X mark will appear at the ends of the curve to indicate that not all samples were retrieved for the time interval of the request. This feature does require some configuration in order to prevent usage of large amounts of memory. A typical way to estimate the maximum number of samples data-logged for a period of time is to divide the time period by the data-collection rate of the signal configured in the data logger. If no dead band is configured, and the signal that is data-logged changes frequently, the result indicates an upper limit of samples data-logged for that period.

Alarms Tab

The **Alarms** tab of the **Edit Trend Pen** dialog box, shown below, sets the

levels at which alarm lines will be drawn (if desired).

The screenshot shows the 'Edit Trend Pen' dialog box with the 'Alarms' tab selected. The 'Low' field is empty, 'LowLow' is 0, 'High' is 100, and 'HighHigh' is 100. Each field has a 'Tags' button and an '=' button. A checkbox 'Show Alarm Lines' is at the bottom left. 'OK' and 'Cancel' buttons are at the bottom right.

Edit Trend Pen Dialog Box: Alarms Tab

The TrendWorX32 Viewer ActiveX allows the connection of OPC Data Access tags to its alarm limits. The OPC-enabled alarm limits will hold the most current value of the alarm limits. The alarm limits can be a constant value, an OPC Data Access tag, or an expression.

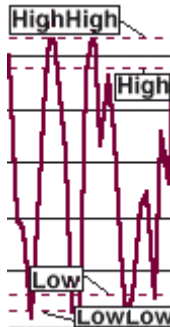
In addition, the alarm limits are interfaced to the Global Alias and Language Alias servers. This functionality is not available in History Trend mode due to the nature of this specific historical replay mode. In addition, several OLE automation methods have been added to support this functionality:

- **void SetPenLoAlarmString(BSTR PenName, BSTR AlarmString, BSTR PenType)**
- **void SetPenLoLoAlarmString(BSTR PenName, BSTR AlarmString, BSTR PenType)**

-
- **void SetPenHiAlarmString(BSTR PenName, BSTR AlarmString, BSTR PenType)**
 - **void SetPenHiHiAlarmString(BSTR PenName, BSTR AlarmString, BSTR PenType)**

If **Show Alarm Lines** is checked, dashed lines at the **Low**, **LowLow**, **High** and **HighHigh** marks will be inserted into the trend. A sample of this is shown below.

Note: If you desire buffering of alarm values, historical replay functionality, etc., you should not use the alarm lines support of the TrendWorX32 Viewer ActiveX, but rather use individual pens to represent each alarm line (limit).



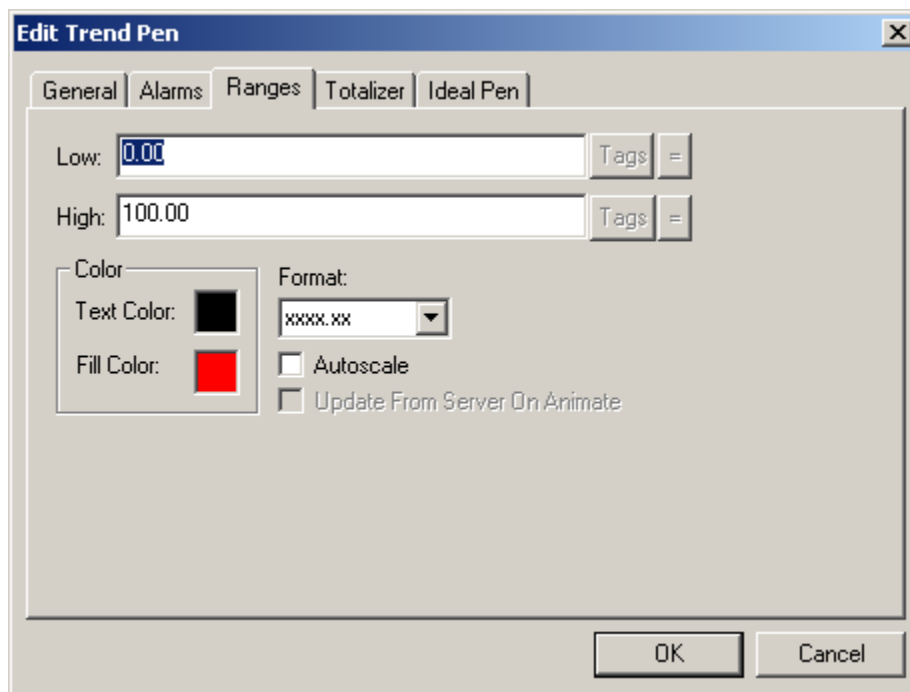
Alarm Lines Shown

Ranges Tab

The **Ranges** tab of the **Edit Trend Pen** dialog box, shown below, deals with the field to the left of the trend display and the relative scaling of the pen curves for drawing. This tab is available only if the **Global Ranges** option is not enabled.

The range values can be constants, expressions, OPC tags, or global aliases. To select an OPC tag for the low or high ranges, click the **Tags** button. To select an expression, click the **=** button to open the Expression Editor. To select a global alias, click the **...** button and select **Global Aliases** from the pop-up menu. This opens the **Global Alias** browser, allowing you to choose from a list of available aliases.

In the Unicode version of TrendWorX32 Viewer, the **Format** field can be language-aliased. The language alias will be used to determine the ranges format on the range scale, if the **Global Ranges** option is not selected. To select an alias to use, click the ... button to the right of the **Format** text box and select either **Global Aliases** or **Language Aliases** from the pop-up menu. This opens the respective alias browsers, allowing you to choose from a list of available aliases.



Edit Trend Pen Dialog Box: Ranges Tab

Ranges Tab Functions

Function	Description
Low	Sets the lowest visible point.
High	Sets the highest visible point.
Tags Button	Launches the OPC Universal Tag Browser .

= Button	Launches the Expression Editor.
... Button	Opens the alias browsers.
Text Color	Clicking the Text Color box will open the Color dialog box so that you can select a color for the text of the field.
Fill Color	Clicking the Fill Color box will open the Color dialog box so that you can select a color for the background of the field.
Format	Specifies how data will be displayed in the Ranges section of the Trend Viewer.
Autoscale	Disables the Low and High fields, and sets these limits automatically according to the values for the pens. Autoscaling, if checked, adjusts the range if the actual value is above or below the set limits. For example, if autoscaling is selected and the ranges are set from 0 (Low) to 200 (High) and the actual value is 250, then the ranges will autoscale to 0 to 280. The adjustment will be around 20 percent above the actual value.
Update From Server on Animate	If this option is enabled, the ranges of the pen are automatically updated from the OPC Data server or the OPC Historical servers upon entering animation mode.

Totalizer Tab

The **Totalizer** function calculates the total amount covered within a certain time period based on the rate represented by the signal; for example, flow (gallons per minute) and speed (miles per minute). The TrendWorX32 Viewer has Totalizer support on a per-pen basis, even for non-circular chart plot types.

Edit Trend Pen Dialog Box: Totalizer Tab

Totalizer Tab Functions

Function	Description
Input Signal - Amount Per Second	The signal represents data in seconds.
Input Signal - Amount Per Minute	The signal represents data in minutes.
Input Signal - Amount Per Hour	The signal represents data in hours.
Input Signal - Amount Per Day	The signal represents data in days.
Totalizer Type - Continuous	When Continuous is selected, the Totalizer will keep counting, starting from 0 until it comes to the end of the trend period, or until the

	integrated value exceeds the Auto Reset Level (a maximum limit).
Totalizer Type - Preload	When Preload is selected, the Totalizer commences counting from any value. This is especially useful when totalizing must be “held” (suspended) while a necessary operation is being performed. For example, the pipes are being cleaned and then started from a known value.
Totalizer Type - Preset	When Preset is selected, you simply specify a preset value. • If you are counting upwards, the value will obviously be positive. • If you are counting downwards, the preset value will be negative. A negative value would be relevant in a situation where a reservoir is being drained, for example. Then the Totalizer function, if enabled for a circular chart, would help monitor the flow rate. • When counting upwards, when the total exceeds the preset value, it is reset. • When counting downwards, the preset value is reset at a point when it becomes negative.
Initial Value	Specifies a counting option.
Enable Totalizer	When Enable Totalizer is checked, the Totalizer will run according to the other configurations on this tab.
Manual Reset	In runtime mode, the Manual Reset button is available in the Totalizer tab. When it is reset, the system returns to its initial value, and everything else is disabled.

Auto Reset Level	Sets the level at which the system will automatically reset, returning to its initial value and disabling everything else.
------------------	--

If you select two pens connected to the same OPC tag and enable the Totalizer function for one, you can view the details—namely that the Totalizer function is enabled—in the output window.

When the signal is 0, the Totalizer will remain at the same value. When it is on, it will show the cumulative value.

The TrendWorX32 Viewer has Totalizer support on a per-pen basis, even for non-circular chart plot types.

Trend Viewer Toolbar

The **Trend Viewer** toolbar is the runtime user interface for the ActiveX. The toolbar offers you access to various runtime TrendWorX32 Viewer ActiveX functions and operations. To view or hide the toolbar, double-click the TrendWorX32 Viewer during runtime. You cannot view the toolbar in configuration mode.



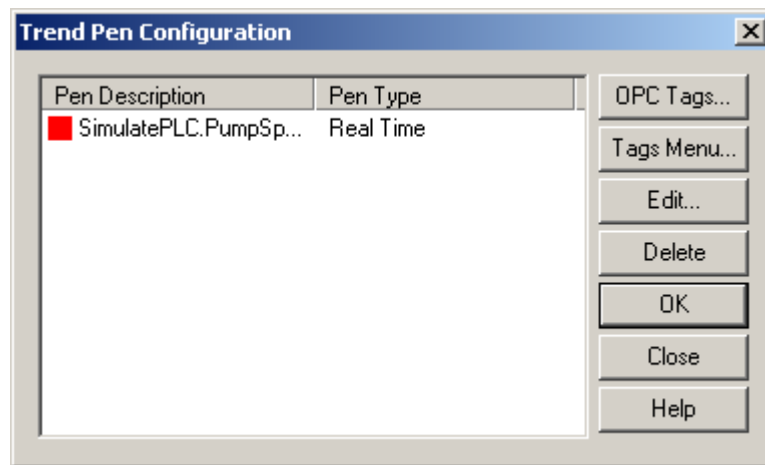
Trend Viewer Toolbar

The **Trend Viewer** toolbar contains the following buttons:

- **Edit pens**
- **Edit trend**
- **Edit period**
- **Freeze mode**
- **Statistics**
- **Zoom trend**
- **Page back**
- **Cursor back**
- **Cursor forward**
- **Page forward**
- **Set right time**
- **Show comments**
- **Print trend**
- **Save trend**

Edit Pens

Clicking the **Edit Pens** button, shown at left, on the **Trend Viewer** toolbar opens the **Trend Pen Configuration** dialog box, shown below.



Trend Pen Configuration Dialog Box

The **Trend Pen Configuration** dialog box contains the following buttons:

- **OPC tags**
- **Tags menu**
- **Edit**
- **Delete**

OPC Tags

To make real-time connections from the **Trend Pen Configuration** dialog box, click the **OPC Tags** button. This opens the **OPC Universal Tag Browser**, which enables you to browse all available OPC servers and their signals.

Tags Menu

Clicking **Tags Menu** in the **Trend Pen Configuration** dialog box opens the **Tags** menu, which contains the following options:

- **Expression editor**
- **Simulation variables**
- **Persistent trend**
- **History tag**

Editing Pens

Clicking **Edit** in the **Trend Pen Configuration** dialog box opens the **Edit Trend Pen** dialog box, which contains the following tabs:

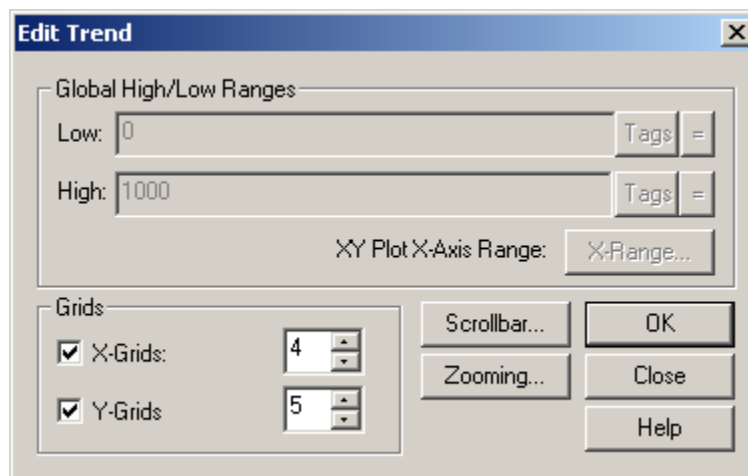
- **General tab**
- **Alarm tab**
- **Ranges tab**
- **Totalizer tab**

Delete Button

You can also delete, modify, and replace pens. To delete one or more pens, select the pen and then click **Delete** on the **Trend Pen Configuration** dialog box. Then click **Yes** to delete the pen.

Edit Trend

Clicking the **Edit Trend** button on the **Trend Viewer** toolbar opens the **Edit Trend** dialog box, shown below.

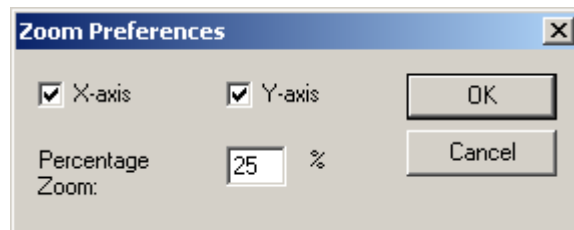


Edit Trend Dialog Box

The **Global High/Low Ranges** fields show the ranges that were assigned in the **Ranges** tab of the **Properties** dialog box. If the **Global Ranges** box was not checked on that tab, then this section of the **Edit Trend** dialog box will be unavailable.

The **Scroll Bar** button allows you to set the time range for how far back you will be able to scroll while in freeze mode.

To select the appropriate settings for the **Zoom Trend** button, click **Zooming** on the **Edit Trend** dialog box. This opens the **Zoom Preferences** dialog box, shown below. You can select how closely you want to zoom and which axis you want to use as a base for the zoom when you are using the point-and-click zoom function. If you use the "rubber band" zooming feature, these options will be ignored. In addition, these options take global effect for all the TrendWorX32 Viewers, and they are not saved on a per-individual Viewer ActiveX basis.

