ProjectWorX32 – Adding Third Party Applications

APPLICATIONS NOTE

April 2010

Description: Guide to adding third party applications to ProjectWorX32.
OS Requirement: Win 2000, XP Pro, Server 2003, Vista, Server 2008, Windows 7
General Requirement: Knowledge of ICONICS ProjectWorX32, Microsoft Visual C++, and programming in a windows environment. You also must have Visual C++ installed on the machine.

Introduction

ICONICS designed ProjectWorX32 with the ability to adopt additional applications or OPC Servers. In order to add a customized item, ProjectWorX32 needs various data to make a user-defined set of behavior and appearance.

The required data is gathered from customized dynamic link library (.dll) files. Each .dll file represents one child of the "Applications" or "OPC Server" node. The .dll file's resource descriptor contains all of the string constants and icons needed to make the item appear in ProjectWorX32 as desired. Internal code in the .dll file defines actions that allow ProjectWorX32 to interface with the product being added to the tree, and perform such actions as launching the application or creating a new document for the application.

Initially, ProjectWorX32 is blind to the capabilities and appearance of the application being added, and the way it appears in the tree. Therefore, all of the .dll files that represent custom applications share a common automation interface, or a common set of exposed functions, through which ProjectWorX32 can discover specifics such as string constants, icons, and code. When the .dll file registers, it informs ProjectWorX32 of its existence (and its interface) and when it un-registers, it clears itself from a list of custom applications.

ProjectWorX32 Add-On AppWizard

1. Locate the file PWXExtensions.awx on your installation CD. It should be in **Tools\ProjectWorX32 Toolkit**
2. Copy the file into the **C:\Program Files\Microsoft Visual Studio\Common\MSDev98\Template** directory.
3. Launch a new instance of Visual C++ and go to **File → New**, you should see the new tool kit listed.

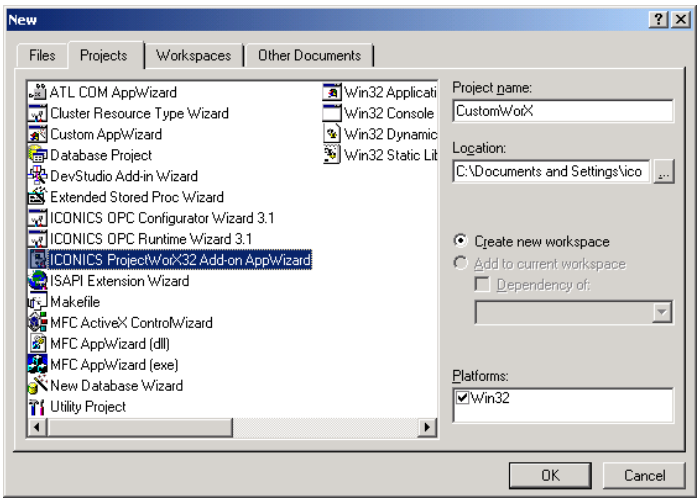

Figure 1 - New ProjectWorX32 Add-on AppWizard Project

4. Select **ICONICS ProjectWorX32 Add-on AppWizard** and give the project a name and a location. Click on **OK** when you are done.

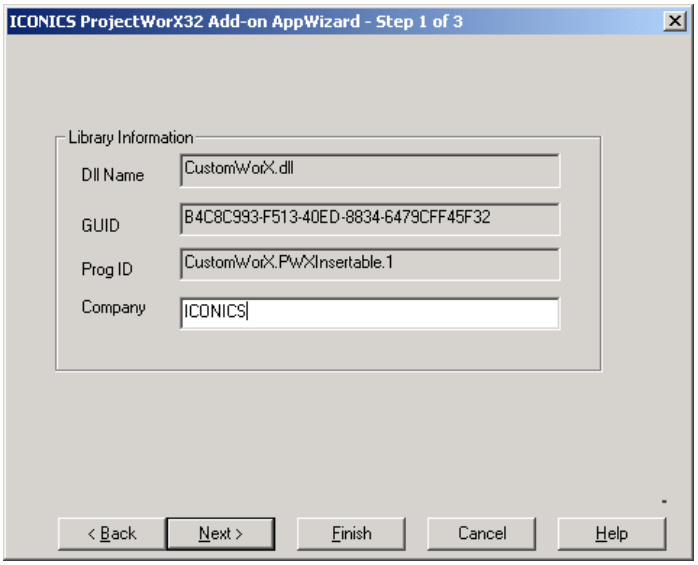

Figure 2 - Creating .dll File

5. In the ProjectWorX32 Add-on AppWizard, enter the company name for the application you wish to add. We will create an imaginary application in this example called CustomWorX32. Click on the **Next** button.
6. Enter a project name as you wish it to appear in the tree-control in ProjectWorX32 and add as many sub-trees as you wish by entering a name for the sub-tree in the **Child Product Name** field and clicking on the **Add** button.

