

**Note**

Hotfixes are not tested for all possible scenarios, so you may experience issues after applying the hotfix.

Always **back up your project files and database** before applying a hotfix.

Table of Contents

- [Prerequisites](#)
- [Installing the hotfix](#)
- [Applying the hotfix to web site projects](#)
 - [Local installation](#)
 - [Remote installation](#)
- [Applying the hotfix to web application projects](#)
 - [Local installation](#)
 - [Remote installation](#)
- [Applying the hotfix to Microsoft Azure](#)
- [Applying the hotfix to MVC projects](#)
- [After applying the hotfix](#)
- [Updating setup files and external utilities](#)
- [Additional notes and workarounds](#)
 - [11.0.3](#)
 - [11.0.5](#)
 - [11.0.8](#)
 - [11.0.9](#)
 - [11.0.10](#)
 - [11.0.15](#)
 - [11.0.19](#)
 - [11.0.23](#)
 - [11.0.25](#)
 - [11.0.27](#)
 - [11.0.29](#)
 - [11.0.35](#)
 - [11.0.39](#)
 - [11.0.42](#)
 - [11.0.47](#)
- [Fixed bugs](#)

Prerequisites

You can only apply the hotfix to complete Kentico projects that use the standard folder structure (the solution file, *GlobalAssemblyInfo.cs*, the *CMS* and *Lib* sub-folders). If you run your website as a deployment of the CMS folder, you need to hotfix the original complete project and then create a new deployment.

Installing the hotfix

1. Download the installer for the required hotfix version from the [Kentico DevNet portal](#).
2. Run the **Hotfix_<version>.exe** file.
3. Go through the hotfix installation procedure.

By default, the installer places the hotfix into the **C:\Program Files\Kentico\<version>\Hotfix<version>** folder.

Applying the hotfix to web site projects

Local installation

1. If your system stores virtual objects on the file system (due to enabled [Deployment mode](#) or source control options), return the files to the database. You can re-enable deployment mode or source control after you apply the hotfix.
2. Run **Kentico Hotfix Utility (Hotfix.exe)** from the location where you installed the hotfix (*C:\Program Files\Kentico\<version>\Hotfix<version>* folder by default).
3. Select your project folder using the **Browse** button or type in the path to the project manually.
4. Use the utility to back up your project files and/or database if you have not already done so.
5. Select which components you want to update. Click **Switch to advanced mode** if you wish to change the settings. We recommend updating all components.
 - **Kentico files** – the files of your web project
 - **Setup files** – the Kentico installer and external utilities (which are separate from the web project)
 - **SQL script** – the database structure and data
6. Choose a method of taking the project offline and back online.
7. Click **Next** to start the hotfix procedure.
8. After the update finishes, click **Next** to view any problems that may have occurred and the instructions to solve them.

For more information, see [Kentico Hotfix and Upgrade Utility](#).

Remote installation

If you cannot access your web project directly (for example when the project is located on a remote hosting server), you can use the Hotfix Utility to create the updated file structure and then upload them to your web project location.

1. Navigate to the location where you installed the hotfix (*C:\Program Files\Kentico\<version>\Hotfix<version>* folder by default).
2. Run **Hotfix.exe** from the command line with the **/deploy** parameter. Set the parameter value based on the installation type you are using:
 - **net46site** – web site project using .NET version 4.6
 - **net47site** – web site project using .NET version 4.7

Example: *Hotfix.exe /deploy=net46site*

This launches the Hotfix Utility, which then creates the files and folders necessary for updating a project running on .NET version 4.6 and saves them into the folder chosen in the first step of the Hotfix Utility wizard.

Additionally, you can provide the **/path** parameter to specify the location where the application creates the updated files. This way you do not need to set the path in the first step of the Hotfix Wizard. You can supply the **/path** parameter in two different formats – **relative**, which represents a folder in the location from which the utility was executed; or **absolute**, which can be any path to a local disk.

Example: *Hotfix.exe /deploy=net46site /path=Deploy*

This launches the Hotfix Utility, which then creates a folder named *Deploy* in the directory from which the utility was executed and copies all files and folders necessary for updating a project running on .NET version 4.6 into that folder.