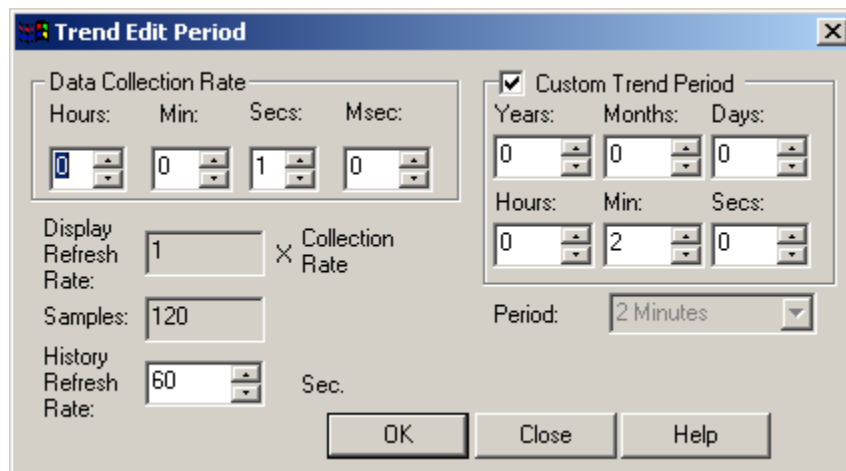


Zoom Preferences Dialog Box

The **Grids** section of the **Edit Trend** dialog box determines how many lines along the x-axis and y-axis make up the grid displayed in the **Trend** window.

Edit Period

Clicking the **Edit Period** button on the **Trend Viewer** toolbar opens the **Trend Edit Period** dialog box, shown below.



Trend Edit Period Dialog Box

The **Data-Collection Rate**, **Display Refresh Rate**, and **Samples** fields are set according to the settings made on the **Rates** tab of the **Properties**

dialog box.

You can set the trend period and/or the data-collection rate using the **Trend Edit Period** dialog box. If you want to select a custom period, the **Custom Trend Period** option should be checked first. Otherwise, the **Period** field will be activated and by default this field is set at 2 minutes but the options of 10 seconds, 30 seconds, 1 minute, 10 minutes, 30 minutes, 1 hour, 3 hours, 8 hours, 1 day, 5 days, 1 week, 1 month, 3 months, 6 months, and 1 year are also available from this list.

During the **Period Change** operation, the TrendWorX32 Viewer ActiveX will try first to increase the number of samples on a per-pen basis to produce a higher trend time period frame for viewing data. If upon doing so it exceeds its maximum number of samples per pen, which is currently set to 7200, it will try to adjust the data resolution (data-collection rate) to produce the desired trend period.

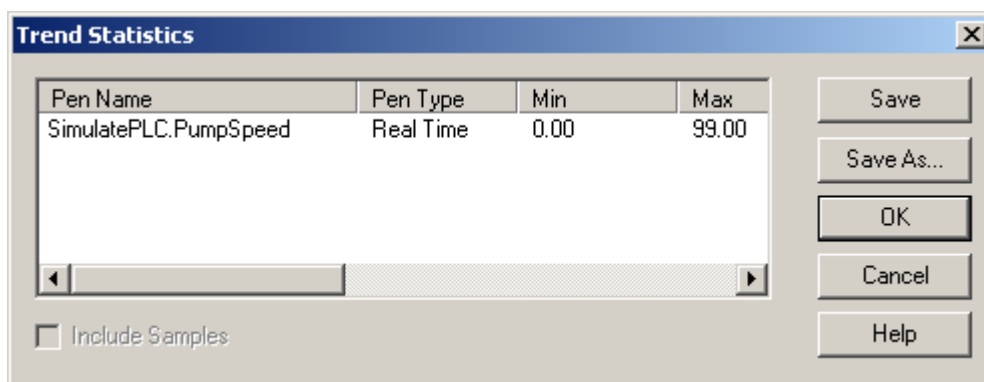
Freeze Mode

Clicking the **Freeze Mode** button on the Trend Viewer toolbar halts the trend display without stopping data collection in the TrendWorX32 Viewer. To unfreeze the display, simply click the **Freeze Mode** button again. While the display is in freeze mode, you can click on the display, and a black vertical cursor line will appear with circles in the color of the each pen where the line intersects the closest sample of the plot for that pen. By holding the left mouse button down, you can move the cursor line and view the sample values, time stamps, and other parameters in the **Details** window. In freeze mode, additional trending functions, such as zooming and paging, are available.

Note: If the TrendWorX32 Viewer has historical pens for which the **Show Comments** option has been enabled, freezing the display will also retrieve the historical comments and batch information (if available).

Statistics

Clicking the **Statistics** button on the **Trend Viewer** toolbar opens the **Trend Statistics** dialog box, shown below.



Trend Statistics Dialog Box

The **Trend Statistics** dialog box 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. When the **Save** or **Save As** buttons are clicked, the **Save As** dialog box opens, and you are prompted to enter a file name for the statistics. The statistics will be saved in a .csv file. Checking **Include Samples** includes each pen's samples in the saved statistics file, along with any historical comments for the history pens (if available).

Zoom Trend

When you click the **Zoom Trend** button on the **Trend Viewer** toolbar in freeze mode, you can zoom in on the **Trend** window and perform the point-and-click zoom function. This enables you to zoom in on the trend display about the current location of the cursor line, using the desired settings for zooming configured in the **Edit Trend** dialog box. Clicking the left mouse button while pressing the **SHIFT** key or right-clicking the mouse inside the trend display zooms out of the trend display.

In addition to the point-and-click zoom function, the TrendWorX32 Viewer ActiveX also supports a "rubber band" or box zoom approach to zooming. While in freeze mode, you can press the **SHIFT** key down and press the left mouse button to select a desired area on which to zoom. Clicking the left mouse button while pressing the **SHIFT** key, or right-clicking the mouse inside the trend display, zooms out of the trend display.

Cursor Back/Cursor Forward

Clicking the **Cursor Back** and **Cursor Forward** buttons on the **Trend Viewer** in freeze mode moves the cursor line back and forth over the trend display. You can also move the cursor by clicking the left mouse button on the trend display, or by pressing the left and right arrow keys on the keyboard. The cursor function is available only when the trend display is in freeze mode.

The cursor enables you to select specific sample data points on the trend display. To get a detailed description about a specific sample data point in the trend, or to view historical comments and batch information, point to the sample using the mouse pointer, as shown in the figure above. Pressing the **SHIFT** key while clicking the **Cursor Back** and **Cursor Forward** buttons moves the cursor left or right, respectively, 10 samples (by default).

Page Back

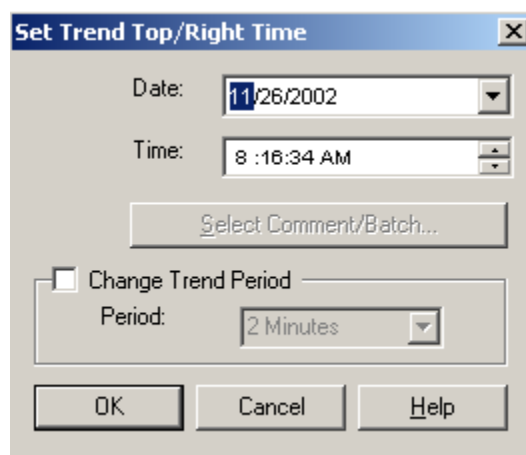
Clicking the **Page Back** button on the **Trend Viewer** toolbar shifts the trend display back from the cursor line by one trend period. This function is available only when the trend display is in freeze mode. The TrendWorX32 Viewer retrieves and displays historical and persistent trend data from the TrendWorX32 Persistent Trending memory buffers.

Page Forward

Clicking the **Page Forward** button on the **Trend Viewer** toolbar shifts the trend display forward from the cursor line by one trend period. This function is available only when the trend display is in freeze mode. The TrendWorX32 Viewer retrieves and displays historical and persistent trend data from the TrendWorX32 Persistent Trending memory buffers.

Set Right Time

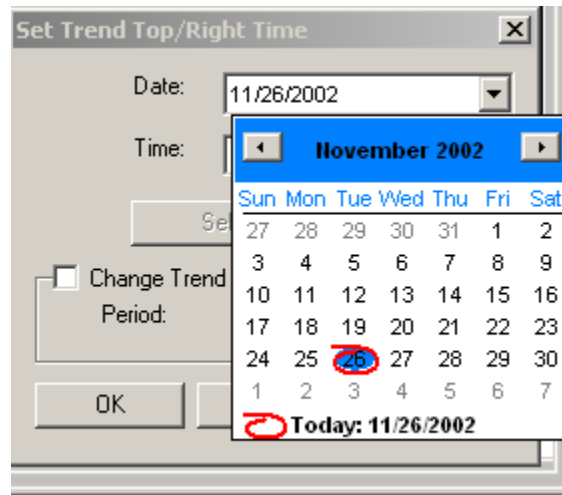
Clicking the **Set Right Time** button on the **Trend Viewer** toolbar opens the **Set Trend Top/Right Time** dialog box, shown below. This dialog box enables you view a specific historical trend period. In freeze mode, this causes the TrendWorX32 Viewer to retrieve and display the selected historical and persistent trend data from the TrendWorX32 Persistent Trending memory buffers.



Set Trend Top/Right Time Dialog Box

To select a desired date for the trend period you want to view, click the down arrow in the **Date** field, and then choose a date from the calendar, as shown in the figure below. To select a desired time, highlight the hours, minutes, or seconds in the **Time** field. You can either type the desired hours, minutes, and seconds or click the arrows to scan the highlighted time

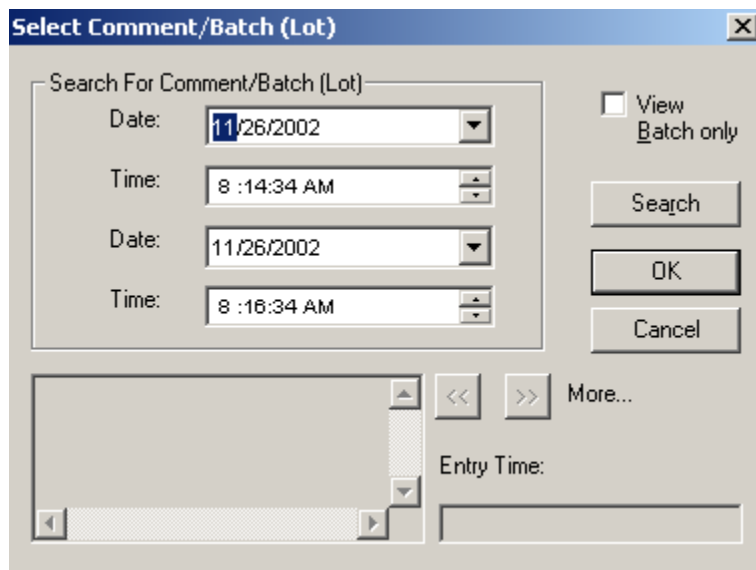
field.



Choosing a Date From the Calendar

The **Set Trend Top/Right Time** dialog box has time and date pick controls as well as an interface to the historical operator comments and batch information. To use this feature, do the following:

1. Select a history pen from the details window.
2. Click the **Set Right Time** button on the **Trend Viewer** toolbar to open the **Set Trend Top/Right Time** dialog box. Click the **Select Comment/Batch** button.
3. This opens the **Select Comment/Batch (Lot)** dialog box, shown below, which allows you to search for a specific batch entry or operator comment within a time range.



Select Comment/Batch (Lot) Dialog Box

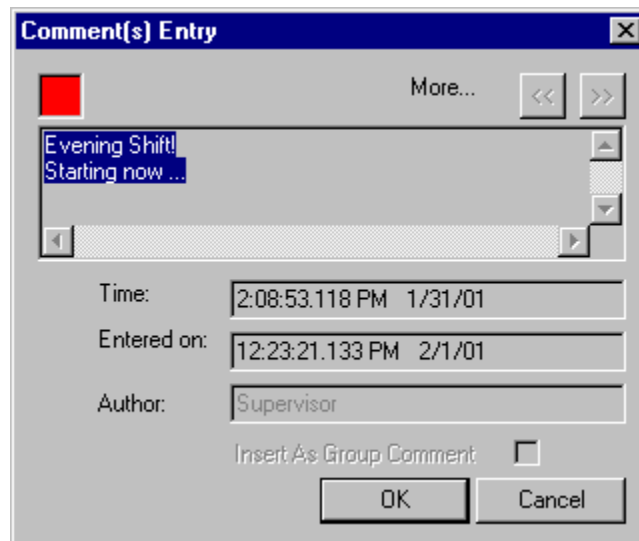
In the **Select Comment/Batch (Lot)** dialog box:

1. Select a start time and date. To select the date, click the down arrow in the **Date** field, and then choose a date from the calendar.
2. Select an end time and date.
3. Click **Search** to initiate the search.
4. Once the search is completed, use the << and >> buttons to scroll through multiple operator comments and batch entries.
5. When you locate the desired one, click **OK**.

Note: Selecting the **View Batch Only** option filters out the historical operator comments so that only the related batch (lot) entries are displayed.

Show Comments

If historical pens are configured in the TrendWorX32 Viewer, clicking the **Show Comments** button on the **Trend Viewer** toolbar places the Viewer in freeze mode and retrieves all historical operator comments and batch information available. You can then select a historical pen from the **Details** window or double-click the comment icon to display the following the **Comment(s) Entry** dialog box, shown below.



Comment(s) Entry Dialog Box

If more than one comment exists in the same area, the << and >> buttons will be enabled. These buttons allow you to scroll through all available comments.

Note: Historical comments are not supported for XY plots and bar plots. Historical comments are integrated into the ToolTip support as well as the historical replay and ideal pen functionality.

Print Trend

Clicking the **Print Trend** button on the **Trend Viewer** toolbar opens the Windows **Print** dialog box, which allows you to print the currently selected TrendWorX32 Viewer. The printout will also include a header that provides some information about the trend, such as the settings and the name of the computer in which the trend originated.

