


The UniView is a templated listing control that supports displaying of hierarchical data from grouped data sources.

### Grouped data sources

Grouped data sources are represented by the *GroupedDataSource* class. A grouped data source takes a standard data source, such as a *DataSet*, and categorizes the data items into hierarchy levels according to the values of specified columns.

You can use the UniView with any bindable data source — not only Kentico data and objects.

 **Tip:** If you want to display pages from Kentico, you can use the [CMSUniView](#) control, which has built-in support for loading pages.

## Getting started

The following is a step-by-step tutorial that shows how to display all pages (*CMS.Menuitem* pages) from the sample Corporate Site using the UniView control:

1. Create a new **Web form** in your web project.
2. Drag the **UniView** control from the toolbox onto the form.
3. Add the following code marked by the **UniView templates** comments between the `<cms:UniView>` tags:

```
<cms:UniView ID="UniView1" runat="server">

    <!-- UniView templates
    ----- --%>

    <ItemTemplate>
        <li>
            <%# HTMLHelper.Encode(Convert.ToString(Eval("NodeName"))) %>
        </li>
    </ItemTemplate>


    <AlternatingItemTemplate>
        <li>
            <font color="#999999">
                <%# HTMLHelper.Encode(Convert.ToString(Eval("NodeName"))) %>
            </font>
        </li>
    </AlternatingItemTemplate>

    <HeaderTemplate>
        <ul>
    </HeaderTemplate>

    <FooterTemplate>
        </ul>
    </FooterTemplate>

    <!-- UniView templates
    ----- --%>

</cms:UniView>
```

 This sets the templates used when displaying the menu items. The control dynamically replaces the `<%# ... %>` inline code with values of the currently displayed record. This is then repeated for every record in the data source.

4. Add the following references to the beginning of the web form's code behind:


```
using System.Data;

using CMS.DocumentEngine;
using CMS.Helpers;
```

5. Add the following code to the **Page\_Load** method:

```
// Creates a DataSet containing all menu item pages in the system
DataSet ds = DocumentHelper.GetDocuments("CMS.MenuItem").Path("/", PathTypeEnum.Children).OrderBy("NodeLevel, NodeOrder");

// Checks that the DataSet isn't empty
if (!DataHelper.DataSourceIsEmpty(ds))
{
    // Binds the DataSet to the UniView control
    this.UniView1.DataSource = ds;
    this.UniView1.DataBind();
}
```

 This code reads pages from the database and provides them to the UniView control as a DataSet.

6. Save the changes to the web form and its code behind file.
7. Right-click the web form in the Solution explorer and select **View in Browser**.

The control displays all *page (menu item)* pages in the system.

- Home
- Products
- News
- Partners
- Community
- Services
- Company
- Media
- Examples
- Mobile
- Other

## Using grouped data sources

A flat list is not ideal for displaying pages that are actually organized in a hierarchy. The following steps show how to set up the UniView to display a hierarchical structure:

1. Add another reference to the beginning of the web form's code behind:

```
using CMS.Base;
```

2. Modify the code of the **Page\_Load** method according to the following:

```
// Creates a DataSet containing all menu item pages in the system
DataSet ds = DocumentHelper.GetDocuments("CMS.MenuItem").Path("/", PathTypeEnum.Children).OrderBy("NodeLevel, NodeOrder");

// Checks that the DataSet isn't empty
if (!DataHelper.DataSourceIsEmpty(ds))
{
    // Creates a GroupedDataSource from the DataSet
    // You need to specify the columns that the data uses to identify parent items, and each item's level in the hierarchy
    GroupedDataSource gpd = new GroupedDataSource(ds, "NodeParentID", "NodeLevel");

    // Specifies the column that the data uses as an identifier (to determine parent-child relationships)
    this.UniView1.RelationColumnID = "NodeID";

    // Binds the DataSet to the UniView control
    this.UniView1.DataSource = gpd;
    this.UniView1.DataBind();
}
```



This code takes the DataSet containing the pages, groups them according to the **NodeID** of their parent page and determines their level in the hierarchy according to their **NodeLevel**.

