

Event categories (classes):

- [ObjectEvents](#)
- [DocumentEvents](#)
- [WorkflowEvents](#)
- [SecurityEvents](#)
- [DataProtectionEvents](#)
- [EventLogEvents](#)
- [StagingEvents](#)
- [IntegrationEvents](#)
- [ImportExportEvents](#)
- [DocumentImportExportEvents](#)
- [ColumnsTranslationEvents](#)
- [ModulePackagingEvents](#)
- [BizFormItemEvents](#)
- [CustomTableItemEvents](#)
- [EcommerceEvents](#)
- [NewsletterEvents](#)
- [WebAnalyticsEvents](#)
- [ContactManagementEvents](#)
- [ActivityEvents](#)
- [AutomationEvents](#)
- [OutputFilterEvents](#)
- [URLRewritingEvents](#)
- [SystemEvents](#)
- [SiteEvents](#)
- [SqlEvents](#)
- [Application events](#)
  - [ApplicationEvents](#)
  - [RequestEvents](#)
  - [SessionEvents](#)
- [Control and page events](#)
  - [AbstractUserControl](#)
  - [CMSPage](#)

## ObjectEvents

**Namespace:** CMS.DataEngine

Contains events that occur for all types of objects in the system. See also: [Handling object events](#)



### Important

Use the following classes to access the object events in the API:

- **ObjectEvents** - events triggered for all object types
- **<name>Info.TYPEINFO.Events** - events triggered only for a specific object type, for example: *UserInfo.TYPEINFO.Events*

Event	Event types	Handler parameters	Description
-------	-------------	--------------------	-------------

Insert	Before After Failure	ObjectEventArgs <ul style="list-style-type: none"> <li>Object (BaseInfo)</li> </ul>	Occurs when a new object is created.  To reliably handle all possible object creation events, you may also need to assign a matching handler to the <b>BulkInsert</b> event. The system triggers the <i>BulkInsert</i> event instead of <i>Insert</i> in certain cases, particularly for operations that can affect multiple objects.
Update	Before After Failure	ObjectEventArgs <ul style="list-style-type: none"> <li>Object (BaseInfo)</li> </ul>	Occurs when the data of an existing object is updated.  To reliably handle all possible object update events, you may also need to assign a matching handler to the <b>BulkUpdate</b> event. The system triggers the <i>BulkUpdate</i> event instead of <i>Update</i> in certain cases, particularly for operations that can affect multiple objects.
Delete	Before After Failure	ObjectEventArgs <ul style="list-style-type: none"> <li>Object (BaseInfo)</li> </ul>	Occurs when an object is deleted.  To reliably handle all possible object deletion events, you may also need to assign a matching handler to the <b>BulkDelete</b> event. The system triggers the <i>BulkDelete</i> event instead of <i>Delete</i> in certain cases, particularly for operations that can affect multiple objects.
BulkInsert	Before After Failure	BulkInsertEventArgs <ul style="list-style-type: none"> <li>InsertedObjects (IEnumerable&lt;IInfo&gt;)</li> <li>TypeInfo (ObjectTypeInfo)</li> </ul>	Occurs when the system performs object creation operations that can affect multiple objects. To reliably handle all possible object creation events, assign matching handlers to both the <b>BulkInsert</b> and <b>Insert</b> events.  In certain cases, the system creates objects by directly inserting data into a database table (for example when <a href="#">importing contacts</a> from CSV files). To cover these scenarios, you also need to assign an appropriate handler to the <b>SqlEvents.BulkInsert</b> event.
BulkUpdate	Before After Failure	BulkUpdateEventArgs <ul style="list-style-type: none"> <li>ChangedColumns (IEnumerable&lt;string&gt;)</li> <li>TypeInfo (ObjectTypeInfo)</li> <li>WhereCondition (IWhereCondition)</li> </ul>	Occurs when the system performs object update operations that can affect multiple objects. To reliably handle all possible object update events, assign matching handlers to both the <b>BulkUpdate</b> and <b>Update</b> events.



BulkDelete	Before After Failure	BulkDeleteEventArgs <ul style="list-style-type: none"><li>TypeInfo (ObjectTypeInfo)</li><li>WhereCondition (IWhereCondition)</li></ul>	Occurs when the system performs object deletion operations that can affect multiple objects. To reliably handle all possible object deletion events, assign matching handlers to both the <b>BulkDelete</b> and <b>Delete</b> events.
GetContent	Execute	ObjectEventArgs <ul style="list-style-type: none"><li>Content (string)</li><li>Object (BaseInfo)</li></ul>	Occurs before the system writes an object's data into <a href="#">smart search indexes</a> (when building or updating indexes).  Allows you to determine exactly how search indexes store object content.  <b>Example:</b> <a href="#">Customizing the content of search indexes</a>
LogChange	Before After Failure	LogObjectChangeEventArgs <ul style="list-style-type: none"><li>Settings (LogObjectChangeSettings)</li></ul>	Occurs when the system logs object changes for the purposes of staging and integration. For example, allows you to exclude specific objects from staging.  <b>Example:</b> <a href="#">Excluding content from staging and integration</a>
CheckPermissions	Before After Failure	ObjectSecurityEventArgs <ul style="list-style-type: none"><li>Object (BaseInfo)</li><li>Permission (PermissionsEnum)</li><li>User (UserInfo)</li><li>SiteName (string)</li><li>Result (AuthorizationResultEnum)</li></ul>	Occurs when the system checks <a href="#">permissions</a> for objects of the given type.



Sort	Before After Failure	<p>ObjectSortEventArgs</p> <ul style="list-style-type: none"> <li>• Object (BaseInfo)</li> <li>• Ascending (bool)</li> <li>• OrderColumn (string)</li> <li>• NameColumn (string)</li> </ul>	Occurs when sorting of objects is requested.
ChangeOrder	Before After Failure	<p>ObjectChangeOrderEventArgs</p> <ul style="list-style-type: none"> <li>• Object (BaseInfo)</li> <li>• NewOrder (int)</li> <li>• RelativeOrder (int)</li> <li>• OrderColumn (string)</li> </ul>	Occurs when a change in the order of objects is requested.
GetData	Before After Failure	<p>ObjectDataEventArgs</p> <ul style="list-style-type: none"> <li>• Data (DataSet)</li> <li>• Query (IObjectQuery)</li> <li>• TotalRecords (int)</li> </ul>	<p>Occurs when the system retrieves object data using the ObjectQuery API.</p> <p>The <b>Before</b> event type is suitable, for example, when you want to retrieve data from an external source. Assign the external data into the <b>Data</b> property of the handler's <i>ObjectDataEventArgs</i> parameter, which is empty by default in the Before event.</p> <p>Use the <b>After</b> event type if you need to edit the data retrieved from Kentico in the <b>Data</b> property.</p> <p>The <b>TotalRecords</b> property provides a number of total records in case paging is used. Use -1 to use the rows count of the <b>Data</b> property.</p>

[> Back to list of event categories](#)

## DocumentEvents

**Namespace:** CMS.DocumentEngine

Contains events fired when changes are made to pages.


