

Contents

- [Installation](#)
- [Upgrade from PostSharp 2.0](#)
- [Additional notes on PostSharp 3.0](#)

PostSharp 3.0 is automatically installed by the [DevForce Code First NuGet Package](#) as of version 7.2.

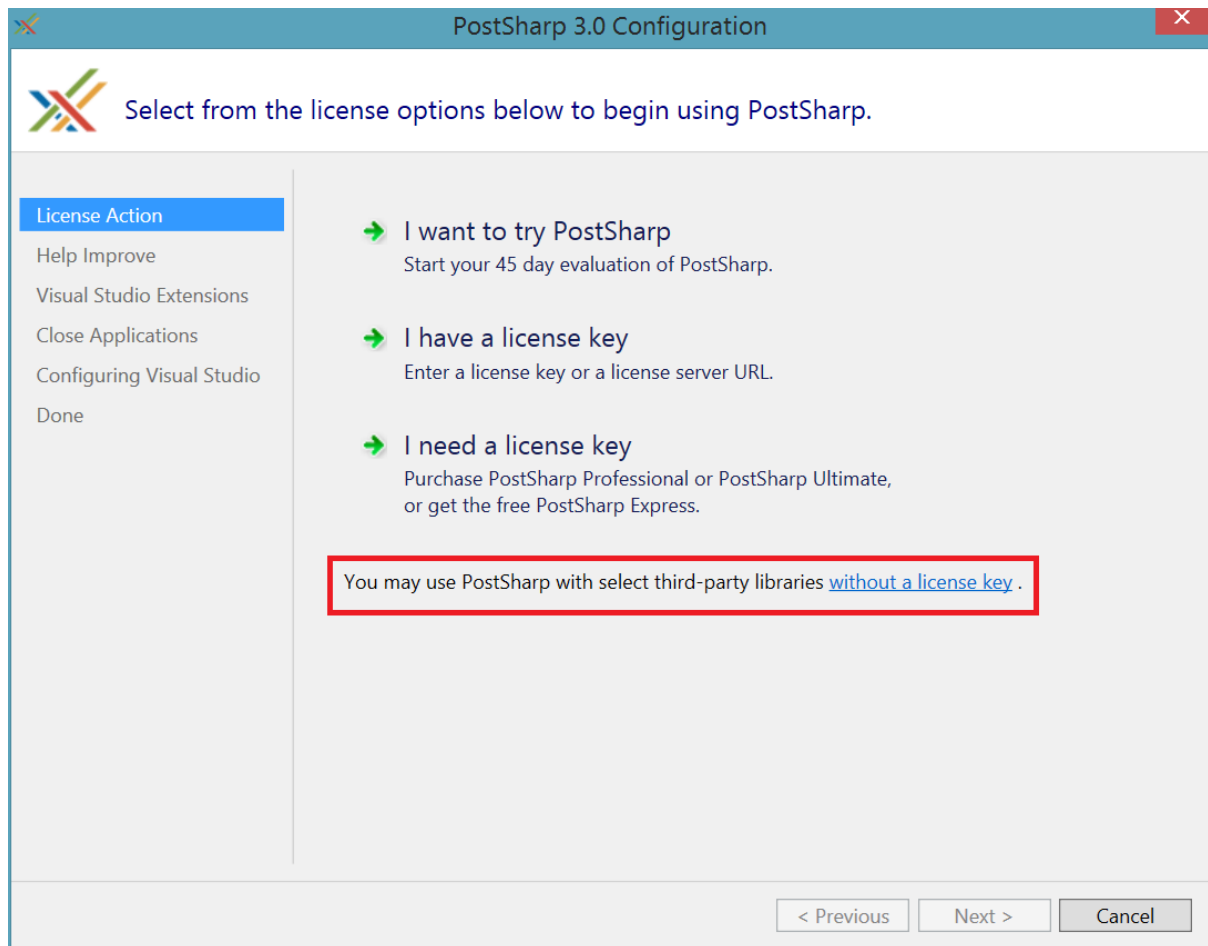
Installation

The first time PostSharp 3.0 is installed you'll see several "PostSharp 3.0 Configuration" windows. Here's some information on how to proceed.

