

Description: Guide to using the symbols library
OS Requirement: Win 2000, XP Pro, Server 2003, Vista, Server 2008, Windows 7
General Requirement: GraphWorX32 must be installed

Introduction

The Symbols Library provides a way for users to quickly build GraphWorX32 displays. Within the symbols library, you will find smart symbols. They contain built-in dynamics that you can add any combinations of OPC tags, local aliases, expression, or global aliases. The symbols can be anything from a simple switch that turns green on true and red on false, to a complicated panel full of switches, dials, and sensors representing a control device on your manufacturing floor.

You can significantly reduce development time by using the symbols library. This document will walk you through the steps of using the Symbols Library to create GraphWorX32 displays.

The Symbols Library

1. Start GraphWorX32 by going to Start → Programs → ICONICS GENESIS32 → GraphWorX32 → GraphWorX32
2. From the menu, select Format → Application Preferences...
3. In the Application Preferences window, click on the Preferences tab, you will see your Symbols Library Styles near the bottom of the Application Settings list.

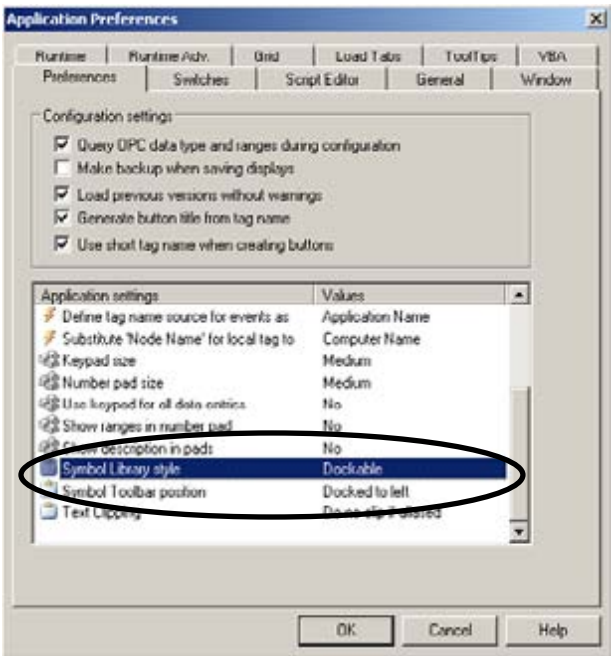


Figure 1 - Application Preferences

4. You can change your Symbols Library Style and docking position here.

NOTE: If you select your Symbols Library to be a standalone application, you will only be able to launch it using the Symbols Icon in the Draw Toolbar.
5. You can click “OK” to exit the Application Preferences window.
6. From the menu, select View → Toggle Symbol Toolbar to bring up the symbols library if it is not already opened.

NOTE: Selecting this option when the symbols toolbar is already displaying will hide the toolbar.
7. You can navigate the symbols toolbar by using the drop down list. The folder icon allows you to go up one directory.
8. You can also personalize your symbols toolbar by clicking on the library icon.

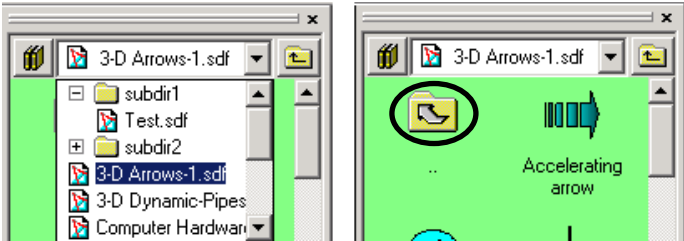


Figure 2 - Navigating the Symbols Library

9. You can also use the folder with and arrow pointing upward to move up one directory.

GraphWorX32 Displays with Symbols

1. In the Symbols toolbar, navigate to 2_Misc → Switches & Sliders.sdf → UpDown1
2. Drag and drop the symbol UpDown1 into the working area of your GraphWorX32 display.
3. You can resize the symbol.
4. Double-click on the symbol to bring up the properties inspector.
5. Select the Digital Selector, you will see two tags, click on each tag and notice how the picture changed as the tags are selected and the show object options change.

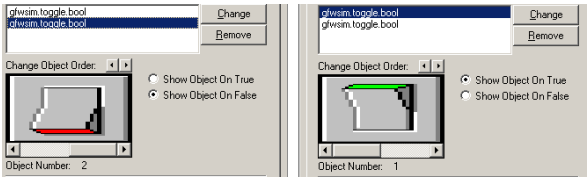


Figure 3 - gfwsin.toggle.bool True and False Display

6. Click on the Pick tab, you will see that it is a Toggle Value Action, the same tag as the data source and the two values it toggles between.
7. Click “OK” to exit the properties. Go into run time, switch work in action. If you mouse over the switch to show the tag, you will notice the tag value is toggling between 1 and 0 automatically.
8. Exit runtime, we will edit the tags so that it will only change value when we click on it.
9. Right-click on the Switch and select “Edit Connections...”