3. Switch to the web form's markup, and add the **SubLevelPlaceholder** control into the UniView's **ItemTemplate** and **AlternatingItemTemplate**:

```
<ItemTemplate>
    <li>
        <%= HTMLHelper.Encode(Convert.ToString(Eval("NodeName"))) %>
        <cms:SubLevelPlaceholder runat="server" ID="plcSub" />
    </li>
</ItemTemplate>

<AlternatingItemTemplate>
    <li>
        <font color="#999999">
            <%= HTMLHelper.Encode(Convert.ToString(Eval("NodeName"))) %>
        </font>
        <cms:SubLevelPlaceholder runat="server" ID="plcSub" />
    </li>
</AlternatingItemTemplate>
```

**i** The **SubLevelPlaceholder** specifies where exactly the control inserts child levels in the output code. For items that have descendants in the hierarchy, the control renders the child level instead of the placeholder (including the header and footer template for the new level).

4. Save the changes.

When you refresh the web form's output, the control displays the pages in a hierarchical structure:

- Home
- Products
  - Smartphones
  - Laptops and Tablets
  - Software
  - E-Books
  - IT Services
  - Memberships
  - Donations
- News
- Partners
  - Silver Partners
  - Gold Partners
- Community
  - Blogs
  - Events
    - Events list
  - Forums
    - Forum search
  - Wiki
    - My contributions
- Services
  - Web Design
  - Web Development
  - Network Administration

## Configuration

You can set the following properties for the UniView control:

Property name	Description	Sample value
AlternatingRange	Indicates how often the <b>AlternatingItemTemplate</b> should be used.	
AlternatingStartPosition	Indicates the item number from which the <b>AlternatingItemTemplate</b> should start being used.	
DataSource	The object from which the list of data items is retrieved.	

HideHeaderAndFooterForSingleItem	If enabled, the UniView does not render the content of the <b>HeaderTemplate</b> and <b>FooterTemplate</b> for levels that only contain a single item.	
HierarchicalDisplayMode	Sets the hierarchical display mode. <i>Inner</i> generates sub-levels inside the level above, <i>Separate</i> generates sub-levels outside of the upper levels.	"Inner" "Separate"
OuterData	Sets the data generated in the <b>HeaderTemplate</b> and <b>FooterTemplate</b> .	
PagerDataItem	Gets or sets the pager data item object.	
PagerForceNumberOfResults	If set, the DataSet containing paged items is not modified by the pager, but the pager itself behaves as if the amount of paged items were identical to this value. The value must be set to -1 for the property to be disabled.	
RelationColumnID	Specifies the name of the column that the source data uses as an identifier (to determine parent-child relationships).	"NodeID"
SelectedItemColumnName	The name of the column that should be used for to find out which item is currently selected.	"DocumentID"
SelectedItemValue	The item whose column specified by the <b>SelectedItemColumn</b> property matches the value of this property will be designated as the currently selected item.  Typically, you will need to insert a <a href="#">Macro expression</a> in order to dynamically retrieve the appropriate value from the current context.	"{% currentpageinfo.documentid %}"
Transformations	Allows you to assign a <b>HierarchicalTransformations</b> object representing a <a href="#">hierarchical transformation</a> . The UniView renders the source data according to the hierarchical transformation (instead of the ItemTemplates).  <b>Note:</b> To use a hierarchical transformation, you need to set the <i>Transformations</i> property before calling the UniView's <b>DataBind</b> method.  See also: <a href="#">Displaying data using hierarchical transformations</a>	
UseNearestItemForHeaderAndFooter	Indicates whether the control provides data to the item templates (or transformations) that display the header and footer content. You can work with the data inside the code of the templates.  <ul style="list-style-type: none"> <li>Header templates use the data of the first item on the given hierarchy level.</li> <li>Footer templates use the data of the last item on the given hierarchy level.</li> </ul> <p>The control ignores this property if the <b>OuterData</b> property is set.</p>	

## Appearance and styling

You can use two different approaches to define the output format of the UniView control:

- Item templates
- A [hierarchical transformation](#) assigned through the **Transformations** property in the API (see the [Displaying data using hierarchical transformations](#) section)

You can define the following item templates within the UniView tags:

Template name	Description	Sample value
---------------	-------------	--------------