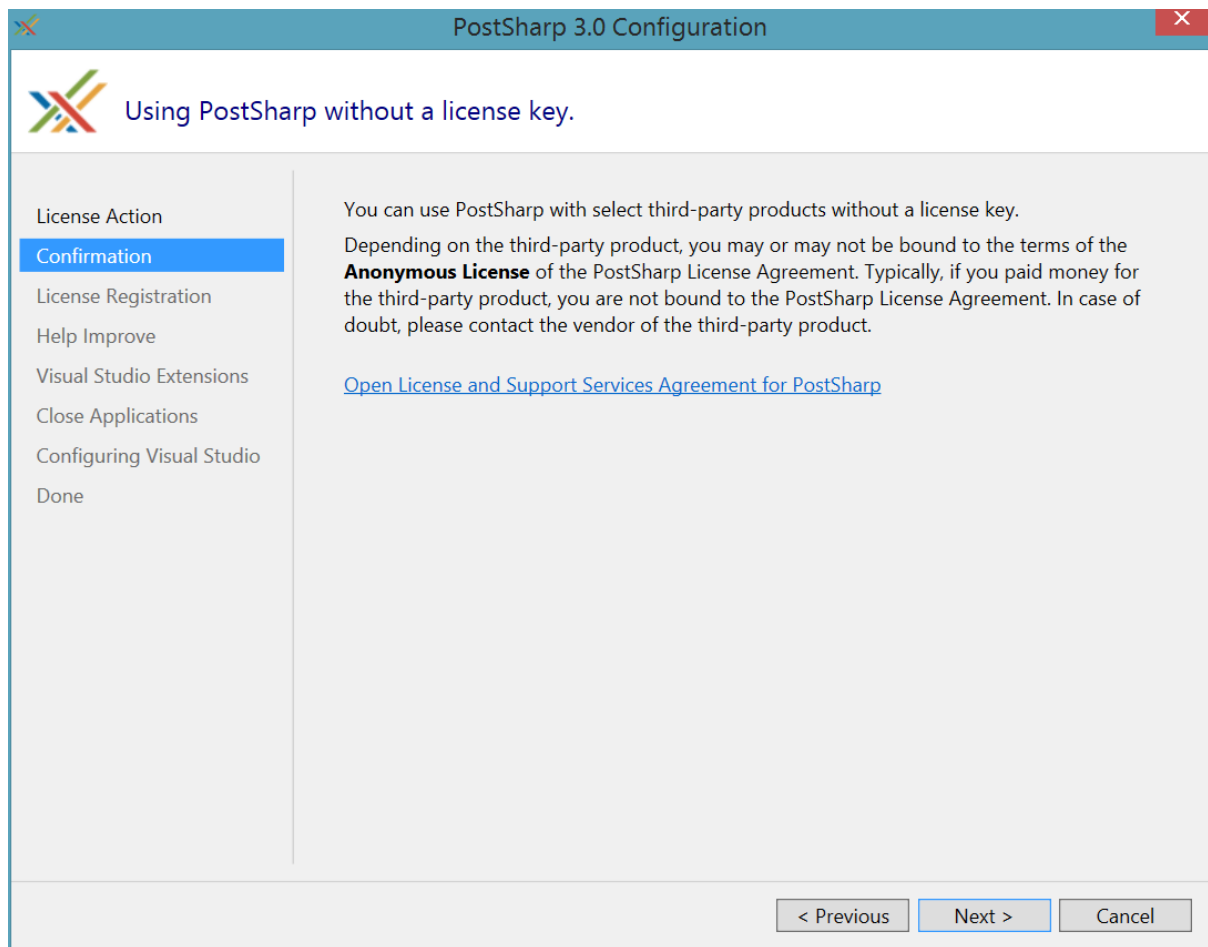
1. License options

Don't worry, you don't need a license key or a 45 day evaluation. DevForce is one of the "select third party libraries" and you do not need a license key to use PostSharp with DevForce.

