GraphWorX64 Scripting – Interacting with a TrendWorX64 Viewer

APPLICATION NOTE

July 2011

Description: A scripting guide demonstrating how to interact with the TrendWorX64 Viewer control.
OS Requirement: Windows Server 2003 x64/Vista x64/ Server 2008 x64/Windows 7 x64/ Server 2008 R2 x64
General Requirement: General knowledge of GraphWorX64 and its scripting capabilities; familiarity with the TrendWorX64 Viewer control.

Introduction

This guide will introduce you to writing scripts that interface with the TrendWorX64 Viewer. It includes an example of how to add new real-time pen into TrendWorX64 Viewer, how to change the title of the chart and how to swap X and Y Axis using GraphWorX64 Scripting.

This document assumes that you have read the *GraphWorX64 Scripting – Quick Start* application note.

Preparing the display

To create this example we will need a display containing a TrendWorX64 Viewer control with a pen, plus four buttons to trigger our script. Start GraphWorX64, create a new display and add then add a TrendWorX64 Viewer and four buttons. Label the buttons, "Add Pen", "Change Title", "Swap XY Axis", and "Toggle Legend". Edit the TrendWorX64 Viewer and add a Plot to the Chart.

It is very important that the trend viewer control has a name. Our scripts will use this name to identify which trend viewer to interact with (in case there is more than one on a display). To name your trend viewer, double-click on it to open the properties then go to the Advanced tab. (You could also click once on the trend viewer and look at the Properties panel.) In the Common section fill in "TwxControl" for the "(Name)" field.

Your display should look something to the one in Figure 1.

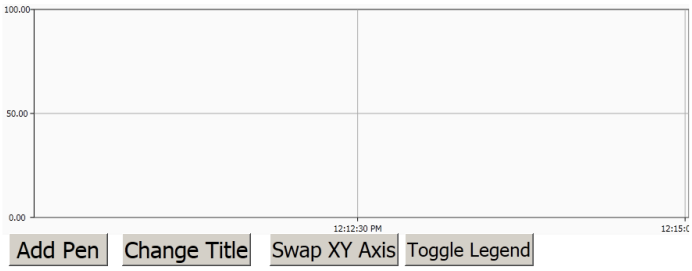

Figure 1 - Display with Trend Viewer and Buttons

Configure the buttons to perform the Run Script command and create a new function from the Events page. (The procedure for this is described in the *GraphWorX64 Scripting – Quick Start* application note.)

Adding References

The following code requires some extra references beyond the defaults. To add these references, open the Script Editor and click on the "Add References" button on the left.

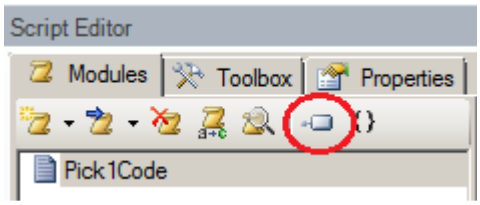

Figure 2 - Add References button

Scroll down to the bottom of the References window and click in the cell for Assembly Codebase in the last empty row. Click the "…" button that will appear when you click in the cell. Browse to one of these files:

C:/Program Files/ICONICS/GENESIS64/Components/IcoWPF.DLL
C:/Program Files/ICONICS/GENESIS64/Components/IcoDataConnector.DLL

Repeat for the other file. When you are done, your references should look something like Figure 3:

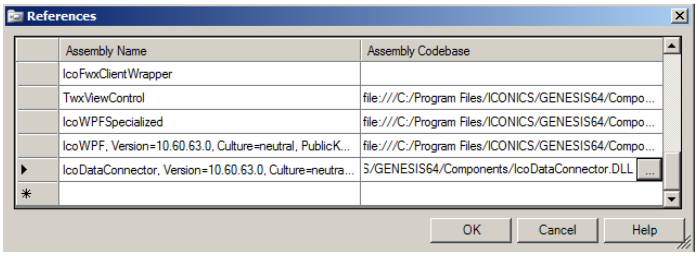

Figure 3 - Added references

Click OK when all the references have been added.

Designing the Script

To add new pen we will use a specific method exposed by the TrendWorX64 Viewer. To do this, we first need to understand a little bit better how the TrendWorX64 Viewer control is organized.

The TrendWorX64 Viewer control can have several tabs, and each tab can have one or more charts in it. For example, in the next figure we have a control with two tabs, with the first tab contains two charts.

