ProjectWorX32 – Deploying Applications

APPLICATIONS NOTE

October 2013

Description: Guide to packing and unpacking projects.
OS Requirement: Win 2000, XP Pro, Server 2003, Vista, Server 2008, Windows 7
General Requirement: You have a ProjectWorX32 project on the system.

Introduction

The pack/unpack projects feature of the ProjectWorX32 console allows you to pack up an entire project (with all included application files) into a single, compressed file. This is especially useful for system administrators who want to configure a project and then send or copy the project to several different computers (i.e. nodes on a network). The packed file can then be unpacked and ran on each computer.

Packing a Project

1. Launch ProjectWorX32 by going to **Start → Programs → ICOINCS Tools → ProjectWorX32**
2. Right-click on the project that you want to pack and select **Pack Project**.

   NOTE: If you need to create a project, please refer to the application note entitled *ProjectWorX32 – ProjectWorX32 Quick Start* or *Workbench32 – Quick Start*.

3. Choose the location where you would like the save the file. Once you click on the **Save** button, the packing will start.
4. When it finishes packing the project, you will get a confirmation dialog. Click on **OK**.
5. The log file will open showing you if there is any errors or warnings for packing the project.

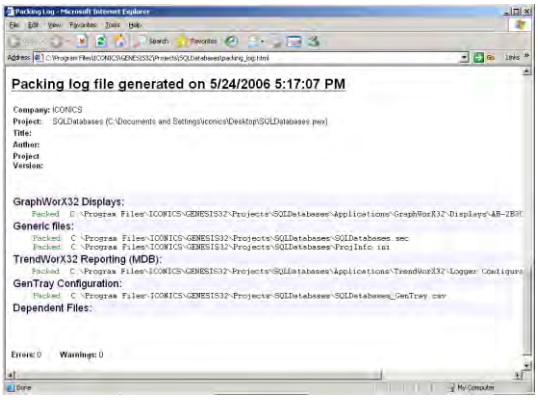

Figure 1 - Pack Project Log

6. You should now have a .pwx file in the location where you specified to save the packed project. You can take this file to any GENESIS32 machine and unpack the project.

Unpacking a Project

1. On the machine where you would like to unpack your project, launch ProjectWorX32.
2. Right-click on **Projects → Unpack Project**

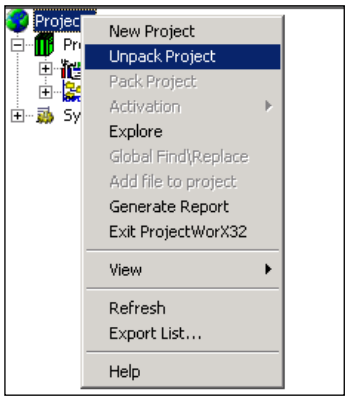

Figure 2 - Unpack Project

3. Select the location of your pack project (.pwx) file and click on **Open**.
4. If the path to the Projects directory on this machine is different from the Projects directory on the original machine, a message window will come up asking if you would like to modify file paths to match the new directory location for the unpacked project. Select **Yes** to change it and **No** to keep the original directory paths.
5. Once the project completes the unpacking process, a confirmation will appear, click on **OK.**
6. A log file, similar to Figure 1 will launch showing errors or warnings for the unpacking.
7. You can now resume editing and running your project as normal.

Incremental Unpack of Existing Projects

ProjectWorX32 includes a feature that allows you to overwrite an existing project by unpacking a newer version of that project. For example, if a system administrator makes changes to a project and needs to deploy an updated version of the project to multiple users on different network nodes, users on each node can unpack and automatically update the project.

When you unpack an updated version of a packaged project (a project with the same name) in the ProjectWorX32 Console, ProjectWorX32 scans each file in the updated project. ProjectWorX32 automatically unpacks any new files in the updated project (i.e. files that have been added to the project since the last version).

Copyright 2013 ICONICS, Inc.