Save Trend

If the TrendWorX32 Viewer is configured to save its configuration into a .v32 file, independently of the container, WebHMI users will have the option in runtime mode to save current settings into the preconfigured file using the **Save Trend** button on the **Trend Viewer** toolbar. In the case of a WebHMI application, this also will be the configuration file to be loaded whenever the HTML page is refreshed.

Note: The **Save Trend** function on the Trend Viewer toolbar is integrated into the Security Server and is ONLY available to WebHMI users. It is not supported in displays that use global aliases.

Historical Replay

The TrendWorX32 Viewer is interfaced through the OPC Historical Data Access specification to the ICONICS TrendWorX32 SQL Server, as well as to other third-party historical servers supporting the specification.

The TrendWorX32 Viewer ActiveX Version 7.0 has been updated to be compliant with the latest OPC HDA release specification 1.2. In general, the OPC HDA specification is an extended OPC specification, which includes an increased number of interfaces and methods, not all of them being mandatory. On the other hand, the TrendWorX32 Viewer ActiveX trend client requires an advanced HDA server implementation to support its rich user interface. Since not all OPC HDA servers will support the TrendWorX32 Viewer's required advanced interface set, and since there is currently no available way to classify OPC HDA server functionality (or equivalently, compliance levels), the TrendWorX32 Viewer ActiveX has integrated minimal OPC HDA server capability information, which will either modify its user interface or warn the user if the third-party OPC HDA server does not support the required interfaces. However, the TrendWorX32 Viewer ActiveX will not compensate for the lack of implementation of used OPC HDA interfaces or methods by combining supported method calls to produce the same end result. For version 7.0, the following additional OPC HDA compliance development effort has been done:

- **Detection of Connection Point Support:** The TrendWorX32 Viewer ActiveX supports two methods of historical replay. In Auto Updating mode (default), mixed real time and historical signals can be configured and updated in real time via asynchronous methods calls. In History Only mode, only historical signals can be configured and they are updated via user interface requests (or VBA automation) and no asynchronous requests will be processed. If the OPC HDA server does not support connection points, then the second historical replay mode should be used.
- **Detection of Max Returned Samples:** If the OPC HDA server does not support return of the number of historical samples configured in the TrendWorX32 Viewer ActiveX based on data collection and the trend display period, a warning message will notify the user to change the trend period settings.
- **Detection of OPC HDA Aggregates:** Each historical pen can utilize a historical data retrieval filter (called "aggregate"). Starting from version 7.0, the TrendWorX32 Viewer will retrieve the OPC HDA server supported aggregates and allow editing of the historical pen to select

the desired aggregate. If the OPC HDA server does not support any aggregates, the only historical data retrieval filter supported will be the **Raw Data** filter.

- **Detection of OPC HDA Server Annotation Support:** The TrendWorX32 Viewer ActiveX trend client will detect if OPC HDA servers support annotations, and it will modify its user interface accordingly.

In terms of historical replay and reporting, the following key changes have been made:

- Data filter selection other than Raw will result in data time stamped at the beginning of each subinterval.
- When retrieving data using data filters other than Raw, subintervals for which there are no data because of no data-logging activity will be marked as “empty” slots at the corresponding time with an empty (VT_EMPTY) value. You can check the returned qualities for further processing.
- The historical qualities supported are as follows:
 - OPCHDA_EXTRADATA 0x0001
 - More data may exist
 - OPCHDA_INTERPOLATED 0x0002
 - Interpolated data value
 - OPCHDA_RAW 0x0004
 - Raw data value
 - OPCHDA_CALCULATED 0x0008
 - Calculated data value
 - OPCHDA_NOBOUND 0x0010
 - No data found to provide upper or lower bound
 - OPCHDA_NODATA 0x0020
 - No data collected. Archiving not active
 - OPCHDA_DATALOST 0x0040
 - Calculation started/stopped/lost
 - OPCHDA_CONVERSION 0x0080
 - Scaling / conversion error
 - OPCHDA_PARTIAL 0x0100

-
- ❑ Aggregate value is for an incomplete interval

Critical Note: Because of the new compliance changes in historical replay and reporting, a version 7.0 historical replay or reporting may not produce identical results to existing 6.1x version reports and or historical replay screen captures for the same signals and time periods.

Two modes of historical replay are supported:

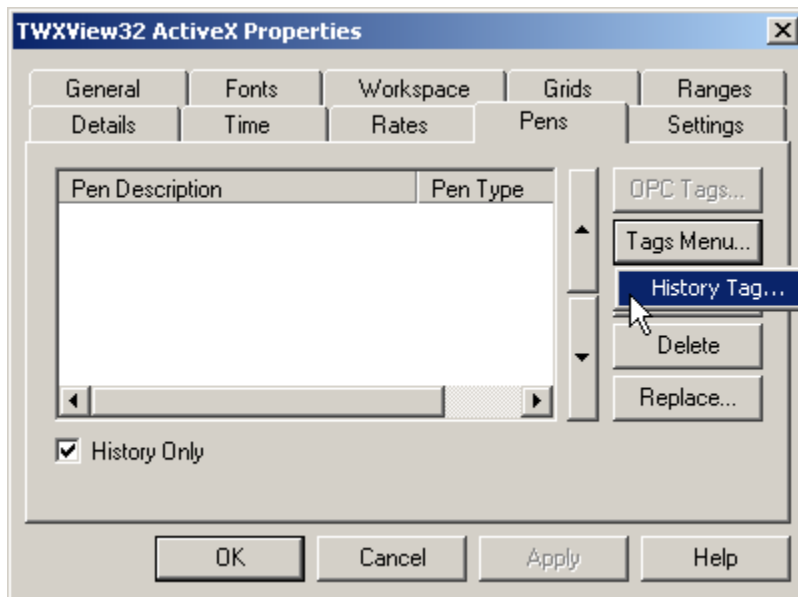
1. Mixed replay of signals
2. History only replay (in freeze mode)

Mixed Replay

In **mixed replay**, which is the default mode, you can connect real-time OPC signals, persistent trending signals, and historical signals. All the signals will auto update, and the historical signals will be refreshed with new historical data as they become available. You can control the frequency of historical data updates by modifying the **History Refresh Rate** option in the **Rates** tab of the **Properties** dialog box of the TrendWorX32 Viewer. This should be set close to the data-logging frequency (i.e. the **Log to Disk** option in the TrendWorX32 Configurator).

History Only Replay

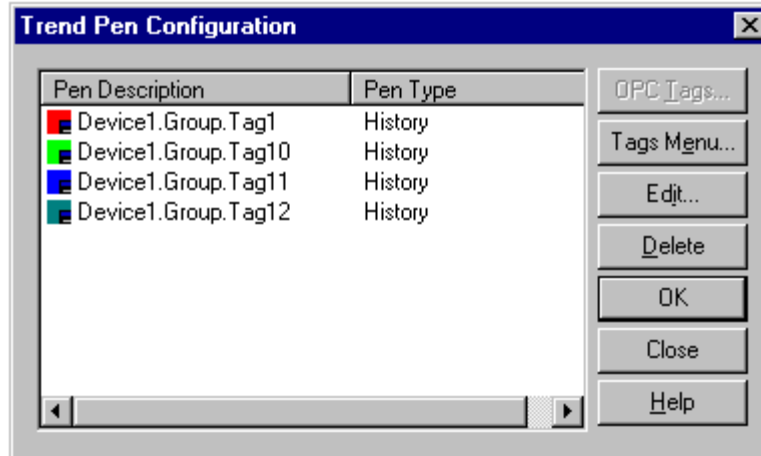
To configure the Viewer for **History Only Replay**, you must first check **History Only** in the **Pens** tab of the **Properties** dialog box, as shown in the figure below, prior to making any signal connections. In this mode, you can only connect to historical data. Once the Viewer is committed to animation (runtime) mode, it will populate the screen with the historical data and automatically set the Viewer into freeze mode. You can then retrieve historical data by scrolling back using the scroll bar. In addition, for finer control over historical replay, you can double-click the time window or click the **Set Right Time** button on the **Trend Viewer** toolbar to set the Viewer's right historical time of reference.



Historical Signal Connections

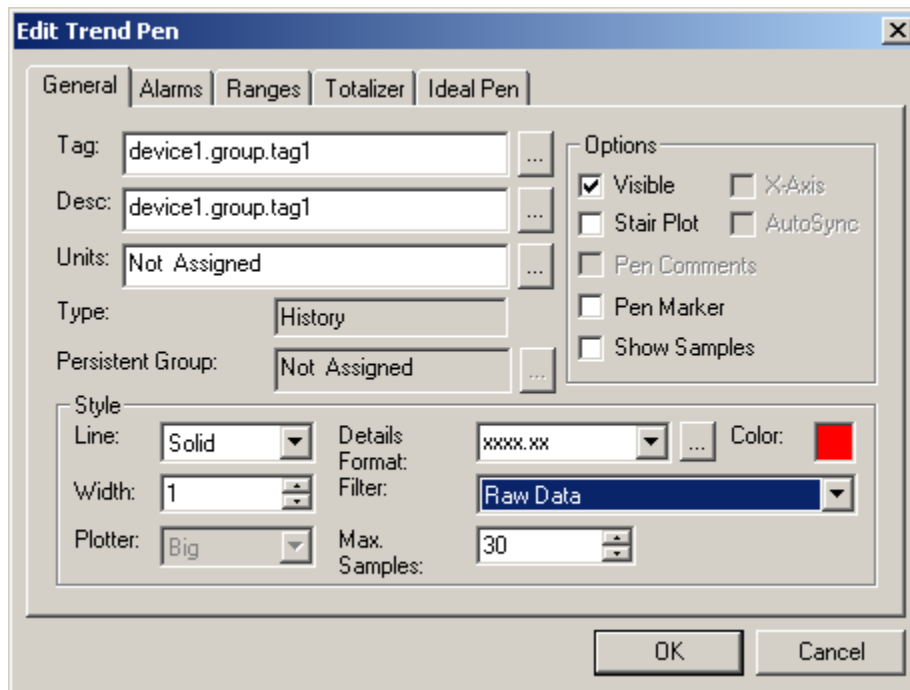
Adding historical pens. To add historical pens, click **Tags Menu** and select **History Tags** to open the **OPC Universal Tag Browser**, which allows you to browse all of the installed historical servers locally or remotely.

If the TrendWorX32 Viewer is in runtime mode, click the **Edit Pens** button on the **Trend Viewer** toolbar to open the **Trend Pen Configuration** dialog box for historical pens, as shown below.



Configuring Historical Pens

Editing historical pens. To edit historical pens, click **Edit** in the **Trend Pen Configuration** dialog box. This opens the **Edit Trend Pen** dialog box, as shown below. When editing a historical pen, you can configure the averaging mechanism used in retrieving historical data. You can always change this option within runtime mode, either through the user interface or OLE Automation, to focus on various aspects of the historical replay.



Editing Historical Pens

Pen Filters

The various choices of the pen filter determine how historical data will be retrieved. For example, if the Viewer is configured to replay data for one hour, with a data resolution of 1 min., the historical data retrieval mechanism will divide one hour's worth of data into bins of 1 min. each, and then it will average the data according to the desired filter. If **Filter** is set to **Average**, it will compute the average of all samples per bin to yield a single pen sample to be drawn on the screen. If **Filter** is set to **Last Sample**, the last sample that falls within each bin will be plotted on the screen.

In addition to the standard data filters, the TrendWorX32 Viewer supports the **Raw Data** filter for historical replay as follows:

1. Select the **Raw Data** filter for a historical pen.
2. Specify the **Max. Samples** to retrieve.

-
3. The upper value of the **Max. Samples** is taken from the following registry setting:

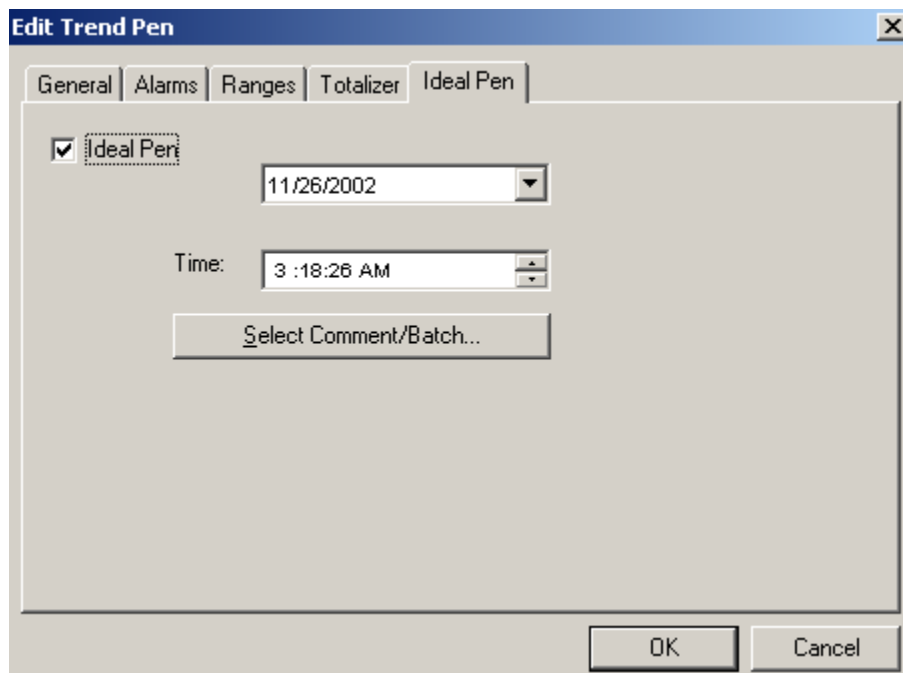
HKEY_LOCAL_MACHINE\SOFTWARE\ICONICS\TWXView32\Data Properties\Max Pen Samples

As a result, the TrendWorX32 Viewer will ask the TrendWorX32 SQL Server to retrieve the Max. Samples number of raw samples (no interpolated/extrapolated) from the historical data. If more data exist for the time interval for which the data were requested, an X mark will appear at the ends of the curve to indicate that not all samples were retrieved for the time interval of the request. This feature does require some configuration in order to prevent usage of large amounts of memory. A typical way to estimate the maximum number of samples data-logged for a period of time is to divide the time period by the data-collection rate of the signal configured in the data logger. If no dead band is configured, and the data-logged signal changes frequently, the result indicates an upper limit of samples data-logged for that period.