**Note:**

Pages under [workflow](#) require a different approach to event handling. Use the events from the [WorkflowEvents](#) class instead of *DocumentEvents*.

For example, to handle the 'update' action for a page under workflow, use the *SaveVersion* event from *WorkflowEvents* instead of the *DocumentEvents.Update* event.

Event	Event types	Handler parameters	Description
Insert	Before After Failure	DocumentEventArgs <ul style="list-style-type: none"> <li>TargetParentNodeID (int)</li> <li>Node (TreeNode)</li> <li>TreeProvider (TreeProvider)</li> </ul>	Occurs when a new page is created.
InsertNewCulture	Before After Failure	DocumentEventArgs <ul style="list-style-type: none"> <li>Node (TreeNode)</li> <li>TreeProvider (TreeProvider)</li> </ul>	Occurs when a new culture version is created for a page.
InsertLink	Before After Failure	DocumentEventArgs <ul style="list-style-type: none"> <li>TargetParentNodeID (int)</li> <li>Node (TreeNode)</li> <li>TreeProvider (TreeProvider)</li> </ul>	Occurs when creating new <a href="#">linked pages</a> .
Update	Before After Failure	DocumentEventArgs <ul style="list-style-type: none"> <li>Node (TreeNode)</li> <li>TreeProvider (TreeProvider)</li> </ul>	Occurs when an existing page is updated.
Move	Before After Failure	DocumentEventArgs <ul style="list-style-type: none"> <li>TargetParentNodeID (int)</li> <li>Node (TreeNode)</li> <li>TreeProvider (TreeProvider)</li> </ul>	Occurs when a page is moved to a different position in the site's content tree.



Copy	Before After Failure	DocumentEventArgs <ul style="list-style-type: none"><li>• TargetParentNodeD (int)</li><li>• IncludeChildren (bool)</li><li>• Node (TreeNode)</li><li>• TreeProvider (TreeProvider)</li></ul>	Occurs when a new copy of a page is created.
Delete	Before After Failure	DocumentEventArgs <ul style="list-style-type: none"><li>• Node (TreeNode)</li><li>• TreeProvider (TreeProvider)</li></ul>	Occurs when a page is deleted.
GetContent	Execute	DocumentSearchEventArgs <ul style="list-style-type: none"><li>• Content (string)</li><li>• Node (TreeNode)</li><li>• Settings (SearchIndexSettingsInfo)</li><li>• IndexInfo (ISearchIndexInfo)</li><li>• SearchDocument (ISearchDocument)</li><li>• IsCrawler (bool)</li></ul>	<p>Occurs before the system writes a page's data into a Smart search index (when building or updating indexes).</p> <p>Allows you to determine exactly how search indexes store page content.</p> <p><b>Examples:</b> <a href="#">Customizing the content of search indexes</a></p>
LogChange	Before After Failure	LogDocumentChangeEventArgs <ul style="list-style-type: none"><li>• Settings (LogDocumentChangeSettings)</li></ul>	<p>Occurs when the system logs page changes for the purposes of staging and integration. For example, allows you to exclude specific pages from staging.</p> <p><b>Example:</b> <a href="#">Excluding content from staging and integration</a></p>
CheckPermissions	Before After Failure	DocumentSecurityEventArgs <ul style="list-style-type: none"><li>• Node (TreeNode)</li><li>• Permission (PermissionsEnum)</li><li>• User (UserInfo)</li><li>• SiteName (string)</li><li>• Result (AuthorizationResultEnum)</li></ul>	<p>Occurs when the system checks <a href="#">permissions</a> for pages.</p> <p><b>Note:</b> This event does not occur on the live site. For checking permissions on the live site, see the <a href="#">AuthorizeDocument</a> event.</p>



Sort	Before After Failure	DocumentSortEventArgs <ul style="list-style-type: none"> <li>ParentNodeId (int)</li> <li>Ascending (bool)</li> <li>SortType (DocumentSortEnum)</li> </ul>	Occurs when sorting of pages is requested.
ChangeOrder	Before After Failure	DocumentChangeOrderEventArgs <ul style="list-style-type: none"> <li>Node (TreeNode)</li> <li>NewOrder (int?)</li> <li>RelativeOrder (bool)</li> <li>OrderType (DocumentOrderEnum)</li> </ul>	Occurs when a change in the order of pages is requested.
GetData	Before After Failure	DocumentDataEventArgs <ul style="list-style-type: none"> <li>Data (DataSet)</li> <li>Query (IDocumentQuery)</li> <li>TotalRecords (int)</li> </ul>	<p>Occurs when the system retrieves page data using the DocumentQuery API.</p> <p>The <b>Before</b> event type is suitable, for example, when you want to retrieve data from an external source. Assign the external data into the <b>Data</b> property of the handler's <i>DocumentDataEventArgs</i> parameter, which is empty by default in the Before event.</p> <p>Use the <b>After</b> event type, when you want to edit the data retrieved from Kentico in the <b>Data</b> property.</p> <p>The <b>TotalRecords</b> property provides a number of total records in case paging is used. Use -1 to use rows count of the <b>Data</b> property.</p>

#### Page attachments

SaveAttachment	Before After Failure	DocumentEventArgs <ul style="list-style-type: none"> <li>Attachment (DocumentAttachment)</li> <li>Node (TreeNode)</li> <li>TreeProvider (TreeProvider)</li> </ul>	Occurs when a file attachment is added to a page or updated.
DeleteAttachment	Before After Failure	DocumentEventArgs <ul style="list-style-type: none"> <li>Attachment (DocumentAttachment)</li> <li>Node (TreeNode)</li> <li>TreeProvider (TreeProvider)</li> </ul>	Occurs when a file attachment is removed from a page.

#### Page security

Authorize Document	Execute	DocumentAuthorizationEventArgs <ul style="list-style-type: none"> <li>• User (UserInfo)</li> <li>• Document (TreeNode)</li> <li>• PermissionName (string)</li> <li>• AuthorizationResult (AuthorizationResultEnum)</li> </ul>	<p>Occurs when the system performs a security check for the permissions of a specific page.</p> <p>You can modify the result of the security check by adding your own conditions and setting the <b>AuthorizationResult</b> property of the handler's <i>DocumentAuthorizationEventArgs</i> parameter.</p>
FilterDataSetByPermissions	Execute	DocumentAuthorizationEventArgs <ul style="list-style-type: none"> <li>• User (UserInfo)</li> <li>• Data (DataSet)</li> <li>• PermissionName (string)</li> </ul>	<p>Occurs when the system filters a set of pages according to the page permissions of a user.</p> <p>You can modify the filtering by adding your own conditions and modifying the page data in the <b>Data</b> property of the handler's <i>DocumentAuthorizationEventArgs</i> parameter.</p>

[> Back to list of event categories](#)

## WorkflowEvents

**Namespace:** CMS.DocumentEngine

Contains events related to [workflow](#) and [versioning](#) actions.