3. Once you have the update files ready, take your website offline. You can do this by copying the **App_Offline.htm** file from the hotfix installation directory into the root of the remote web project.
4. Upload the update files created by the Hotfix Utility into your remote web project's **CMS** folder (overwrite the original files).



Important

If you have previously modified some of the Kentico project files (such as web parts or administration interface files), **DO NOT OVERWRITE** these files. You could lose your modifications.

You need to compare the modified files with the new versions and make appropriate changes manually.

5. Manually execute the hotfix SQL scripts on your Kentico database. The script files are located in the **SQL** folder of your hotfix installation directory.
 - If you do not use separated databases – run **Hotfix_separated.sql** first and then **Hotfix_default.sql**.
 - If you use [separated databases](#) – run **Hotfix_separated.sql** on the separated database and **Hotfix_default.sql** on your main database.
 - Note: The script files may be empty if the given hotfix does not require any SQL scripts for the related tables (skip the file in these cases).
6. Bring your website back online (delete the **App_Offline.htm** file).

Applying the hotfix to web application projects

Local installation

1. If your system stores virtual objects on the file system (due to enabled [Deployment mode](#) or source control options), return the files to the database. After you apply the hotfix, re-enable deployment mode or source control and rebuild the project.
2. Run **Kentico Hotfix Utility (Hotfix.exe)** from the location where you installed the hotfix (*C:\Program Files\Kentico\<version>\Hotfix<version>* folder by default).
3. Select your project folder using the **Browse** button or type in the path to the project manually.
4. Use the utility to back up your project files and/or database if you have not already done so.
5. Select which components you want to update. Click **Switch to advanced mode** if you wish to change the settings. We recommend updating all components.
 - **Kentico files** – the files of your web project
 - **Setup files** – the Kentico installer and external utilities (which are separate from the web project)
 - **SQL script** – the database structure and data
6. Choose a method of taking the project offline and back online.
7. Click **Next** to start the hotfix procedure.
8. After the update finishes, click **Next** to view any problems that may have occurred and the instructions to solve them.
9. Open the project solution in Visual Studio and build the solution.

For more information, see [Kentico Hotfix and Upgrade Utility](#).

Remote installation

If you cannot access your web project directly (for example when the project is located on a remote hosting server), you can use the Hotfix Utility to create the updated file structure and then upload them to your web project location.

1. Navigate to the location where you installed the hotfix (*C:\Program Files\Kentico\<version>\Hotfix<version>* folder by default).
2. Run **Hotfix.exe** from the command line with the **/deploy** parameter. Set the parameter value based on the installation type you are using:
 - **net46app** – web site project using .NET version 4.6
 - **net47app** – web site project using .NET version 4.7

Example: `Hotfix.exe /deploy=net46app`

This launches the Hotfix Utility, which then creates the files and folders necessary for updating a project running on .NET version 4.6 and saves them into the folder chosen in the first step of the Hotfix Utility wizard.

Additionally, you can provide the **/path** parameter to specify the location where the application creates the updated files. This way you do not need to set the path in the first step of the Hotfix Wizard. You can supply the **/path** parameter in two different formats – **relative**, which represents a folder in the location from which the utility was executed; or **absolute**, which can be any path to a local disk.

Example: `Hotfix.exe /deploy=net46app /path=Deploy`

This launches the Hotfix Utility, which then creates a folder named *Deploy* in the directory from which the utility was executed and copies all files and folders necessary for updating a project running on .NET version 4.6 into that folder.

3. Once you have the update files ready, take your website offline. You can do this by copying the **App_Offline.htm** file from the hotfix installation directory into the root of the remote web project.
4. Upload the update files created by the Hotfix Utility into your remote web project's **CMS** folder (overwrite the original files).

**Important**

If you have previously modified some of the Kentico project files (such as web parts or administration interface files), DO NOT OVERWRITE these files. You could lose your modifications.

You need to compare the modified files with the new versions and make appropriate changes manually.