In the case of bar plots, the historical replay results in a time-averaged sample across the preconfigured trend period in configuration mode. That is, if the period is set to one hour, the bar plot type will display the averaged sample over the last hour.

Ideal Pens

Any historical pen can be configured as an ideal pen, and there is no limitation on the number of ideal pens per TrendWorX32 Viewer. As shown in the figure below, an **Ideal Pen** configuration tab is automatically inserted in the **Edit Trend Pen** dialog box to allow you to select ideal pens and to set their starting time and date. Upon entering animation mode, the TrendWorX32 Viewer will populate the ideal pens with historical data starting at the selected time and date, while all other nonideal historical pens will update with current data from the TrendWorX32 historical databases. To select a desired date for the trend period you want to view, click the down arrow in the **Date** field, and then choose a date from the calendar. This feature allows for powerful comparisons of past historical data with current historical data for the same historical pen.



Ideal Pen Configuration

You can set (or reset) any historical pen as an ideal pen at any time during animation mode, and you can modify its starting time and date. The **Ideal Pen** has time and date pick controls as well as an interface to the historical operator comments and batch information. To use this feature:

1. Select a history pen from the **Details** window of the trend display.
2. Right-click to display the pop-up menu.
3. Select **Edit** to display the **Edit Trend Pen** dialog box.

Alternatively, you can:

1. Click the **Edit Pens** button on the **Trend Viewer** toolbar to open the **Trend Pen Configuration** dialog box.
2. Select a historical pen from the list.
3. Click **Edit** to display the **Edit Trend Pen** dialog box.

In the **Select Comment/Batch (Lot)** dialog box:

1. Select a start time and date. To select a desired date for the trend period you want to view, click the down arrow in the **Date** field, and then choose a date from the calendar.
2. Select an end time and date.
3. Click **Search** to initiate the search.
4. Once the search is completed, use the << and >> buttons to scroll through multiple operator comments and batch entries.
5. When you locate the desired one, click **OK**.

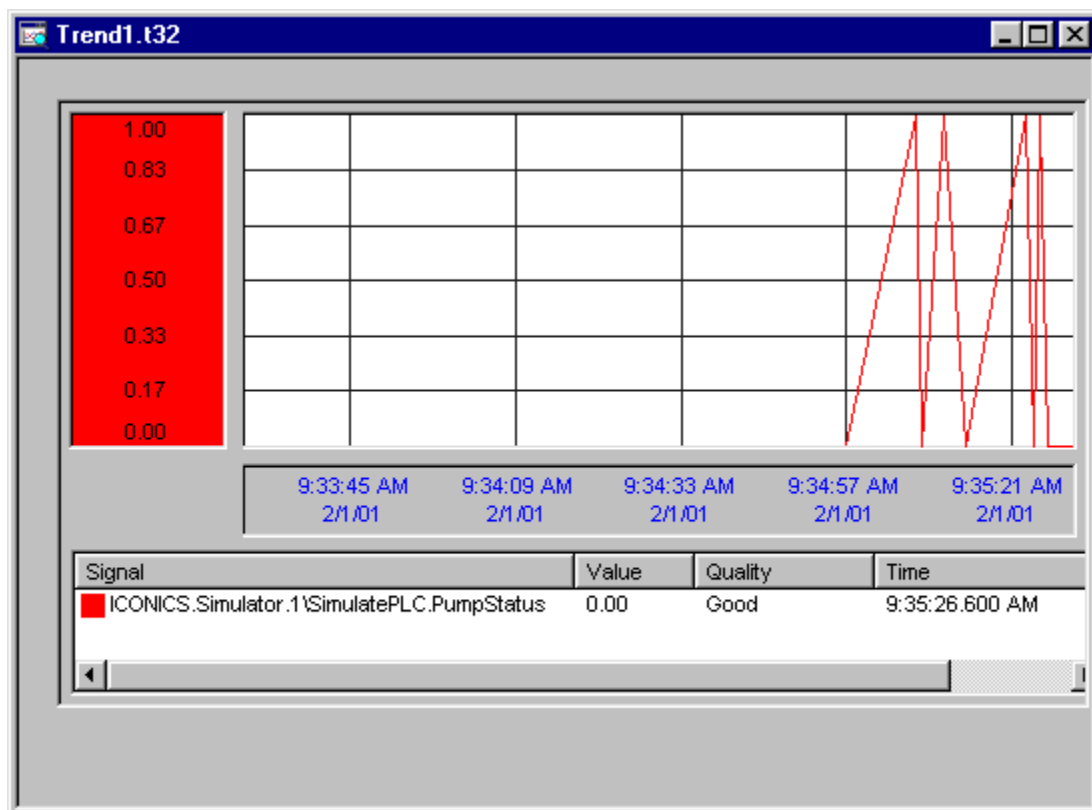
Note: Selecting the **View Batch Only** option filters out the historical operator comments so that only the related batch (lot) entries are displayed.

Editing Historical Pens

During runtime mode, you can edit a historical pen and change the filter type to obtain a different plot. For example, you can use **Max** to view the trend at its peaks. Note that the **MaxTime** and **MinTime** options retrieve the actual maximum and minimum values within the resample interval, as well as the actual time stamps at which these maximum and minimum values occurred.

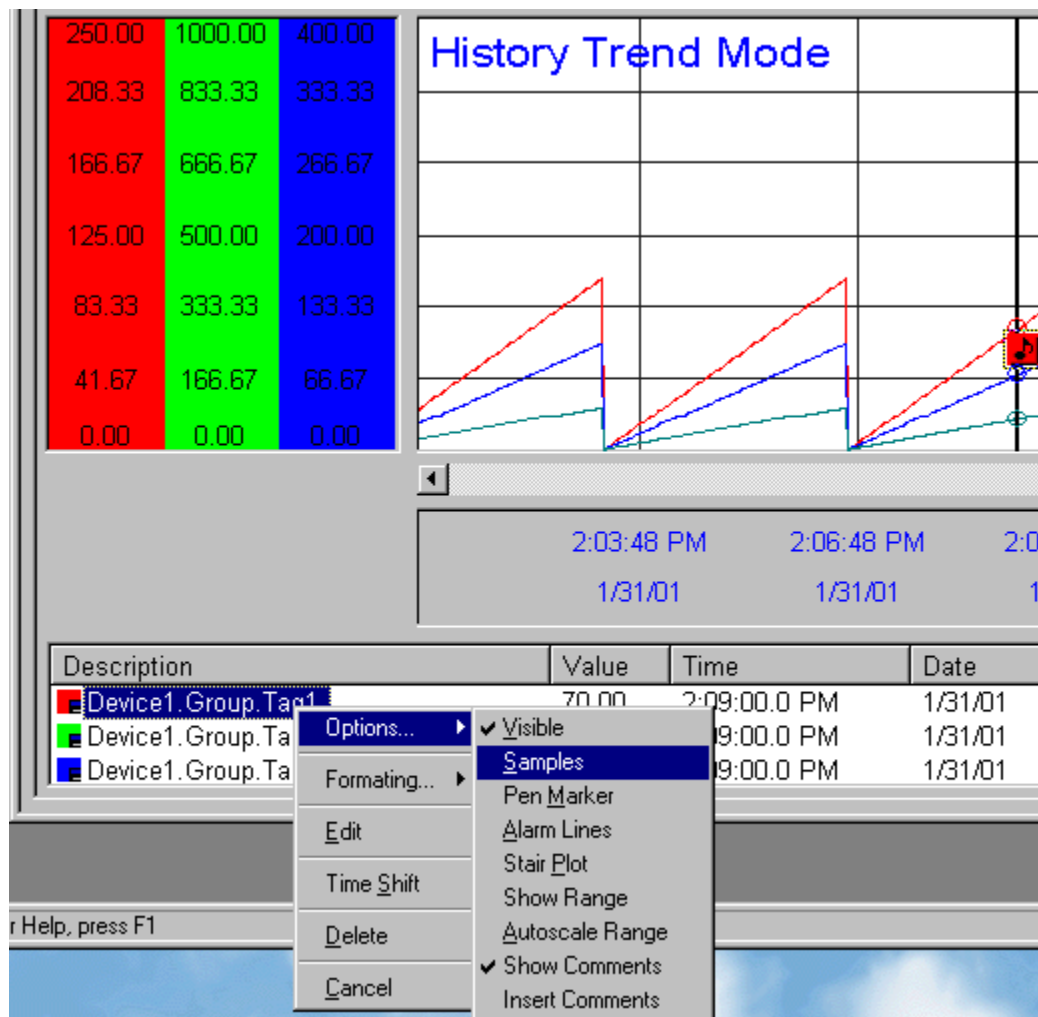
Quality Status

The Viewer has integrated quality status support for both Historical Data Access and OPC Data Access tags. If **Quality** is checked in the **Details** tab of the **Properties** dialog box, the Viewer will display quality status information for each sample in the **Details** window of the trend display, as shown in the figure below.



Quality Status in the Details Window

In addition to the standard OPC Data Access qualities, the Viewer supports the Historical Data Access quality specification to indicate time intervals where data logging stopped, actual (raw) data samples, or calculated samples. During runtime mode, the **Details** window supports extended functionality on a per-pen basis or a group of pens (if selected).



Details Window Menu

Options for Pens

As shown in the figure above, you can select single or multiple pens from the **Details** window. When you right-click a pen, a pop-up menu opens, as shown in the figure above. The **Options** submenu lists the following trend display options and actions for the selected pen:

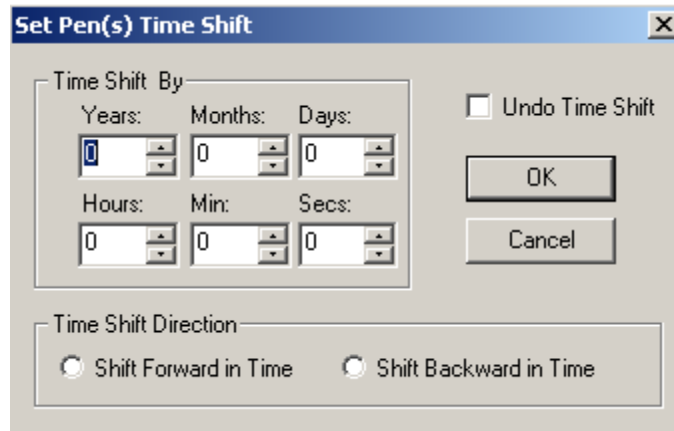
- **Visible:** Shows/hides the pen in the Details view of the Trend Viewer.

-
- **Samples:** Retrieves historical data samples.
 - **Pen marker:** Shows/hides the pen marker in the Details view of the Trend Viewer.
 - **Alarm lines:** Shows/hides the dashed alarm lines at the **Low**, **LowLow**, **High** and **HighHigh** in the Details view of the Trend Viewer.
 - **Stair plot:** Uses a stepwise draw method to plot the trend. This is ideal for plotting slowly changing signals or setpoints.
 - **Show range:** Displays the range values for the pen in the Trend Viewer.
 - **Autoscale range:** Disables the **Low** and **High** fields, and sets these limits automatically according to the values for the pens. Autoscaling, if checked on the **Ranges** tab, adjusts the range if the actual value is above or below the set limits. For example, if autoscaling is selected and the ranges are set from 0 (Low) to 200 (High) and the actual value is 250, then the ranges will autoscale to 0 to 280. The adjustment will be around 20 percent above the actual value.
 - **Show comments:** If historical pens are configured in the TrendWorX32 Viewer, selecting **Show Comments** places the Viewer in freeze mode and retrieves all historical operator comments and batch information available.
 - **Insert comments:** Displays the **Comment(s) Entry** dialog box for historical pens. This option is available only in freeze mode.

Note that the same functionality is also available for real-time trending.

Time Shifting

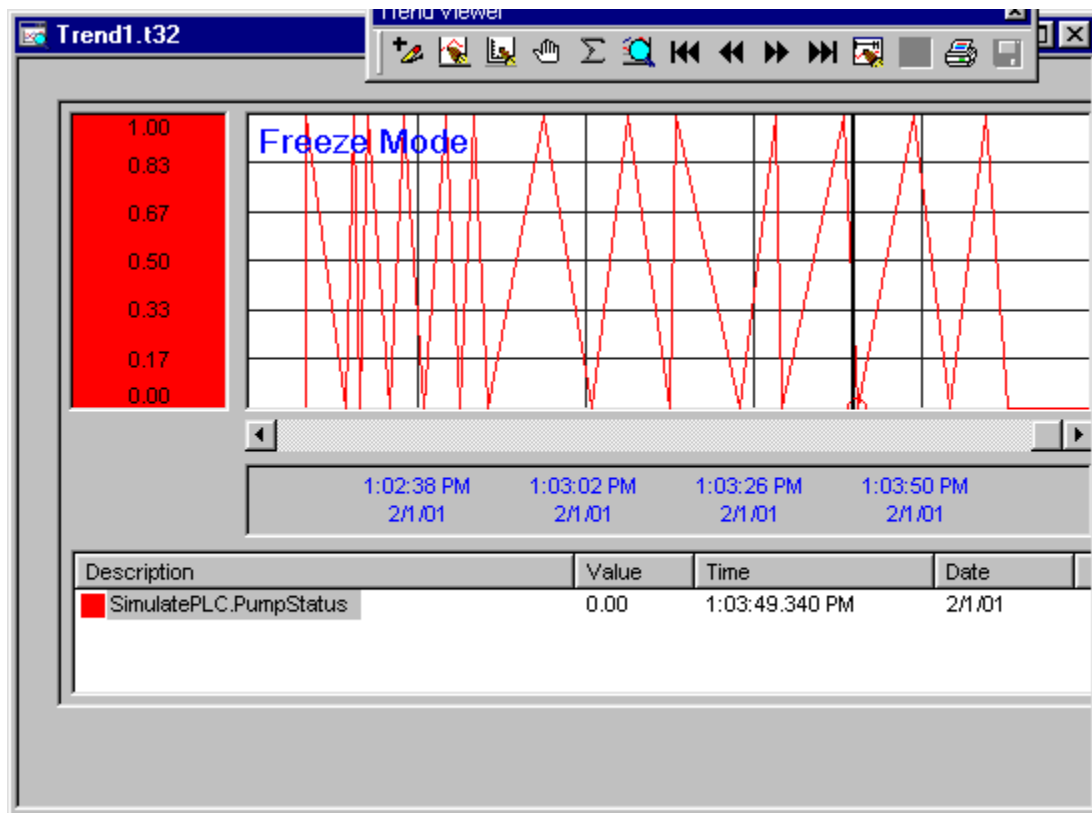
One feature of historical replay is the time shifting of pens in freeze mode. This allows for direct comparison of pens regardless of the time axis settings. To use time shifting, select the desired pen(s) from the **Details** window. Then right-click and select **Time Shift** from the pop-up menu. This opens the **Set Pen(s) Time Shift** dialog box, shown below.



