

WiX v4 is here! [Let us help. \(/services/conversion-offer/?utm\\_source=firegiant.com&utm\\_medium=Display&utm\\_campaign=conversion\\_offer&utm\\_term=let+us+help&utm\\_content=announcement\\_bar\)](/services/conversion-offer/?utm_source=firegiant.com&utm_medium=Display&utm_campaign=conversion_offer&utm_term=let+us+help&utm_content=announcement_bar)

[Products \(/products/\)](/products/)[Pricing \(/services/\)](/services/)[Documentation \(/docs/\)](/docs/)[Blog \(/blog/\)](/blog/)[Contact Us \(/contact-us/\)](/contact-us/)[Home \(/\)](#) / [WiX Toolset v3 Tutorial \(/docs/wix/v3/tutorial/\)](/docs/wix/v3/tutorial/)/ [Upgrades and Modularization \(/docs/wix/v3/tutorial/upgrades-and-modularization/\)](/docs/wix/v3/tutorial/upgrades-and-modularization/) / [Checking for Oldies](#)

# Checking for Oldies

The first task when doing any kind of update or upgrade is to make sure we have the previous version we want to change.

Windows Installer relies on the `UpgradeCode` attribute of the `Product` tag to do that. Therefore, it is very important to always include an `UpgradeCode`, even if you don't plan your current program to be upgraded later. You never

know and once it's out there, you can't provide it any more.

Keep the same `UpgradeCode` GUID as long as you want the products to be upgraded by the same upgrade version. In the usual case, this would mean one code for all 1.x versions, another one for the 2.x versions, etc:

```
<?xml version='1.0' encoding='windows-1252'?>
<Wix xmlns='http://schemas.microsoft.com/wix/2006/wi'>
  <Product Name='Foobar 1.0' Id='PUT-GUID-1-HERE' UpgradeCode='PUT-GUID-2-HERE'
    Language='1033' Codepage='1252' Version='1.0.0' Manufacturer='Acme Ltd.'>
    <Package Description="Acme's Foobar 1.0 Installer" Compressed='yes' />

    ...
  </Product>
</Wix>
```

Our current [SampleUpgrade \(/system/files/samples/SampleUpgrade.zip\)](/system/files/samples/SampleUpgrade.zip) now consists of two installation packages. Both are based on the simple UI installer, `SampleWixUI`.

The second version of `SampleUpgrade` is meant to replace one of the deployed files with a newer version. As we will now consider this a *minor upgrade*, we change the `Version`. Cosmetic changes to the human readable `Name`


## 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 \(/docs/heatwave/\)](/docs/heatwave/)

 [HeatWave Build Tools \(/docs/heatwave-build-tools/\)](/docs/heatwave-build-tools/)

 [FG-WiX v3 Additional Features \(/docs/fg-wix/\)](/docs/fg-wix/)

 [WiX Toolset v4 Tutorial \(/docs/wix/tutorial/\)](/docs/wix/tutorial/)

 [WiX Toolset v3 Tutorial \(/docs/wix/v3/tutorial/\)](/docs/wix/v3/tutorial/)

▶ [Getting Started \(/docs/wix/v3/tutorial/getting-started/\)](/docs/wix/v3/tutorial/getting-started/)

▶ [User Interface \(/docs/wix/v3/tutorial/user-interface/\)](/docs/wix/v3/tutorial/user-interface/)

and `Description` are obvious:

```
<Product Name='Foobar 1.0.1' Id='PUT-GUID-1-HERE'
  UpgradeCode='PUT-GUID-2-HERE'
    Version='1.0.1' Manufacturer='Acme Ltd.' Language='1033' Codepage='1252' >
  <Package Id='*' Keywords='Installer' Description
    ="Acme's Foobar 1.0.1 Updater" ... >
```

We also need a description of what product versions we plan to replace with this upgrade. The `Id` attribute of the `Upgrade` tag refers back to the `UpgradeCode` GUID of the original installation package (the older `SampleUpgrade` in our case). The internal `UpgradeVersion` tag describes the details of the versions to be updated. `OnlyDetect` tells the installer not to remove the previous product (remember, we're doing a minor upgrade, we keep the old version of the product in place but replace one file. If we were doing a major upgrade, we would probably *remove* 1.0.0 and *install* 1.1.0 instead).