5. Manually execute the hotfix SQL scripts on your Kentico database. The script files are located in the **SQL** folder of your hotfix installation directory.
 - If you do not use separated databases – run **Hotfix_separated.sql** first and then **Hotfix_default.sql**.
 - If you use [separated databases](#) – run **Hotfix_separated.sql** on the separated database and **Hotfix_default.sql** on your main database.
 - Note: The script files may be empty if the given hotfix does not require any SQL scripts for the related tables (skip the file in these cases).
6. Open the project in Visual Studio and build the solution.
7. Bring your website back online (delete the **App_Offline.htm** file).

Applying the hotfix to Microsoft Azure

In the case of websites hosted on Microsoft Azure, you need to apply the hotfix to your local copy of the web application and then redeploy.

1. Run **Kentico Hotfix Utility (Hotfix.exe)** from the location where you installed the hotfix (*C:\Program Files\Kentico\<version>\Hotfix<version>* folder by default).
2. Select the folder containing your local version of the project using the **Browse** button or type in the path to the project manually.
3. Use the utility to back up your project files and/or database in case you have not already done so.
4. Select whether you want to update files, database or both. If your database is hosted on SQL Azure, you need to update it manually.
5. Click **Next** to start the hotfix procedure.
6. After the update finishes, click **Next** to view any problems that may have occurred and the instructions to solve them.
7. Open the project's solution in Visual Studio and rebuild it.
8. Redeploy the application to Microsoft Azure.
9. If your website's database is hosted on SQL Azure, manually execute the hotfix SQL scripts against it. The script files are located in the **SQL** folder of your hotfix installation directory.
 - If you do not use separated databases – run **Hotfix_separated.sql** first and then **Hotfix_default.sql**.
 - If you use [separated databases](#) – run **Hotfix_separated.sql** on the separated database and **Hotfix_default.sql** on your main database.
 - Note: The script files may be empty if the given hotfix does not require any SQL scripts for the related tables (skip the file in these cases).

Applying the hotfix to MVC projects

If your website runs as a [separate ASP.NET MVC application](#) (in addition to the Kentico administration application), use the following approach to apply the hotfix:

1. [Install the hotfix](#) onto a local machine where you have your MVC and Kentico applications.
2. Apply the hotfix to the Kentico application (and the database) according to the instructions in the relevant section above.
3. Open your MVC application in Visual Studio.
4. Right-click your MVC project in the **Solution Explorer** and select **Manage NuGet Packages**.
5. Update the **Kentico.Libraries** NuGet package to the version that matches the applied hotfix.
6. Redeploy the local applications to your hosting server.

After applying the hotfix

All users who work with the Kentico administration interface should **clear the cache** in their browsers after applying the hotfix. Otherwise, some features in the administration interface may not be displayed correctly.

After applying the hotfix to a development instance that has [continuous integration](#) enabled, you need to run complete serialization for all objects to recreate the content of the *CIRepository* folder (use the **Continuous integration** application in the Kentico administration interface). Apply the hotfix separately for each instance in your development environment. After one developer commits the hotfixed changes to the source control, other developers CANNOT commit or load changes until they apply the hotfix to their own instance.

Updating setup files and external utilities

Hotfixes provide bug fixes for all files related to the Kentico installation, which includes the installer and other external utilities, such as Kentico Installation Manager and Kentico Service Manager.

To update these files and utilities, you need to apply the hotfix to the setup files:

1. When running the hotfix utility, click **Switch to advanced mode** in the **Change confirmation** step.
2. Select the **Setup files** checkbox.
3. Click **Next**.
4. In the **Running applications** step, stop any applications that are currently running from the setup files.
5. Click **Next** and continue through the rest of the hotfix procedure.

The hotfix makes any required changes to the files in the directory where you installed your Kentico setup files.

Additional notes and workarounds

11.0.3

- REST – The [REST service](#) limits the length of request data in JSON format to 2097152 characters (4 MB of Unicode string data) by default, which is not sufficient in certain cases (for example when creating attachment objects with large binary data). If you need to adjust the limit, apply the hotfix and add the **CMSRestMaxJsonLength** key to the *appSettings* section of your project's web.config file.