Set Pen(s) Time Shift Dialog Box

You can then choose the interval and direction for the time shift. When you click **OK**, the selected pens will be redrawn to display refreshed historical or real-time data. The time shift operation can be undone at any time by clicking **Undo Time Shift**.

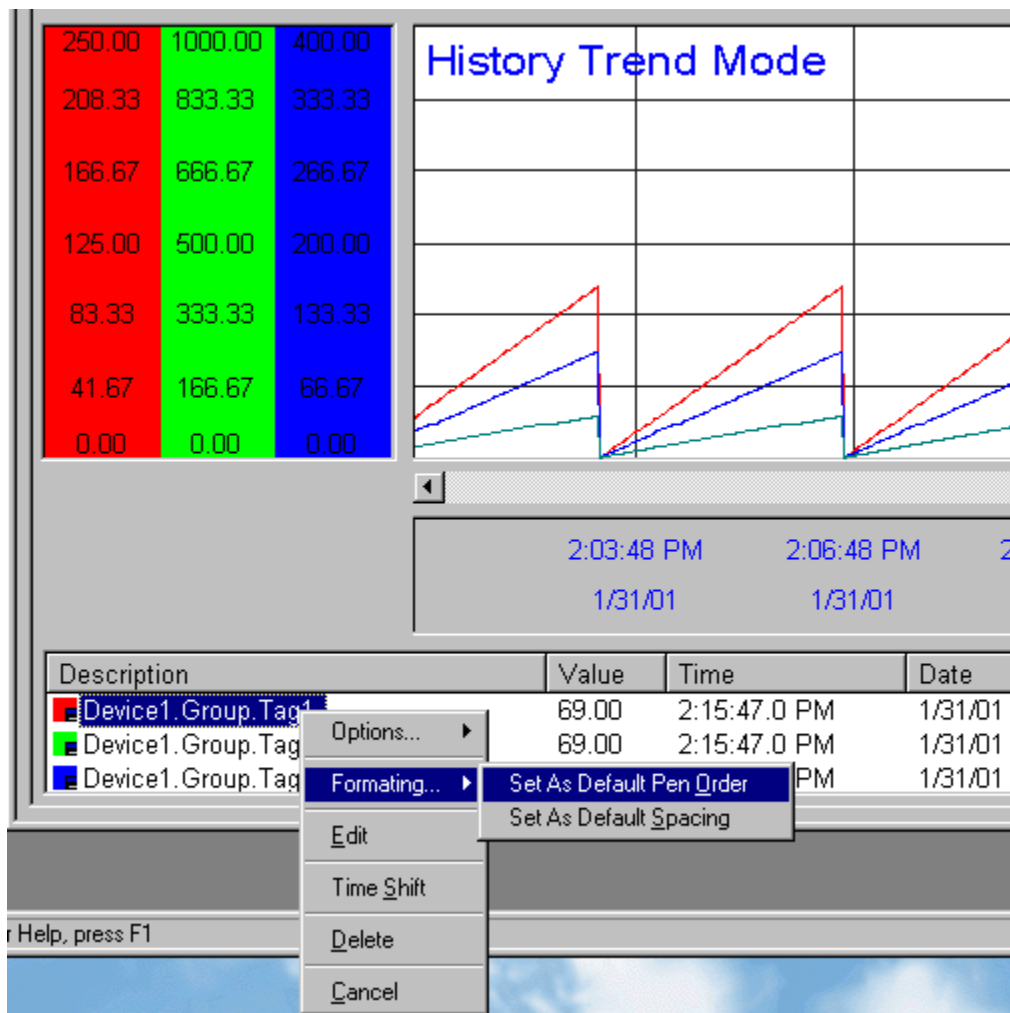
Graphical time shifting. There is also a graphical interface for performing time shifting. Again, you need to select the desired pen(s) from the **Details** window. To perform graphical time shifting, press the **CTRL** key and move the mouse within the trend display area while pressing the left mouse button. The time shifting cursor appears, and you can shift backward or forward in time based on which direction you move the mouse.



Graphical Time Shifting

Formatting Pens

If you do not select any pens but still right-click inside the **Details** window, the formatting options appear in the pop-up menu, as shown in the figure below. You can elect to restore the default settings for the pens.



Pen Formatting

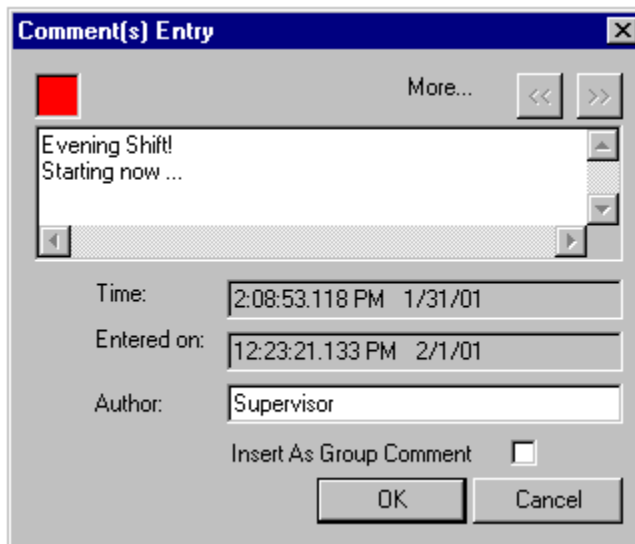
Operator Comments

As an operator, you can insert historical comments (operator comments) into any historical pen or group of pens. To perform this task, do the following:

1. Move the cursor to where you want the comment to be inserted.

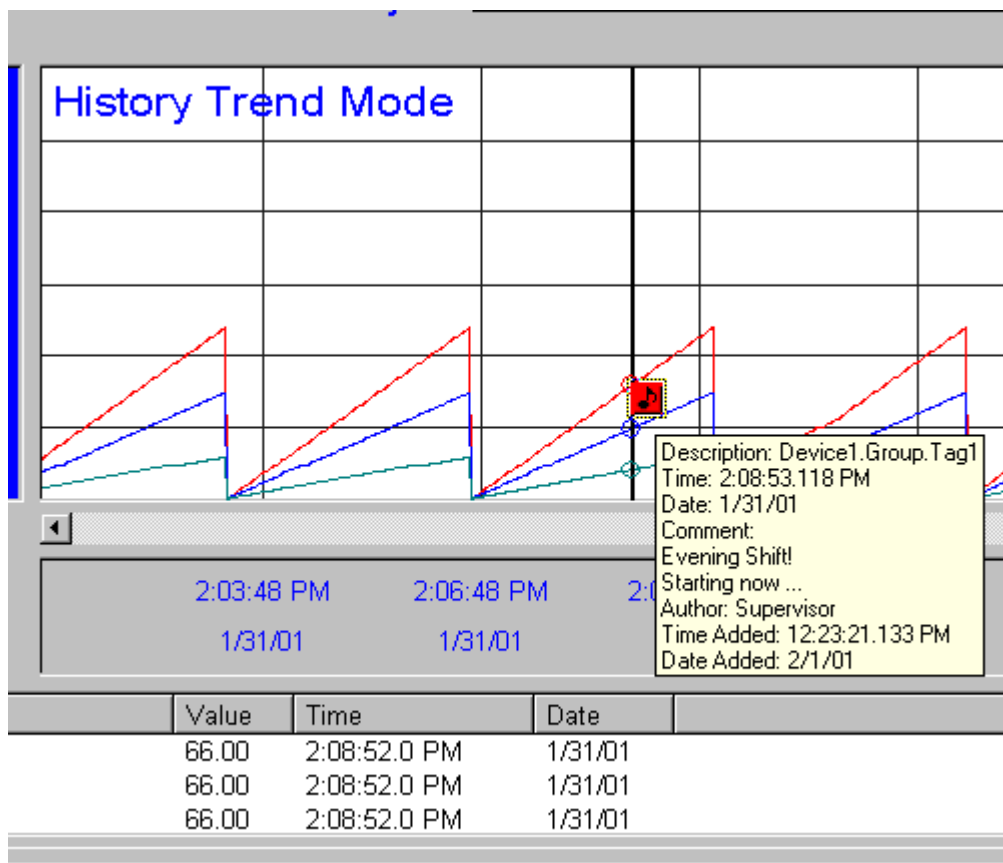
-
2. Select a historical pen in the **Details** window.
 3. Right-click to open the pop-up menu.
 4. From the **Options** submenu, select **Insert Comment**. This opens the **Comment(s) Entry** dialog box, shown below.
 5. Type in a comment (it can be multiline).
 6. Type in a name for the operator entering the comment (by default is the currently logged-in user).
 7. Select **Insert As Group Comment**, if desired, to enter the comment for all the historical tags that belong to the same TrendWorX32 Configurator logging group as the currently selected historical pen.
 8. Click **OK**.

Note: Historical comments are integrated into the ICONICS Security Server. When the operator opens the **Comment(s) Entry** dialog box, the Viewer ActiveX will get the currently logged-in user(s) within the ICONICS Security Server. If users are logged in, the Viewer ActiveX will set the author of the comment to be the logged-in user(s), and it will not allow the operator to modify the author field. If there are no logged-in users, the **Comment(s) Entry** dialog box will use the current Windows user and Node name, and it will allow the operator to modify the comment author field.



Comment(s) Entry Dialog Box

If the **Show Comments** option is enabled for the historical pen, the inserted comment will be visible on the screen, as shown below.



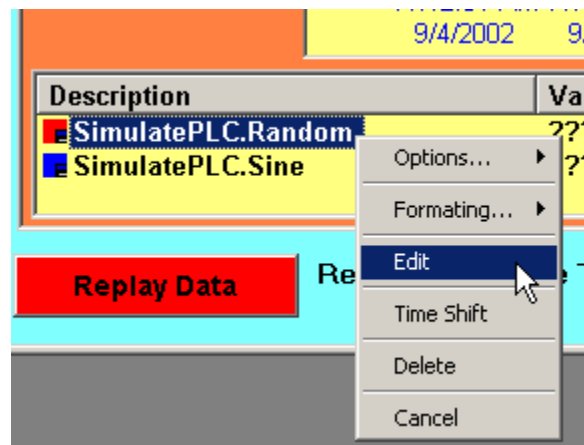
Operator Comments in the Trend Display

Editing Pens During Runtime Mode

To edit a pen during runtime mode, right-click on the pen in the Trend Viewer and select from the following pop-up menu items, as shown in the figure below:

- **Options:** Specifies various options and actions for viewing the selected pen during runtime mode.

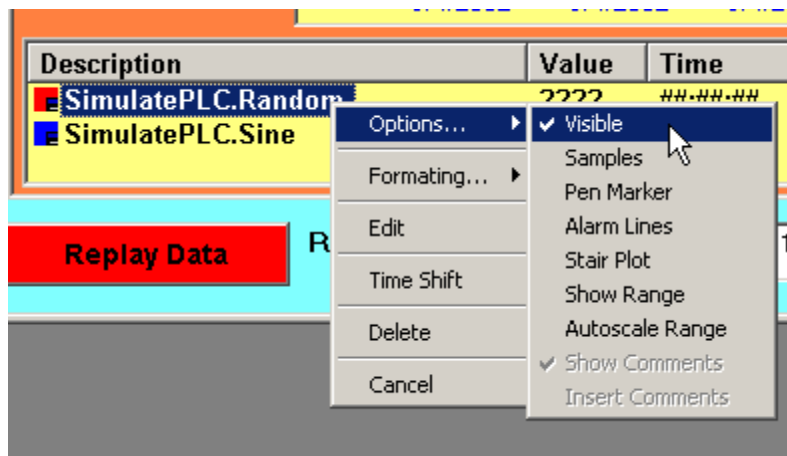
-
- **Formatting:** Restores the default settings for the pen during runtime mode.
 - **Edit:** Opens the **Edit Trend Pen** dialog box.
 - **Time Shift:** Sets the interval and direction for the time shift.
 - **Delete:** Removes the pen from the Trend Viewer.



Editing Pens During Runtime Mode

Pen Options During Runtime Mode

To view the options for pens during runtime mode, right-click the desired pen(s) in the **Details** window and select **Options** from the pop-up menu, as shown in the figure below.



Setting Pen Options in Runtime Mode

The **Options** submenu lists the following trend display options and actions for the selected pen:

- **Visible:** Shows/hides the pen in the Details view of the Trend Viewer.
- **Samples:** Retrieves historical data samples.
- **Pen marker:** Shows/hides the pen marker in the Details view of the Trend Viewer.
- **Alarm lines:** Shows/hides the dashed alarm lines at the **Low**, **LowLow**, **High** and **HighHigh** in the Details view of the Trend Viewer.
- **Stair plot:** Uses a stepwise draw method to plot the trend. This is ideal for plotting slowly changing signals or setpoints.
- **Show range:** Displays the range values for the pen in the Trend Viewer.
- **Autoscale range:** Disables the **Low** and **High** fields, and sets these limits automatically according to the values for the pens. Autoscaling, if checked on the **Ranges** tab, adjusts the range if the actual value is above or below the set limits. For example, if autoscaling is selected and the ranges are set from 0 (Low) to 200 (High) and the actual value is 250, then the ranges will autoscale to 0 to 280. The adjustment will be around 20 percent above the actual value.

-
- **Show comments:** If historical pens are configured in the TrendWorX32 Viewer, selecting **Show Comments** places the Viewer in freeze mode and retrieves all historical operator comments and batch information available.
 - **Insert comments:** Displays the **Comment(s) Entry** dialog box for historical pens. This option is available only in freeze mode.