Event	Event types	Handler parameters	Description
Approve	Before After Failure	WorkflowEventArgs <ul style="list-style-type: none"> <li>• Document (TreeNode)</li> <li>• PreviousStep (WorkflowStepInfo) - only in the <i>After</i> event</li> <li>• TreeProvider (TreeProvider)</li> </ul>	Occurs when a page is approved to the next step.
Reject	Before After Failure	WorkflowEventArgs <ul style="list-style-type: none"> <li>• Document (TreeNode)</li> <li>• PreviousStep (WorkflowStepInfo) - only in the <i>After</i> event</li> <li>• TreeProvider (TreeProvider)</li> </ul>	Occurs when a page is rejected back to a previous step





Publish	Before After Failure	<p>WorkflowEventArgs</p> <ul style="list-style-type: none"> <li>Document (TreeNode)</li> <li>PublishedDocument(TreeNode) - only in the <i>After</i> event</li> <li>PreviousStep (WorkflowStepInfo) - only in the <i>After</i> event</li> <li>TreeProvider (TreeProvider)</li> </ul>	Occurs when a page reaches the published step.
Archive	Before After Failure	<p>WorkflowEventArgs</p> <ul style="list-style-type: none"> <li>Document (TreeNode)</li> <li>PreviousStep (WorkflowStepInfo) - only in the <i>After</i> event</li> <li>TreeProvider (TreeProvider)</li> </ul>	Occurs when a page is archived.
Action	Before After Failure	<p>WorkflowActionEventArgs&lt;TreeNode, BaseInfo, WorkflowActionEnum&gt;</p> <p>Requires references to:</p> <ul style="list-style-type: none"> <li>CMS.WorkflowEngine</li> <li>CMS.DataEngine</li> </ul>	Occurs when executing <a href="#">advanced workflow action steps</a> .
Page versioning			
SaveVersion	Before After Failure	<p>WorkflowEventArgs</p> <ul style="list-style-type: none"> <li>Document (TreeNode)</li> <li>VersionHistory (VersionHistoryInfo)</li> <li>VersionNumber (string)</li> <li>VersionComment (string)</li> <li>TreeProvider (TreeProvider)</li> </ul>	Occurs when the edited version of a page is updated.
SaveAttachmentVersion	Before After Failure	<p>WorkflowEventArgs</p> <ul style="list-style-type: none"> <li>Document (TreeNode)</li> <li>Attachment (DocumentAttachment)</li> <li>TreeProvider (TreeProvider)</li> </ul>	Occurs when an attachment is updated or inserted for the edited version of a page.
RemoveAttachmentVersion	Before After Failure	<p>WorkflowEventArgs</p> <ul style="list-style-type: none"> <li>Document (TreeNode)</li> <li>Attachment (DocumentAttachment)</li> <li>TreeProvider (TreeProvider)</li> </ul>	Occurs when an attachment is removed for the edited version of a page.
Content locking			
CheckOut	Before After Failure	<p>WorkflowEventArgs</p> <ul style="list-style-type: none"> <li>Document (TreeNode)</li> <li>TreeProvider (TreeProvider)</li> </ul>	Occurs when a page is checked-out.

CheckIn	Before After Failure	WorkflowEventArgs <ul style="list-style-type: none"> <li>Document (TreeNode)</li> <li>TreeProvider (TreeProvider)</li> </ul>	Occurs when a page is checked-in.
UndoCheckOut	Before After Failure	WorkflowEventArgs <ul style="list-style-type: none"> <li>Document (TreeNode)</li> <li>TreeProvider (TreeProvider)</li> </ul>	Occurs when a user performs the undo check-out action.

[> Back to list of event categories](#)

## SecurityEvents

**Namespace:** CMS.Membership

Contains security and authentication events.

Event	Event types	Handler parameters	Description
Authenticate	Execute	AuthenticationEventArgs <ul style="list-style-type: none"> <li>User (UserInfo)</li> <li>UserName (string)</li> <li>Password (string)</li> <li>SiteName (string)</li> </ul>	Occurs when authenticating users.  <b>See also:</b> <a href="#">Handling custom external authentication</a>
AuthenticationRequested	Execute	AuthenticationRequestEventArgs <ul style="list-style-type: none"> <li>RequestedUrl (string)</li> </ul>	Occurs when a user accesses a restricted section of Kentico and the user's authentication is required.
MultiFactorAuthenticate	Execute	AuthenticationEventArgs <ul style="list-style-type: none"> <li>User (UserInfo)</li> <li>Passcode (string)</li> <li>UserName (string)</li> </ul>	Occurs when a user tries to sign in to Kentico with <a href="#">multi-factor authentication</a> enabled.  <b>See also:</b> <a href="#">Handling custom multi-factor authentication</a>
AuthorizeResource	Execute	AuthorizationEventArgs <ul style="list-style-type: none"> <li>User (UserInfo)</li> <li>ResourceName (string)</li> <li>PermissionName (string)</li> <li>Authorized (bool)</li> </ul>	Occurs when checking the permissions for modules.
AuthorizeClass	Execute	AuthorizationEventArgs <ul style="list-style-type: none"> <li>User (UserInfo)</li> <li>ClassName (string)</li> <li>PermissionName (string)</li> <li>Authorized (bool)</li> </ul>	Occurs when performing security checks for object type or page type permissions.

AuthorizeUIElement	Execute	AuthorizationEventArgs <ul style="list-style-type: none"> <li>User (UserInfo)</li> <li>ElementName (string)</li> <li>PermissionName (string)</li> <li>Authorized (bool)</li> </ul>	Occurs when checking permissions for UI elements.
SignOut	Before After Failure	SignOutEventArgs <ul style="list-style-type: none"> <li>User (UserInfo)</li> <li>SignOutUrl (string)</li> </ul>	Occurs when a user signs out from a website.

[> Back to list of event categories](#)

## DataProtectionEvents

**Namespace:** CMS.DataProtection

Contains events related to protection of personal data in the system.

Event	Event types	Handler parameters	Description
RevokeConsentAgreement	Execute	RevokeConsentAgreementEventArgs <ul style="list-style-type: none"> <li>ConsentAgreement (ConsentAgreementInfo)</li> <li>Consent (ConsentInfo)</li> <li>Contact (ContactInfo)</li> </ul>	Occurs when a consent agreement is revoked. When a contact's consent agreement is revoked, this event can be used to perform custom actions, such as terminating tracking of the contact or deleting <a href="#">personal data</a> related to the contact.  <b>See also:</b> <a href="#">Revoking consent agreements</a> , <a href="#">Working with consents</a>

[> Back to list of event categories](#)

## EventLogEvents

**Namespace:** CMS.EventLog

Contains events that occur when logging records into the system's [event log](#).

Event	Event types	Handler parameters	Description
LogEvent	Before After Failure	LogEventArgs <ul style="list-style-type: none"> <li>Event (EventLogInfo)</li> </ul>	Occurs when the system logs a record into the event log. You can use the <i>Before</i> event to customizing the data logged for events or cancel logging for certain types of events. The <i>After</i> event allows you to perform custom actions after events are successfully logged, for example duplicate certain types of events into a custom log.  <b>See also:</b> <a href="#">Customizing event logging</a>

[> Back to list of event categories](#)

## StagingEvents

**Namespace:** CMS.Synchronization

Contains events that occur when using [content staging](#) for pages and objects.