Example

```
<add key="CMSRestMaxJsonLength" value="2147483647" />
```

11.0.5

- Cultures – The **Related pages** web part did not take the **Combine with default culture** setting into account and always displayed the default culture version for untranslated related pages. After applying the hotfix, the web part only displays related pages that exist in the visitor's currently selected culture if the *Combine with default culture* setting is disabled.

11.0.8

- Facebook integration – The hotfix adds support for app secret proof parameters when communicating with integrated Facebook apps. If your website uses the [Facebook page integration](#) features or [Facebook authentication](#), we strongly recommend applying the hotfix and then securing your Facebook app:
 1. Open the Facebook App editing page at <https://developers.facebook.com/apps>.
 2. Select your app and view the **Settings -> Advanced** tab.
 3. Scroll down to the **Security** section and switch the **Require App Secret** option to Yes.
 4. Click **Save Changes**.

11.0.9

- The [SharePoint data source](#) web part did not correctly handle multiselect fields with complex types. The hotfix ensures that the data source processes such fields into strings consisting of the type's properties separated by semicolon (;) characters, and different entries separated by newline characters (environment specific). To display data from multiselect SharePoint fields, you need to manually parse the values within the [transformations](#) applied to the data source's output. For example:

```
<%# String.Join(" and ", String.Split(Eval<string>("MultiselectField"), Environment.NewLine).Select(l => String.Format("{1} ({0})", String.Split(l, ";")))) %>
```

11.0.10

- The original implementation of the **reCAPTCHA** form control will stop working after March 31, 2018. The hotfix updates the control to use reCAPTCHA v2. Please register your site again at <https://www.google.com/recaptcha/admin>, select the *reCAPTCHA v2* type, and copy the new API keys into the corresponding Kentico settings. For more information, see [Spam protection \(CAPTCHA\)](#).

11.0.15

- Licensing – After evaluating feedback from partners and clients, Kentico has decided to change the license limitations on the maximum number of enabled [products](#) and make them less restricted. Applying the hotfix removes the limit completely for *Kentico CMS Ultimate* license editions and increases the limit to 500 products for *Kentico CMS Base* editions. For more information about license editions, see the [Configure Your License](#) page on kentico.com.

11.0.19

- LinkedIn integration – Effective May 18th, 2018, the LinkedIn API will no longer work with the original OAuth 1.0 implementation in Kentico. The hotfix updates the system to use OAuth 2.0 authentication for [LinkedIn company management](#) and [LinkedIn authentication](#) functionality. After applying the hotfix, you need to manually perform the following:
 - If you use LinkedIn authentication:
 - Sign in to <https://www.linkedin.com/secure/developer>.
 - Select your app on the My applications page.
 - On the **Authentication** tab of the Application Settings, add the URLs of all website pages that provide LinkedIn authentication (i.e. pages that display the [LinkedIn logon web part](#)). Enter the full absolute URL into the **Authorized Redirect URLs** field (under *OAuth 2.0*), including the protocol, domain, virtual directory and trailing slash (if necessary). For example: <https://mydomain.com/Special-pages/SignIn>
 - Click **Add** for each URL.
 - Click **Update** after you add all required redirect URLs.
 - If you use LinkedIn company profile management:
 - Sign in to <https://www.linkedin.com/secure/developer>.
 - Select your app on the My applications page.
 - On the **Authentication** tab of the Application Settings, enter the following value into the **Authorized Redirect URLs** field (under *OAuth 2.0*):
`<your website's domain>/CMSModules/SocialMarketing/Pages/LinkedInCompanyAccessTokenDialog.aspx`



The redirect URL value must be absolute, including the protocol, domain and virtual directory. For example:
<https://mydomain.com/CMSModules/SocialMarketing/Pages/LinkedInCompanyAccessTokenDialog.aspx>

If your website's administration interface is available on multiple domains, add all of the possible options.

- Click **Add**.
- Click **Update**.

6. Generate new authentication tokens for all LinkedIn company profiles registered in Kentico:
 - a. Open the **LinkedIn** application in Kentico.
 - b. Select the **Company profiles** tab.
 - c. Edit the company profile and switch to the **General** tab.
 - d. Click **Reauthorize** next to **Company**.
 - e. In the opened dialog, enter your LinkedIn account credentials and click **Allow access**.
 - f. Click **Save**.