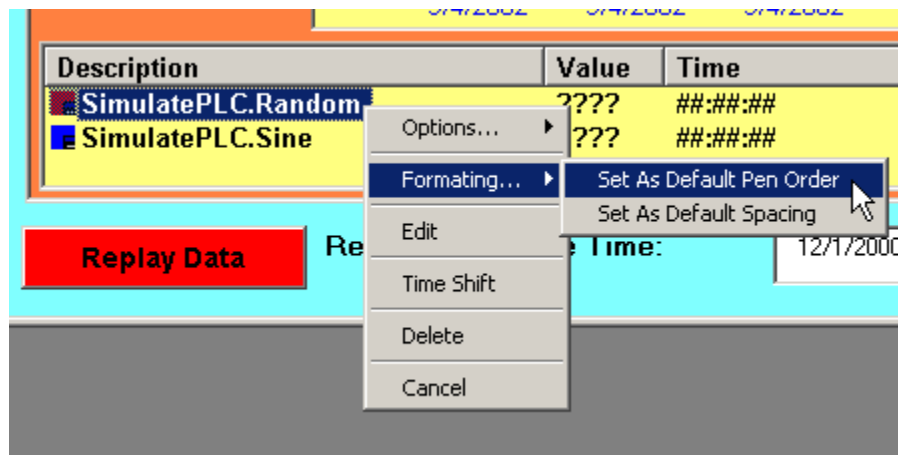
Note: The same functionality is also available for real-time trending.

Formatting Pens During Runtime Mode

To restore the default settings for a pen during runtime mode, right-click the desired pen(s) in the **Details** window and select **Formatting** from the pop-up menu, as shown in the figure below. Choose from the following:

- **Set As Default Pen Order:** Restores the default order (arrangement) of the pen in the Details window during runtime mode.
- **Set As Default Spacing:** Restores the default spacing parameters format for the pen in the Details window during runtime mode.

Note: The TrendWorX32 Viewer ActiveX supports reordering of pens in configuration mode. In the **Pens** tab, select the trend pen(s) you would like to reposition, as shown in the figure below. The **Spacing** button on the **Details** tab allows you to configure a default format for the Details window to be recalled during runtime mode.

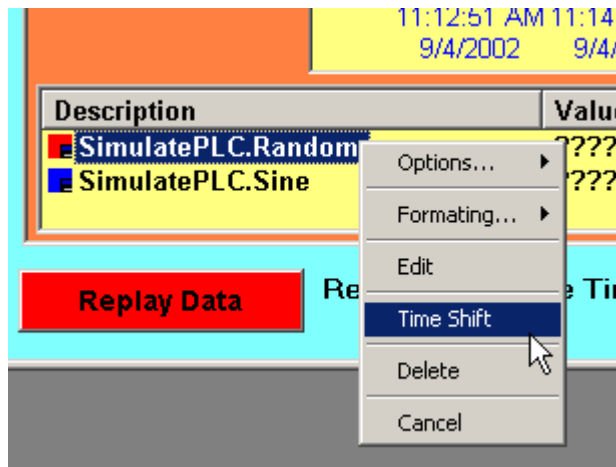


Formatting Pens During Runtime Mode

Time Shifting Pens During Runtime Mode

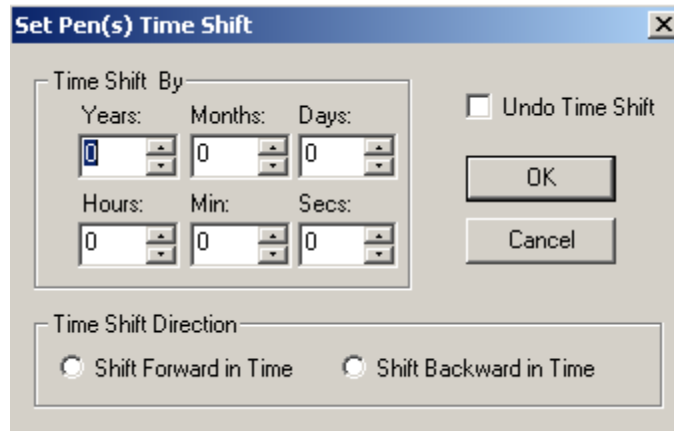
One feature of historical replay is the time shifting of pens in freeze mode. This allows for direct comparison of pens regardless of the time axis settings. To enable time shifting:

1. Right-click the desired pen(s) in the **Details** window and select **Time Shift** from the pop-up menu, as shown in the figure below.



Time Shifting Pens During Runtime Mode

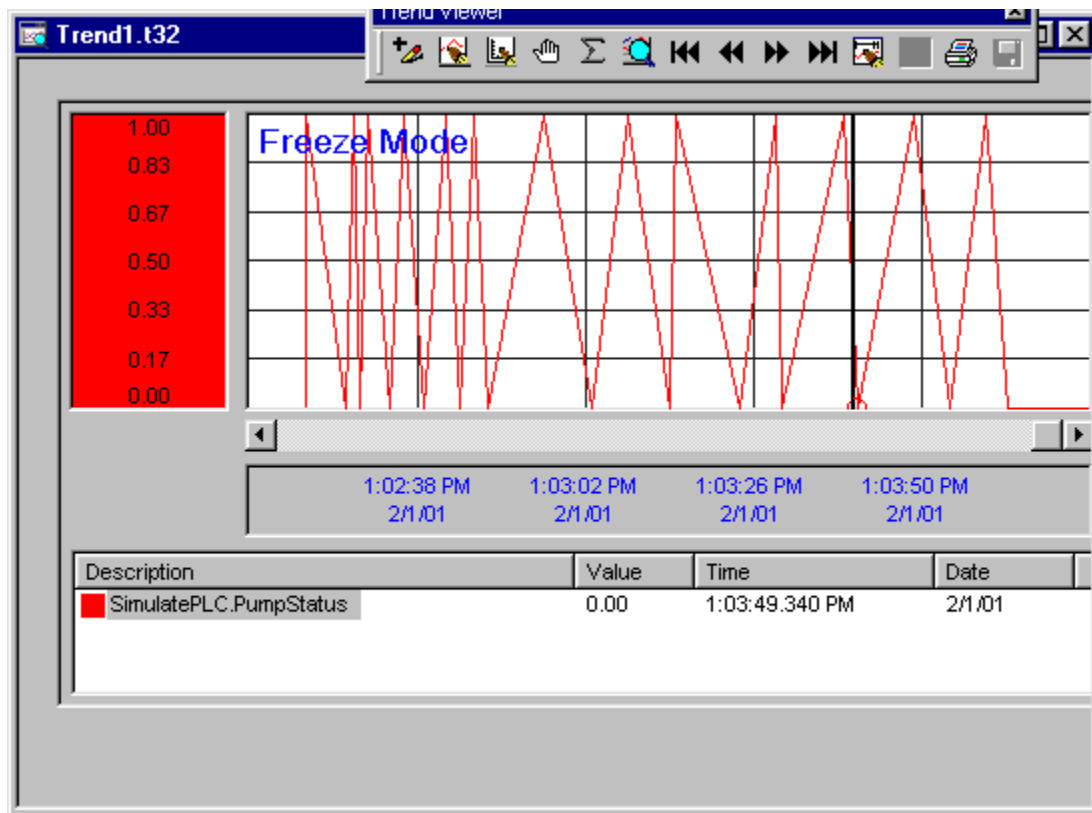
2. This opens the **Set Pen(s) Time Shift** dialog box, as shown below. Choose the interval and direction for the time shift. When you click **OK**, the selected pens will be redrawn to display refreshed historical or real-time data. The time shift operation can be undone at any time by clicking **Undo Time Shift**.



Setting the Time Shift Parameters for Pens

Graphical Time Shifting

There is also a graphical interface for performing time shifting. Again, you need to select the desired pen(s) from the **Details** window. To perform graphical time shifting, press the **CTRL** key and move the mouse within the trend display area while pressing the left mouse button. The time shifting cursor appears, and you can shift backward or forward in time based on which direction you move the mouse, as shown in the figure below.



Graphical Time Shifting

OLE Automation

This section describes the OLE Automation interfaces for the TrendWorX32 Viewer.

Interfaces

Each ActiveX exposes a set of objects, which are used to configure and automate the control. Each object categorizes an area of functionality. The following objects are exposed by the TrendWorX32 Viewer ActiveX:

- **IOTitle**
- **IORanges**
- **IOTimeSettings**
- **IODetails**
- **IOPen**
- **IOGrids**

Properties and Methods for the TrendWorX32 Viewer ActiveX
Properties and Methods for Supporting Objects

Properties and Methods for the TrendWorX32 Viewer ActiveX

The following properties and methods are categorized by the object in which they exist.

Properties
Methods
Events Fired

Viewer: Properties

boolean AutoStartRuntime	Toggles the runtime mode when the container application switches from design mode to active mode. Default: TRUE
boolean ShowTitle	Determines whether the main title is displayed. Default: FALSE
boolean ShowRanges	Determines whether the ranges field is displayed. Default: TRUE
boolean ShowDetails	Determines whether the details field is displayed. Default: TRUE
BSTR PlotType	Determines plot type. Values are: time plot, XY plot, logarithmic plot, strip chart, bar plot, and circular chart.

	Default: time plot
BSTR BorderEdge	Determines trend border area settings. Values are: sunken, raised, etched, and bumped. Default: Sunken
BSTR BorderOptions	Determines trend border area options. Values are: rectangle, left, bottom, right, top, top left, top right, bottom left, and bottom right. Default: Rectangle
OLE_COLOR DispColor	Determines trend curve area color.
OLE_COLOR BackColor	Determines background color.
LPFONTDISP TimeFont	Dispatches interface to the time field font.
LPFONTDISP DetailsFont	Dispatches interface to the detail field font.
LPFONTDISP MainFont	Dispatches interface to the main (title) font.
boolean ShowTimeInfo	Determines if the time field is displayed. Default: TRUE
boolean MainToolbar	Enables/disables Trend Viewer toolbar. Default: TRUE
boolean FreezeMode	Enables/disables freeze mode.
short NumberOfPens	Returns number of currently configured

	pens.
LPFONTDISP RangesFont	Dispatch interface to the range field font.
BSTR FileName	Gets/sets file name for saving configuration data. "Set" is not supported while in runtime mode.
boolean URLPathUsed	Enables/disables use of URL path for asynchronous downloads. Default: FALSE
BSTR URLPathName	Gets/sets URL path name.
BSTR StatisticsFile	Gets/sets the .csv/.txt file used for statistical information storage. "Set" is not supported while in runtime mode.
boolean StatFileAppend	Enables/disables append mode for statistical file. Default: FALSE
boolean GlobalRanges	Enables/disables the use of global ranges. Not supported in runtime (animation) mode. Default: FALSE
boolean ShiftGrids	Enables/disables the shifting of grids during runtime. Not supported in runtime (animation) mode. Default: TRUE

double StartUp	Gets/sets initial offset for plotting of curves. Default: 0
short CursorStep	Gets/sets the cursor step for stepping in freeze mode. Default: 10
LPDISPATCH Title	Dispatch interface to Title object.
LPDISPATCH Details	Dispatch interface to Details object.
LPDISPATCH Ranges	Dispatch interface to ranges object.
LPDISPATCH TimeSettings	Dispatch interface to Time Settings object.
LPDISPATCH XGrids	Dispatch interface to X Grids object.
LPDISPATCH YGrids	Dispatch interface to Y Grids object.
boolean RestoreRanges	Enables/disables restoring of ranges modified during runtime. Not supported in runtime (animation) mode Default: FALSE
long LanguageID	Gets/sets the Language ID for loading a multilingual resource .dll.
boolean FillBackColor	Enables/disables the color fill of trend display areas. This should be used prior to and after printing (within the container) to conserve

	color ink on the printer.
long ScrollBarRange	Sets the range of the scrollbar (in seconds). Default: 10,800 seconds (3 hours)
boolean FullChartReset	Resets Circular Chart after a full trend period elapses. "Set" is not supported while in runtime mode. Default: FALSE
long ReadyState	Signals controls state changes in loading configuration data. Used for asynchronous download of configuration data
short Appearance	Determines TrendWorX32 Viewer ActiveX appearance (3D edge or not). Default: TRUE
short BorderStyle	Determines TrendWorX32 Viewer ActiveX appearance (border or not). Default: TRUE
OLE_COLOR ForeColor	Determines title color and main display text color.
boolean UseLocalSettings	Sets the Local Settings support. If the TrendWorX32 Viewer is configured to save its configuration into a .v32 file, independently of the container, you will have the option in runtime mode to save current settings into the preconfigured file. In the case of a Web HMI application, this will also be the configuration file to be loaded whenever

	the HTML page is refreshed. Default: FALSE
--	--

Viewer: Methods

boolean AddRealTimePen(BSTRPenName)	Adds a real-time pen, which can be an OPC data tag, expression tag, and internal simulation point. For OPC data tags, the "PenName" has to match exactly the string that the Tag Browser would produce for the same tag.
boolean SetXAxis (BSTR PenName)	Used in XY plots to select a pen for x-axis. The "PenName" here represents the description text for the desired pen, as it would appear in the Details window.
VARIANT GetPenNames()	Returns a variant string array of all pen description text as it appears in the Details window of the Viewer ActiveX.
boolean EnterAnimate()	Enters runtime (Animate) mode.
boolean ExitAnimate()	Exits runtime (Animate) mode.
boolean GetAnimateMode()	Returns TRUE if the Viewer is currently in animation mode. Otherwise it returns FALSE.
boolean DeletePen (BSTR PenName, BSTR PenType)	Deletes the pen that has "PenName" as its description text. Valid types are: real time, history, and custom.

void SetPenHiRangeString (BSTR PenName, BSTR RangeString, BSTR PenType)	Sets the point name for high range of the pen with description text matching "PenName." Not supported for historical pens
void SetPenLoRangeString (BSTR PenName, BSTR RangeString, BSTR PenType)	Sets the point name for low range of the pen with description text matching "PenName." Not supported for historical pens
LPDISPATCH GetPen (BSTR PenName, BSTR PenType)	Returns dispatch interface of the pen with description text matching "PenName." You must release the interface when done. Valid types are: real time, history, and custom.
boolean SetDetailsFromPen (BSTR PenName, BSTR PenType)	Populates details information from the pen with description text matching "PenName." Valid types are: real time, history, and custom.
boolean ComputeStat()	Computes and stores statistical info for the pens configured in the Viewer ActiveX. Supported only in runtime (animation) mode.
void SetHiRangeString (BSTR RangeString)	Sets point name for high range of range object. Not supported in history-only mode.
void SetLowRangeString (BSTR RangeString)	Sets point name for low range of range object. Not supported in history-only mode.
boolean ReplacePoint(BSTR PointName, BSTR OldString,	Use to replace pen point names and pen descriptions if desired ("DoPenText" set to