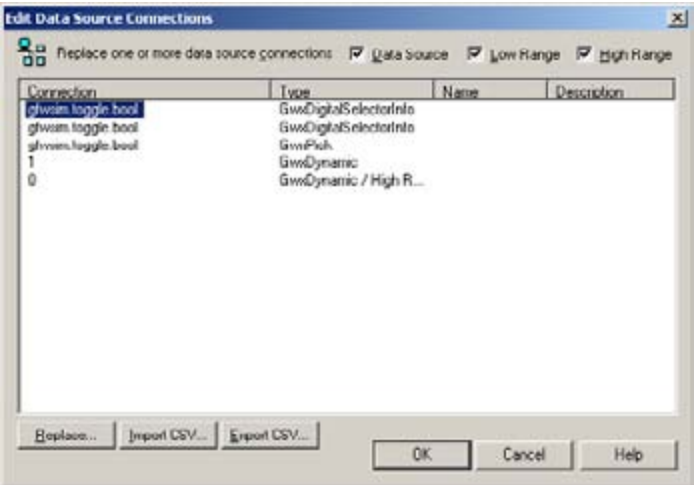


Figure 4 - Edit Data Source Connections

10. Select one of the gfwsm.toggle.bool and click on the “Replace...” button.
11. In the Replace Connection window, click on the ... button and select Unified Data Browser.
12. Browse to the BIT tag from ICONICS.Simulator.1 → SimulatePLC → OUTPUTS → BIT, and click on “OK”.

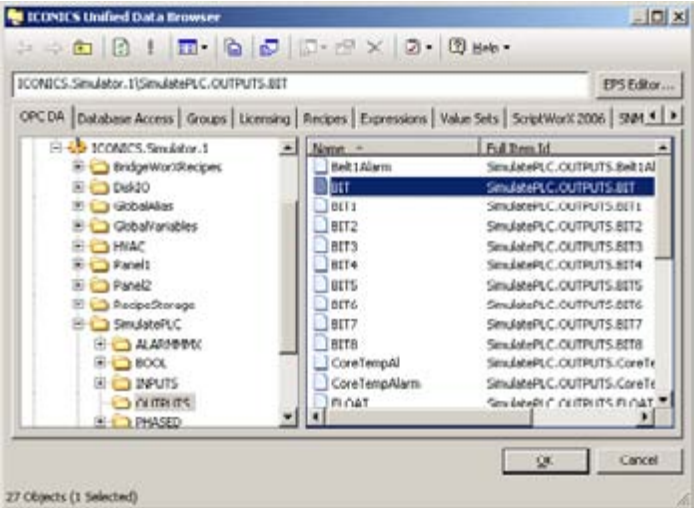


Figure 5 - Unified Data Browser

13. In the Replace Connections window, select the All option in the upper left-hand corner, and then Click on “OK”.



Figure 6 - Edit All Connections in the Symbol

14. In the Edit Data Source Connections window, you should see all of your connections changed to ICONICS.Simulator.1\SimulatePLC.OUTPUTS.BIT. Click on “OK”.

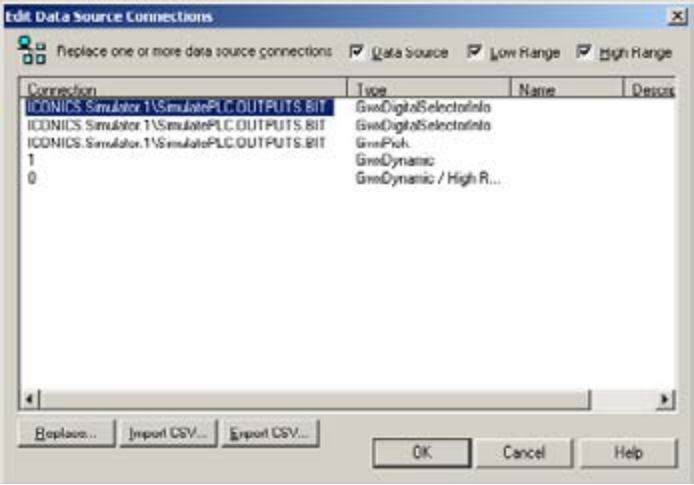


Figure 7 - Changed Data Source Connections

15. Enter runtime.
16. Your switch will now only change when you click on it.
17. You can edit the connections to connect to any other OPC points that you desire. You can also use expressions, local aliases, global aliases, local variables, or simulation variables in the data source.