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/)

 Home (/) / WiX Toolset v3 Tutorial (/docs/wix/v3/tutorial/)
 / COM, Expression Syntax, Miscellanea (/docs/wix/v3/tutorial/com-expression-syntax-miscellanea/) / XML

XML

More and more programs use XML-based configuration files these days instead of the older .ini style. WiX contains prefabricated custom actions to modify such files during installation and uninstallation. Let's suppose that we install a Settings.xml file together with our application. Originally, the file only contains the outermost tags:

```
<settings>
</settings>
```

We want the installer to add new nodes, one of those with attribute values:

```
<settings>
  <add key="a_key" value="a_value">key_item
    <inside>inside_item</inside>
  </add>
</settings>
```

To achieve this, we can use the `XmlFile` tag. Apart from its usual attributes like `Id` and `File,` its `Action, Name` and `Value` attributes will determine what to do in the XML file while the `ElementPath` specifies where to do it. This last attribute uses the standard XPath specification language (http://www.w3.org/TR/xpath) to describe the XML node the actual operation will be applied to.

- Action=createElement a new element named `Name` will be created as a child of the one specified in `ElementPath`
- Action=setValue If `Name` is omitted, the inner text of the element specified in `ElementPath` will be set to `Value`. If both `Name` and `Value` are specified, the node specified in `ElementPath` will receive an attribute named `Name` with a value of `Value`.
- Action=deleteValue If both `Name` and `Value` are omitted, the inner text of the element specified in `ElementPath` will

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/)

be deleted. If `Name` is specified, the attribute named `Name` will be deleted from the node specified in ElementPath.

So, to carry out the modifications we planned, we need the following entries (note that we have to provide our own sequencing, this is important to ensure that the changes will be removed in the proper reverse order upon uninstallation):

```
<Component Id='Settings' Guid='YOURGUID-574D-4A9
A-A266-5B5EC2C022A4'>
  <File Id='XmlSettings' Name='settings.xml' Dis
kId='1' Source='settings.xml' Vital='yes' />
  <util:XmlFile Id='XmlSettings1' File='[INSTALL
DIR]settings.xml'
    Action='createElement' Name='add' ElementPat
h='//settings' Sequence='1' />
  <util:XmlFile Id='XmlSettings2' File='[INSTALL
DIR]settings.xml'
    Action='setValue' Name='key' Value='a_key' E
lementPath='//settings/add' Sequence='2' />
  <util:XmlFile Id='XmlSettings3' File='[INSTALL
DIR]settings.xml'
    Action='setValue' Name='value' Value='a_valu
e' ElementPath='//settings/add' Sequence='3' />
  <util:XmlFile Id='XmlSettings4' File='[INSTALL
DIR]settings.xml'
    Action='setValue' Value='key_item' ElementPa
th='//settings/add' Sequence='4' />
  <util:XmlFile Id='XmlSettings5' File='[INSTALL
DIR]settings.xml'
    Action='createElement' Name='inside' Element
Path='//settings/add' Sequence='5' />
  <util:XmlFile Id='XmlSettings6' File='[INSTALL
DIR]settings.xml'
    Action='setValue' Value='inside_item' Elemen
tPath='//settings/add/inside' Sequence='6' />
</Component>
```

If there are several nodes of the same name in the XML file, we can use the usual XPath format of `'node[@attr="value"]'` to specify which one we refer to. As `ElementPath` expects a formatted string, we have to

escape the brackets with backslashes (`[\[]` and `[\]]`) to keep them from being evaluated as properties.

As this functionality resides in the standard utilities module, we have to link against that:

```
light.exe -ext WixUtilExtension -out Sample.msi
Sample.wixobj
```

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

✉ 🐦 📡 in 🐙 ▶

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/)