

Many pages contain components ([web parts](#) or controls) that load and display data from the Kentico database, or other external sources. For example, when displaying a page with a list of news articles, the system retrieves text from the fields of news pages stored in the database, and then formats the data on the page.

Communication with storage spaces and processing of data are common weak points in the performance of pages. **Content caching** helps the system maximize the efficiency of data. The content cache saves data loaded by page components into the application's memory. Components then re-use the cached data on future requests.

The following types of web parts and controls support content caching:

- Dedicated data sources
- Repeaters and viewers with built-in data sources
- Navigation components

## Configuring content caching

You can set up content caching on two levels:

- Globally for entire sites (all data components on the given site)
- For individual instances of web parts or controls



### **We recommend caching all possible content.**

You can ensure that components do not display outdated content by setting [cache dependencies](#). Most non-custom data sources have default dependencies that automatically clear the content cache whenever the stored data is modified.

The cache is shared between web parts with the same setup on different pages. For example, featured products listings. This can improve the performance of first page loads even when [higher-level cache](#) is used.

### **To enable content caching globally:**

1. Open the **Settings** application.
2. Select the **System -> Performance** category.
3. Type a number of minutes into the **Cache content (minutes)** setting. The value determines how long the content cache retains data, and must be greater than 0.
4. **Save** the settings.

With content caching enabled globally, the system caches the structured data of all page components by default.

### **To enable content caching for individual web part instances:**

1. Open the **Pages** application.
2. Edit the page containing the web part on the **Design** tab.
3. **Configure** the web part instance (double-click).
4. Type a number of minutes into the **Cache minutes** property (in the **System settings** category). The value determines how long the cache stores the web part's data, and must be greater than 0.

#### **Recommended settings:**

- 1 to 60 minutes, depending on the nature of the content.
  - In combination with [partial caching](#) on the same page, the interval should be about 10 times the interval set for the output cache.
5. (Optional) [Add dependencies](#) via the **Cache dependencies** property. This allows you to automatically clear the cache based on changes in the source data.
  6. Click **Save & Close**.

The system caches the data loaded by the given web part instance.

### **To disable content caching for specific web parts when content caching is enabled globally:**


1. **Configure** the web part instance on the **Design** tab.
2. Type **0** (zero) into the **Cache minutes** property (in the **System settings** category).
3. Click **Save & Close**.

The web part instance reloads data from the data source without caching.

## Setting dependencies for content cache

Cache dependencies allow the system to automatically clear cached data when related objects are modified. Web parts with data sources provide default dependencies for the content cache, and you can also add your own custom dependencies for individual instances:

1. Open the **Pages** application.
2. Edit the page containing the web part on the **Design** tab.
3. **Configure** the web part instance (double-click).
4. Add the dependencies into the **Cache dependencies** property (in the **System settings** category). For more information and examples, see [Setting cache dependencies](#).

 If you leave the **Use default cache dependencies** box checked, the system automatically clears the web part's content cache:

- Whenever the loaded data changes (depends on the web part)
- When you modify the property configuration of the given web part instance

Default dependencies do NOT cover data loaded via external or custom data sources. For example, [custom query data sources](#) only clear the content cache when the query itself changes, not the loaded data.

5. Click **Save & Close**.

The system deletes the web part's content cache whenever the specified objects change. With the cache cleared, the web part reloads the data from the source the next time a visitor opens the page.


## Sharing the content cache between components

The content cache stores the data loaded by page components (web parts or controls) under cache keys. By default, the system generates a unique cache key name for each component. The default name contains variables such as the web part ID, the name of the user viewing the page, or the code name of the language selected for the page.

If you have multiple web part instances that load exactly the same data, you can share the cached content:

1. Open the **Pages** application.
2. **Configure** one of the web parts on the **Design** tab.
3. Enter a custom key name into the **Cache item name** property (in the **System settings** category).
4. Copy the key name into the **Cache item name** property of all web part instances that load the same data.


When a visitor opens a page containing one of the web parts, the system loads the data and saves it into the cache under the specified key. While the cached content is valid, other web parts to which you assigned the same *Cache item name* retrieve the data directly from the cache. This setup optimizes loading of content from the database (or other source) and avoids redundant keys in the cache.





 **Tip:** You can use [macro expressions](#) to create dynamic cache item names based on variables such as query string parameters, or other context data.

## Caching the page output of web parts

In addition to content caching, [web parts](#) also support *Partial output caching*. The partial cache stores the full HTML output code of web part instances.

See [Caching portions of the page output](#) for more information.

 Partial caching has the following advantages and disadvantages when compared with content caching:

-  Better efficiency — the partial cache allows the system to directly send the web part's HTML output to the browser, without loading data or processing the web part at all
-  Usable by all web parts with visible output, not just web parts that load structured data
-  Not suitable for web parts whose output changes very frequently (for example lists with filtering)
-  No default dependencies on the data loaded by web parts — you need to set custom partial cache dependencies to ensure that the content of web parts is up-to-date

**Note:** Set up partial caching for the web part that actually renders the content on the page (this may not always be the same web part as the data source).