11.0.23

- Web farms – The system creates and processes redundant web farm synchronization tasks in environments where multiple web farm instances share a single file system. A common example is when running Kentico as a [scaled out Azure Web App](#) (i.e., the instance is utilizing multiple web farm servers). If you run Kentico in this type of scenario, apply the hotfix, add the **CMSSharedFileSystem** configuration key to the *appSettings* section of your project's *web.config* file, and redeploy the project:

```
<add key="CMSSharedFileSystem" value="true" />
```

The key signals Kentico that it is running on shared storage, and configures web farm behavior accordingly.



Important: If you have the *CMSSharedFileSystem* key enabled, use [local search indexes](#), and run Kentico as a [scaled out Azure Web App](#), you need to apply hotfix **11.0.43** or newer. Otherwise search indexing will not work correctly.

11.0.25

- Email marketing – [Email widgets](#) did not reflect the selected UI culture when storing their properties, which could lead to an error when editing the widget. For example, an error occurred when the preferred UI culture used a different date format than the *en-US* culture. After applying the hotfix, you need to manually replace the affected widgets in your marketing emails and re-save their configuration.

11.0.27

- Geolocation – By default, the [geolocation feature](#) in Kentico 11 uses MaxMind's GeoLite or GeoIP Legacy Databases, which will be [discontinued](#) in the future. If you wish to integrate MaxMind's newer GeoIP2 Databases, use the following manual process:
 1. Apply hotfix 11.0.27 or newer to your Kentico instance.
 2. Open your web project in Visual Studio. For sites created using the [MVC development model](#), perform all of the following steps in the MVC project that provides the live site.
 3. Right-click the project in the **Solution Explorer** and select **Manage NuGet Packages**.
 4. Install the **Kentico.Geolocation.Update-v11** package.



Note: The package also installs several required dependencies, such as the **MaxMind.GeoIP2** package and version 11.0.2 of the **Newtonsoft.Json** package. You need to accept all license agreements and ensure that these packages work correctly with any customizations you may have in your project. If you have any custom projects within the solution that use the same packages, you need to consolidate the package versions.

5. Download the free [GeoLite2 City](#) database or purchase the [GeoIP2 City](#) database (provides more accuracy and details).
6. Place the database file into your project's *~/App_Data/CMSModules/WebAnalytics/MaxMind/* folder.
7. Open your project's *web.config* file, and make sure the **CMSSGeoIPLocationFileName** key (in the *appSettings* section) is set according to the name of your geolocation database file (*GeoLite2-City.mmdb* or *GeoIP2-City.mmdb*).

Example

```
<add key="CMSSGeoIPLocationFileName" value="GeoIP2-City.mmdb" />
```

8. (Optional) If you also wish to use the GeoLite2 ASN database to determine the organization name of visiting contacts, download the [GeoLite2 ASN](#) database file (*GeoLite2-ASN.mmdb*) and place it into your web project's ~
/App_Data/CMSModules/WebAnalytics/MaxMind/ folder.

11.0.29

- E-commerce – When displaying [coupon codes](#) added to a customer's shopping cart, it was not possible to adjust the appearance of codes that are no longer valid (for example after products are removed from the cart and the total price falls under the value required by the coupon code's discount). After applying the hotfix, coupon code transformations provide a new **IsApplied** data property, which can be used to evaluate whether codes are still valid. You can adjust your site's transformations, and use the new property to add formatting options for invalid coupon codes. For example:

Text / XML transformation

```
<div class="label">
  <span {% if (!IsApplied) {"style='text-decoration: line-through'"} %}>
    {% HTMLEncode(Code) %}
  </span>
</div>
```

11.0.35

- Pages – [Page aliases](#) containing [wildcards](#) were processed in an incorrect order in certain cases. For example, if a page had two aliases with paths like /page/{param} and /page/{p1}-{p2}, accessing the URL path /page/value1-value2 resulted in the first alias being selected instead of the second (the value of the *param* parameter was *value1-value2*, and the *p1* and *p2* parameters were not set). After applying the hotfix, you need to manually resave all page aliases where this problem occurs.