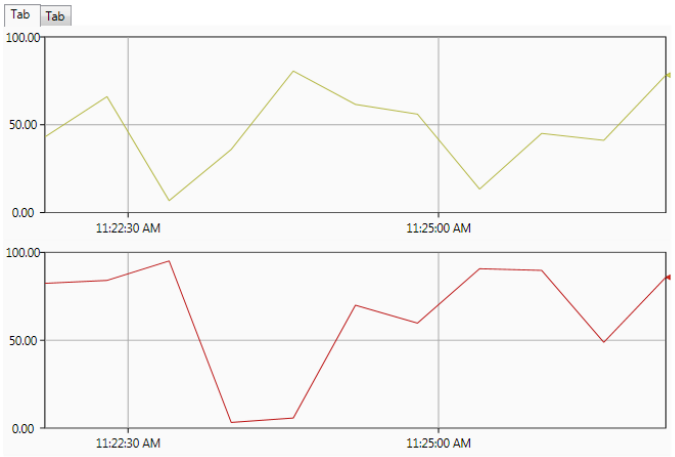

Figure 4 - A More Complicated TrendWorX64 Viewer

For this example we will use a TrendWorX64 Viewer control with just one tab and one chart, but it is important for you to understand that in a real application you may have a more complex scenario to work with.

Before writing the actual code, let's review what we need to do in our scripts:

1. First, we need to get a reference to the TrendWorX64 Viewer control.
2. Then, we need to get a reference to its first tab.
3. From the tab, we take a reference to the chart.
4. From the chart, we take a reference to the series
5. Once we have the series' reference, we can invoke the method to add new pen, change the title, or swap the axis.

Writing the Add Pen script

Now that we know what needs to be done we can start writing the code to implement the script. This code should go with the "Add Pen" button.

First we need to know how to get a reference to the TrendWorX64 Viewer control. We will declare a variable of the correct type, then use the GetObjectByName method to retrieve the reference to it.

```
var viewer : Ico.Twx.TwxViewControl;
viewer =  Ico.Twx.TwxViewControl
    (ThisConfiguration.
    GetObjectByName("TwxControl").ToDependencyOb
    ject());
```

Now we will get the reference to the first (and only) available tab. With this next line we are accessing the first element of the Items property of the viewer control.

```
var tab : Ico.Windows.Controls.MultipleTabItem;
tab = Ico.Windows.Controls.MultipleTabItem
    (viewer.Items[0]);
```

Using a similar approach we get a reference to the first chart of the chosen tab.

```
var chart : Ico.Twx.TwxChartComponent;
chart = Ico.Twx.TwxChartComponent
    (tab.Items[0]);
```

And then get the first series of that chart.

```
var series : Ico.Twx.TwxSeries;
series = Ico.Twx.TwxSeries(chart.Series[0]);
```

We are now ready to create a pen object and add that new pen to that series.

```
var pen : Ico.Twx.TwxChartPen = new
    Ico.Twx.TwxChartPen();
series.Pens.Add(pen);
```

Now we must create a new data connection configuration and add it to the created pen to tell the pen what data source to use.

```
var config :
    Ico.Data.DataConnector.DataConnectorConfigur
    ation;
config = new
    Ico.Data.DataConnector.DataConnectorConfigur
    ation();
config.PointName = "localsim:sine";
pen.DataConnectionConfiguration = config;
```

The code is complete. We can save our display, go to runtime mode, then press the button add our new pen.

> **NOTE:** Wait a moment if you do not see the pen immediately. It may need a few seconds to begin drawing its line.

> **NOTE:** See the last page for the completed script.

Writing the Change Title Script

This example, which goes with the "Change Title" button, is much simpler than the first.

As in the first example, create a reference to the viewer, then we only need to call this line to change the title:

```
viewer.Header = "New Title";
```

The code is complete. Save our display, go to runtime mode, click on the Change Title button to change the title of the viewer.

> **NOTE:** See the last page for the completed script.

Writing the Swap Axis Script

This example goes with the "Swap XY Axis" button. As in the first example, get a reference to the viewer, tab, and chart. Then call this line to swap the axes:

```
chart.SwapAxes = !chart.SwapAxes;
```

Save the display, go into runtime, click on swap axis button and you should now see the time axis being vertical.

NOTE: See the last page for the completed script.

Writing the Toggle Legend Script

This example goes with the "Toggle Legend" button. As in the first example and the Swap Axis example, get a reference to the viewer, tab, and chart. Then call this line to toggle the legend:

```
chart.ShowLegend = !chart.ShowLegend;
```

NOTE: See the last page for the completed script.