AlternatingItemTemplate	Template used for alternating items.	<pre> &lt;li&gt; &lt;font colo r=" #999 999" &gt; &lt;% # Eval ("No deName") %&gt; &lt; /font t&gt; &lt;cms : SubLevel PlaceHolder runat=" server" ID=" plcSub" /&gt; &lt; /li&gt; </pre>
FirstItemTemplate	Template for the first item on every level in the hierarchy. Only applied to levels that contain more than one item.	
FooterTemplate	Template rendered at the end of every level (after the last item on the level). Can be used to close encapsulating elements from the <b>HeaderTemplate</b> .	<pre> &lt; /ul&gt; </pre>
HeaderTemplate	Template rendered at the beginning of every level (before the first item on the level). Allows you to visually separate or style individual levels.	<pre> &lt;ul&gt; </pre>

ItemTemplate	Template used for all standard items, that are not covered by a specialized template (e.g. alternating items, first items).	<pre>&lt;li&gt; &lt;%# Eval ( "No deNa me" ) %&gt; &lt;cms : SubL evel Plac eHol der runa t=" serv er" ID=" plcS ub" /&gt; &lt; /li&gt;</pre>
LastItemTemplate	Template for the last item on every level in the hierarchy. Only applied to levels that contain more than one item.	
SeparatorTemplate	Template rendered between items on the same level. The UniView does not place the separator between items on different hierarchy levels (i.e. between a parent item and its child).	
SingleItemTemplate	Template applied in cases where there is only one item on a level in the hierarchy.	

## Setting the location of sublevels

When displaying hierarchical data, you can add a placeholder that specifies the position of sublevels inside the code of item templates:

```
<cms:SubLevelPlaceholder runat="server" ID="plcSub" />
```

For items that have descendants in the hierarchy, the placeholder is replaced by the child level under the given item (including the header and footer for the new level). If you do not add the sublevel placeholder, the system automatically renders child levels after the code of parent items.

You can add the placeholder into any type of item template (Item, AlternatingItem, FirstItem, LastItem, SingleItem).

**Note:** To use the sublevel placeholder, the **HierarchicalDisplayMode** property of the control must be set to **Inner** (this is the default state).

## Displaying data using hierarchical transformations

The following example demonstrates how to use [hierarchical transformations](#) with the UniView control. The sample scenario displays a hierarchy of [forum posts](#) from the Kentico database, based on the forum thread structure. You can use the same approach for any type of hierarchical data.

✓ **Tip:** The [CMSUniView](#) and [QueryUniView](#) controls allow you to use hierarchical transformations without writing custom code. However, the CMS controls can only display pages or other hierarchical Kentico data. The approach described below is primarily intended for displaying:

- External hierarchical data
- Customized or composite Kentico data sources

1. Create a new **Web form** in your web project.
2. Add the **UniView** control onto the form:

```
<cms:UniView ID="UniViewForumPosts" runat="server" />
```

3. Switch to the web form's code behind file and add the following references:

```
using System.Data;

using CMS.Forums;
using CMS.Helpers;
using CMS.Base;
using CMS.DocumentEngine.Web.UI;
using CMS.PortalEngine;
```

4. Set up the UniView control in the web form's **Page\_Load** method:



```
protected void Page_Load(object sender, EventArgs e)
{
    // Loads a DataSet containing all forum posts, ordered according to the
    PostLevel and PostTime
    DataSet forumPosts = ForumPostInfoProvider.GetForumPosts().OrderBy
    ("PostLevel, PostTime").Columns("PostID, PostForumID, PostParentID, PostIDPath,
    PostLevel, PostSubject, PostText, PostTime");

    // Checks that the DataSet isn't empty
    if (!DataHelper.DataSourceIsEmpty(forumPosts))
    {
        // Creates a GroupedDataSource from the forumPosts DataSet
        GroupedDataSource groupedForumPosts = new GroupedDataSource(forumPosts,
        "PostParentID", "PostLevel");

        // Assigns the grouped data source to the UniView
        UniViewForumPosts.DataSource = groupedForumPosts;

        // Sets the display mode of the UniView
        UniViewForumPosts.HierarchicalDisplayMode = HierarchicalDisplayModeEnum.
        Inner;

        // Specifies the column that the data uses as an identifier (to determine
        parent-child relationships)
        UniViewForumPosts.RelationColumnID = "PostID";

        // Gets the hierarchical transformation from the system
        TransformationInfo ti = TransformationInfoProvider.GetTransformation("CMS.
        Root.ForumPosts");

        // Checks that the transformation exists
        if (ti != null)
        {
            // Checks that the transformation is hierarchical
            if (ti.TransformationIsHierarchical)
            {
                // Stores the structure of the hierarchical transformation into a
                HierarchicalTransformations object
                HierarchicalTransformations transformation = new
                HierarchicalTransformations("PostID");
                transformation.LoadFromXML(ti.
                TransformationHierarchicalXMLDocument);

                // Assigns the hierarchical transformation to the UniView control
                UniViewForumPosts.Transformations = transformation;
            }
        }

        // Binds the UniView's data
        UniViewForumPosts.DataBind();
    }
}
```