For packaged files that already exist (files with the same name) inside the project, ProjectWorX32 also checks the date/time of the files. If the date/time of a packaged file is the same as the date/time of the existing file of the same name, nothing happens. If a packaged file is newer (more recent date/time stamp) than the existing file of the same name, ProjectWorX32 automatically unpacks the newer file and overwrites the existing file. If a packaged file is older than the existing file of the same name, ProjectWorX32 warns you of this fact and gives you the option to overwrite the file.

> **NOTE:** Projects containing Microsoft SQL Server databases will not be updated. When unpacking SQL-based configuration files, the files will not be overwritten even if the packed project contains newer files of the same name.

SQL Configuration Databases in a Project

ProjectWorX32 makes it possible to import configuration databases that reside inside Microsoft SQL Server databases. Several GENESIS32 and BizViz applications, such as MobileHMI, AlarmWorX32 Multimedia, and ReportWorX, use SQL configuration databases. Universal Data Link (.udl) files are used to connect to the Microsoft SQL Server database. When you unpack a project that contains a SQL Server configuration database, ProjectWorX32 checks to see whether SQL Server is present on the local machine. If it detects the SQL Server, it tries to connect to the server via integrated security.

If SQL Server is not present on the local machine, then ProjectWorX32 ask you if you want to scan the network for an instance of SQL Server, as shown in the figure below. Click **Yes** to search for a SQL Server.

In the SQL Server Login dialog box, select a server from the drop-down list and enter your login ID and password (if applicable). Then click **OK**.

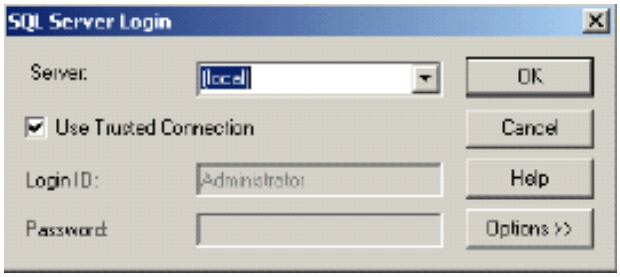

Figure 3 - SQL Server

ProjectWorX32 checks to see if any existing databases have the same name as the database being unpacked. If so, you are prompted with a dialog that shows you the databases that already exists as shown in Figure 4. To continue unpacking the project, you can follow the Database Creation Wizard for each of the checked databases, or simply uncheck the ones you do not wish to create and they will be skipped.

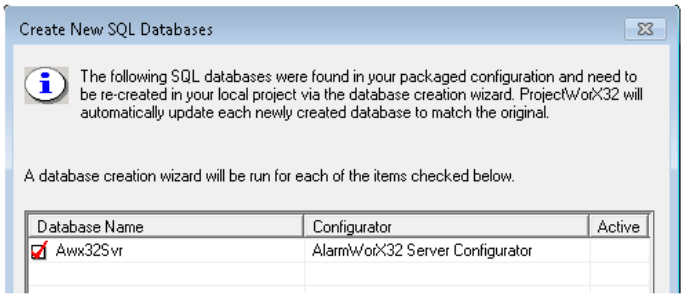

Figure 4 – Database already exists with same name

Log Files

When projects are packed or unpacked, a packing log or an unpacking log is created in the project folder, as shown in the figure below. These files contain useful information about what has been packed\unpacked, including any errors and warnings issued during the process.

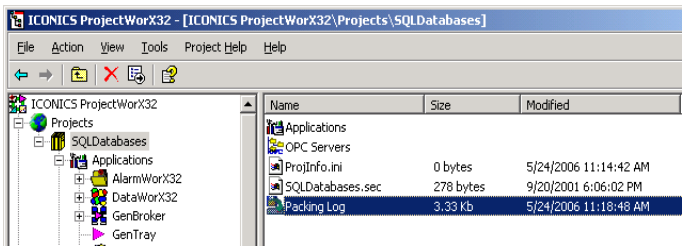

Figure 5 - Log File

Any referenced files in a project are also packed/unpacked. When unpacking the project, all the referenced files (which are listed with a warning in the packing log file) are placed in the same location as they were initially. ProjectWorX32 tries to recreate the locations necessary for referenced external files (i.e. files that are located in a directory outside the project folder). Referenced files that are located in a subfolder within the project folder are unpacked into the subfolder.