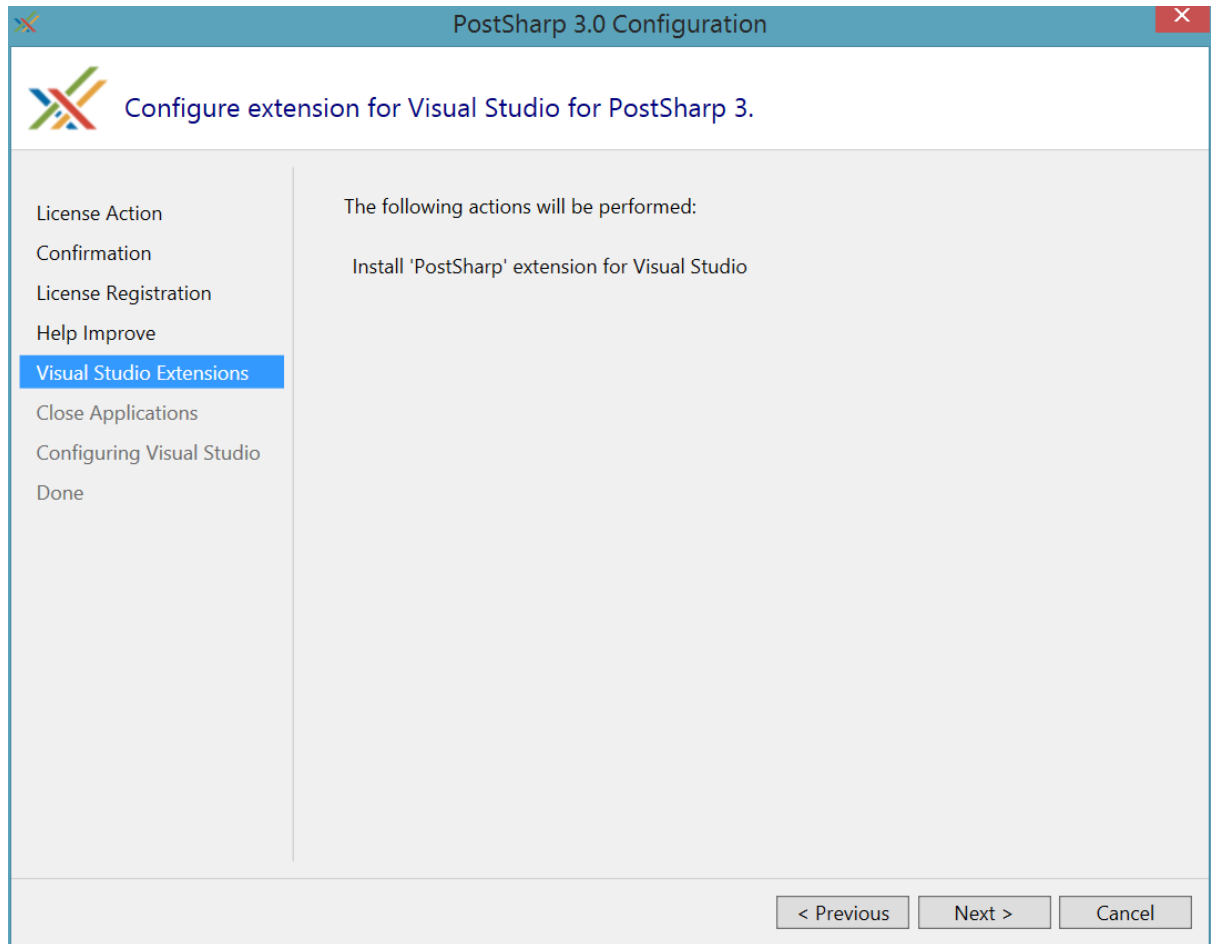
Click the link **without a license key** to continue. Note that if you cancel the prompt you'll see it again the next time you open Visual Studio.