5. Save the changes to the web form and its code behind file.
6. Define the [hierarchical transformation](#) in the Kentico administration interface (named *CMS.Root.ForumPosts*).

**i** For example, you can create the following transformation for this scenario:

- a. Open the **Page types** application.
- b. Edit the **Root** page type.
- c. Open the **Transformations** tab.
- d. Click **New hierarchical transformation**.
- e. Name the transformation *ForumPosts*.
- f. Click **Save**.

Add 4 sub-transformations:

- **Transformation type:** Item transformation
- **Page types:** All (empty)
- **Level:** 0
- **ASCX transformation code:**

```
<div style="background-color:#FFFBEB;">
  <a href="<%# ForumFunctions.GetPostURL(Eval("PostIDPath"), Eval
("PostForumID")) %>"><%# Eval("PostSubject", true) %></a>
  <p><%# HTMLEncode(LimitLength(StripTags(RemoveDynamicControls
(RemoveDiscussionMacros(Eval("PostText")))), 400)) %></p>
</div>
```

- **Transformation type:** Item transformation
- **Page types:** All (empty)
- **Level:** 1
  - **Apply to sublevels:** yes (checked)
- **ASCX transformation code:**

```
<div>
  <a href="<%# ForumFunctions.GetPostURL(Eval("PostIDPath"), Eval
("PostForumID")) %>"><%# Eval("PostSubject", true) %></a>
  <p><%# HTMLEncode(LimitLength(StripTags(RemoveDynamicControls
(RemoveDiscussionMacros(Eval("PostText")))), 400)) %></p>
</div>
```

- **Transformation type:** Header transformation
- **Level:** All (empty)
  - **Apply to sublevels:** yes (checked)
- **Transformation code:**

```
<div style="margin-left:30px">
```

- **Transformation type:** Footer transformation
- **Level:** All (empty)
  - **Apply to sublevels:** yes (checked)
- **Transformation code:**

```
</div>
```

7. Return to Visual Studio, right-click the web form in the Solution explorer and select **View in Browser**.

The page displays the forum posts according to the hierarchical transformation:

#### Products requests

In this thread, you can post requests for new products that you would like to be available in our web shop. We will monitor this thread regularly so you have a good chance for your requested products to appear in our range soon.

##### Apple iPad 2

Hi, how about the new iPad 2? I've already got the first version, but I just can't wait to the new one, it seems advanced and trendy!!!

##### RE: Apple iPad 2

Hi Trevor, good news for you, the iPad 2 is already available. You can order it right now in the Products section, under Laptops and Tablets. And the price is only \$510.99, nothing but a great deal for such a masterpiece hardware!

##### RE: Apple iPad 2

Wow, don't let me be mistaken, but you seem to be the very first web shop to have it in stock. Ordered already, can't wait for it arrive :-)))

##### More smartphones

Hi, I would like to see a wider range of smartphones in your e-shop. The current range is really impressive, but some more models would still be appreciated.

##### RE: More smartphones

Hi Jane, we are working on this at the moment. If everything goes well, you should be able to see many more models from various suppliers soon. You can expect more cost-savvy models, more manufacturers and a larger variety of operating systems.

#### Services feedback

In this thread, you can express your opinions on the services we provide. Whether you have experience with our services in web design, web development, network administration or the recently introduced consulting, any type of feedback is highly appreciated!

##### Consulting exceeding all expectations

Hi, I must say that your consultants are in a class of their own. Throughout my whole career, I have never met consultants with this level of expertise, knowledge and natural willingness to help us solve our problems. What really persuaded me of the uniqueness of the service you provide was the ability to go extra mile. They not only helped us overcome the hurdles that we ordered them for, but...