

Kentico CMS 5.5 Transformations

Transformations overview

Transformations in Kentico CMS are pieces of code that determine how documents, or certain parts of them, are rendered by viewers. They take raw data from the Kentico CMS database and *transform* it into a specified output HTML. This makes them an essential tool when displaying documents and document related data on the pages of a website. **Their functionality is very similar to that of templates used by standard ASP.NET templated controls, such as the Repeater.** The main difference is that **transformations are stored separately in the database and are therefore easily reusable.** Transformations are categorized according to the data that they are supposed to display (document types, custom tables, etc.). **The code of a transformation can contain a mix of HTML elements, embedded controls, standard ASP.NET data binding expressions and methods, such as Eval() or special methods designed to be used in transformations.** The use of transformations is supported by all Kentico CMS web parts and controls that are intended to display data, no matter if the data comes from content documents, system objects or your own data source.

Transformation methods (CMSAbstractTransformation)

Data <ul style="list-style-type: none"> • Eval(string columnName) • Eval(string columnName, bool encode) • EvalCDATA(string columnName) • EvalCDATA(string columnName, bool encapsulate) • GetEditableValue(string controlId) <ul style="list-style-type: none"> • DataItem { get; } • DataItemIndex { get; } • DataRowView DataRowView { get; } • DisplayIndex { get; } • DocumentCustomData { get; } • EditableItems { get; } • NodeCustomData { get; } 	Document links and URLs <ul style="list-style-type: none"> • GetAbsoluteUrl(string relativeUrl) • GetAbsoluteUrl(string relativeUrl, int siteId) • GetAbsoluteUrl(string relativeUrl, object siteNameObj) • GetAbsoluteUrl(string relativeUrl, string domainName) <ul style="list-style-type: none"> • GetAttachmentIconUrl(object attachmentExtension, string iconSet) • GetAttachmentUrl(string attachmentFileName, string nodeAliasPath) <ul style="list-style-type: none"> • GetBlogCommentUrl(object documentIdObj) <ul style="list-style-type: none"> • GetDocumentLink() • GetDocumentLink(bool encodeName) <ul style="list-style-type: none"> • GetDocumentUrl() • GetDocumentUrl(object nodeGuidColumn, object nodeAlias) <ul style="list-style-type: none"> • GetFileIconUrl(object attachmentGuidColumn) • GetFileIconUrl(object attachmentGuidColumn, string iconSet) <ul style="list-style-type: none"> • GetFileUrl(object attachmentGuidColumn) • GetFileUrl(object attachmentFileName, object attachmentDocumentId) <ul style="list-style-type: none"> • GetForumPostUrl(object postIdPath, object forumId) • GetMediaFileUrl(object fileGUID, object fileName) • GetMessageBoardUrl(object documentIdObj) • GetProductUrl(object skuGUID, object skuName, object siteName) <ul style="list-style-type: none"> • GetUrl(object aliasPath, object urlPath) • GetUrl(object aliasPath, object urlPath, object siteName) <ul style="list-style-type: none"> • GetUserAvatarImageUrl(object avatarId, object userId, int maxSideSize, int width, int height)
Membership <ul style="list-style-type: none"> • GetAge(object dateOfBirth, string unknownAge) • GetGender(object genderObj) <ul style="list-style-type: none"> • GetGroupAvatarImage(int maxSideSize, object alt) • GetGroupAvatarImage(object avatarId, int maxSideSize, object alt) • GetGroupAvatarImage(object avatarId, int maxSideSize, int width, int height, object alt) <ul style="list-style-type: none"> • GetUserAvatarImage(int maxSideSize, string alt) • GetUserAvatarImage(object avatarId, object userId, int maxSideSize, object alt) • GetUserAvatarImage(object avatarId, object userId, int maxSideSize, int width, int height, object alt) <ul style="list-style-type: none"> • GetGroupProfileUrl(object groupNameObj) • GetMemberProfileUrl(object memberNameObj) • GetUserProfileUrl(object userNameObj) <ul style="list-style-type: none"> • GetBadgeImage(int badgeld) • GetBadgeName(int badgeld) 	Text <ul style="list-style-type: none"> • HTMLEncode(string text) <ul style="list-style-type: none"> • EnsureMaximumLineLength(object textObj, int maxLength) • LimitLength(object textObj, int maxLength, string padString) <ul style="list-style-type: none"> • RemoveDiscussionMacros(object inputText) • StripTags(object inputText)
Date & time <ul style="list-style-type: none"> • FormatDateTime(object dateTime, string format) • GetCustomDateTime(object dateTime, string timeZoneName) • GetDate(string dateTimeField) • GetDateTime(string dateTimeField) • GetDateTime(string dateTimeField, string formattingString) • GetSiteDateTime(object dateTime) • GetUserDateTime(object dateTime) • GetDateTimeString(object dateTime, bool showTooltip) 	General <ul style="list-style-type: none"> • IfCompare(object value, object comparableValue, object falseResult, object trueResult) • IfEmpty(object value, object emptyResult, object nonEmptyResult) • GetNotEmpty(object columnsObj) • GetColumnName(string columns) • GetColumnName(string columns, bool allowEmptyColumn) <ul style="list-style-type: none"> • GetBlogCommentsCount(object documentIdObj, string nodeAliasPath) • GetBlogCommentsCount(object documentIdObj, string nodeAliasPath, bool includingTrackbacks)
Images <ul style="list-style-type: none"> • GetImage(object attachmentGuidColumn) • GetImage(object attachmentGuidColumn, int maxSideSize) • GetImage(object attachmentGuidColumn, int width, int height) • GetImage(object attachmentGuidColumn, object maxSideSize, object width, object height) • GetImage(object attachmentGuidColumn, object maxSideSize, object width, object height, object alt) <ul style="list-style-type: none"> • GetImageByUrl(object imageUrl) • GetImageByUrl(object imageUrl, int width, int height) • GetImageByUrl(object imageUrl, int maxSideSize) • GetImageByUrl(object imageUrl, object maxSideSize, object width, object height, object alt) <ul style="list-style-type: none"> • IfImage(object attachmentGuidColumn, object imageUrl, object notImage) 	Syndication (RSS, Atom, XML) <ul style="list-style-type: none"> • GetAtomDateTime(object dateTime) • GetRSSDateTime(object dateTime) <ul style="list-style-type: none"> • GetProductUrlForFeed(object skuGUID, object skuName, object siteName) • GetDocumentUrlForFeed() • GetForumPostUrlForFeed(object postIdPath, object forumId) • GetMediaFileUrlForFeed(object fileGUID, object fileName) • GetMessageBoardUrlForFeed(object documentIdObj) • GetBlogCommentUrlForFeed(object documentIdObj)
Smart search <ul style="list-style-type: none"> • SearchResultUrl(bool absolute) • SearchHighlight(string text, string startTag, string endTag) • GetSearchImageUrl(string noImageUrl, int maxSideSize) • GetSearchValue(string columnName) 	Microsoft SharePoint <ul style="list-style-type: none"> • SplitSharePointField(string value, int index) • GetSharePointFileUrl(string server, string name)