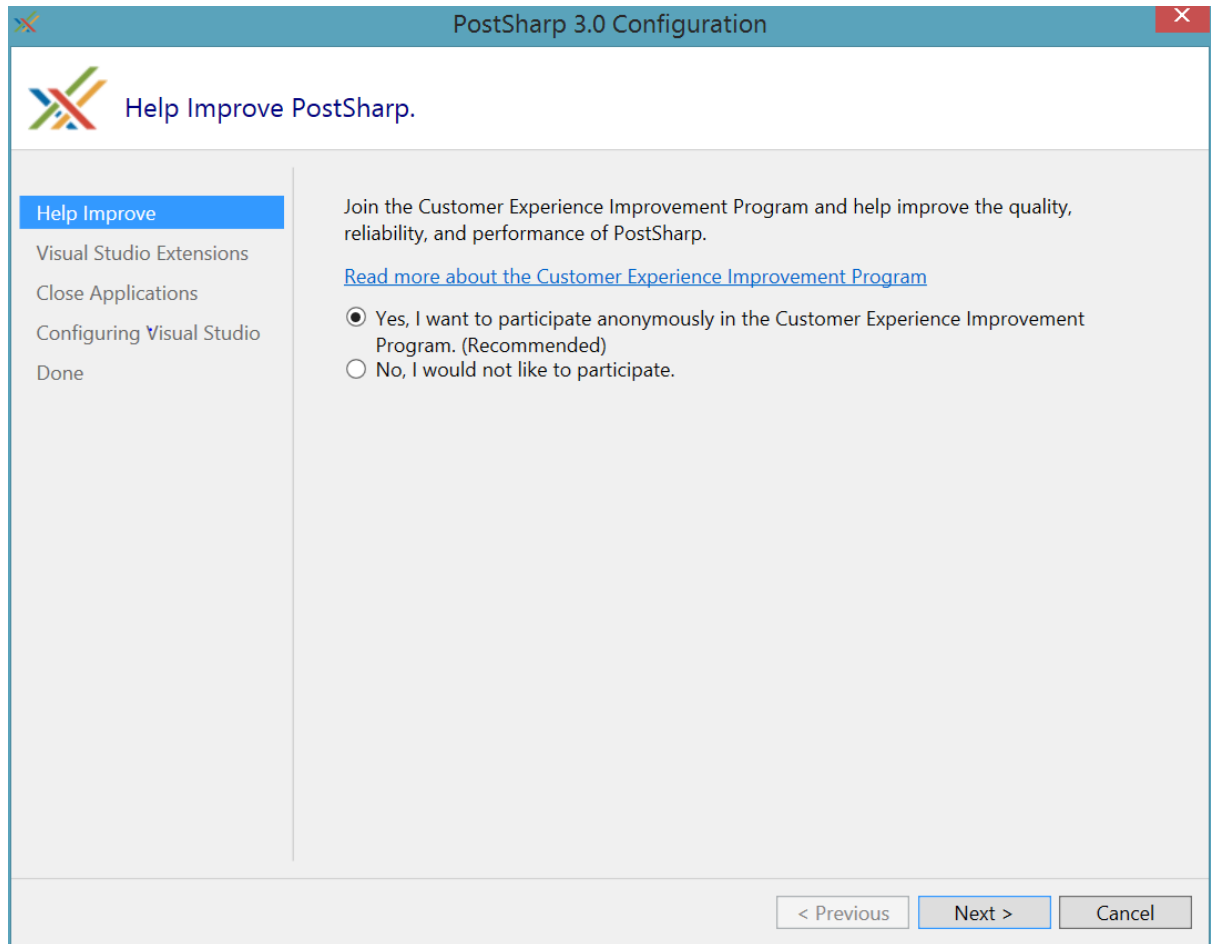
You'll then see a confirmation that you're continuing without a license key. Click **next**.



