

Install components and IDE extensions into the Delphi IDE

In this tip I draw together in one place information on how to install components into the Delphi IDE for various Win32 versions of Delphi. The notes also cover how to install IDE extensions or make simple units available in design time packages. The tip assumes you are installing source code, not pre-compiled units or packages.

Before you start copy all the files needed to install the component, IDE extension or unit into some suitable location on your hard disk. Files should include `.pas` files and any associated `.dfm`, `.res`, `.dofm` and, for components, `.dcr` files.

You then proceed differently for different versions of Delphi:

- ▶ *Installation with Delphi 2005 and later*
- ▶ *Installation with Delphi 3 to 7*
- ▶ *Installation with Delphi 2*

Installation with Delphi 2005 and later

To add a component or IDE extension to the IDE it must be included in a design package. You can either create a new design package or add the component to an existing one. These instructions assume you will be installing into an existing package. See the Delphi help for information on how to create a new package. If your component came with its own package, use that.

1. Open the required package. Select *File | Open* menu and choose a suitable package source file (`.dpk` extension). You may decide to install into the "user" package that is often supplied with Delphi.
2. The selected package (with `.bpl` extension) will now be displayed in the Project Manager. If the package has not been installed yet right click the `.bpl` entry in the Project Manager and select *Install*. Dismiss any resulting message dialogue.
3. Now right click the `.bpl` entry again and select *Add*. What happens next depends on the version of Delphi.
 - ▶ For Delphi 2009 and earlier the "Add" dialogue box will be displayed. Click the dialogue's *Browse* button then select the `.pas` file(s) to be installed from the resulting file open dialogue. The full path to the file(s) should now be displayed in the "Add" dialogue's *Unit file name* edit box. Click *OK* to add the file(s) to the package.
 - ▶ For Delphi 2010 and later a simple file dialogue box will open from where you can choose the required `.pas` file(s). Just press *OK* to add the file(s) to the package.
4. Back in the Project Manager, check that the `.pas` file(s) you added appear under the package's "Contains" node. Also check that any associated `.dcr` files also appear. If any `.dcr` file does not appear you need to display the package's source file and add the line `{ $R FileName.dcr }` to the code between the package name and the *requires* section. To display the source right click the package name in the project manager and select *View Source*.
5. If you are installing an IDE enhancement, like a property editor, make sure the `designide` package is included in the package's *requires* section.
6. Now right click the `.bpl` entry again and select *Compile*. If no errors are displayed the package has been recompiled correctly.
7. When adding components the package may need to be (re)installed. Right click the `.bpl` entry again and select *Install*. If all goes well a message should be displayed indicating that the component has been registered and it will appear in the appropriate tab of the Tool Palette (which will be named "DelphiDabbler" when installing one of the components from the *DelphiDabbler Code Library*).

Should a message appear saying that the installation has failed you will need to un-install the package and try again. How this is done depends on the version of Delphi.

- ▶ In Delphi 2009 and earlier select the *Components | Install Packages* menu option then select the required package in the resulting dialogue box. Press the *Remove* button followed by *OK*. The package will now be uninstalled.
- ▶ In Delphi 2010 and later the `.bpl` entry's context menu has an *Uninstall* option. Simply click that option to uninstall the package.

8. Close the package project using the *File | Close* menu item. Accept any prompts to save the package project.

9. If you wish you can now delete the `.pas` file. Other files must be left in place.

Building for 64 bit Windows (Delphi XE2 and later)

If the components etc. are compatible with 64 bit Windows add to a package as described above. Now add a Windows 64 bit target to the package project and rebuild.

Installation with Delphi 3 to 7

To add a component or IDE extension to the IDE it must be included in a design package. You can either create a new design package or add the component to an existing one. These instructions assume you will be installing into an existing package. See the Delphi help for information on how to create a new package. If your component came with its own package, use that.

1. Open the required package. Select *File | Open* menu, make sure that files with the `*.dpk` extension are being displayed, and choose a suitable package source file. You may decide to install into the "user" package. This is found in the `Lib` sub-directory of your Delphi installation and is named `dclusr.dpk` or `dclusrXX.dpk` where `XX` is a version number.
2. The package editor will now be displayed. If its *Install* button is not disabled then click it to ensure the package is installed into Delphi. Dismiss any resulting message dialogue.
3. Now click the package editor's *Add* button. The "Add" dialogue box will be displayed. Click the dialogue's *Browse* button then select the required `.pas` file(s) from the resulting file open dialogue. The full path to the file should now be displayed in the "Add" dialogue's *Unit file name* edit box. Click *OK* to add the file(s) to the package.
4. Click the package editor's *Compile* button. The package will now be recompiled and, if no errors are displayed, the package has been recompiled correctly. When adding components a message may be displayed at this stage indicating that the component has been registered. In some cases it is necessary to click the *Install* button after compiling before the component will be registered and the message displayed. Any components will now appear appropriate tab of the Tool Palette (which will be named "DelphiDabbler" when installing one of the components from the *DelphiDabbler Code Library*).
5. Now close the package editor window. You will probably be prompted to save the package. Agree to this and the package source file will be updated with information about the new component.
6. If you wish you can now delete the `.pas` file. Other files must be left in place.

Installation with Delphi 2

1. Select the *Component | Install* menu option to display the "Install Components" dialogue box.
2. Click the *Add* button to display the "Add Module" dialogue box.
3. Click the *Browse* button and choose the required `.pas` file from the resulting standard file open dialogue box and click *Open*.
4. This returns you to the "Install Components" dialogue box. Click *OK* here to rebuild the component library. If you were installing component it should now appear on the component palette (which will be named "DelphiDabbler" when installing one of the components from the *DelphiDabbler Code Library*).

5. If you wish you can now delete the `.pas` file. Other files must be left in place.

Author:	Peter Johnson
Added:	2009-10-06
Last updated:	2010-02-22

Copyright © Peter Johnson (*DelphiDabbler*) 2002-2018