PointName, BSTR OldString, BSTR NewString, boolean DoPenText)	TRUE). "PointName" is the actual (existing) point name. "OldString" is a substring of the "PointName." "NewString" is its substitute string, and if "DoPenText" is set to TRUE, the corresponding pen description text will be replaced as well. Not supported in runtime (animation) mode.
void SetPointValue(BSTR PointName, double PointValue, DATE PointTDate)	Use to feed custom data for an assigned pen. "PointName" is the full point name by which the pen connection was made. Supported only in runtime (animation) mode.
void ChangeWinPeriod (long NewPeriod)	Changes the trend period into the desired "NewPeriod," given in seconds. Not supported for bar plots and XY plots.
boolean SetTrendRightTDate (DATE RightTDate)	In freeze mode, sets the trend's right time and date.
boolean AddRealTimeBgPen(BSTR PenName, BSTR TWXBgGroupName)	Adds a real-time pen from Persistent Trending, which can be an OPC data tag, expression tag, and internal simulation point. For OPC data tags, the "PenName" has to match exactly the string that the Tag Browser would produce for the same tag. "TWXBgGroupName" is the name of the configured Persistent Group.
void DeleteAllPens ()	Deletes all pens from the TrendWorX32 Viewer ActiveX.

boolean AddCustomPen (BSTR PenName)	Adds a custom pen for displaying custom data. You are responsible for supplying data to be displayed by the pen, using the "SetPenValues" method.
Boolean SetPenValues (BSTR PenName, VARIANT* PenValues, VARIANT* PenTDates, VARIANT* PenMSecs, VARIANT* PenQuality)	Fills a custom pen with user data. "PenName" is the description text of the pen, as it appears in the Details window. "PenValues" is an array of doubles. "PenTDates" is an array of date values corresponding to the "PenValues." "PenMSecs" is an array of milliseconds subdivisions corresponding to the "PenTDates" values. "PenQuality" is an array of OPC-defined qualities.
DATE GetRightTime()	Returns the Viewer ActiveX "right" time and date. Supported only in runtime (animation) mode.
boolean SetHistTrendMode(Boolean Mode)	Toggles history-only mode of the Viewer ActiveX. Supported only in configuration mode.
boolean GetHistTrendMode()	Retrieves the status of history replay mode of the Viewer ActiveX.
boolean AddHistoryPen(BSTR PenName)	Adds a history pen to the Viewer ActiveX. The "PenName" string should be the same as the one produced by the Tag Browser when browsing historical servers.
boolean UpdateHistoryPens()	Forces an update of all historical signals. Typically this would be called after setting

	the Viewer's right reference time. Supported only in freeze mode
boolean SetPenFilter(BSTR PenName, short PenFilter)	Sets the data filter type for a pen with description text matching "PenName." Used for historical pens. Valid "PenFilter" types are: OHDA_RAW = 0 OHDA_AVG = 1 OHDA_MIN = 2 OHDA_MAX = 3 OHDA_STDEV = 4 OHDA_LAST = 5 OHDA_MINTIME = 6 OHDA_MAXTIME = 7
void RefreshDetails()	Forces a refresh of the Details window. Since each pen is an individual object, any change in the pen's properties through OLE Automation may not be directly updated in the Details window. Using this ensures that all pen property changes will be reflected properly in the Details window. It is supported only in runtime (animate) mode
void ShowStat()	Shows the statistical window of the Viewer. Supported only in runtime (animation) mode.
boolean DoZoom(short nPercent, boolean bYaxis, boolean bXaxis)	Zoom in on the Viewer. If both "bYAxis" and "bXAxis" are set, then the zoom action will be applied to both axes. For an ordinary time plot, this means both range and time zooming. The "nPercent" parameter controls the zoom factor. If set

	to 0 (zero), it zooms out of the Viewer. Supported only in freeze mode.
boolean LoadSaveFile(BSTR FileName, short Load)	Loads or saves a Viewer ActiveX configuration file.
void SetFirstUpdateMode(boolean bMode)	If "bMode" is TRUE, in freeze mode the Viewer ActiveX will not update historical pens upon connection. You will have to call the "UpdateHistoryPens" method.
long GetPenCount()	Returns the number of configured pens.
LPGDISPATCH GetPenByIndex(long PenIndex)	Returns the pen object, where "PenIndex" is the order of the pen as it appears in the Details window.
boolean ShowPenInRanges(long PenIndex)	Fills in the ranges display with the ranges of the pen, where "PenIndex" is the order of the pen as it appears in the Details window. Not supported if Global Ranges option is enabled.
long AddRealTimePen2(BSTR PenName)	Adds a real-time pen, which can be an OPC data tag, expression tag, and internal simulation point. For OPC data tags, the "PenName" has to match exactly the string that the Tag Browser would produce for the same tag. It returns the index of the pen added.
long AddHistoryPen2(BSTR PenName)	Adds a history pen to the Viewer ActiveX. The "PenName" string should be the same as the one produced by the Tag Browser when browsing historical servers. It returns the index of the pen added.

long AddRealTimeBgPen2(BSTR PenName, BSTR TWXBgroupName)	Adds a real-time pen from Persistent Trending, which can be an OPC data tag, expression tag, and internal simulation point. For OPC data tags, the "PenName" has to match exactly the string that the Tag Browser would produce for the same tag. "TWXBgroupName" is the name of the configured Persistent Group. It returns the index of the pen added.
long AddCustomPen2 (BSTR PenName)	Adds a custom pen for displaying custom data. It returns the index of the pen added.
boolean SetPenTimeShift(long PenIndex, long TimeShift)	Sets the time shift for the pen with index "PenIndex" by "TimeShift" seconds. Not supported for XY plots. Supported only in freeze mode.
long GetPenTimeShift(long PenIndex)	Returns the time shift for the pen with index "PenIndex" in seconds. Not supported for XY plots. Supported only in freeze mode.
boolean RefreshIdealPen(long PenIndex)	Refreshes the ideal pen, with index "PenIndex," with new historical data based on its ideal pen settings.
boolean DeletePen2(long PenIndex)	Deletes the pen with index "PenIndex."
boolean SetPenFilter2(long PenIndex, short PenFilter)	Sets the data filter type for a pen with index "PenIndex." Used for historical pens. Valid "PenFilter" types are: OHDA_RAW = 0 OHDA_AVG = 1 OHDA_MIN = 2

	OHDA_MAX = 3 OHDA_STDEV= 4 OHDA_LAST = 5 OHDA_MINTIME = 6 OHDA_MAXTIME = 7
void ToggleMainToolbar()	Toggles Trend Viewer toolbar if enabled. Supported only in runtime (animation) mode.
void ToggleMainToolbar2(long top, long left)	Toggles Trend Viewer toolbar if enabled and positions it at the top, left coordinates. Supported only in runtime (animation) mode.
void AboutBox().	Displays the Viewer's About Box.
void Refresh()	Standard refresh method. Call after visual modifications.
boolean PrintTrend(boolean bShowPrinterDlg)	Prints the TrendWorX32 Viewer display. If "bShowPrinterDlg" is set to TRUE, the Windows Print dialog box will be evoked. Otherwise, the default Windows printer will be used.
boolean SetInkSaveMode(boolean bMode)	Sets the ink saving mode during printouts. If TRUE, during the printing process all of the Viewer background area colors will be ignored to save ink.
boolean GetInkSaveMode()	Returns the current status of the ink-saving mode.
boolean ResetPenTotalizer(long PenIndex)	Resets a pen's Totalizer function support. The "PenIndex" is the order of the pen.

void SetUTCReplayMode(boolean bMode)	Sets the UTC historical replay mode. Supported only for a history chart Viewer. If TRUE, all curves are drawn using UTC time stamping.
boolean GetUTCReplayMode()	Returns TRUE if the Viewer is replaying historical data in UTC mode.
boolean GetSampleInfo(long PenIndex, VARIANT* SampleValue, DATE* SampleTDate, VARIANT* SampleMSecs, VARIANT* SampleQual, VARIANT* SampleHDAQual, VARIANT* HistoricalSample)	Returns sample information for the current location of the cursor in freeze mode or zoom mode for the pen with index "PenIndex."
void SetCircularPrintMode(boolean bAutoPrint)	Sets the automatic print mode of circular charts upon a full trend period reset. If "bAutoPrint" is set to TRUE, the Viewer will print trend display upon full chart reset.
boolean GetCircularPrintMode()	Returns the current automatic print mode of circular charts.
void MoveCursorForward()	In freeze mode, moves the cursor one step forward in time.
void MoveCursorBackwards()	In freeze mode, moves the cursor one step backward in time.
boolean MoveCursorToTime(DATE timeDate)	In freeze mode, moves the cursor to a particular position on the time axis.
long ChangeTrendDataRate(long	Changes Viewer data-collection rate in

trendDataRate)	animation (runtime) mode. "TrendDataRate" is the desired data-collection rate in milliseconds.
boolean GetPenValues(BSTR PenName, VARIANT* PenValues, VARIANT* PenTDates, VARIANT* PenMSecs, VARIANT* PenQuality)	Returns in variant arrays the current samples in memory buffers of the pen with point name (or description) "PenName."
void SetPenComments(long PenIndex, boolean Mode)	Sets the historical comment and batch information mode of a historical pen with index "PenIndex." If "Mode" is TRUE, it enables the display of historical comments.
boolean GetPenComments(long PenIndex, VARIANT* SourceTDates, VARIANT* SourceMsec, VARIANT* Notes, VARIANT* NoteTDates, VARIANT* NoteMSecs, VARIANT* Authors)	Retrieves the associated (if any) historical operator comments and batch information with the pen having index "PenIndex." Returns variant arrays with all the information, including authors. Supported only in animation runtime mode and for a history pen.
boolean InsertPenComment(long PenIndex, DATE CommentTDate, BSTR Author, boolean GroupComment, BSTR CommentText)	Inserts a historical comment for the historical pen with index "PenIndex." If "GroupComment" is set to TRUE, the inserted comment will be visible by all historical signals of the same data-logging group as the historical pen.
boolean ShowComments()	Toggles in animation (runtime) mode and batch information. If used, the Viewer will be placed in freeze mode, and all historical comments as well as batch information for the historical pens will be retrieved.
void SetPenLoAlarmString(BSTR PenName, BSTR AlarmString, BSTR PenType)	The TrendWorX32 Viewer ActiveX allows the connection of OPC Data Access tags to its alarm limits. The OPC-enabled alarm

void SetPenLoLoAlarmString(BSTR PenName, BSTR AlarmString, BSTR PenType) void SetPenHiAlarmString(BSTR PenName, BSTR AlarmString, BSTR PenType) void SetPenHiHiAlarmString(BSTR PenName, BSTR AlarmString, BSTR PenType)	limits will hold the most current value of the alarm limits. The alarm limits can be a constant value, an OPC Data Access tag, or an expression. In addition, the alarm limits are interfaced to the Global Alias and Language Alias servers. This functionality is not available in History Trend mode due to the nature of this specific historical replay mode. These OLE automation methods have been added to support this functionality.
---	--

ProjectWorX32 Support

Several new OLE Automation methods have been added to facilitate ProjectWorX32 deployment and WebHMI enhanced support:

1. **ReplaceStringInString**(
StringToReplace as String,
OldSubstring as String,
NewSubstring as String
) as String

2. **ReplaceHost**(
OldHostName as String,
NewHostName as String
) as Long

3. **ReplaceFilePath**(
OldSubstring as String,
NewSubstring as String
) as Long

4. **ReplaceTag**(
OldSubstring as String,
NewSubstring as String

) as Long

5. ReplaceHostEx(

OldHostNameSubstring as String,
NewHostNameSubstring as String
MatchCase as Boolean,
MatchWholeWord as Boolean
)
as Long

The **ReplaceHostEx** with parameters **MatchCase=False & MatchWholeWord=True** works exactly the same as the **ReplaceHost()** function.

Events Fired

ReadyStateChange	Fired when the Viewer has finished initializing its state.
InAnimateMode	Fired when the Viewer has entered automation mode.
UnableToCreateThread	Fired when the Viewer cannot create a background trend upon entering animation mode.
OutOfAnimateMode	Fired when the Viewer exits automation mode.
DatabaseChanged	Fired when there are configuration changes, such as adding or deleting a pen.

ErrorInLoadingData	Fired when the Viewer encounters errors upon loading its configuration data through a .v32 file.
RightTimeChanged	Fired when the "right" reference time of the Viewer changed while in freeze mode.
void CursorChangedPos()	Fired in freeze mode as the cursor changes position. You can call the "GetSampleInfo" method to retrieve detailed sample information on a per-pen basis.

Properties and Methods for Supporting Objects

Title Objects

Time Settings Objects

Ranges Objects

Pen Objects

Details Objects

Grids Objects

Title Objects

Title Objects: Properties

Title Objects: Methods

Using Visual Basic To View Title Objects

Title Objects: Properties

boolean ShowTitle

Determines whether the main title is displayed.

Default: TRUE

BSTR TitleText

The main title of the Viewer ActiveX.

Title Objects: Methods

None

Time Settings Objects

Time Settings Objects: Properties

Time Settings Objects: Methods

Using Visual Basic To View Time Settings Objects

Time Settings Objects: Properties

boolean ShowDate	Determines whether date information is displayed in the Viewer's time field. Default: TRUE
boolean ShowMilliSecs	Shows milliseconds divisions on time axis. Default: FALSE
boolean ShowTimeInfo	Determines whether the time field is displayed. Default: TRUE
BSTR BorderEdge	Visual appearance of time display. Values are: sunken, raised, etched, and bumped. Default: Sunken
BSTR BorderOptions	Visual appearance options of time display. Values are: rectangle, left, bottom, right, top, top left, top right, bottom left, and bottom right. Default: Rectangle
BSTR DateFormat	Not used in the current version.
BSTR Size	Size for time display. It can be large, normal, or small.