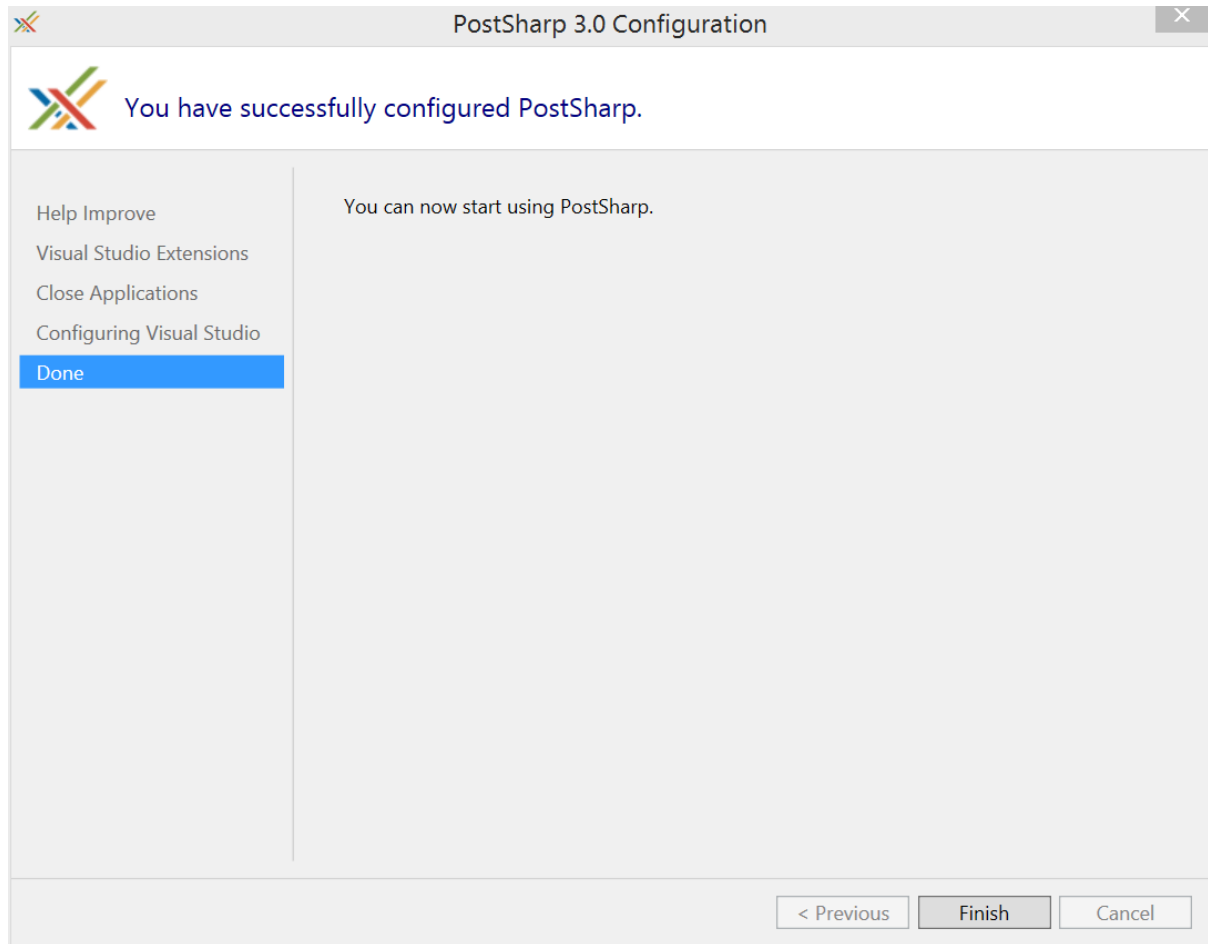
2. The PostSharp VS Extension is required. Click **next**. When prompted, you will need to close Visual Studio and restart once the VSIX is installed.



3. Help improve PostSharp. You may see this popup during installation or upon re-opening Visual Studio.



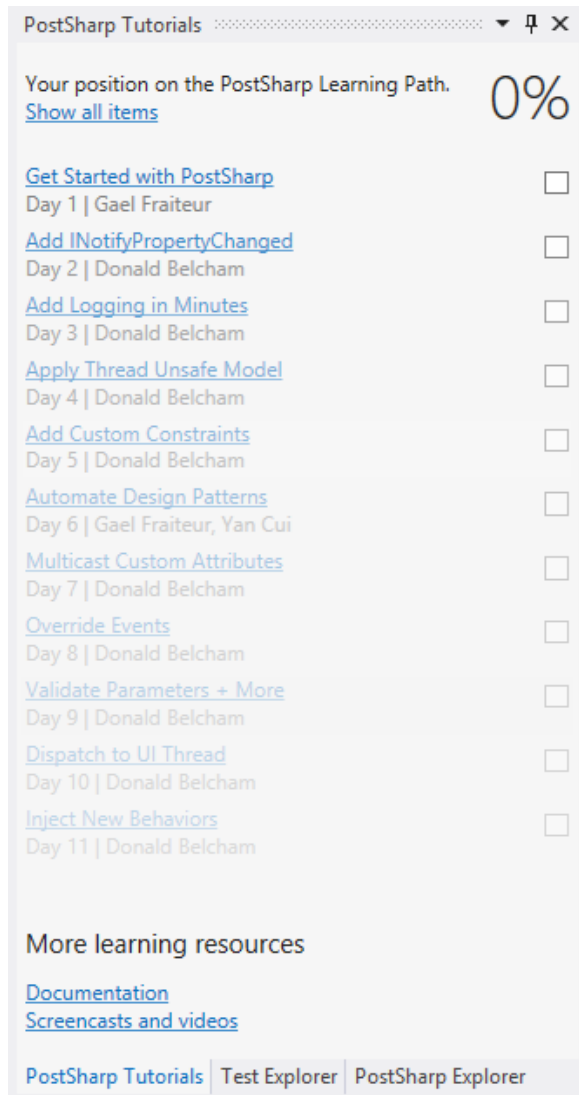
4. Done! You can now start using DevForce Code First.



Once installed, you should not see PostSharp configuration prompts again when using Visual Studio.