`Minimum` and `Maximum` specify the range of versions we are supposed to update with this upgrade.

`IncludeMaximum` and `IncludeMinimum` specify whether the bound value is actually included in the range or not ( `IncludeMaximum=yes` meaning to find versions *less than or equal to* the version specified in `Maximum` while `IncludeMaximum=no` only finds those *less than* the `Maximum` ). Although these attributes have their default values, we don't rely on those here and always include

► [Events and Actions \(/docs/wix/v3/tutorial/events-and-actions/\)](/docs/wix/v3/tutorial/events-and-actions/)

▼ [Upgrades and Modularization \(/docs/wix/v3/tutorial/upgrades-and-modularization/\)](/docs/wix/v3/tutorial/upgrades-and-modularization/)

📄 [Checking for Oldies \(/docs/wix/v3/tutorial/upgrades-and-modularization/checking-for-oldies/\)](/docs/wix/v3/tutorial/upgrades-and-modularization/checking-for-oldies/)

📄 [Replacing Ourselves \(/docs/wix/v3/tutorial/upgrades-and-modularization/replacing-ourselves/\)](/docs/wix/v3/tutorial/upgrades-and-modularization/replacing-ourselves/)

📄 [Patchwork \(/docs/wix/v3/tutorial/upgrades-and-modularization/patchwork/\)](/docs/wix/v3/tutorial/upgrades-and-modularization/patchwork/)

📄 [Fragments \(/docs/wix/v3/tutorial/upgrades-and-modularization/fragments/\)](/docs/wix/v3/tutorial/upgrades-and-modularization/fragments/)

📄 [Mergers \(/docs/wix/v3/tutorial/upgrades-and-modularization/mergers/\)](/docs/wix/v3/tutorial/upgrades-and-modularization/mergers/)

► [Net and .NET \(/docs/wix/v3/tutorial/net-and-net/\)](/docs/wix/v3/tutorial/net-and-net/)

► [COM, Expression Syntax, Miscellanea \(/docs/wix/v3/tutorial/com-expression-](/docs/wix/v3/tutorial/com-expression-)

them---it's better to spell them out for the sake of clarity and self-documentation.

```
<Upgrade Id='PUT-GUID-2-HERE'>
  <UpgradeVersion OnlyDetect='yes' Property='SELFFOUND'
    Minimum='1.0.1' IncludeMinimum='yes'
    Maximum='1.0.1' IncludeMaximum='yes' />
  <UpgradeVersion OnlyDetect='yes' Property='NEWERFOUND'
    Minimum='1.0.1' IncludeMinimum='no' />
</Upgrade>
```

Using an `Upgrade` tag in our source file triggers the inclusion of a new standard action,

`FindRelatedProducts`. It is scheduled to run before `AppSearch`, if any. If necessary, we can re-schedule it inside the `InstallExecuteSequence` tag.

`FindRelatedProducts` works its way through our `Upgrade` tag and checks for each version listed there. If one is found, its `Product` GUID will be appended to the property specified in the `UpgradeVersion` `Property` attribute (`SELFFOUND` and `NEWERFOUND` in our example). Needless to say, the Installer can only find products that have been installed using an `.msi` package with `UpgradeCode` specified---so now you know why it's so important to always specify one.

If you develop localized software packages, you can also specify a `Language` attribute in both `UpgradeVersion` and

[syntax-miscellanea/](#)

► [SQL \(/docs/wix/v3/tutorial/sql/\)](/docs/wix/v3/tutorial/sql/)

► [User Interface Revisited \(/docs/wix/v3/tutorial/user-interface-revisited/\)](/docs/wix/v3/tutorial/user-interface-revisited/)

► [Transforms \(/docs/wix/v3/tutorial/transforms/\)](/docs/wix/v3/tutorial/transforms/)

► [Standard Libraries \(/docs/wix/v3/tutorial/standard-libraries/\)](/docs/wix/v3/tutorial/standard-libraries/)

► [Localization \(/docs/wix/v3/tutorial/localization/\)](/docs/wix/v3/tutorial/localization/)

Product tags. Use the usual Windows decimal language identifier to specify the language. If specified, FindRelatedProducts will only locate products with matching languages.