How it works

1. Web part sets the properties of a templated control inside it, including the transformation name for items and optionally for selected items. The transformation name is in format `<classname>.<transformationname>`

e.g. `cms.news.default`

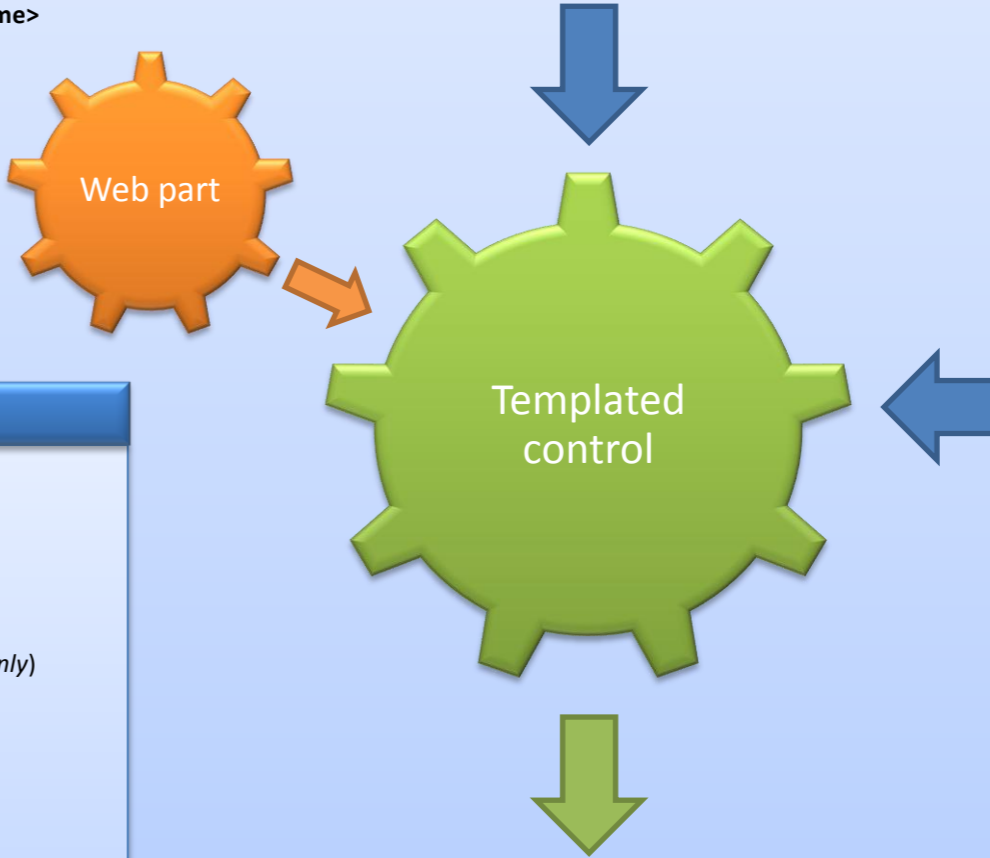
Templated controls

- Document controls (from Content)**
 - CMSCalendar
 - CMSRepeater
 - CMSDataList
 - CMSDataGrid (Selected item only)
 - GoogleSiteMap
 - CMSViewer (XSLT transformation only)
- Query controls (general query)**
 - QueryRepeater
 - QueryDataList
- Basic controls (need data source)**
 - BasicRepeater
 - BasicDataList
 - BasicDataGrid
 - BasicCalendar

Web parts using transformations are always based on one of the above listed templated controls from the CMS.Controls library.

NewsID	NewsTitle	NewsReleaseDate	NewsSummary
1	Your first news	2008-01-10 05:00:00.000	<p>Here you can enter the s...
2	Your second news	2008-01-15 00:00:00.000	<p>Here you can enter the s...

2. The templated control loads the data depending on current control settings. The data is loaded for specific document types (document controls) or with dependence on a specific query (query controls). The data can be also provided by a data source control or web part.


Transformation

```

<div class="newsItem">
  <p>
    <strong>
      <%# Eval("NewsTitle") %>
    </strong>
  </p>
  <p>
    <%# StripTags(Eval("NewsSummary")) %>
  </p>
</div>
  
```

3. The templated control loads the transformation definition and evaluates the content through the standard data binding process. Eval is the most used method in transformations. If the data contains HTML code that is not guaranteed to be safe and shouldn't be directly rendered, you can use a Eval method override with the encode parameter. The transformation is inherited from the CMSAbstractTransformation class. The available methods are listed in the Transformation methods box.

Result (View/HTML)

Your first news

Here you can enter the summary of the news item. You can use WYSIWYG editor to format the text, insert links and images. The summary will be used in news item preview.

```

<div class="newsItem">
  <p>
    <strong>
      Your first news
    </strong>
  </p>
  <p>
    Here you can enter the summary of the news item.....
  </p>
</div>
          
```

4. The data is transformed to the required format depending on the used transformation and the output is rendered as standard HTML code. You have complete control over the markup.

Selected item transformation