5. PostSharp Tutorials

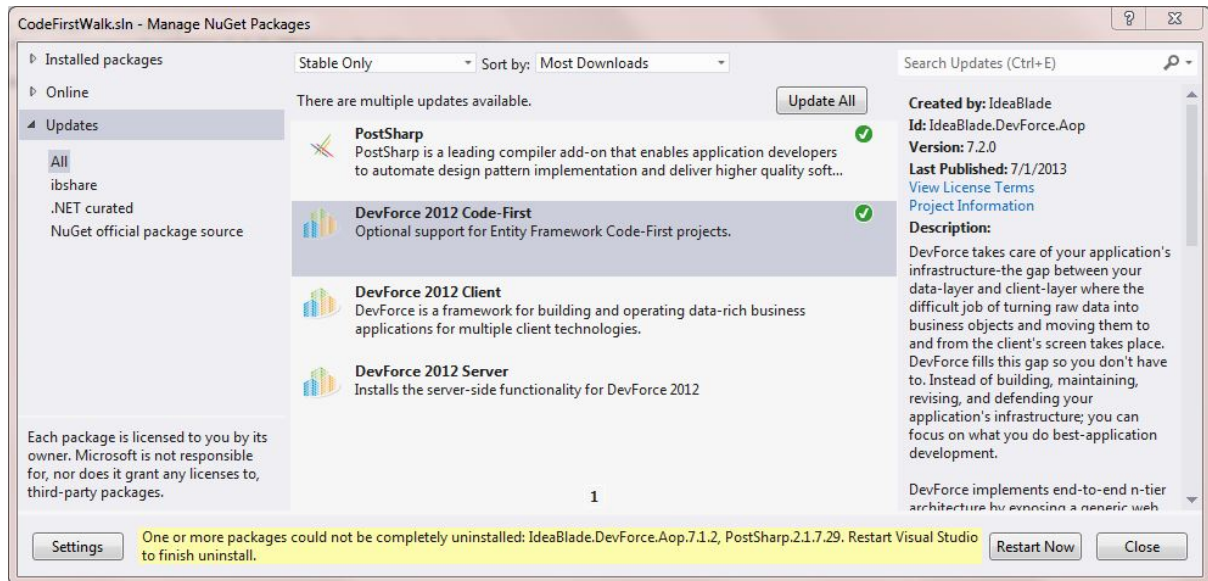
There's no need to go through the tutorials to use DevForce Code First. If you wish to use PostSharp in your own application development we encourage you to explore PostSharp. You will need your own license for use outside of DevForce applications. Please see the [PostSharp web site](#) for more information.



Upgrade from PostSharp 2.0

Prior to version 7.2.0, DevForce installed PostSharp 2.0 as a dependency for its Code First support. When you upgrade a solution using an earlier version to DevForce 7.2.0 or later, PostSharp 2.0 will be uninstalled.

You'll receive a prompt to restart Visual Studio after the NuGet packages are updated.



Additional notes on PostSharp 3.0

1. Assembly versioning

In PostSharp 2.0, the assembly version of the PostSharp.dll (or PostSharp.SL.dll) was always 2.1.0.0, regardless of the file version or NuGet package version. This meant that applications could use a different PostSharp version than what DevForce had been compiled for.

This is no longer true in PostSharp 3.0.

In PostSharp 3.0 the assembly version now corresponds to the NuGet package version. Except for .NET 4.5 projects, which allow for assembly binding redirection, your projects must now use the same PostSharp version as DevForce.

For example, DevForce 7.2.0 requires the PostSharp 3.0.31 NuGet package, and is compiled with that version. If a later version of PostSharp becomes available, you will not be able to upgrade to it without also upgrading to a later DevForce release supporting that package.

2. If the PostSharp NuGet package is not installed to a project but that project references the PostSharp.dll (for example, a WPF project which references a Code First model project), the project reference should be updated:

- For .NET 4.5 projects, be sure to reference the *PostSharp.dll* in the *packages\PostSharp.3.0.xx\lib\net20* folder.
- All other project types - Silverlight, Windows Store and Windows Phone - should reference the *PostSharp.dll* found in the *packages\PostSharp.3.0.xx\lib\portable-windows8+net40+sl4+wp7* folder.
- Note that the PostSharp.SL.dll used in version 2.0 no longer exists.

3. See [here](#) for more information on configuring and deploying PostSharp.