Event	Event types	Handler parameters	Description
GetChildProcessingType	Execute	StagingChildProcessingTypeEventArgs <ul style="list-style-type: none"> <li>ObjectType (string)</li> <li>ParentObjectType (string)</li> <li>ProcessingType (IncludeToParentEnum)</li> </ul>	<p>Occurs on the <b>target server</b> when processing create or update staging tasks that include the data of child or binding objects (once for each child or binding type).</p> <p>For example, staging tasks for roles include the data of binding objects, such as user-role and role-permission relationships. When processing a create or update staging task for a role, the <i>GetChildProcessingType</i> event triggers separately for each binding type included in the role data.</p> <p>You can use the event to customize how the system handles child and binding object data during staging. For more information, see <a href="#">Customizing staging of child and binding objects</a>.</p>
LogTask	Before After Failure	StagingLogTaskEventArgs <ul style="list-style-type: none"> <li>Object (BaseInfo)</li> <li>Task (StagingTaskInfo)</li> </ul>	<p>Occurs when creating staging tasks on the source server after a page or object is changed in the system.</p> <p>See also:</p> <ul style="list-style-type: none"> <li><a href="#">Excluding content from staging and integration</a></li> <li><a href="#">Automatically synchronizing staging and integration tasks</a></li> </ul>
Synchronize	Before After Failure	CMSEventArgs	Occurs on the source server when synchronizing staging tasks – once per synchronize action, even when processing multiple tasks.
SynchronizeTask	Before After Failure	StagingTaskEventArgs <ul style="list-style-type: none"> <li>Task (StagingTaskInfo)</li> </ul>	Occurs on the source server when synchronizing staging tasks – separately for every processed task.

ProcessTask	Before After Failure	StagingSynchronizationEventArgs <ul style="list-style-type: none"> <li>TaskType (TaskTypeEnum)</li> <li>ObjectType (string)</li> <li>TaskData (DataSet)</li> <li>TaskBinaryData (DataSet)</li> <li>TaskHandled (bool)</li> <li>SyncManager (ISyncManager)</li> </ul>	Occurs on the <b>target server</b> when processing incoming staging tasks.  The processing of the staging tasks then also triggers the corresponding <b>DocumentEvents</b> or <b>ObjectEvents</b> (Insert, Update, Delete, etc.).
-------------	----------------------------	--	---

[> Back to list of event categories](#)

## IntegrationEvents

**Namespace:** CMS.Synchronization

Contains events that occur when synchronizing pages or objects between Kentico and other applications using the [integration bus](#).

See also:

- [Excluding content from staging and integration](#)
- [Automatically synchronizing staging and integration tasks](#)

Event	Event types	Handler parameters	Description
LogInternalTask	Before After Failure	IntegrationTaskEventArgs <ul style="list-style-type: none"> <li>Task (IntegrationTaskInfo)</li> <li>Object (BaseInfo)</li> </ul>	Occurs when creating <b>outgoing</b> integration tasks after a page or object in the system is changed.
LogExternalTask	Before After Failure	IntegrationTaskEventArgs <ul style="list-style-type: none"> <li>Task (IntegrationTaskInfo)</li> <li>Object (BaseInfo)</li> </ul>	Occurs when the system creates <b>incoming</b> tasks provided by an integration connector.

[> Back to list of event categories](#)

## ImportExportEvents

**Namespace:** CMS.CMSImportExport

Contains events related to [import and export](#) actions.

Event	Event types	Handler parameters	Description
Import	Before After Failure	ImportBaseEventArgs <ul style="list-style-type: none"> <li>Settings (SiteImportSettings)</li> </ul>	Occurs at the start of the import process.
ImportObject	Before After Failure	ImportEventArgs <ul style="list-style-type: none"> <li>Settings (SiteImportSettings)</li> <li>Object (BaseInfo)</li> <li>ParentObject (BaseInfo)</li> <li>Parameters (ImportParameters)</li> </ul>	Occurs once for every imported object.
ImportObjects	Before After Failure	ImportDataEventArgs <ul style="list-style-type: none"> <li>Settings (SiteImportSettings)</li> <li>Data (DataSet)</li> <li>ObjectType (string)</li> <li>SiteObjects (bool)</li> <li>ParentImported (bool)</li> </ul>	Occurs when importing sets of objects.
ImportBinding	Before After Failure	ImportEventArgs <ul style="list-style-type: none"> <li>Settings (SiteImportSettings)</li> <li>Object (BaseInfo)</li> <li>ParentObject (BaseInfo)</li> <li>Parameters (ImportParameters)</li> </ul>	Occurs once for every imported binding object.
ImportBindings	Before After Failure	ImportEventArgs <ul style="list-style-type: none"> <li>Settings (SiteImportSettings)</li> <li>Object (BaseInfo)</li> <li>ParentObject (BaseInfo)</li> <li>Parameters (ImportParameters)</li> </ul>	Occurs when importing binding objects.

ImportChild	Before After Failure	ImportEventArgs <ul style="list-style-type: none"> <li>Settings (SiteImportSettings)</li> <li>Object (BaseInfo)</li> <li>ParentObject (BaseInfo)</li> <li>Parameters (ImportParameters)</li> </ul>	Occurs once for every imported child object.
ImportChildren	Before After Failure	ImportEventArgs <ul style="list-style-type: none"> <li>Settings (SiteImportSettings)</li> <li>Object (BaseInfo)</li> <li>ParentObject (BaseInfo)</li> <li>Parameters (ImportParameters)</li> </ul>	Occurs when importing child objects.
GetImportData	Before After Failure	ImportGetDataEventArgs <ul style="list-style-type: none"> <li>Settings (SiteImportSettings)</li> <li>Data (DataSet)</li> <li>Object (BaseInfo)</li> <li>ObjectType (string)</li> <li>SiteObjects (bool)</li> <li>SelectionOnly (bool)</li> </ul>	Occurs when importing data. Allows you to work with the data set in the import package created for a specific object types. <ul style="list-style-type: none"> <li>Use the <b>Before</b> event type to modify the data in the import package (for example add data from an external source).</li> <li>Use the <b>After</b> event type to modify the data after the import package has been loaded.</li> </ul>
ProcessFileOperation	Before After Failure	ImportProcessFileOperationEventArgs <ul style="list-style-type: none"> <li>Operation (FileOperation)</li> <li>Settings (SiteImportSettings)</li> </ul>	Occurs when processing file/folder operations in the import package.
ImportCanceled	Execute	ImportBaseEventArgs <ul style="list-style-type: none"> <li>Settings (SiteImportSettings)</li> </ul>	Occurs when a user cancels the import process.
ImportError	Execute	ImportErrorEventArgs <ul style="list-style-type: none"> <li>Exception (Exception)</li> <li>Settings (SiteImportSettings)</li> </ul>	Fires when an error occurs during the import process.

