

Resource strings store the text displayed in the Kentico administration interface (and on the live site in some cases).

- All default resource strings are stored in the **cms.resx** file, which is located in the project's **CMSResources** folder.
- You can define site specific resource string files, as described in [Creating site specific resource strings](#).
- To create resource string files for custom modules, see an example in [Example - Creating a packageable module](#).
- You can edit the resource strings that the system stores in the database in the **Localization** application on the **Resource strings** tab.

### Resource string priority

When loading resource strings, the system uses the following priority:

1. database (Localization application) – highest priority
2. site specific resx files
3. resx files of custom modules
4. custom.resx
5. cms.resx

If there are duplicate strings with the same key in all five sources, the system uses the one stored in the database.

To change the priorities, you can add the following key to your web.config:

```
<add key="CMSUseSQLResourceManagerAsPrimary" value="false" />
```

When this key is added, the priorities are as follows:

1. site specific resx files
2. resx files of custom modules
3. custom.resx
4. cms.resx
5. database

## Modifying the default UI strings

If you want to modify text in the Kentico administration interface (including web part dialogs), use one of the following options:

- Override resource strings in the **Localization** application.
- Create a **custom.resx** file in the project's **CMSResources** folder and store your strings in this file.
  - The file's content must have a valid XML structure for the [.resx file format](#), including header information. You can copy the general structure from the default **cms.resx** file.
  - To customize strings in non-English resource files, your custom file must use a name in format **custom.<culture code>.resx** (for example, **custom.fr-fr.resx** for French).

In both cases, the keys used to identify the strings must be the same as in the default **cms.resx** file.

## Accessing the Localization application

Editing of resource strings in the **Localization** application is only possible for two types of users:

- Users with the Global administrator [privilege level](#)
- Users who belong to [roles](#) with the [Localize strings permission](#) for the **Localization** module



**Note:** The *Localize strings* permission allows users to edit or override all global resource strings. You cannot use the permission model to restrict access to a certain subset of resource strings, for example on Kentico instances containing multiple independent websites.

## Adding your own strings

If you need to translate custom strings used on your website such as form labels, display names of objects or other static text into other languages, you can create a new resource string:

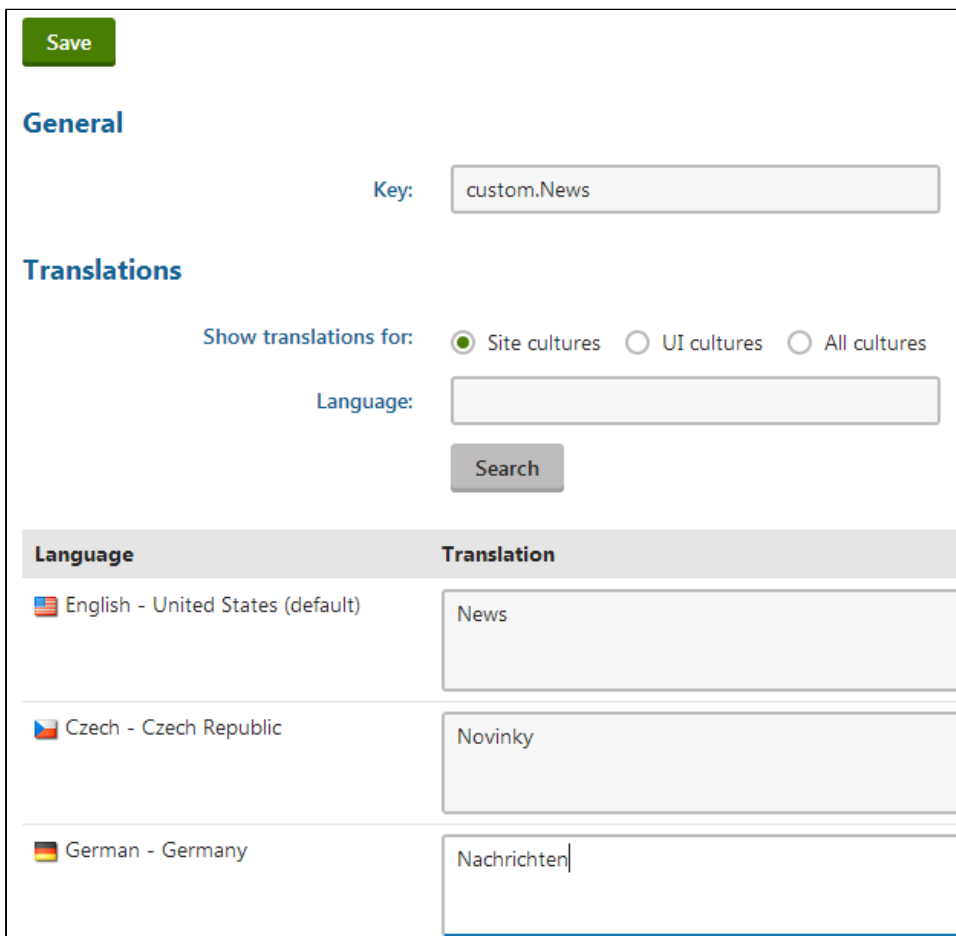
1. Open the **Localization** application.
2. Choose the default culture in the **Culture** selector.
3. Click **New string**.
4. Type the name of the resource string into the **Key** field.
5. Type the text for the key into the **Translation** field of the corresponding language.
6. Click **Save**.

A new string in the default culture is now displayed in the list.




✓ You can also create new resource strings directly when editing text fields in the administration interface. See [Localizing text fields](#).

## Translating resource strings into other languages

1. Open the **Localization** application.
2. Edit (✎) the resource string.
3. If you do not see the desired language, use the **Show translation for** radio buttons to change the category of cultures.
4. Translate the string into the desired language in the corresponding **Translation** field.



The screenshot shows the 'Translations' section of the Localization application. At the top, there is a 'Save' button. Below it, the 'General' section shows the 'Key' field with the value 'custom.News'. The 'Translations' section has three radio buttons for 'Show translations for': 'Site cultures' (selected), 'UI cultures', and 'All cultures'. Below these is a 'Language' dropdown menu and a 'Search' button. A table displays the translations for the selected key:

Language	Translation
 English - United States (default)	News
 Czech - Czech Republic	Novinky
 German - Germany	Nachrichten

5. Click **Save**.

The resource string and its translation are now created and stored in the database.

## Using resource strings in transformations

If you need to display resource strings within [transformations](#), use the following code:

- **ASCX** transformations – call the **Localize** transformation method:

```
<%# Localize("Text containing localization expressions: {$stringKey$}") %>
```

- **Text / XML** transformations – use localization string [macro expressions](#) or the **GetResourceString** macro method:

```
{ $stringKey$ }  
- OR -  
{% GetResourceString("stringKey") %}
```

## Loading resource strings in the API

If you need to retrieve the value of a resource string in your custom code, use the **CMS.Helpers.ResHelper.GetString** method.

```
using CMS.Helpers;  
  
...  
  
// Loads the value of the 'stringKey' resource string (in the default culture)  
string localizedResult = ResHelper.GetString("stringKey");
```