After the check has run, we can take the appropriate actions based on the existence and value of the properties involved:

```
<CustomAction Id='AlreadyUpdated' Error='Foobar
1.0 has already been updated to 1.0.1 or newer.'
/>
<CustomAction Id='NoDowngrade' Error='A later ve
rsion of [ProductName] is already installed.' />

<InstallExecuteSequence>
  <Custom Action='AlreadyUpdated' After='FindR
elatedProducts'>SELFFOUND</Custom>
  <Custom Action='NoDowngrade' After='FindRela
tedProducts'>NEWERFOUND</Custom>
</InstallExecuteSequence>
```

For some strange reason, small updates and minor upgrades cannot be run simply by clicking on the .msi file. They give the error: Another version of this product is already installed.

We know, stupid. Anyway, you have to start it with the command:

```
msiexec /i SampleUpgrade.msi REINSTALL=ALL REINST  
TALLMODE=vomus
```

Don't ask me how this would fare with the average user. You'd better start it from an Autorun.inf file or devise an outer Setup.exe shell to launch it.

As you can see, this upgrade version will behave properly in both directions. It will replace the earlier package but it will also handle the case where the program might be upgraded even further in the future (eg, 1.0.2 and later). It will not downgrade a possible later version back to 1.0.1. To be absolutely correct and foolproof, we have to plan that early. Even the very first installer we ship should already have this safety lock built in:

```
<Upgrade Id='PUT-GUID-2-HERE'>  
  <UpgradeVersion OnlyDetect='yes' Property='NE  
WERFOUND' Minimum='1.0.0' IncludeMinimum='no' /  
>  
</Upgrade>  
  
<CustomAction Id='NoDowngrade' Error='A later ve  
rsion of [ProductName] is already installed.' />  
  
<InstallExecuteSequence>  
  <Custom Action='NoDowngrade' After='FindRela  
tedProducts'>NEWERFOUND</Custom>  
</InstallExecuteSequence>
```

[◀ Upgrades and Modularization \(/docs/wix/v3/tutorial/upgrades-and-modularization/\)](/docs/wix/v3/tutorial/upgrades-and-modularization/)  
[up \(/docs/wix/v3/tutorial/upgrades-and-modularization/\)](/docs/wix/v3/tutorial/upgrades-and-modularization/)  
[Replacing Ourselves ▶ \(/docs/wix/v3/tutorial/upgrades-and-modularization/replacing-ourselves/\)](/docs/wix/v3/tutorial/upgrades-and-modularization/replacing-ourselves/)

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



## **PRODUCTS (/products/)**

- › [HeatWave \(/docs/heatwave/\)](/docs/heatwave/)
- › [HeatWave Build Tools \(/products/heatwave-build-tools/\)](/products/heatwave-build-tools/)
- › [FireGiant WiX v3 \(/products/fg-wix/\)](/products/fg-wix/)

## **SERVICES (/services/)**

- › [Support Programs \(/services/support-programs/\)](/services/support-programs/)
- › [Code Review \(/services/code-review/\)](/services/code-review/)
- › [Phone Consultation \(/services/phone-consultation/\)](/services/phone-consultation/)
- › [Custom Development \(/services/custom-development/\)](/services/custom-development/)

## **DOCUMENTATION (/docs/)**

- › [HeatWave \(/docs/heatwave/\)](/docs/heatwave/)
- › [WiX v4 Tutorial \(/docs/wix/tutorial/\)](/docs/wix/tutorial/)
- › [WiX v3 Tutorial \(/docs/wix/v3/tutorial/\)](/docs/wix/v3/tutorial/)
- › [FireGiant WiX v3 \(/docs/fg-wix/\)](/docs/fg-wix/)

**© 2023 FIREGIANT (/)**

- › [About Us \(/about-firegiant/\)](/about-firegiant/)
- › [Why WiX \(/why-wix/\)](/why-wix/)
- › [Leadership Team \(/about-firegiant/leadership-team/\)](/about-firegiant/leadership-team/)
- › [Privacy Policy \(/about-firegiant/privacy-policy/\)](/about-firegiant/privacy-policy/)