Event	Event types	Handler parameters	Description
-------	-------------	--------------------	-------------

Export	Before After Finished Failure	ExportEventArgs <ul style="list-style-type: none"> <li>Settings (SiteExportSettings)</li> <li>SiteObjects (bool)</li> </ul>	Occurs at the start of the export process.
ExportObjects	Before After Failure	ExportEventArgs <ul style="list-style-type: none"> <li>Settings (SiteExportSettings)</li> <li>Data (DataSet)</li> <li>ObjectType (string)</li> <li>SiteObjects (bool)</li> </ul>	Occurs when exporting objects.
GetExportData	Before After Failure	ExportGetDataEventArgs <ul style="list-style-type: none"> <li>Settings (SiteExportSettings)</li> <li>Data (DataSet)</li> <li>ObjectType (string)</li> <li>WhereCondition (string)</li> <li>ChildData (bool)</li> <li>SelectionOnly (bool)</li> <li>Parameters (QueryDataParameters)</li> </ul>	Occurs when exporting data. Allows you to work with the data set in the export package created for specific object types. <ul style="list-style-type: none"> <li>Use the <b>Before</b> event type to extract data from the export package.</li> <li>Use the <b>After</b> event type to modify the data in the export package.</li> </ul>
SingleExportSelection	Execute	SingleExportSelectionEventArgs <ul style="list-style-type: none"> <li>Settings (SiteExportSettings)</li> <li>InfoObject (BaseInfo)</li> <li>SelectedObject (List&lt;string&gt;)</li> </ul>	Occurs when selecting related objects during single object export. Allows you to attach related objects to the export package.
ExportCanceled	Execute	ExportBaseEventArgs <ul style="list-style-type: none"> <li>Settings (SiteExportSettings)</li> </ul>	Occurs when a user cancels the export process.



ExportError	Execute	ExportErrorEventArgs <ul style="list-style-type: none"> <li>Exception (Exception)</li> <li>Settings (SiteExportSettings)</li> </ul>	Fires when an error occurs during the export process.
-------------	---------	---	---

[> Back to list of event categories](#)

## DocumentImportExportEvents

**Namespace:** CMS.DocumentEngine

Contains events related to [import and export](#) of pages.

Event	Event types	Handler parameters	Description
ExportDocuments	Before After Failure	DocumentsExportEventArgs <ul style="list-style-type: none"> <li>Settings (SiteExportSettings)</li> <li>Data (DataSet)</li> <li>ObjectType (string)</li> <li>SiteObjects (bool)</li> </ul>	Occurs when exporting pages.
ImportDocument	Before After Failure	DocumentImportEventArgs <ul style="list-style-type: none"> <li>Document (TreeNode)</li> <li>Settings (SiteImportSettings)</li> </ul>	Occurs once for every imported page.
ImportDocuments	Before After Failure	DocumentImportEventArgs <ul style="list-style-type: none"> <li>Document (TreeNode)</li> <li>Settings (SiteImportSettings)</li> </ul>	Occurs when importing pages.

[> Back to list of event categories](#)

## ColumnsTranslationEvents

**Namespace:** CMS.DataEngine

When transferring data between instances of Kentico (via [import/export](#) or [staging](#)), the IDs of objects can differ between the environments. Kentico automatically "translates" the ID values for system objects, but not for custom objects. For example, when you stage a [custom table](#) or [page type](#) that contains a custom field storing object IDs, the values may not match the IDs of the objects on the target server. The *ColumnsTranslationEvents* allow you to ensure that the IDs are translated correctly.

**See also:** [Deploying objects with custom ID fields](#)

Event	Event types	Handler parameters	Description
-------	-------------	--------------------	-------------

RegisterRecords	Execute	ColumnsTranslationEventArgs <ul style="list-style-type: none"> <li>• ObjectType (string)</li> <li>• Data (IDataContainer)</li> <li>• TranslationHelper (TranslationHelper)</li> </ul>	Occurs when exporting objects. Allows you to provide additional information for the translation of custom ID fields.
TranslateColumns	Execute	ColumnsTranslationEventArgs <ul style="list-style-type: none"> <li>• ObjectType (string)</li> <li>• Data (IDataContainer)</li> <li>• TranslationHelper (TranslationHelper)</li> </ul>	Occurs when importing objects. Allows you to translate the values of custom ID fields using the provided information.

[> Back to list of event categories](#)

## ModulePackagingEvents

**Namespace:** CMS.Modules

Contains events that occur when the system generates [installation packages for custom modules](#).

Event	Event types	Handler parameters	Description
BuildNuSpecManifest	Before After Failure	BuildNuSpecManifestEventArgs <ul style="list-style-type: none"> <li>• Manifest (NuGet.Manifest)</li> <li>• NuSpecBuilder (CMS.Modules.NuSpecBuilder)</li> <li>• ResourceName (string)</li> </ul>	<p>Occurs when the system creates the <i>.nuspec</i> file for module NuGet packages. The file is an XML manifest containing package metadata. See the <a href="#">.nuspec reference</a> for more information.</p> <p>The event allows you to customize the package metadata for your modules:</p> <ul style="list-style-type: none"> <li>• The <b>Before</b> event type allows you to adjust the default nuspec content via the Kentico API – use the members of the <i>NuSpecBuilder</i> instance provided by the corresponding property of the handler's <i>BuildNuSpecManifestEventArgs</i> parameter.</li> <li>• Use the <b>After</b> event type if you wish to modify or completely replace the resulting nuspec manifest, which is accessible as an instance of the <i>NuGet.Manifest</i> class via the corresponding property of the handler's <i>BuildNuSpecManifestEventArgs</i> parameter.</li> </ul> <p>The <b>ResourceName</b> property of the handler's <i>BuildNuSpecManifestEventArgs</i> parameter contains the code name of the module for which the package is being created.</p>

[> Back to list of event categories](#)

## BizFormItemEvents

**Namespace:** CMS.OnlineForms

Contains events related to [form](#) actions. See [Working with form data using the API](#) for an example.



Event	Event types	Handler parameters	Description
Insert	Before After Failure	BizFormItemEventArgs <ul style="list-style-type: none"><li>Item (BizFormItem)</li></ul>	Occurs when inserting new form items (data records of on-line forms).
Update	Before After Failure	BizFormItemEventArgs <ul style="list-style-type: none"><li>Item (BizFormItem)</li></ul>	Occurs when updating form data records.
Delete	Before After Failure	BizFormItemEventArgs <ul style="list-style-type: none"><li>Item (BizFormItem)</li></ul>	Occurs when deleting data records from forms.

[> Back to list of event categories](#)

## CustomTableItemEvents

**Namespace:** CMS.CustomTables

Contains events related to custom table actions. See [Custom table internals and API](#).

Event	Event types	Handler parameters	Description
Insert	Before After Failure	CustomTableItemEventArgs <ul style="list-style-type: none"><li>Item (CustomTableItem)</li></ul>	Occurs when inserting new custom table items (data records stored in a custom table).
Update	Before After Failure	CustomTableItemEventArgs <ul style="list-style-type: none"><li>Item (CustomTableItem)</li></ul>	Occurs when updating custom table items.
Delete	Before After Failure	CustomTableItemEventArgs <ul style="list-style-type: none"><li>Item (CustomTableItem)</li></ul>	Occurs when deleting custom table items.