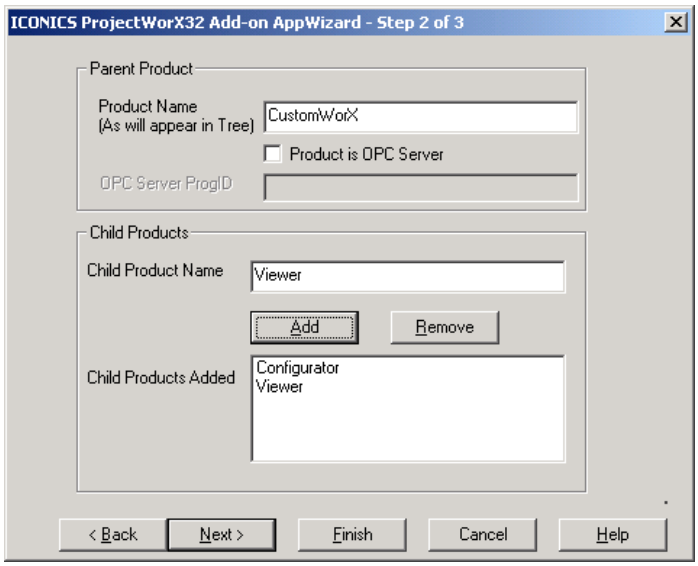

Figure 3 - Configuring ProjectWorX32 Tree-Control

NOTE: If this is an OPC Server, you should also check the **Product is OPC Server** option. You must enter the OPC Server ProgID in this case.

7. Click on **Next** when you are done.
8. You will configure each of the child products next. In the properties of the sub-tree, there are four menu items you can configure. Choose which menus to include and the caption for the item. Below is a description of each of the menu options.

   Add New Document – If have checked this option, then the user can create a new document by right-clicking on the item in the sub-tree. If this option is not enabled at this time, then the user will be responsible, in the future, for supplying code in the .dll file that will programmatically handle the creation of a new document.

   Template File – This is only available if **Add New Document** is enabled. You can specify a template file, which is essentially an empty document for the custom application, residing in the Genesis32\bin\PWXTemplates directory. By enabling this option, ProjectWorX32 will look for the specified file in that location and make a copy of it every time a new document is requested.

   Add Existing Document – Allows the user to add existing files to the project.

   Make Main Document – Allows the user to make a configuration file active similar to that of AlarmWorx32 and TrendWorX32 configuration files.

9. ProjectWorX32 has the ability to interface with Microsoft Visual Source Safe and put files under version control. This ability can be extended to your custom application by selecting the **Enable Source Control Menus** button.

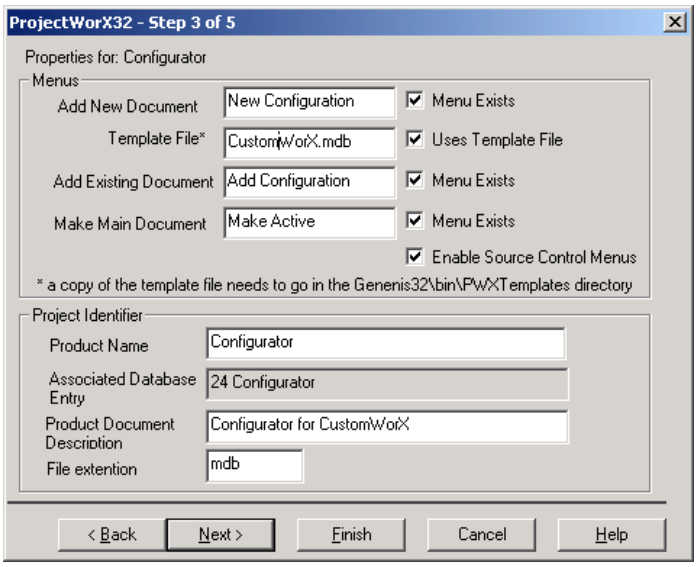

Figure 4 - Sub-tree Properties

10. In the **Product name** field, enter the name of the sub-tree as you would want to see it in ProjectWorX32. You will also need to enter a description for the product and a file extension associated with it.
11. Click on **Next** when you are done. Repeat steps 9-11 for each sub-tree you are creating.
12. Click on **Finish** when you are done configuring.
13. The wizard will generate a new work space for the project. You should watch the screen and acknowledge any message dialogs that come up.
14. Build the project in your new work space.
15. Launch ProjectWorX32 and you should see your new application as part of the tree-control.

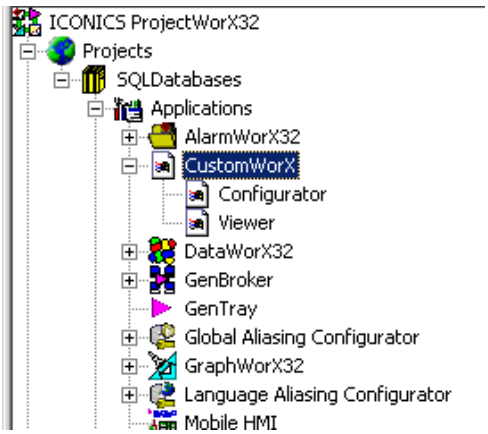

Figure 5 - ProjectWorX32 Tree with CustomWorX