11.0.39

- E-commerce – Certain payment gateways (e.g. [PayPal](#)) require data with separate tax values for the purchased items and the shipping costs, instead of a single overall tax value. Applying the hotfix ensures that payments using the default PayPal provider no longer fail if the site is configured to include tax in prices. If you have customized your site's [tax calculation process](#) by creating a custom **ITaxCalculationService** implementation, you need to manually update your code after applying the hotfix – when preparing **TaxCalculationResult** objects, do not use the setter of the **TotalTax** property. Instead, you need to separately set the new **ItemsTax** and **ShippingTax** properties.

11.0.42

- Facebook integration – Due to changes in the Facebook API and updated security requirements, the initial [Facebook page publishing](#) functionality in Kentico 11 no longer works and [Facebook authentication](#) requires configuration changes in the related Facebook app. To use the features, you need to apply the hotfix and manually perform the following changes:
 - If you use Facebook authentication:
 1. Log in to <https://developers.facebook.com/apps>.
 2. Click the Facebook app that you use for your website.
 3. Open the **Settings** tab of the app's **Facebook Login** product.
 4. In the **Valid OAuth redirect URIs** field, enter the URLs of all pages from your website where you wish to allow Facebook authentication (i.e. pages that display the *Facebook Connect logon* web part).




The redirect URI values must be absolute, including the protocol, domain and virtual directory. For example: <https://mydomain.com/Special-pages/SignIn>

If your website is available on multiple domains, add all of the possible options.

See also: [Strict URI Matching](#)

5. Click **Save Changes**.

6. If you wish to retrieve user birthday and location data from Facebook, you additionally need to submit your app for **App Review** and request the **user_birthday** and **user_location** permissions.
 - If you use Facebook page publishing:
 1. Log in to <https://developers.facebook.com/apps>.
 2. Click the Facebook app that you use for your website.
 3. Open the **Settings** tab of the app's **Facebook Login** product.
 4. Enter the following values into the **Valid OAuth redirect URIs** field:
 - `<your website's domain>/CMSModules/SocialMarketing/Pages/FacebookPageAccessTokenDialog.aspx`
 - `<your website's domain>/CMSModules/SocialMarketing/Pages/FacebookPageAccessTokenDialog.aspx?redirected=1`
-  The redirect URI values must be absolute, including the protocol, domain and virtual directory. For example:
<https://mydomain.com/CMSModules/SocialMarketing/Pages/FacebookPageAccessTokenDialog.aspx>

If your website's administration interface is available on multiple domains, add all of the possible options.
5. Click **Save Changes**.
 6. Switch to the **App Review** tab and make sure your app has the following permissions as approved items:
 - **manage_pages** and **publish_pages** – required to retrieve Page Access Tokens and [post_content](#) from Kentico to Facebook pages.
 - **read_insights** – required to retrieve [Facebook Insights statistics](#) about published posts and the related pages.
 7. If your app is missing any of the required permissions, create a submission, add the permission items, and submit your app for review.
 8. In Kentico, open the **Facebook** application, select the **Pages** tab, and check that your pages are valid:
 - Existing pages with a valid access token should remain functional.
 - When adding new pages or renewing the access token, you now need to set the **Facebook page ID** instead of the originally used page URL. You can find the Page ID value on the *About* tab of the given page on Facebook.

11.0.47

- E-commerce – Applying hotfix 11.0.39 or newer introduced a change in the e-commerce API, which could cause undetected broken functionality for sites with a [customized tax calculation process](#). After applying hotfix 11.0.47, a **runtime and compilation error** now occurs for projects with custom code that attempts to set the **TotalTax** property of **TaxCalculationResult** objects. In these cases, you need to update your code to set the new **ItemsTax** and **ShippingTax** properties instead.

Fixed bugs

For a full list of bugs covered by the hotfix, open the [Hotfixes page](#) on DevNet and click **Fixed bugs** for the appropriate Kentico version.