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/)
/ Events and Actions (/docs/wix/v3/tutorial/events-and-actions/) / Control Your Controls

Control Your Controls

Granted, what we've just done was not elegant. We should check right when the user enters the key, display a warning and give the user a chance to re-enter the key instead of aborting the installation in a later phase. Let's see how we can accomplish this.

We can link custom actions to two kinds of user interface controls, push buttons and check boxes. We can use the `Publish` tag we already know to accomplish this. The `Value` attribute carries the name of the custom action:

```
<Control Id="..." Type="PushButton" ...>
    <Publish Event="DoAction" Value="CheckingPI
D">1</Publish>
</Control>
```

This will cause the custom action to call our DLL when the user presses the Next button of the User Information page. The custom action will be linked to this UI event, thus we no longer need to schedule it in the `InstallExecuteSequence` tag. However, the custom action definition remains in the source:

```
<CustomAction Id='CheckingPID' BinaryKey='CheckP
ID' DllEntry='CheckPID' />
```

To warn the user, we need a message box. This is just another dialog, similar to our previously created User Information page. We can put it into a separate source file as a fragment and refer to it using a `DialogRef` tag, as earlier. Just to illustrate that there is another solution as well, we now specify it directly in our main source file, right inside the `UI` section.

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/)

```
<Dialog Id="InvalidPidDlg" Width="260" Height="8
5" Title="[ProductName] Setup" NoMinimize="yes">
    <Control Id="Icon" Type="Icon" X="15" Y="15"
Width="24" Height="24" ToolTip="Information ico
n" FixedSize="yes" IconSize="32" Text="Exclam.ic
o" />
    <Control Id="Return" Type="PushButton" X="10
0" Y="57" Width="56" Height="17" Default="yes" C
ancel="yes" Text="&amp;Return">
        <Publish Event="EndDialog" Value="Retur
n">1</Publish>
    </Control>
    <Control Id="Text" Type="Text" X="48" Y="15"
Width="194" Height="30" TabSkip="no">
        <Text>
            The user key you entered is invalid.
Please, enter the key printed on the label
            of the jewel case of the installatio
n CD.
        </Text>
    </Control>
</Dialog>
```

We also have to update the User Information page because we have to call our custom action and the new message box from this dialog:

```
<Control Id="Next" Type="PushButton" X="236" Y="
243" Width="56" Height="17" Default="yes" Text
="[ButtonText_Next]">
    <Publish Event="DoAction" Value="CheckingPI
D">1</Publish>
    <Publish Event="SpawnDialog" Value="InvalidP
idDlg">PIDACCEPTED = "0"</Publish>
    <Publish Event="NewDialog" Value="SetupTypeD
lg">PIDACCEPTED = "1"</Publish>
</Control>
```

Now, when the user presses the Next button, the function in the DLL is called (every time because the `1` condition evaluates to *true*). The function in our DLL will check the `PIDKEY` property and set `PIDACCEPTED` to signal whether the key was accepted. If it was, we go on to `SetupTypeDlg`. If it wasn't, we display our error message.

There is only one small item left, as we mentioned an icon in the message box, we also have to provide it in the installer:

```
<Binary Id="Exclam.ico" SourceFile="Exclam.ico"
/>
```

The full source can be downloaded as SampleAskKey (/system/files/samples/SampleAskKey.zip).

By the way, it's not necessarily a nice and safe thing to have the user key appear in the log file. To avoid this, use

the `Hidden` attribute:

```
<Property Id="PIDKEY" Hidden='yes' />
```

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/)