[> Back to list of event categories](#)

## EcommerceEvents

**Namespace:** CMS.Ecommerce

Contains events related to the E-commerce solution.

Event	Event types	Handler parameters	Description
-------	-------------	--------------------	-------------

NewOrderCreated	Execute	NewOrderCreatedEventArgs <ul style="list-style-type: none"> <li>NewOrder (OrderInfo)</li> </ul>	<p>Fires when a new <a href="#">order</a> is created.</p> <p>The event triggers only when the order is created based on an existing shopping cart. For example, the event does not trigger if an order is created within code using the <i>OrderInfoProvider</i> API.</p>
OrderPaid	Execute	OrderPaidEventArgs <ul style="list-style-type: none"> <li>Order (OrderInfo)</li> </ul>	Fires when an order becomes paid.

[> Back to list of event categories](#)

## NewsletterEvents

**Namespace:** CMS.Newsletters

Contains events related to [Email marketing](#).

Event	Event types	Handler parameters	Description
ResolveMacros	Before After Failure	ResolveMacrosEventArgs <ul style="list-style-type: none"> <li>MacroResolver (MacroResolver)</li> <li>TextToResolve (string)</li> <li>Newsletter (NewsletterInfo)</li> <li>IssueInfo (IssueInfo)</li> </ul>	Occurs when all email feed text macros are resolved.
GenerateQueueItems	Before After Failure	GenerateQueueItemsEventArgs <ul style="list-style-type: none"> <li>Subscriber (SubscriberInfo)</li> <li>Issue (IssueInfo)</li> <li>EmailAddressBlocker (IEmailAddressBlocker)</li> <li>GeneratedEmails (HashSet&lt;string&gt;)</li> </ul>	<p>Occurs when a campaign email is sent to one or more recipients. You can use this event to manually add emails to the queue for custom recipients.</p> <p>Note: The <i>GeneratedEmails</i> property is used only with Newsletter email feeds and its primary purpose is to check duplicates in the newsletter recipients.</p>
GeneratePreview	Before After Failure	GeneratePreviewEventArgs <ul style="list-style-type: none"> <li>Subscriber (SubscriberInfo)</li> <li>Issue (IssueInfo)</li> <li>PreviewHtml (string)</li> </ul>	Occurs when an email preview is generated for a recipient.



SubscriberUn subscribes	Execute	UnsubscriptionEventArgs <ul style="list-style-type: none"> <li>Newsletter (NewsletterInfo)</li> <li>Email (string)</li> <li>IssueID (int?)</li> </ul>	Occurs when a recipient unsubscribes from an email feed.
SubscriberOp ensEmail	Execute	LinksEventArgs <ul style="list-style-type: none"> <li>IssueInfo (IssueInfo)</li> <li>NewsletterInfo (NewsletterInfo)</li> <li>AdditionalParameters (NameValueCollection)</li> </ul>	Occurs when a recipient opens an email and the open email tracking feature is enabled.
SubscriberCli cksTrackedLi nk	Execute	LinksEventArgs <ul style="list-style-type: none"> <li>IssueInfo (IssueInfo)</li> <li>NewsletterInfo (NewsletterInfo)</li> <li>AdditionalParameters (NameValueCollection)</li> </ul>	Occurs when a recipient clicks an email link and the email link tracking feature is enabled.

[> Back to list of event categories](#)

## WebAnalyticsEvents

**Namespace:** CMS.WebAnalytics

Event	Event types	Handler parameters	Description
CheckAn alyticsC onsent	Execute	CheckAnalyticsC onsentEventArgs <ul style="list-style-type: none"> <li>ContactId (int)</li> <li>HasConsent (bool)</li> </ul>	Occurs before the system logs any <a href="#">web analytics</a> for visitors (before any processing of the visitor's personal data).  Allows you to disable logging of web analytics based on consent. Assign a handler to the event, create a condition according to your website's consent requirements, and set the <b>HasConsent</b> property of the handler's <b>CheckAnalyticsConsentEventArgs</b> parameter based on the result.  <b>Example:</b> <a href="#">Logging web analytics based on consent</a>
LogConv ersion	Before After Failure	CMSEventArgs<L ogRecord>	Occurs when a custom conversion is logged. You can access the data of the custom conversion via the <i>Parameter</i> property of the <i>CMSEventArgs</i> parameter.
ProcessL ogRecord	Before After Failure	CMSEventArgs<L ogRecord>	Occurs when the web analytics log is processed.



Process Analytics Service	Before After Failure	AnalyticsJSEvent Args <ul style="list-style-type: none"> <li>QueryParameters (Dictionary&lt;string, string&gt;)</li> </ul>	Occurs when the system logs any web analytics.
InsertAnalyticsJS	Before After Failure	AnalyticsJSEvent Args <ul style="list-style-type: none"> <li>QueryParameters (Dictionary&lt;string, string&gt;)</li> </ul>	Occurs when the web analytics JavaScript snippet is inserted into the page.
CampaignLaunched	Execute	CMSEventArgs<CampaignInfo>	Occurs when a campaign is launched (whether it was originally scheduled or launched directly).
CampaignUTMChanged	Execute	CMSEventArgs<CampaignUTMChangedEventArgs>	Occurs when the UTM parameter is changed for a campaign.

[> Back to list of event categories](#)

## ContactManagementEvents

**Namespace:** CMS.ContactManagement

Event	Event types	Handler parameters	Description
ContactInfosDeleted	Execute	ContactInfosDeletedHandlerEventArgs <ul style="list-style-type: none"> <li>DeletedContactIds (IEnumerable&lt;int&gt;)</li> <li>DeletedContactGUIDs (IEnumerable&lt;Guid&gt;)</li> </ul>	Occurs when two or more selected contacts are deleted.  When a single contact is deleted, the <i>TypeInfoEvents.Delete</i> event can be used on <i>ContactInfo.TYPEINFO</i> .

[> Back to list of event categories](#)

## ActivityEvents

**Namespace:** CMS.Activities

Event	Event types	Handler parameters	Description
ActivityBulkInsertPerformed	Before After Failure	CMSEventArgs<List<ActivityInfo>>	Fires when the system processes <a href="#">on-line marketing activities</a> that are logged in memory, and bulk inserts them to the database.

ActivityProcessedInLogService	Execute	CMSEventArgs<IActivityInfo>	Fires when an <a href="#">on-line marketing activity</a> is logged and processed. At this point, the activity is only stored in the application memory (activities are regularly processed and saved in bulk to the database – see the <i>ActivityBulkInsertPerformed</i> event).
-------------------------------	---------	-----------------------------	---

[> Back to list of event categories](#)

## AutomationEvents

**Namespace:** CMS.Automation

Contains events related to [marketing automation](#) actions.