	Default: Normal
long DataUpdatePeriod	This is the data-collection rate in milliseconds. "Set" is not supported in runtime (animation) mode.
long DataTotalPeriod	This is the trend period in seconds. "Set" is not supported in runtime (animation) mode.
long Samples	This is the number of samples per pen. "Set" is not supported in runtime (animation) mode.
long DataDisplayMult	Gets/sets drawing intervals relative to data collection.
OLE_COLOR FillColor	Color for filling time area.
OLE_COLOR TextColor	Color for text displayed.
long HistoryRefreshRate	Gets/sets the history refresh rate for which historical pens will be refreshed, regardless of the data-collection rate. Ideally, it should be configured to be as close as possible to the logging-to-disk interval set in the TrendWorX32 Configurator.

Time Settings Objects: Methods

None

Ranges Objects

Ranges Objects: Properties

Ranges Objects: Methods

Using Visual Basic To View Ranges Objects

Ranges Objects: Properties

boolean OverHighRange	Gets/sets override for high range. If set TRUE, the high-range value will be computed from a tag or expression.
boolean OverLowRange Short VisiblePens	Gets/sets override for low range. If set TRUE, the low-range value will be computed from a tag or expression. Gets/sets the number of visible pen ranges on the ranges display area (if the Global Ranges option is not selected).
boolean ShowRanges	Determines whether the range field is displayed. Default: TRUE
BSTR BorderEdge	Visual appearance of range display. Values are: sunken, raised, etched, and bumped. Default: Sunken
BSTR BorderOptions	Visual appearance options of range display. Values are: rectangle, left, bottom, right, top, top left, top right, bottom left, and bottom right. Default: Rectangle
BSTR NumberFormat	Gets/sets number format for range

	values.
BSTR Position	Gets/sets position of range display. It can be right or left.
BSTR XNumberFormat	Gets/sets number format for x-axis range values in XY plots.
double XHigh	Gets/sets high-range value for x-axis in XY plots.
double XLow	Gets/sets low-range value for x-axis in XY plots.
double High	Gets/sets high-range value.
double Low	Gets/sets low-range value.
OLE_COLOR FillColor	Gets/sets fill color of range display.
OLE_COLOR TextColor BSTR Size	Size for time display. It can be large, normal, or small. Default: Normal

Ranges Objects: Methods

None

Pen Objects

Pen Objects: Properties

Pen Objects: Methods

Using Visual Basic To View Pen Objects

Pen Objects: Properties

boolean Autoscale	Autoscales the pen.
boolean EnableAlarmLines	Enables drawing of alarm lines.
boolean OverHighRange	Gets/sets override for high range. If set TRUE, the high-range value will be computed from a tag or expression.
boolean OverLowRange	Gets/sets override for low range. If set TRUE, the low-range value will be computed from a tag or expression.
boolean PenVisible	Determines whether a pen is to be drawn.
boolean PenXAxis	Gets/sets a pen x-axis in XY-plots.
boolean ShowMarkers	If TRUE, a triangular marker is displayed at the position of the last known sample value.
boolean ShowSamples	Indicates real samples.
boolean Stairplot	Enables step-wise plotting mode.
boolean UpdateServerRanges	Controls the initial update of the ranges from the server.
BSTR DetailsFormat	The format type for the details.

BSTR PenPointName	The pen signal point name.
BSTR PenStyle	Gets/sets the pen style, (e.g. solid, dotted, dashed).
BSTR PenText	Pen description.
BSTR PenUnits	Pen engineering units.
BSTR RangeFormat	The format type for showing ranges.
BSTR TWXBgGroupName	The Persistent Trend group name.
double AlarmHigh	High alarm value for pen.
double AlarmHiHi	High-high alarm value for pen.
double AlarmLoLo	Low-low alarm value for pen.
double AlarmLow	Low alarm value for pen.
double HighLimit	High limit value.
double LowLimit	Low limit value.
long NumberOfSamples	Gets number of samples for the pen.
OLE_COLOR DetailColor	The fill color to be used in the Details window.
OLE_COLOR PenColor	Gets/sets the color associated with the pen.

OLE_COLOR RangeColor	The fill color to be used in the range window.
OLE_COLOR RangeTextColor	The color to be used for text in the range window.
short HistoryFilter	Returns the filter type for the historical data.
short PenWidth	The width of the pen in logical units. Default: 1
BSTR PenType	Returns the type of the pen, such as real-time, history, and custom. Read only property.
short HistoryFilter	Returns the history filter for the pen. Possible values are: Raw = 0 Average = 1 Minimum = 2 Maximum = 3 Standard deviation = 4 Last sample = 5 Minimum with actual time = 6 Maximum with actual time = 7
double AutoResetLevel	Gets/sets the autoreset value for a pen in configuration mode, if the totalizer mode is enabled.
double PreLoadValue	Gets/sets the preload value for a pen in configuration mode, if the totalizer mode is enabled.
double PresetValue	Gets/sets the preset value for a pen in configuration mode, if the totalizer mode is

	enabled.
double IntegrationFactor	Gets/sets the Integration factor value for a pen in configuration mode, if the totalizer mode is enabled. Possible values are: 1.0 (integration constant computed in seconds) 60.0 (integration constant computed in minutes) 3660.0 (integration constant computed in hours) 86400.0 (integration constant computed in days)
long IntegrationDir	Gets/sets the integration direction value for a pen in configuration mode, if the totalizer mode is enabled. Possible values are: 0 (Totalizer counts upward from 0.) 1 (Totalizer counts downward to 0.)
long TotalizerMode	Gets/sets the totalizer mode value for a pen in configuration mode if the totalizer mode is enabled. Possible values are: 0 (continuous mode) 1 (preload mode) 2 (preset)
boolean EnableTotalizer	Enables/disables the totalizer mode for a pen in configuration mode.
long PlotterMark	Gets/sets the totalizer plotter mark size for the pen's curve, if the totalizer mode is enabled. Possible values are:

	No marker = 0 Tiny = 1 Small = 2 Medium = 3 Big = 4 Huge = 5
long TotalizerUnits	Gets/sets the totalizer units value for a pen in configuration mode, if the totalizer mode is enabled. Possible values are: 0 (integration factor computed in seconds; units are seconds) 1 (integration factor computed in minutes; units are minutes) 2 (integration factor computed in hours; units are hours) 3 (integration factor computed in days; units are days)
boolean IdealPenMode	Gets/sets the ideal pen mode for a historical pen.
boolean AutoTimeSync	Gets/sets the autosync mode for a real time pen.

Pen Objects: Methods

boolean GetPenStat(double* Max, double* Min, double* Avg, double* StdDev, long* NumSamples)	Returns the associated statistical values for the pen using the visible samples at the current state of the pen.
boolean IsHistoryPen()	Returns TRUE if the pen is historical.

boolean SetIdealDate(DATE Date)	Sets the ideal date for a historical ideal pen. This is the reference starting time of the ideal pen.
DATE GetIdealDate()	Sets the Ideal date for a historical ideal pen. This is the reference starting time of the ideal pen.

Details Objects

Details Objects: Properties

Details Objects: Methods

Using Visual Basic To View Details Objects

Details Objects: Properties

boolean ShowDetails	Determines whether the details field is displayed. Default: TRUE
boolean ShowPenDate	Determines whether a pen's date information is displayed.
boolean ShowPenHiHiLoLoAlarm	Determines whether a pen's "HiHi" and "LoLo" alarm information is displayed.
boolean ShowPenHiLo	Determines whether a pen's "Hi/Lo" information is displayed.
boolean ShowPenHiLoAlarm	Determines whether a pen's "Hi/Lo" alarm information is displayed.
boolean ShowPenPoint	Determines whether a pen's information is displayed.
boolean ShowPenText	Determines whether a pen's text is displayed.
boolean ShowPenTime	Determines whether a pen's time information is displayed.
boolean ShowPenUnits	Determines whether a pen's units are displayed.

boolean ShowPenValidity	Determines whether signal quality will be displayed.
boolean ShowPenValue	Determines whether a pen's value is displayed.
boolean ShowToolTips	TRUE if plot ToolTips are to be shown for the currently focused pen (when in freeze mode).
boolean ShowXAxis	Determines whether x-axis values are shown for XY plots.
BSTR BorderEdge	Visual appearance of the Details window.
BSTR BorderOptions	Visual appearance of the Details window.
Short Visible Pens	Gets/sets the number of visible pen details to be drawn in the Details window.
BSTR NumberFormat	Gets/sets the number format of the pen.
BSTR Size	Gets/sets the size of the Details window.
OLE_COLOR FillColor	The background color of the Details window.
OLE_COLOR TextColor	The color of all text in the Details window.
boolean ShowHeader	Enables/disables the display of the header in the Details window.

boolean ShowFilter	Enables/disables the display of the data filter in the Details window.
boolean ShowMSecs	Enables/disables the display of milliseconds in the time column of the details window.
boolean ShowPenStyle	Enables/disables the display of the pen's style in the Details window, rather than the squares filled in the color of the pen.

Details Objects: Methods

None

Grid Objects

Grid Objects: Properties

Grid Objects: Methods

Using Visual Basic To View Grid Objects

Grid Objects: Properties

boolean ShowGrids	Determines whether grids are displayed in the Viewer's main display area. Default: TRUE.
BSTR GridStyle	Gets/sets grid style, such as solid, dotted, and dashed.
OLE_COLOR GridColor	The color of the grid.
short GridNumber	Gets/sets grid number
short GridWidth	The grid width in logical units.
short SubGridNumber	Gets/sets the subgrid number for the Viewer in configuration mode, if the circular chart mode is enabled.
short SubGridWidth	Gets/sets the subgrid width for the Viewer, if the circular chart mode is enabled.
BSTR SubGridStyle	Gets/sets the subgrid style for the Viewer, if the circular chart mode is enabled.
boolean ShowSubGrid	Enables/disables the display of the subgrids number for the Viewer, if the circular chart mode is enabled.

OLE_COLOR SubGridColor	Gets/sets the subgrid color the for Viewer, if the circular chart mode is enabled.
short GridsBetweenLabels	Gets/sets the grids between labels value in the Viewer, if the circular chart mode is enabled.
short CircularEdgeMode	Gets/sets the circular edge appearance for the Viewer, if the circular chart mode is enabled. Possible values are: 0 (No edge) 1 (Default) 2 (Thick)

Grid Objects: Methods

None

Code Samples

Using the Component Inside Visual Basic

The TrendWorX32 Viewer ActiveX easily integrates into any Visual C++ or Visual Basic container application. The following are some code samples for using the TrendWorX32 Viewer ActiveX inside Visual Basic:

- a. Disable Global Ranges and animate the component

```
With TWXViewer1
    .GlobalRanges = False
    .EnterAnimate
End With
```

- b. Freeze the Trend

```
With TWXViewer1
    .FreezeMode = True
End With
```

- c. Unfreeze the Trend

```
With TWXViewer1
    .FreezeMode = False
End With
```

- d. Modify X- axis Grid Style

```
Dim xGrids As Object
'Get XGrids Interface
Set xGrids = TWXViewer1.xGrids
With xGrids
    .GridStyle = "Dotted"
    .GridNumber = 8
End With
```

```
' Call Refresh method
TWXViewer1.Refresh
'Release Interface
Set xGrids = Nothing
```

- e. A full script

```
' Get Pen Names
Dim penNames As Variant
Dim Count As Integer
Dim penDel As String
```

```

penNames = TWXViewer1.GetPenNames
'Show to the user
For Count = 1 To UBound(penNames)
    penDel = penNames(Count)
    MsgBox (penDel)
Next Count
' Now disable pen drawing for last pen
Dim pen As Object
'Get Pen's Interface
Set pen = TWXViewer1.GetPen(penDel, "Real Time")
MsgBox ("Hiding Last Pen !")
'Hide Pen
With pen
    .PenVisible = False
End With

'Call refresh method
TWXViewer1.Refresh
MsgBox ("Showing Last Pen !")
'Show pen again
With pen
    .PenVisible = True
End With

'Call Refresh method
TWXViewer1.Refresh
MsgBox ("Deleting Last Pen !")
'Now delete this pen
'Release Pen Interface
Set pen = Nothing
'Delete pen
TWXViewer1.DeletePen penDel, "Real Time"
'Call refresh method
TWXViewer1.Refresh

f. Exit Runtime (Animate Mode)
    With TWXViewer1
        .ExitAnimate
    End With

```

Note: All of the OLE Automation objects exposed by the TrendWorX32 Viewer ActiveX are already created when the component is installed. The programmer should get their interface and not try to create them.

When you programmatically obtain an interface to one of TrendWorX32 objects, the reference count for that interface is increased. Therefore, you should always release the corresponding interface once it is not needed anymore. In addition, you should *always* check for "null" (or "nothing" in Visual Basic) interface pointers. Depending on the state of the TrendWorX32 Viewer component and the configured security settings, you may not have access to a certain object, and the returned interface in this case is a "null" interface.

VBA Examples

Note: TrendWorX32 includes a set of VBA examples under the "ICONICS\GENESIS-32\Examples\TrendWorX32 Examples" directory. The following folders located inside the "Examples" directory are associated with the Viewer.

Aliasing. The files in this folder demonstrate the use of the OLE Automation interface of the TrendWorX32 Viewer ActiveX within a VBA-enabled application, such as GraphWorX32, in aliasing.

Real-Time Trend Configuration. The files in this folder demonstrate the use of the OLE Automation interface of the TrendWorX32 Viewer ActiveX within a VBA-enabled application, such as GraphWorX32. It demonstrates how to add pens "on the fly" by reading the pen configuration from a Microsoft Access database (.mdb) file and utilizing the OLE Automation properties of the TrendWorX32 ActiveX through VBA scripts.

TrendWorX32 VBA Examples. This folder contains several TrendWorX32 displays that demonstrate the rich VBA OLE Automation support of the TrendWorX32 Viewer Trend ActiveX Control.

TrendWorX32 Data Logging and Reporting. This folder contains sample TrendWorX32 data-logging configuration, a sample historical database, as well as sample reporting and historical replay displays.