Conclusion

These simple examples show some important concepts, like how to get a reference to a TrendWorX64 Viewer control. This is very similar to the procedure for getting a reference to other controls as well. You've also learned about how the TrendWorX64 Viewer is structured and how to interact with its elements.

Feel free to experiment with the other available methods. You can find these by using the Intelli-sense built into the Script Editor, or by browsing the GENESIS64 API Reference Help.

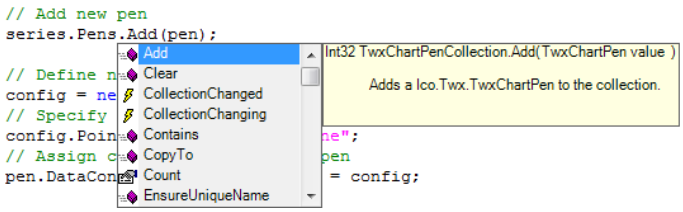

Figure 5 - Intelli-sense Showing Available Methods and Properties

Complete code

Here is the full code of the scripts we created above, along with some comments. (The code continues to the next page.)

```
function Pick1_CommandExecuted(sender : System.Object, cmdArgs : Ico.Gwx.CommandExecutionEventArgs)
{
	// Get a reference to the TWX Viewer
	var viewer : Ico.Twx.TwxViewControl;
	viewer =  Ico.Twx.TwxViewControl (ThisConfiguration.
GetObjectByName("TwxControl").ToDependencyObject());

	// Get a reference to the first (and only) tab.
	var tab : Ico.Windows.Controls.MultipleTabItem;
	tab = Ico.Windows.Controls.MultipleTabItem (viewer.Items[0]);

	// Get a reference to the first (and only) chart
	var chart : Ico.Twx.TwxChartComponent;
	chart = Ico.Twx.TwxChartComponent (tab.Items[0]);

	// Get a reference to the first (and only) series
	var series : Ico.Twx.TwxSeries;
	series = Ico.Twx.TwxSeries(chart.Series[0]);

	// Create a pen and add it to the series.
	var pen : Ico.Twx.TwxChartPen = new Ico.Twx.TwxChartPen();
	series.Pens.Add(pen);

	// Link the pen to a value
	var config : Ico.Data.DataConnector.DataConnectorConfiguration;
	config = new Ico.Data.DataConnector.DataConnectorConfiguration();
	config.PointName = "localsim:sine";
	pen.DataConnectionConfiguration = config;

}
```

Complete code, continued…

```
function Pick2_CommandExecuted(sender : System.Object, cmdArgs : Ico.Gwx.CommandExecutionEventArgs)
{
	// Get a reference to the TWX Viewer
	var viewer : Ico.Twx.TwxViewControl;
	viewer =  Ico.Twx.TwxViewControl (ThisConfiguration.
GetObjectByName("TwxControl").ToDependencyObject());

	// Change the viewer title
	viewer.Header = "New Title";
}

function Pick3_CommandExecuted(sender : System.Object, cmdArgs : Ico.Gwx.CommandExecutionEventArgs)
{
	// Get a reference to the TWX Viewer
	var viewer : Ico.Twx.TwxViewControl;
	viewer =  Ico.Twx.TwxViewControl (ThisConfiguration.
GetObjectByName("TwxControl").ToDependencyObject());

	// Get a reference to the first (and only) tab.
	var tab : Ico.Windows.Controls.MultipleTabItem;
	tab = Ico.Windows.Controls.MultipleTabItem (viewer.Items[0]);

	// Get a reference to the first (and only) chart
	var chart : Ico.Twx.TwxChartComponent;
	chart = Ico.Twx.TwxChartComponent (tab.Items[0]);

	// Swap the axes
	chart.SwapAxes = !chart.SwapAxes;
}


function Pick4_CommandExecuted(sender : System.Object, cmdArgs : Ico.Gwx.CommandExecutionEventArgs)
{
	// Get a reference to the TWX Viewer
	var viewer : Ico.Twx.TwxViewControl;
	viewer =  Ico.Twx.TwxViewControl (ThisConfiguration.
GetObjectByName("TwxControl").ToDependencyObject());

	// Get a reference to the first (and only) tab.
	var tab : Ico.Windows.Controls.MultipleTabItem;
	tab = Ico.Windows.Controls.MultipleTabItem (viewer.Items[0]);

	// Get a reference to the first (and only) chart
	var chart : Ico.Twx.TwxChartComponent;
	chart = Ico.Twx.TwxChartComponent (tab.Items[0]);

	// Toggle the legend
	chart.ShowLegend = !chart.ShowLegend;
}
```