Event	Event types	Handler parameters	Description
MoveToNextStep	Before After Failure	AutomationEventArgs <ul style="list-style-type: none"> <li>InfoObject (BaseInfo)</li> <li>StateObject (AutomationStateInfo)</li> <li>PreviousStep (WorkflowStepInfo)</li> </ul>	Occurs when an automation process moves to the next step.
MoveToPreviousStep	Before After Failure	AutomationEventArgs <ul style="list-style-type: none"> <li>InfoObject (BaseInfo)</li> <li>StateObject (AutomationStateInfo)</li> <li>PreviousStep (WorkflowStepInfo)</li> </ul>	Occurs when an automation process moves back to the previous step.
Action	Before After Failure	WorkflowActionEventArgs<BaseInfo, AutomationStateInfo, AutomationActionEnum>  Requires references to: <ul style="list-style-type: none"> <li>CMS.WorkflowEngine</li> <li>CMS.DataEngine</li> </ul>	Occurs when the system executes a marketing automation action step.
ProcessTrigger	Before After Failure	AutomationProcessTriggerEventArgs <ul style="list-style-type: none"> <li>Options (IEnumerable&lt;TriggerOptions&gt;)</li> <li>TriggerInfo (ObjectWorkflowTriggerInfo)</li> </ul>	Occurs when processing triggers for automation processes. The system evaluates triggers periodically in bulk for a certain number of related contacts (represented by the collection in the <i>Options</i> property of the <i>AutomationProcessTriggerEventArgs</i> parameter).  Evaluates the trigger condition and allows you to, for example, insert data into the condition.

[> Back to list of event categories](#)

## OutputFilterEvents

**Namespace:** CMS.OutputFilter

Contains events related to [output filters](#).

Event	Event types	Handler parameters	Description
SaveOutputToCache	Before After Failure	OutputCacheEventArgs <ul style="list-style-type: none"> <li>Output (CachedOutput)</li> <li>ViewModel (ViewModelOnDemand)</li> <li>FallbackToRegularLoad (bool)</li> </ul>	Fires when an output filter saves the output to the cache.
SendCacheOutput	Before After Failure	OutputCacheEventArgs <ul style="list-style-type: none"> <li>Output (CachedOutput)</li> <li>ViewModel (ViewModelOnDemand)</li> <li>FallbackToRegularLoad (bool)</li> </ul>	Fires before content from the <a href="#">output cache</a> is sent as a response. Occurs only if the cached output is found, and if the cached output gives an opportunity to the user to change the output or to bypass caching entirely.
CurrentFilter.OnBeforeFiltering		<ul style="list-style-type: none"> <li>filter (ResponseOutputFilter)</li> <li>finalHtml (ref string)</li> </ul>	Fires before output filtering occurs. See <a href="#">Making custom modifications to output HTML</a> .
CurrentFilter.OnAfterFiltering		<ul style="list-style-type: none"> <li>filter (ResponseOutputFilter)</li> <li>finalHtml (ref string)</li> </ul>	Fires after output filtering occurs. See <a href="#">Making custom modifications to output HTML</a> .

[> Back to list of event categories](#)

## URLRewritingEvents

**Namespace:** CMS.URLRewritingEngine

Contains events related to [URL rewriting](#) and page redirection.

Event	Event types	Handler parameters	Description
-------	-------------	--------------------	-------------





PageNotFound	Execute	URLRewriterPageNotFoundEventArgs <ul style="list-style-type: none"> <li>PageNotFoundHandled (bool)</li> <li>RelativePath (string)</li> <li>SiteName (SiteNameOnDemand)</li> <li>ViewMode (ViewModeOnDemand)</li> </ul>	Occurs when the system redirects users to a <a href="#">Page not found error page</a> (after requesting a URL that returns a 404 HTTP status code).
ProcessRewritingResult	Before After Failure	URLRewritingEventArgs <ul style="list-style-type: none"> <li>Parameters (URLRewritingParams)</li> </ul>	<p>Occurs at the end of the rewriting process for all URLs. If there are redirects during the rewriting, the event is only triggered for the final URL at the end of the process.</p> <p>Use the <b>Before</b> event if you need to perform custom rewriting. To avoid the default rewriting steps, set the <i>URLRewritingEventArgs.Parameters.Status</i> property to <i>RequestStatusEnum.PathRewritten</i>.</p> <p>Use the <b>After</b> event to perform additional custom actions after the default rewriting.</p>
ProcessABTest	Execute	ProcessABTestEventArgs <ul style="list-style-type: none"> <li>PageInfo (PageInfo)</li> <li>ReturnedPageInfo (PageInfo)</li> <li>SiteName (String)</li> <li>ViewMode (ViewModeEnum)</li> </ul>	Occurs when the system starts rewriting the URL of a page with a running <a href="#">A/B test</a> .

[> Back to list of event categories](#)

## SystemEvents

**Namespace:** CMS.Base

Contains general system events.

Event	Event types	Handler parameters	Description
Exception	Execute	SystemEventArgs <ul style="list-style-type: none"> <li>Exception (Exception)</li> <li>LogException (bool)</li> </ul>	Fires when an unhandled exception occurs in the system.

[> Back to list of event categories](#)

## SiteEvents

**Namespace:** CMS.SiteProvider

Contains events related to site actions.

Event	Event types	Handler parameters	Description
Delete	Before After Failure	SiteDeletionEventArgs <ul style="list-style-type: none"> <li>User (UserInfo)</li> <li>Context (CMSActionContext)</li> <li>Info (DeletionInfo)</li> </ul>	Occurs when a site is deleted from the system.

[> Back to list of event categories](#)

## SqlEvents

**Namespace:** CMS.DataEngine

Contains events related to the execution of queries and other database operations.

Event	Event types	Handler parameters	Description
BulkInsert	Before After Failure	BulkInsertDataEventArgs <ul style="list-style-type: none"> <li>Connection (IDataConnection)</li> <li>InsertSettings (BulkInsertSettings)</li> <li>SourceData (DataTable)</li> <li>TargetTable (string)</li> </ul>	Occurs when the system creates objects by directly inserting data into a database table (for example when <a href="#">importing contacts</a> from CSV files).
ExecuteQuery	Before After Failure	ExecuteQueryEventArgs <DataSet> <ul style="list-style-type: none"> <li>Connection (IDataConnection)</li> <li>Query (QueryParameters)</li> <li>Result (DataSet)</li> </ul>	Occurs when the system executes a database query.  See also: <a href="#">Working with database queries in the API</a>
ExecuteNonQuery	Before After Failure	ExecuteQueryEventArgs <int> <ul style="list-style-type: none"> <li>Connection (IDataConnection)</li> <li>Result (DataSet)</li> </ul>	Occurs when the system executes a non-query database operation.

ExecuteScalar	Before After Failure	ExecuteQueryEventArgs<object> <ul style="list-style-type: none"> <li>• Connection (IDataConnection)</li> <li>• Query (QueryParameters)</li> <li>• Result (DataSet)</li> </ul>	Occurs when the system executes a scalar query.
ExecuteReader	Before After Failure	ExecuteQueryEventArgs< <a href="#">DbDataReader</a> > <ul style="list-style-type: none"> <li>• Connection (IDataConnection)</li> <li>• Result (DataSet)</li> </ul>	Occurs when the system runs a database DataReader.

