

Code Upgrade Tool

Kentico 12 to Xperience 13

Introduction

The **Xperience Code upgrade tool** provides a way to analyze projects and helps convert code that uses the Xperience API to version Xperience 13.

The tool has the following functionality:

- Detects custom code that is no longer valid in Xperience 13
- Provides recommendations for each occurrence of invalid code
- Automatic conversion of most code to the Xperience 13 API

If you encounter any problems, contact us at support@kentico.com.

Tip: Check the [DevNet portal](#) for the latest version of the code upgrade tool.

Prerequisites

The machine where you plan to run the code upgrade tool must have the following installed:

- [Microsoft .NET Framework 4.8](#) or newer
- Visual Studio 2019 with [Build Tools for Visual Studio 2019](#) (including *Web development build tools*)

The solution that you wish to analyze must compile without errors.

Analyzing Kentico 12 projects

You can run the code upgrade tool for any ASP.NET project that uses the Kentico API in its code. This includes:

- Kentico administration projects (i.e. projects that provide the administration interface for a separate MVC live site)
- Projects that integrate the Kentico API through NuGet packages (for example MVC live site projects)

Limitation: The code upgrade tool does not process Razor view files in MVC projects (files with the `.cshtml` extension). You need to update the code of your views manually after upgrading the Kentico NuGet packages for the MVC project.

Basic detection

Run **CodeUpgrade.exe** from the **command line**, with the path to your project's solution file as the parameter (for example `WebApp.sln` for Kentico projects).

For example:

```
CodeUpgrade.exe C:\inetpub\wwwroot\Kentico12\WebApp.sln
```

The tool generates a **csv** file containing a list of custom code occurrences in your project that are no longer valid in Xperience 13. The information will help you update your custom code after you perform the upgrade.

Each listed occurrence in the csv file has one of the following statuses:

- **Detected** - an occurrence of invalid code that the tool cannot resolve automatically. See the *Instructions* value of the occurrence for more information.
- **CanBeResolved** - a detected code occurrence that the tool can resolve automatically if you run it with the `-resolve` parameter (see the *Automatic conversion* section).
- **Resolved** - invalid code that the tool automatically resolved in the specified output folder. Only used when running the tool with the `-resolve` parameter.

Automatic conversion

You can also use the tool to provide automatic resolutions for most types of code expressions. Add the **-resolve** parameter when running *CodeUpgrade.exe*.

For example:

```
CodeUpgrade.exe -resolve C:\inetpub\wwwroot\Kentico12\WebApp.sln
```

If the tool detects code issues that can be automatically resolved, it creates copies of the given files under the **Converted** folder in the tool directory. All resolvable issues in the files are updated to be compatible with the Xperience 13 API.

Optional parameters

You can also add the following parameters when running the tool:

- **-vstoolspath<path>** - sets the path to the Visual Studio build tools required to analyze the solution. You need to set the path if your build tools are installed in a different directory than the default (*C:\Program Files (x86)\Microsoft Visual Studio\2019\BuildTools\MSBuild\Microsoft\VisualStudio\v16.0*).
- **-output=results.csv** - sets the name of the csv file containing the tool's results.
- **-verbose=0** - determines how detailed the tool's output is in the console (enter values from 0 to 3). Does not affect the output file.

For more details about the available parameters, run the tool without any parameters.

Limiting the scope of the analysis

For customized Xperience files, the tool reports all detected issues, including the default Kentico 12 code. You can avoid reporting of these false positives by enclosing your custom code into **#region** directives with a specific name. For ASPX markup in the administration project, use the following syntax: `<%-- #region Name --%> ... <%-- #endregion --%>`

To analyze only the code placed inside the regions, run the tool with an additional parameter:

- **-customregion=<regionName>**

Implementing the API changes

Use the output of the tool to update your custom code AFTER you upgrade the project to Xperience 13. See [Upgrading to Xperience 13](#) for more information.

The csv output file lists all occurrences of invalid custom code, including:

- The path of the affected file and line number
- The exact source code of the expression
- A set of instructions on how to resolve the issue

If you generated automatic conversions for your code, use the following approaches to insert the new code into your upgraded project:

- **Fully custom files** - directly replace the files using the converted files created by the tool.
- **Customized Kentico files** - use a code comparison tool and merge the Xperience 13 files with the matching files in the *Converted* output folder.