Templated controls for documents allow you to setup the **selected item transformation**. This transformation is used if the currently displayed document is of the **same document type as is defined in the control's class name property**. You can leverage this and use a single control for both list and detail views.

Custom functions

In some cases, you may need to process values using your own code. In these cases you can create your own function to process data. It is recommended to create your own class in the App_Code folder (For example ~/App_Code/Global/MyFunctions.cs). This custom function can be called as follows:

```

<%# MyFunctions.DoSomething(Eval("NewsTitle")) %>
  
```

XSLT transformations

Kentico CMS supports two types of transformations. ASCX and XSLT transformations. XSLT type transformations need to be in **valid XML format** and can contain **standard XSL elements**. The transformed XML is a DataSet serialized into the XML.

Please note: XSLT transformation are case sensitive, which means that the match and select attributes must contain correct case values.

```

<xsl:stylesheet xmlns:xsl="http://www.w3.org/1999/XSL/Transform" version="1.0">
  <xsl:output method="xml" omit-xml-declaration="yes" />
  <xsl:template match="CMS.News">
    Title:<xsl:value-of disable-output-escaping="yes" select="NewsTitle" />
    Summary:<xsl:value-of disable-output-escaping="yes" select="NewsSummary" />
  </xsl:template>
</xsl:stylesheet>
  
```

Localized transformations

In some cases, you may need to display a different text in transformations based on the **currently displayed language**. You can achieve this by creating another transformation with the name ending with the culture code of that particular language. (For an existing **default** transformation, a localized version named **default_en-gb** can be created, which will be used for the en-GB language).

Context menus

When editing transformations displaying **Users** or **Groups**, you can enclose the transformation code within one of the following controls, which enables context menus after a right-click on the selected user or group.

The context menus that are loaded are the following:

- ~/CMSAdminControls/ContextMenus/UserContextMenu.ascx
- ~/CMSAdminControls/ContextMenus/GroupContextMenu.ascx

Examples

Users:

```

<cms:usermenucontainer runat="server" ID="userMenuElem" MenuID="userContextMenu"
  Parameter='<%# Eval("UserID").ToString() %>' ContextMenuCssClass="UserContextMenu">
  ... transformation code ...
</cms:usermenucontainer>
  
```

Groups:

```

<cms:groupmenucontainer runat="server" ID="groupMenuElem" MenuID="groupContextMenu"
  Parameter='<%# Eval("GroupID").ToString() %>' ContextMenuCssClass="UserContextMenu">
  ... transformation code ...
</cms:groupmenucontainer>
  
```