WiX v4 is here! Let us help. (/services/conversion-offer/?utm_source=firegiant.com&utm_medium=Display&
utm_campaign=conversion_offer&utm_term=let+us+help&utm_content=announcement_bar)

FIREGIANT (/)

Products (/products/)   Pricing (/services/)   Documentation (/docs/)   Blog (/blog/)

Contact Us (/contact-us/)

Patchwork

It wouldn't be terribly effective to create an upgrade
installation package with megabytes of files in it just
because one or two small files inside have to be renewed.
In cases like this, patching has always been a better
solution. Patches basically contain the difference between
the old and the new versions and can automagically turn
the old file on the user's computer into the new one. Patch

packages can also contain new files to be deployed.

The WiX toolset can also create patch installation packages (.msp files). These are created from two normal installation packages: the original one with the old, erroneous files and a new one with the fixed files. In the downloadable SamplePatch (/system/files/samples /SamplePatch.zip) we have two very simple installation packages, without user interface. Both install a single file that will change between the original and patched versions. The details of the source files must be completely familiar by now. Our Error.wxs and Fixed.wxs source files will only differ in their `Source` file references:

```
<File Id='FoobarEXE' Name='FoobarAppl10.exe' DiskId='1' Source='Error\FoobarAppl10.exe' KeyPath='yes' />
```

versus

```
<File Id='FoobarEXE' Name='FoobarAppl10.exe' DiskId='1' Source='Fixed\FoobarAppl10.exe' KeyPath='yes' />
```

The patch will be created from a third source file. It is, just like the earlier ones, an XML file but the contents are different from the files we've created so far:

About FireGiant

Now you can install with confidence. FireGiant arms you with accurate support from the creators of WiX.

Because there are no questions we can't answer or problems we can't solve, you can unleash the full functionality of WiX without a doubt.

Get Supported › (/services/support-programs

📓 HeatWave (/docs/heatwave/)

📓 HeatWave Build Tools (/docs/heatwave-build-tools/)

📓 FG-WiX v3 Additional Features (/docs/fg-wix/)

📓 WiX Toolset v4 Tutorial (/docs /wix/tutorial/)

📓 WiX Toolset v3 Tutorial (/docs/wix/v3 /tutorial/)

▸ Getting Started (/docs/wix/v3/tutorial /getting-started/)

▸ User Interface (/docs/wix/v3/tutorial /user-interface/)

```xml
<?xml version='1.0' encoding='windows-1252'?>
<Wix xmlns='http://schemas.microsoft.com/wix/2006/wi'>
    <Patch AllowRemoval='yes' Manufacturer='Acme Ltd.' MoreInfoURL='www.acmefoobar.com'
        DisplayName='Foobar 1.0.1 Patch' Description='Small Update Patch' Classification='Update'>

        <Media Id='5000' Cabinet='Sample.cab'>
            <PatchBaseline Id='Sample' />
        </Media>

        <PatchFamily Id='SamplePatchFamily' Version='1.0.0.0' Supersede='yes'>
            <ComponentRef Id='MainExecutable' />
        </PatchFamily>

    </Patch>
</Wix>
```

The `Classification` attribute can be *Hotfix*, *Security Rollup*, *Critical Update*, *Update*, *Service Pack*, or *Update Rollup*. `AllowRemoval` controls whether the user can later uninstall the patch or not. A `PatchFamily` tag encompasses the items to be patched. `Supersede` controls whether this actual patch supersedes all previous patches in this same family.

Building will be a little bit more complicated than our previous projects. First, we build the two base packages the usual way. Each goes into its own folder:

```
candle.exe Error.wxs
light.exe -out Error\Product.msi Error.wixobj

candle.exe Fixed.wxs
light.exe -out Fixed\Product.msi Fixed.wixobj
```

Then we use another WiX tool, Torch, to create a transform between the two installation packages. The command line arguments instruct the program to use WiX's own formats, .wixpdb and .wixmst rather than those of Windows Installer (.msi and .mst).

```
torch.exe -p -xi Error\Product.wixpdb Fixed\Product.wixpdb -out Patch.wixmst
```

We also have to build our patch package using the usual WiX compiler and linker but this time, the output format will be different from the usual one, .wixmsp. There is no need to tell the linker to produce this kind of file, it will determine it automatically based on the contents of the source file.

```
candle.exe Patch.wxs
light.exe Patch.wixobj
```

And finally, we build the actual Windows Installer patch package from the result of the previous step and the transform we created a while ago. Pyro, the WiX tool responsible for patch creation, needs not only the name

of the transform file but the corresponding `PatchBaseline/@Id` attribute as well on the command line:

```
pyro.exe Patch.wixmsp -out Patch.msp -t Sample P
atch.wixmst
```

The Patch.msp will be the patch installer actually distributed. To test it, first install the original package (Error/Product.msi), then apply the patch:

```
msiexec /p Patch.msp
```

... and check that the file has indeed been changed to the new version. Then, go into the Programs and Features Control Panel tool, choose *Show Updates*, and remove first the patch (the changed file will revert back to the original one) and then the program itself.

CONTACT US (/contact-us/) | BLOG (/blog/)

PRODUCTS (/products/)

> HeatWave (/docs/heatwave/)

> HeatWave Build Tools (/products/heatwave-build-tools/)

> FireGiant WiX v3 (/products/fg-wix/)

SERVICES (/services/)

> Support Programs (/services/support-programs/)

> Code Review (/services/code-review/)

> Phone Consultation (/services/phone-consultation/)

> Custom Development (/services/custom-development/)

DOCUMENTATION (/docs/)

> HeatWave (/docs/heatwave/)

> WiX v4 Tutorial (/docs/wix/tutorial/)

> WiX v3 Tutorial (/docs/wix/v3/tutorial/)

> FireGiant WiX v3 (/docs/fg-wix/)

© 2023 FIREGIANT (/)

> About Us (/about-firegiant/)

> Why WiX (/why-wix/)

> Leadership Team (/about-firegiant/leadership-team/)

> Privacy Policy (/about-firegiant/privacy-policy/)