[> Back to list of event categories](#)

## Application events

Kentico provides a set of events that you can use to customize the global behavior of the application.

### ApplicationEvents

**Namespace:** CMS.Base

Contains events related to the application life cycle.

The following table lists the events that **occur once in the application's life cycle, in order of execution**:

Event	Event types	Handler parameters	Description
Preinitialized	Execute	<a href="#">EventArgs</a>	Occurs when the application finishes pre-initialization (preparation of things unrelated to the database).  Handlers need to be assigned in the <b>PreInit</b> method of <a href="#">custom modules</a> .
Initialized	Execute	<a href="#">EventArgs</a>	Occurs when the application finishes initialization and starts (including all modules and the database connection). Handle this event to perform standard application start events.
PostStart	Execute	<a href="#">EventArgs</a>	Occurs at the end of the first application request. Allows you to perform asynchronous tasks needed for application startup.  Running long operations within this event's handlers can negatively affect the response time of the first request when starting the application.
End	Execute	<a href="#">EventArgs</a>	Occurs when the application ends.

The following events occur as needed during the application life cycle:

Event	Event types	Handler parameters	Description
Error	Execute	<a href="#">EventArgs</a>	Fires when an error occurs in the application.
SessionStart	Execute	<a href="#">EventArgs</a>	Occurs when an HTTP session starts for a client.

SessionEnd	Execute	<a href="#">EventArgs</a>	Occurs when an HTTP session ends for a client. May not be raised if the application is configured to use a session state mode different than <i>InProc</i> .
GetVaryByCustomString	Execute	GetVaryByCustomStringEventArgs <ul style="list-style-type: none"> <li>• Result (string)</li> <li>• Context (HttpContext)</li> <li>• Custom (string)</li> </ul>	Occurs when the application processes variables in custom names of <a href="#">output cache</a> keys.

## RequestEvents

**Namespace:** CMS.Base

Contains events related to handling of requests. See also: [ASP.NET Application Life Cycle Overview](#)

The following table lists the request events **in order of execution**:

Event	Event types	Handler parameters	Description
Prepare	Execute	<a href="#">EventArgs</a>	Occurs before the request processing begins.
Begin	Execute	<a href="#">EventArgs</a>	Occurs when the request processing starts.
Authenticate	Execute	<a href="#">EventArgs</a>	Occurs when authenticating requests.
Authorize	Execute	<a href="#">EventArgs</a>	Occurs when authorizing requests. If the application loads the response from the <a href="#">output cache</a> , the life cycle skips to the <b>End</b> event.
PostAuthorize	Execute	<a href="#">EventArgs</a>	Occurs after the request's user is authorized.
PostResolveRequestCache	Execute	<a href="#">EventArgs</a>	Occurs when ASP.NET bypasses execution of the current handler and allows a caching module to serve the request from the cache. Not used by the Kentico <a href="#">output caching</a> (see the <i>Authorize</i> event).  The system performs <a href="#">URL rewriting</a> during the event. After this event, the application can access the context of the displayed page and current page (page info).
MapRequestHandler	Execute	<a href="#">EventArgs</a>	Occurs when a handler is selected to respond to the request.
PostMapRequestHandler	Execute	<a href="#">EventArgs</a>	Occurs after the application has mapped the request to the appropriate event handler.
AcquireRequestState	Execute	<a href="#">EventArgs</a>	Occurs when acquiring the state (for example session state) that is associated with the current request.
PostAcquireRequestState	Execute	<a href="#">EventArgs</a>	Occurs after the request state (for example session state) that is associated with the current request has been obtained.
End	Execute	<a href="#">EventArgs</a>	Occurs at the end of the request processing.
RunEndRequestTasks	Execute	<a href="#">EventArgs</a>	Allows you to execute background tasks at the end of the request processing. Running long tasks in this event's handlers can negatively affect the response time of requests.
Finalize	Execute	<a href="#">EventArgs</a>	Occurs when the request is finalized. Use to cleanup and release any resources used by the request. Running long tasks in this event's handlers can negatively affect the response time of requests.

### ✓ Handling other **HttpApplication** events

If you wish to handle **HttpApplication** events that are not available in the Kentico *RequestEvents* class, you need to create and register a custom HTTP module. You can implement the module in the *App\_Code* folder of web site projects or as part of a custom assembly.

See [Walkthrough: Creating and Registering a Custom HTTP Module](#) for more information.

## SessionEvents

**Namespace:** CMS.Membership

Contains events related to session management.

Event	Event types	Handler parameters	Description
UpdateSession	Before After Failure	CMSEventArgs	Occurs when an HTTP session is updated.
UpdateSessionData	Execute	CMSEventArgs<SessionInfo> <ul style="list-style-type: none"> <li>Parameter (SessionInfo)</li> </ul>	Occurs when the data of an HTTP session is updated.

[> Back to list of event categories](#)

## Control and page events

You can handle events related to the life cycle of pages and controls. The events are defined directly in the base classes of system pages and controls. You can leverage the events to customize the user interface. The events are direct delegates.

### AbstractUserControl

**Namespace:** CMS.Base.Web.UI

Contains events that occur during the user control life cycle.

Event	Handler parameter	Handler return type
OnBeforeUserControlInit	<a href="#">EventArgs</a>	bool
OnAfterUserControlInit	<a href="#">EventArgs</a>	void
OnBeforeUserControlLoad	<a href="#">EventArgs</a>	bool
OnAfterUserControlLoad	<a href="#">EventArgs</a>	void
OnBeforeUserControlPreRender	<a href="#">EventArgs</a>	bool
OnAfterUserControlPreRender	<a href="#">EventArgs</a>	void
OnBeforeUserControlRender	<a href="#">HtmlTextWriter</a>	bool
OnAfterUserControlRender	<a href="#">HtmlTextWriter</a>	void



Return *true* in Before handlers to perform the default system actions for the given event.

## CMSPage

**Namespace:** CMS.UIControls

Contains events that occur during the system page life cycle.

Event	Handler parameter	Handler return type
OnBeforePagePreInit	<a href="#">EventArgs</a>	bool
OnAfterPagePreInit	<a href="#">EventArgs</a>	void
OnBeforePageInit	<a href="#">EventArgs</a>	bool
OnAfterPageInit	<a href="#">EventArgs</a>	void
OnBeforePageLoad	<a href="#">EventArgs</a>	bool
OnAfterPageLoad	<a href="#">EventArgs</a>	void
OnBeforePagePreRender	<a href="#">EventArgs</a>	bool
OnAfterPagePreRender	<a href="#">EventArgs</a>	void
OnBeforePageRender	<a href="#">HtmlTextWriter</a>	bool
OnAfterPageRender	<a href="#">HtmlTextWriter</a>	void



Return *true* in Before handlers to perform the default system actions for the given event.

[> Back to list of event categories](#)