WiX Reusable Installer Template

Work to do:

Document “how to use” completely.

Check into Git

Add a choose components window in MSI UI?

Add a choose components window in Bundle UI?

How to use:

BASIC SETUP

1. Make a clone of the WRIT GitRepo. (sample project)
2. Place the cloned repo (WixInstaller folder) into your project folder.
3. Copy BuildBase.bat, BuildPatch.bat, CommonVersion.cs, certpass.txt and Sample.build into the root project folder.
4. Rename Sample.build to <ProjectName>.build. Edit BuildBase.bat and BuildPatch.bat to reference the new <ProjectName>.build filename.
5. Edit the <ProjectName>.build file as follows:
   1. Change <project name="Sample" default="BuildCompleteRelease"> to reference the project name instead of “Sample”
   2. Change <property name="ProductName" value="Sample" /> to reference your project name instead of “Sample”.
   3. Edit the current release version number. The version number will always have four values, but do not edit the fourth one – it must be a “1”. The number is defined in parts as follows: <property name="VersionSeg1" value="1" /> <property name="VersionSeg2" value="0" /> <property name="VersionSeg3" value="0" />
   4. Edit the Copyright Year and Manufacturer as needed.
   5. Edit the CFG and Platform variables to match you build configuration.
   6. Edit the CertPath to reflect the relative path from the “WixInstaller\BaseInstallerBuild\ folder to your code signing certificate file.
   7. Generate a GUID for each new base release installer that is built. Note that each base installer must increment the third part of the version number. It is up to you to define the new number (VersionSeg3) and generate a product id GUID for it. Patch updates will automatically increment the fourth number.
   8. If you are making a first build of a product that will be upgraded in the future, then generate a new UpgradeCodeGuid and a new CompGGS GUID. Do not use the GUIDS that come by default in this file.
   9. Check the path for property MSBUILD to ensure that it is pointing at a valid compiler.
   10. Check the value for “Installers.Base.Dir” – this is where the build files will be located relative to the location of this file.
   11. You may need to extensively change the <target name=”Clean”> in order to make the clean target work correctly with your project.
   12. Under the <target name=”CopyFilesToInstallation1”> target you may also need to change this section so that it copies your compiled files from the proper output directory, using the proper CFG and Platform properties.
   13. The <target name=”UpdateVersion”> target will need to set the correct version number in your AssemblyInfo file.
   14. Save your changes.
6. Put your licensing information in the file WixInstaller/BaseInstallerBuild/TemplateLicense.htm
7. In the WiXInstaller\resources folder there are several graphics files used by the installer. Customize these as you wish but do not change the names or dimensions of the images.
8. Open a command prompt and run the BuildBase.bat file to build your installer. The installer will be created in the BuildDir folder if all goes well.

Note: One of John Hatton’s big issues was that inevitably projects need to “downgrade” a file in the installation. Perhaps a new version of a dll was included, but it is unstable. How does the product installer move backwards to a more stable version? The current installer project under “Sample” can build a base installer that removes the “KeyPath” attributes from harvested files during the heat.exe process. This allows a later base installer to overwrite the higher versioned file with a lower versioned file. Patch files do NOT have this capability. The “KeyPath” hack results in an error during the patch creation process.